Description of Functions 11/2003 Edition

sinumerik

Basic Machine
SINUMERIK 840D/810D/840Di

SIEMENS

SIEMENS

SINUMERIK 840D/840Di/ SINUMERIK 810D

Description of Functions Basic Machine (Part 1)

Description of Functions

Valid for

Control	*Software Version*
SINUMERIK 840D	6
SINUMERIK 840DE (export version)	6
SINUMERIK 840D powerline	6
SINUMERIK 840DE powerline	6
SINUMERIK 840Di	3
SINUMERIK 840DiE (export version)	3
SINUMERIK 810D	3
SINUMERIK 810DE (export version)	3
SINUMERIK 810D powerline	6
SINUMERIK 810DE powerline	6

11.2003 Edition

Various Interface Signals **A2**

Axis Monitoring, Protection Zones **A3**

Continuous Path Mode, Exact Stop and Look Ahead **B1**

Acceleration **B2**

Diagnostic Tools **D1**

Interactive Programming **D2**

Travel to Fixed Stop **F1**

Velocities, Setpoint/ Actual Value Systems, Closed-Loop Control **G2**

Auxiliary Function Output to PLC **H2**

Mode Group, Channel, Program Operation, Reset Response **K1**

Axes, Coordinate Systems, Frames **K2**

Communication **K4**

EMERGENCY STOP **N2**

Transverse Axes **P1**

PLC Basic Program **P3**

Reference Point Approach **R1**

Spindles **S1**

Feeds **V1**

Tool Offset **W1**

Abbreviations **A**

Definition of Terms **B**

References **C**

Index

Complete Index (FB1, FB2, FB3)

SINUMERIK® Documentation

Printing history

Brief details of this edition and previous editions are listed below.

The status of each edition is shown by the code in the “Remarks” column.

Status code in the “Remarks” column:

A . . . . . New documentation.
B . . . . . Unrevised reprint with new order no.
C . . . . . Revised edition with new status.
If factual changes have been made on the page in relation to the same software version, this is indicated by a new edition coding in the header on that page.

Edition	Order No.	Remarks
06.94	6FC5 297-0AC20-0BP0	**A**
08.94	6FC5 297-0AC20-0BP1	**C**
02.95	6FC5 297-2AC20-0BP0	**C**
04.95	6FC5 297-2AC20-0BP1	**C**
12.95	6FC5 297-3AC20-0BP0	**C**
03.96	6FC5 297-3AC20-0BP1	**C**
08.97	6FC5 297-4AC20-0BP0	**C**
12.97	6FC5 297-4AC20-0BP1	**C**
12.98	6FC5 297-5AC20-0BP0	**C**
08.99	6FC5 297-5AC20-0BP1	**C**
04.00	6FC5 297-5AC20-0BP2	**C**
10.00	6FC5 297-6AC20-0BP0	**C**
09.01	6FC5 297-6AC20-0BP1	**C**
11.02	6FC5 297-6AC20-0BP2	**C**
11.03	6FC5 297-6AC20-0BP3	**C**

This book is part of the documentation on CD-ROM **(DOCONCD)**

Edition	Order No.	Remarks
11.03	6FC5 298-6CA00-0BG4	**C**

Trademarks
SIMATIC®, SIMATIC HMI®, SIMATIC NET®, SIROTEC®, SINUMERIK® and SIMODRIVE® are trademarks of Siemens. Other product names used in this documentation may be trademarks which, if used by third parties, could infringe the rights of their owners.

Further information is available on the Internet under:
http://www.ad.siemens.de/mc

This publication was produced with Interleaf V7.

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights, including those created by patent grant or registration of a utility model or design, are reserved.

© Siemens AG, 1994–2003. All rights reserved

Other functions not described in this documentation might be executable in the control. However, no claim can be made regarding the availability of these functions when the equipment is first supplied or for service cases.

We have checked that the contents of this document correspond to the hardware and software described. Nonetheless, differences might exist and therefore we cannot guarantee that they are completely identical. The information contained in this document is, however, reviewed regularly and any necessary changes will be included in the next edition. We welcome suggestions for improvement.

Subject to changes without prior notice.

Order No. 6FC5297-6AC20-0BP3
Printed in Germany

Siemens Aktiengesellschaft

Preface

Reader's note

The SINUMERIK documentation is structured in three levels:

- General documentation
- User documentation
- Manufacturer/service documentation.

This documentation is intended for machine tool manufacturers. It gives a detailed description of the functions available in the SINUMERIK controls.

The descriptions of functions are only valid for the software versions specified. For new software versions, the relevant descriptions are available on request. Earlier descriptions of functions are only partly applicable for new software versions.

For detailed information on further publications on SINUMERIK 840D/840Di/ 810D, as well as on publications applicable to all SINUMERIK control systems (e.g. universal interface, measuring cycles ...), please contact your regional Siemens branch office.

Note

Other functions not described in this documentation might be executable in the control. This does not, however, represent an obligation to supply such functions with a new control or when servicing.

Hotline

If you have any questions about the control, please contact the hotline:

A&D Technical Support
Phone.: ++49-180-5050-222
Fax: ++49-180-5050-223
Email: adsupport@siemens.com

Please send any questions about the documentation (suggestions for improvement, corrections) to the following fax number or email address:
Fax: ++49-9131-98-2176
Email: motioncontrol.docu@erlf.siemens.de

Fax form: see reply form at the end of the manual.

Internet address SINUMERIK

http://www.ad.siemens.de/mc

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D powerline

The

- SINUMERIK 840D powerline and
- SINUMERIK 840DE powerline

with enhanced performance are available as of 09.2001. You will find a list of available **powerline** modules in the Hardware Reference Manual /PHD/ in Section 1.1.

SINUMERIK 810D powerline

The

- SINUMERIK 810D powerline and
- SINUMERIK 810DE powerline

with enhanced performance are available as of 12.2001. You will find a list of available **powerline** modules in the Hardware Reference Manual /PHC/ in Section 1.1.

Objective

The Descriptions of Functions provide the information required to configure and start-up the control.

Target groups

The Descriptions of Functions include information for:

- the configuring engineer
- the PLC programming engineer for the creation of the PLC user program with the signals listed
- the start-up engineer after system configuration and installation
- the service technician for checking and interpreting the status displays and alarms.

Information on using this manual

This Function Manual is structured as follows:

- Overall table of contents (main headings) of the manual
- Descriptions of functions in alphanumeric sequence according to the Function Description codes
- Appendix with Abbreviations, Definition of Terms and References
- Index of all Descriptions of Functions.

Note

The footer on each page contains the following information:
Part of the Description of Functions / Manual / Section – Page

---

© Siemens AG, 2003. All rights reserved

If you require more information about a particular function, you can find the function and its sorting code on the inside front page of the manual.

If you require information on a certain term, please refer to the index in the appendix. The index lists the code of the Description of Functions, the Section number and the page on which information relevant to the term you are seeking can be found.

Definitions on the “effectiveness, data format, input limits etc.” of various signals and data are to be found in the individual Descriptions of Functions in Chapters 4 and 5.
An explanation of this definition is to be found in the following under “Technical Comments”.

!

Important

This documentation is valid for the:

- SINUMERIK 840D control, SW 6
- SINUMERIK 810D control, SW 6
- SINUMERIK 840Di control SW 3.

Specification of software version

The software versions specified in this documentation refer to the SINUMERIK 840D control system. The equivalent software version for the SINUMERIK 810D (if the function is enabled, refer to /BU/, Catalog NC 60) is not explicitly specified. In this case, the following applies:

Table 1-1 Corresponding software version

SINUMERIK 840D		SINUMERIK 810D	SINUMERIK 810D powerline	SINUMERIK 840Di
6.5 (09.03)	corresponds to	–	6.4 (11.02)	3.1 (09.03)
5.3 (04.00)	corresponds to	3.3 (04.00)	3.3 (04.00)	1.1 (07.00)
3.7 (03.97)	corresponds to	1.7 (03.97)	1.7 (03.97)	–

© Siemens AG, 2003. All rights reserved

Explanation of the symbols used in this documentation



Important

This symbol always appears in the documentation when important information is being conveyed.



Ordering Data Option

In this documentation, you will find this symbol with a reference to an order data option. The function described is executable only if the control contains the designated option.



Machine Manufacturer

This symbol appears in this documentation whenever the machine manufacturer can influence or modify the described functional behavior. Please observe the information provided by the machine manufacturer.



Danger

Indicates an imminently hazardous situation which, if not avoided, **will** result in death or serious injury or in substantial property damage.



Warning

Indicates a potentially hazardous situation which, if not avoided, **could** result in death or serious injury or in substantial property damage.



Caution

Used with the safety alert symbol indicates a potentially hazardous situation which, if not avoided, **may** result in minor or moderate injury or in property damage.

Caution

Used without safety alert symbol indicates a potentially hazardous situation which, if not avoided, **may** result in property damage.

© Siemens AG, 2003. All rights reserved

Notice

Used without the safety alert symbol indicates a potential situation which, if not avoided, **may** result in an undesirable result or state.

Technical information

Notations

The following notations and abbreviations are used in this documentation:

- PLC interface signals –> IS “Signal name” (signal date)
  Example – IS “MMC CPU1 ready” (DB10, DBX108.2) means that the signal is filed in data block 10, data byte 108, bit 2
  – IS “Feedrate/spindle override” (DB31–48, DBB0) means that the signals are located referring to spindles and axes in the data blocks 31 to 48, data block byte 0.
- Machine data –> MD: MD_NAME (English designation)
- Setting data –> SD: SD_NAME (English designation)
- The symbol “≙” means “corresponds to”.

Explanation of terms and abbreviations in Chapters 4 and 5

Chapters 4 and 5 of each Description of Functions contain a description of the data and signals which are important for the relevant function. Some of the terms and abbreviations used in this tabulated, explanatory text are explained in more detail below.

Values in the table

The machine data used in the Description of Functions are always values for an NCU 572.

The values for another NCU (for example, NCU 570, NCU 571, NCU 573) are in the list manual.

References: /LIS/ “Lists”

Default value

The machine/setting data is automatically preset to this value during start-up. If the default values depending on the channels are different, these are specified by “ / ”.

Range of values (minimum and maximum values)

Specifies the input limits. If no value range is defined, the input limits are defined by the data type and the field is indicated by “***”.

Effectiveness of changes

Changes to machine data, setting data, or similar parameters do not necessarily take immediate effect in the control. For this reason, the conditions for activating a new setting are always specified. The following list shows the applied conditions in prioritized order:

- POWER ON (po) “RESET” key on front panel of NCU module or disconnection/connection of power supply
- NEW_CONF (cf) – “New configuration” function via PLC interface
  – “RESET” key on control unit
- RESET (re) “RESET” key on control unit
- Immediately (im) after value entry

© Siemens AG, 2003. All rights reserved

Protection level

There are protection levels 0 to 7. Levels 0 to 3 (4 to 7) can be enabled by entering a password (keyswitch setting). The operator only has access to information on the level for which he is authorized and for all lower levels. Various protection levels are assigned to machine data as standard.

The table only lists the write protection level. However, there is a fixed assignment between write and read levels:

Write protection level	Read protection level
0	0
1	1
2	4

References: /BA/, Operator's Guide
/FB/, A2, "Various Interface Signals"

Unit

The unit refers to the default settings of the machine data SCALING_FACTOR_USER_DEF_MASK and SCALING_FACTOR_USER_DEF.
If the MD is not governed by any particular physical unit, then the symbol "–" is entered in the unit field.

Data type

The following data types are used in the control system:

- DOUBLE
  Real or integer values (decimal values or integers)
  Input limits from $+/-4.19*10^{-307}$ to $+/-1.67*10^{308}$
- DWORD
  Integer values
  Input limits from $-2{,}147*10^{9}$ to $+2{,}147*10^{9}$
- BOOLEAN
  Possible input values: true or false and/or 0 or 1
- BYTE
  Integer values from –128 to +127
- STRING
  Consisting of max. 16 ASCII characters (capital letters, digits and underscore).

Data management

The explanations of the PLC interface in the individual Function Descriptions are based on a theoretical maximum number of components:

- 4 operating mode groups (associated signals stored in DB11)
- 8 channels (associated signals stored in DB21–30)
- 31 axes (associated signals stored in DB31–61).

For the number of components which can actually be implemented in the relevant software version, please refer to
References: /BU/, "Order Document" Catalog NC 60

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Various Interface Signals (A2)

1	**Brief Description**		**1/A2/1-3**
2	**Detailed Description**		**1/A2/2-5**
	2.1	General	1/A2/2-5
	2.2	Ready signals to PLC (DB10)	1/A2/2-7
	2.3	Alarm signals to PLC (DB10 and DB21, ...)	1/A2/2-8
	2.4	New interface signals in DB10 for SINUMERIK 840Di	1/A2/2-9
	2.5	Access protection using password and keyswitch	1/A2/2-10
	2.5.1	Password	1/A2/2-11
	2.5.2	Keyswitch settings (DB10, DBX56.4 to 7)	1/A2/2-12
	2.5.3	MMC machine data for protection levels	1/A2/2-13
	2.5.4	Tool geometry and restricted wear input	1/A2/2-14
	2.6	Interface signals for operator panel (DB19)	1/A2/2-15
	2.7	Interface signal from and to channel (DB21, ...)	1/A2/2-19
	2.7.1	Interface signals to channel	1/A2/2-19
	2.8	Interface signals from and to axis/spindle (DB31, ...)	1/A2/2-19
	2.8.1	Interface signals to axis/spindle	1/A2/2-19
	2.8.2	Interface signals to from axis/spindle	1/A2/2-30
	2.9	Interface signals for digital drives (DB31, ...)	1/A2/2-32
	2.9.1	Signals to axis/spindle	1/A2/2-32
	2.9.2	Signals from axis/spindle	1/A2/2-33
	2.10	Screen settings	1/A2/2-35
	2.11	General functions	1/A2/2-37
	2.11.1	Settings for involute interpolation (SW 6 and higher)	1/A2/2-37
	2.11.2	Activate DEFAULT memory	1/A2/2-40
	2.12	Reading and writing PLC variables (SW 4 and higher)	1/A2/2-41
3	**Supplementary Conditions**		**1/A2/3-45**
4	**Data Descriptions (MD, SD)**		**1/A2/4-47**
	4.1	Machine data for operator panel	1/A2/4-47

© Siemens AG, 2003. All rights reserved

4.2 General machine data 1/A2/4-58
4.3 Channel-specific machine data 1/A2/4-60
4.4 Axis/spindle-specific NCK machine data 1/A2/4-61
4.5 Machine data for involute (SW 6 and later) 1/A2/4-64
4.6 System variables (SW 4 and later) 1/A2/4-65

5 Signal Descriptions 1/A2/5-67

5.1 NC-specific signals 1/A2/5-67
5.1.1 Signals from PLC to NC (DB10) 1/A2/5-67
5.1.2 Signals from NC to PLC (DB10) 1/A2/5-68
5.2 Signals to/from operator panel (DB19) 1/A2/5-71
5.2.1 Signals to operator panel (PLC → MMC) 1/A2/5-71
5.2.2 Signals from operator panel (MMC → PLC) 1/A2/5-77
5.3 Channel-specific signals 1/A2/5-83
5.3.1 Signals to channel 1/A2/5-83
5.3.2 Signals from channel 1/A2/5-84
5.4 Axis/spindle-specific signals 1/A2/5-85
5.4.1 Signals to axis/spindle 1/A2/5-85
5.4.2 Signals from axis/spindle 1/A2/5-97

6 Example 1/A2/6-113

6.1 Controller parameter set switchover 1/A2/6-113

7 Data Fields, Lists 1/A2/7-115

7.1 Interface signals 1/A2/7-115
7.2 Machine data 1/A2/7-118
7.3 System variables 1/A2/7-121
7.4 Alarms 1/A2/7-121

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

Contents

This Description of Functions describes the functionality of various interface signals which are of general relevance but are are not included in the other existing Descriptions of Functions.

Asynchronous events

The CNC offers the possibility of activating specific reactions in response to asynchronous events on the machine. The PLC user program reports these events via the PLC interface of the NC. The NC processes this PLC interface cyclically and triggers the relevant internal reactions.

Signaling states

The NC also signals various functional states to the PLC via the interface which the PLC user program can then process on a machine-specific basis.

Read/write PLC variable (SW 4 and higher)

For highspeed exchange of information between the PLC and NC, a memory area is reserved in the interface memory of these modules (dualport RAM). Variables of any type (I/O, DB, DW, flags) may be exchanged within this memory area.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 General

Interface PLC/NCK

The exchange of signals and data between the PLC user program and the NCK, MMC and MCP areas is organized by the basic PLC program.

The PLC/NCK interface consists of a data interface and a function interface.

Tasks from the PLC user program to the NCK can be transferred via the function interface (e.g. to traverse concurrent axes). The function interface is generated by function calls (FC). For further information about PLC function calls, please refer to
References: /FB/, P3, “Basic PLC Program”.

We distinguish between the following signal and data groups:

- Cyclic signal exchange of control and status signals
- Eventdriven signal exchange (auxiliary and G functions), see
  References: /FB/, H2, “Output of Auxiliary Functions to PLC”

For an overview of all PLC/NCK interface signals, function blocks used by manufacturers as well as assigned data blocks, please refer to
References: /LIS/, Lists

Cyclic signal exchange

The control and status signals of the PLC/NCK interface are transferred cyclically by the basic PLC program (OB1).

The signals can be subdivided into the following groups:

- General NC-specific signals
- Mode group-specific signals
- Channel-specific signals
- Axis/spindle-specific signals.

The structure of the interface and the functions of the basic program are described in detail in
References: /FB/, P3, “Basic PLC Program”.

As already explained, this description of functions describes the control and status signals of the PLC interface which are of universal significance and cannot be assigned to other existing Descriptions of Functions (e.g. IS “Servo enable”). Only these PLC interface signals are therefore dealt with on the following pages.

© Siemens AG, 2003. All rights reserved

General signals (DB10)

From PLC to NC:

- Signals for influencing the CNC inputs and outputs
- Keyswitch signals (and password).

From NC to PLC:

- Actual values of CNC inputs
- Setpoints of CNC outputs
- Ready signals from NC, NCK, MMC
- NC status signals (alarm signals).

OP-specific signals (DB10)

From PLC to NC:

- Signals for influencing the CNC inputs and outputs
- Keyswitch signals (and password).

From NC to PLC:

- Actual values of CNC inputs
- Setpoints of CNC outputs
- Ready signals from NC, NCK, MMC
- NC status signals (alarm signals).

Channel-specific signals (DB21, ...)

From PLC to NC:

- Control signal “Delete distance-to-go”

From NC to PLC:

- NC status signals (NCK alarm active).

Axis/spindle-specific signals (DB31, ...)

From PLC to NC:

- Control signals to axis/spindle (e.g. follow-up mode, servo enable, ...)
- Control signals to SIMODRIVE 611D (bytes 20, 21).

From NC to PLC:

- Status signals from axis/spindle (e.g. position controller active, current controller active, ...)
- Status signals from SIMODRIVE 611D or SIMODRIVE 611U in conjunction with 840 Di (bytes 93, 94)

Note

In Chapter 5 you will find a list of the interface signals and data described in this Description of Functions.
For an overview diagram of the PLC/NCK interface, please refer to:
References: /FB/, P3, “Basic PLC Program”

© Siemens AG, 2003. All rights reserved

2.2 Ready signals to PLC (DB10)

Note

For further information about the interface signals listed below, please refer to Chapter 5.

NCK CPU Ready (DB10, DBX104.7)

The NCK CPU is ready and sending ready signals cyclically to the PLC.

MMC2 CPU Ready DB10, DBX108.1)

MMC2 (see FB B3) is ready and sending signals cyclically to the NCK. Signal is valid for connecting MMC2 to OPI and MPI.

MMC CPU1 Ready (DB10, DBX108.2) (DB10, DBX108.3)

(DB10, DBX108.2) MMC to MPI
(DB10, DBX108.3) MMC to OPI, standard link
The MMC CPU is ready and sending signals cyclically to the NCK.

611D Ready (DB10, DBX108.6)

The 611D signals to the PLC via the NCK that all installed drives are ready to operate. IS “Drive Ready” (DB31, ... DBX93.5) (group signal) is active on all axes and spindles. 611U signals “Drive Ready” to the PLC (DB31, ...DBX93.5), when the drive is ready. 611D Ready is only output with the drive 611D.

NC Ready (DB10, DBX108.7)

The control system is ready.

© Siemens AG, 2003. All rights reserved

2.3 Alarm signals to PLC (DB10 and DB21, ...)

Note

For further information about the interface signals listed below, please refer to Section 5.1 (for DB10) and Section 5.3 (for DB21, ...).

MMC Alarm is active (DB10, DBX103.0)

The MMC sends this signal to the PLC to indicate that at least one MMC alarm (between alarm numbers 100000 and 105999) is active.

Air temperature alarm (DB10, DBX109.6)

The ambient temperature or fan monitoring function has responded.

NCK battery alarm (DB10, DBX109.7)

The battery voltage has dropped below the lower limit value. The control can still be operated. Loss of voltage would lead to the loss of data in the volatile memory areas.

NCK alarm is active (DB10, DBX109.0)

The control sends this signal to the PLC to indicate that at least one NCK alarm is active. The channel-specific interface can be scanned to see which channels are involved and whether this will cause a processing stop.

Channel-specific NCK alarm is active (DB21, ... DBX36.6)

The control system sends this signal to the PLC to indicate that at least one NCK alarm is active for the affected channel. To what extent this may influence whether the current program run will be interrupted or aborted can be determined from IS “NCK alarm with processing stop is active”.

NCK alarm with processing stop is active (DB21, ... DBX36.7)

The control sends this signal to the PLC to indicate that at least one NCK alarm, which has interrupted or aborted the current program run (processing stop), is active for the affected channel.

© Siemens AG, 2003. All rights reserved

2.4 New interface signals in DB10 for SINUMERIK 840Di

PC Shutdown (DB10,...DBX57.3)

If the signal “PC Shutdown” is set, NC and PLC will shut down automatically. Then WINDOWS NT will shut down. SINUMERIK 840Di or the PC can now be turned off.

The PC can be shut down using, for example, an external pushbutton before the power is switched off or the capacity of a UPS (uninterruptible power supply) runs out.

PC OS fault (DB10,...DBX109.4)

If WINDOWS NT detects a fatal exception error (blue screen) during the operation of NC and PLC, the interface signal “PC OS fault” (PC Operating System fault) is set.

A fatal exception error of WINDOWS NT has no influence on the SINUMERIK 840Di NC and PLC. The execution of parts programs and of the PLC user program is continued.

Note

To shut down the whole system (NC, PLC and WINDOWS NT) after a fatal exception error (blue screen), you must first shut down the SINUMERIK 840Di (PLC and NC) by setting the interface signal “PC Shutdown”.

As long as the SINUMERIK 840Di (NC and PLC) is still in operation after a fatal exception error has occurred, it is not possible to restart WINDOWS NT.

© Siemens AG, 2003. All rights reserved

2.5 Access protection using password and keyswitch

Access authorization

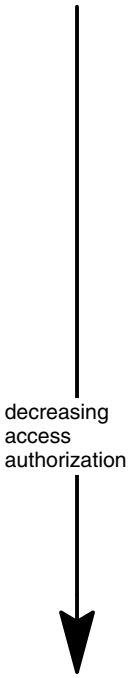
Access to programs, data and functions is user-oriented and controlled via 8 hierarchical protection levels. These are divided into (see Table 2-1):

- 4 password levels for Siemens, machine manufacturer and end user
- 4 keyswitch positions for end user

This provides a multilevel safety concept for controlling access rights.

Table 2-1 Access protection

Protection level	Type	User	Access to (examples)
0	Password	Siemens	All functions, programs and data
1	Password	Machine manufacturer: development	Defined functions, programs and data; e.g.: entering options
2	Password	Machine manufacturer: commissioner	Defined functions, programs and data; e.g.: bulk of machine data
3	Password	End user: service	Assigned functions, programs and data
4	Keyswitch pos. 3	End user: programmer setter	Less than protection level 0 to 3; defined by machine manufacturer or end user
5	Keyswitch pos. 2	End user: qualified user who does not program	Less than protection level 0 to 3; defined by end user
6	Keyswitch pos. 1	End user: trained user who does not program	Example: only program selection, entering tool wear data, and zero offsets
7	Keyswitch pos. 0	End user: semi-skilled user	Example: no inputs or program selection possible, only operation of machine control panel



Access features

- Protection level 0 provides the greatest number of access rights, protection level 7 the least.
- If certain access rights are granted to a protection level, these protection rights automatically apply to any higher protection levels.
- Conversely, protection rights for a certain protection level can only be altered from a higher protection level.

© Siemens AG, 2003. All rights reserved

- Access rights for protection levels 0 to 3 are permanently assigned by Siemens and cannot be altered (default).
- Access rights can be set by querying the current keyswitch positions and comparing the passwords entered. When a password is entered it overwrites the access rights of the keyswitch position.
- Options can be protected on each protection level. However, option data can only be entered in protections levels 0 and 1.
- Access rights for protection levels 4 to 7 are only suggestions and can be altered by the machine tool manufacturer or end user.

2.5.1 Password

Set password

For the four possible password levels with their respective access authorization, passwords can be entered in the DIAGNOSIS area by pressing the **SET PASSWORD** soft key.
References: /BA/, “Operator’s Guide”

Resetting password

Please note that a password is valid as long as the access authorization is canceled again selectively by means of soft key **DELETE PASSWORD**.
Access authorization is therefore not automatically deleted during POWER ON!

Possible characters

The password can be up to eight characters in length. We recommend that you confine yourself to the characters available on the operator panel when defining the password. Where a password consists of less than eight characters, the additional characters are interpreted as blanks.

Default passwords

The following default passwords have been set for protection levels 1 to 3:

Protection level 1	SUNRISE
Protection level 2	EVENING
Protection level 3	CUSTOMER

Note

These passwords are entered as default on power up of the system in start-up mode (NCK start-up switch is at position 1).

These passwords should be changed to ensure effective access protection.

© Siemens AG, 2003. All rights reserved

2.5.2 Keyswitch settings (DB10, DBX56.4 to 7)

Keyswitch

The keyswitch on the SINUMERIK 840D/840Di/810D has four positions to which protection levels 4 to 7 are assigned. The keyswitch has three keys of different colors that can be removed in different settings (see table 2-2). The key positions can be assigned to functions by the machine tool manufacturer or end user. User-oriented access to programs, data and functions can be set in the machine data (see Subsection 2.5.3).

The keyswitch positions are transferred to the PLC interface (IS “Keyswitch position 0 to 3” (DB10, DBX56.4 to 7) and can be evaluated by the PLC user program.
Keyswitch position 0 represents the fewest access rights and position 3 the most. For example, all data that can be changed in keyswitch positions 0, 1 or 2 can also be changed in position 3.

Keyswitch applications

Access to certain data areas can be disabled with the keyswitch. Unintentional changes to geometry data (e.g. zero offsets) or activation of program control functions (e.g. selecting dry run feedrate) by the operator are therefore ruled out.

Table 2-2 Keyswitch setting 0 to 3

Keyswitch pos.	Retraction pos.	DB10, DBB56	Protection level
Position 0 	–	Bit 4	7
Position 1 	0 or 1 black key	Bit 5	6
Position 2 	0 or 1 or 2 green key	Bit 6	5
Position 3 	0 or 1 or 2 or 3 red key	Bit 7	4

Influencing the PLC user program

The PLC IS interface signals “Keyswitch position 0 to 3” can be defined either directly via the keyswitch on the machine control panel or from the PLC user program. Only one interface signal must be set. If more than one interface signal is set at the same time, the control sets keyswitch position 3.

Further information

See Chapter 5

© Siemens AG, 2003. All rights reserved

2.5.3 MMC machine data for protection levels

Lockable data areas

The machine manufacturer or end user can assign the necessary protection levels for individual functions and data areas by means of MMC machine data. Different protection levels can be defined for read and write access of some of the data types.
Below is a list of the most important MMC machine data for the protection levels:

MMC machine data	Access to
USER_CLASS_READ_TOA	Read all tool offsets
USER_CLASS_WRITE_TOA	Write all tool offsets
USER_CLASS_WRITE_TOA_GEO	Write tool geometry (incl. type and cutting edge)
USER_CLASS_WRITE_TOA_WEAR	Write tool wear (without limiting value)
USER_CLASS_WRITE_TOA_ADAPT	Write tool adapter geometry values
USER_CLASS_WRITE_ZOA	Write settable zero offsets
USER_CLASS_OVERSTORE_HIGH	Extended overstore enabled
USER_CLASS_WRITE_PRG_CONDIT	Enable program modification
USER_CLASS_DRF_INPUT	Enter DRF offset
USER_CLASS_WRITE_SEA	Write setting data
USER_CLASS_READ_PROGRAM	Read parts programs
USER_CLASS_WRITE_PROGRAM	Write/edit parts programs
USER_CLASS_SELECT_PROGRAM	Enable program selection
USER_CLASS_TEACH_IN	Enable TEACH IN
USER_CLASS_PRESET	Enable PRESET
USER_CLASS_CLEAR_RPA	Delete all R parameters
USER_CLASS_WRITE_RPA	Write R parameters

Default setting

These MMC machine data are preset with protection level 7 during standard installation. This means that these data areas and functions can also be influenced at keyswitch position 0. If required, these protection levels can be changed by the machine manufacturer or end user. Protection levels 0 to 3 can also be entered.

© Siemens AG, 2003. All rights reserved

2.5.4 Tool geometry and restricted wear input

Protection levels

The protection levels described in Section 2.5 should be used as follows:
MD 9201: USER_CLASS_WRITE_TOA_GEO = 5
MD 9202: USER_CLASS_WRITE_TOA_WEAR = 6
i.e. higher access rights must prevail for tool geometry input than for wear input. The person entering the wear must have at least protection level 6 in this example.

Restricting the wear input

The difference between the old and new wear value may not exceed the value in MD 9450: MM_WRITE_TOA_FINE_LIMIT assume set value.

© Siemens AG, 2003. All rights reserved

2.6 Interface signals for operator panel (DB19)

Operator panel inhibit (DB19, DBX0.0)

The entire operator panel is disabled by means of IS “Operator panel inhibit” = 1.

Darken screen (DB19, DBX0.1)

The screen is darkened or brightened (e.g. for screen saving purposes).

- **Via the keyboard (“automatic screen darkening”)**
  If a key is not pressed for a time period corresponding to the setting in MMC MD 9006: DISPLAY_BLACK_TIME (default = 3 minutes), the screen is automatically darkened.
  The screen is switched bright again the next time a key is pressed on the operator panel (this first keystroke does not trigger any operator action).

  The precondition for automatic screen darkening is:

  – IS “Darken screen” = 0

  – MMC MD 9006: DISPLAY_BLACK_TIME > 0

- **By the PLC user program via IS “Darken screen”**
  The screen can also be darkened (1 signal) or brightened (0 signal) by means of IS “Darken screen” directly from the PLC.

  As soon as IS “Darken screen” is set to 1, the following applies:

  – It is no longer possible to switch the screen bright again on the keyboard (see above).

  – The first keystroke on the operator panel already triggers an operator action.

Note

As soon as the screen is darkened by means of IS “Darken screen”, the keyboard should be inhibited **simultaneously** by IS “Key disable” = 1 to prevent the operator from entering any keyboard commands unintentionally!

Key disabled (DB19, DBX0.2)

IS “Key disable” can be applied to disable (1 signal) or enable (0 signal) the connected keyboard (see Section 5.2).

© Siemens AG, 2003. All rights reserved

Cancel alarms (DB19.DBX0.3) and Recall alarms (DB19.DBX0.4)

IS “Cancel alarms” and “Recall alarms” can be used to initiate central clearing of the “Cancel alarms” and “Recall alarms” (e.g. MMC alarms). When the central clear error key is pressed on the machine control panel, the application resets bit DB19.DBX0.3 for “Cancel alarms” (and DB19.DBX0.4 for “Recall alarms”). The master control receives the bits and transmits the Cancel (and Recall) key to the MBDDE, which then acknowledges all Cancel alarms (and Recall alarms) of the NCK and MMC. The PLC application can acknowledge the PLC alarms itself. PowerOn and Reset alarms remain active on the NCK until the cause of the error has been rectified.

The master control sets bits DB19.DBX20.3 (DB19.DBX20.4) in order to acknowledge the clear request. The PLC application then clears all of the involved bits again.

Actual value in WCS (DB19, DBX0.7)

Actual values are displayed on the basis of two different coordinate systems:

- **Machine coordinate system (MKS)**
  The machine coordinate system defines the relationships of the machine axes. In the MCS all machine and special axes are displayed.
- **Workpiece coordinate system (WKS)**
  The assignments between the workpiece and machine axes are made by means of special transformations (frame definitions, zero offsets). The workpiece is always described in a Cartesian coordinate system. In the WCS, all geometry and special axes are displayed.

Actual value display

It is possible to select which coordinate system is to be displayed

- by the operator using the soft keys “Actual values MCS” or “Actual values WCS”.

References: /BA/, “Operator’s Guide”

- via the PLC with IS “Actual value in WCS”.

If the PLC sets IS “Actual value in WCS” to “1”, then the actual values in the workpiece coordinate system (WCS) are always displayed **when the machine area is selected**. Within the machine area, the operator can switch the actualvalue display between the WCS and the MCS with the above soft keys.

Actual values in WCS

When you select “WCS”, the geometry and special axes, and their actual positions and distances to go, are displayed in the workpiece coordinate system in the “Position” window.

Actual values in the MCS

When you select “MCS”, the geometry and the special axes with their actual position and distances-to-go are displayed in the “Position” window in the machine coordinate system (see Section 5.2).

© Siemens AG, 2003. All rights reserved

Controlling the RS232C (MMC 100 only) (DB19.DBB12)

This byte for controlling the RS232 interface is the incoming byte from PLC–>MMC. It can be used to initiate the tasks “RS232 On”, “RS232 Off”, “RS232 external”, “RS232 Stop” etc. The tasks refer to the user control file stored in DB19.DBB14 and DB19.DBB15.

Controlling file transfer via hard disk (MMC 103 only) (DB19.DBB13)

This byte for controlling file transfer via hard disk is the request byte from PLC–>MMC. It can be used to initiate the tasks “Select”, “Load” and “Unload”. The tasks refer to the user control file stored in DB19.DBB16 and DB19.DBB17.

Controlling the RS232C (MMC 100 only) (DB19.DBB14)

This byte for controlling the RS232 interface defines the PLC index which specifies the axis, channel or TO number for the standard control file. This file is handled from PLC–>MMC in accordance with the task stored in DB19.DBB12.

Controlling the RS232C (MMC 100 only) (DB19.DBB15)

This byte for controlling the RS232 interface defines the line of the standard or user control file on which the file to be transferred is specified.

Controlling file transfer via hard disk (MMC 103 only) (DB19.DBB16)

This byte for controlling file transfer via hard disk defines the index for the control file (job list). This file is handled from PLC–>MMC in accordance with the task stored in DB19.DBB13.

Controlling file transfer via hard disk (MMC 103 only) (DB19.DBB17)

This byte for controlling file transfer via hard disk indicates which line in the user control file contains the control file to be transferred.

Controlling the RS232C (MMC 100 only) (DB19.DBB24)

This byte for controlling the RS232 interface is the acknowledgment byte from MMC–>PLC and contains the current status of data transfer for “RS232 ON”, “RS232 OFF”, “RS232 external”, “RS232 Stop” etc. or indicates if a data transfer error occurred.

Controlling the RS232C (MMC 100 only) (DB19.DBB25)

The data transfer error values from MMC–>PLC are output in this byte for controlling the RS232 interface.

© Siemens AG, 2003. All rights reserved

Controlling file transfer via hard disk (MMC 103 only) (DB19.DBB26)

This byte for controlling file transfer via hard disk is the acknowledgment byte from MMC–>PLC and contains the current status of data transfer for “Select”, “Load” or “Unload” or indicates if a data transfer error occurred.

Controlling file transfer via hard disk (MMC 103 only) (DB19.DBB27)

This byte for controlling file transfer via hard disk is used to output the data transfer error values from MMC–>PLC.

© Siemens AG, 2003. All rights reserved

2.7 Interface signal from and to channel (DB21, ...)

2.7.1 Interface signals to channel

Delete distance-to-go (channel-specific) (DB2128, DBX6.2)

The IS “Delete distance-to-go (channel-specific)” can only be applied to path axes.
As the edge of the interface signal rises, the distance-to-go of all axes in the geometry grouping is deleted and the axes thus brought to a standstill with ramp stop. The next program block is then started.

For a more detailed description, see Section 5.3.

2.8 Interface signals from and to axis/spindle (DB31, ...)

2.8.1 Interface signals to axis/spindle

Drive test travel enable (DB31, ... DBX1.0)

Drive test travel enable function
Axes that are started without operator intervention, e.g. by a function generator, require a safety handshake executed in the following sequence:

The NC first requests permission to travel from the PLC with:

- IS “Drive test travel request” (DB31, ... DBX61.0) = 1

The PLC user program is checked to ensure that the relevant axes can be safely moved. Axes with a mechanical brake, for example, must be actuated beforehand. Once all travel conditions are fulfilled, the PLC returns the following ready confirmation to the NC:

- IS “Drive test travel enable” (DB31, ... DBX1.0) = 1

The function generator in the NC can now be started.

Axis/spindle disable (DB31, ... DBX1.3)

Axis disable function
If IS “Axis/spindle disable” is set, then the position setpoint is kept constant. Axis travel is disabled. If the signal is applied only briefly before the end of the traversing block, the axis is not started again for the rest of the block. When the disable is canceled, the next movement does not start again until the next traversing block for the axis. Any remaining distance from the previous block is traversed in the new block as well as the new movement.

Note

Attention must be paid to avoid incorrect positioning/contouring errors due to remaining uncompleted blocks.

© Siemens AG, 2003. All rights reserved

Spindle disable function
If IS “Spindle disable” is set, a “0” speed setpoint is output to the speed controller for the relevant spindle in openloop control mode or, in positioning mode, the position setpoint is kept constant analogously to the “axis disabled” state. The movement of the spindle is thus disabled.

For a more detailed description, see Section 5.4.

Follow-up mode (DB31, ... DBX1.4)

If an axis/spindle is operating in follow-up mode, then its setpoint position is made to track the current actual value position. As shown in Fig. 2-1, the position setpoint in follow-up mode is not defined by the interpolator but derived from the actual position value. As actual position value acquisition of the axis continues, it is not necessary to rereference the axis when follow-up mode has been canceled.
Zero speed control, clamping and position monitoring are not active during follow-up mode.

Effect

The IS “Follow-up mode” is only of relevance if the drive servo enable has been removed (e.g. by IS “Servo enable” = 0 signal or because of a fault in the control), or because servo enable is being reissued.

- IS “Follow-up mode” = 1 (follow up; see Fig. 2-3)

  If “Servo enable” is removed the position setpoint of the axis concerned is continuously corrected to the actual value. This state is signaled to the PLC by means of IS “Follow-up mode active” (DB31, ... DBX61.3) = “1”. If the “Servo enable” is enabled again, a controlinternal repositioning operation is initiated (REPOSA: linear approach with all axes) to the last programmed position if a parts program is active. Otherwise, all further axis movements start at the, possibly changed, new actual position.

  Clamping or zero speed monitoring **is inactive**.

- IS “Follow-up mode” = 0 (Hold; see Fig. 2-2)

  If “Servo enable” is removed, the old position setpoint is maintained. If the axis is pushed out of position, a following error between position setpoint and actual value results which is corrected when “Servo enable” is set. All subsequent axis movements start at the setpoint position which existed before “Servo enable” was removed.
  IS “Follow-up mode active” is set set to a 0 signal during the “Hold” state.

  Clamping or zero speed monitoring **remains active**.

If IS “Servo enable” (DB31, ... DBX2.1) is set (1 signal), IS “Follow-up mode” (DB31, ... DBX1.4) has no effect.

Example: Clamping an axis

The following example is given to explain the positioning behavior of the axis/spindle when “Servo enable” is set (see Figs. 2-2 and 2-3).

As the Y axis can be pushed out of its actual position Y_1 to clamping position Y_k by mechanical or electrical influences, “Servo enable” for the Y axis is first removed.

© Siemens AG, 2003. All rights reserved

Hold

When IS “Servo enable” is set, IS “Follow-up mode” switches to 0 (Fig. 2-2):

When closed-loop control is reactivated (by setting IS “Servo enable”) the “old” setpoint position Y_1 is immediately approached again because the setpoint position has not been corrected to the actual value position (Fig. 2-2). In the next traversing block the path contour is traversed exactly as defined in the parts program (advantage!).

Caution: This return movement to the old setpoint position is made directly by the position controller (i.e. without acceleration characteristic) and can place a load on the mechanical system, depending on the distance.

Note

To prevent activation of alarm 25040 “Zero speed monitoring” during the clamping process, it must be replaced by IS “Clamping in progress” (DB31, ... DBX2.3) from the clamping monitoring function.

Follow-up

When IS “Servo enable” is set, IS “Follow-up mode” switches to 1 (Fig. 2-3):

When the closed-loop control system is reactivated, a control-internal repositioning operation is performed (REPOSA: linear approach with all axes) to the last programmed position if a parts program is active.

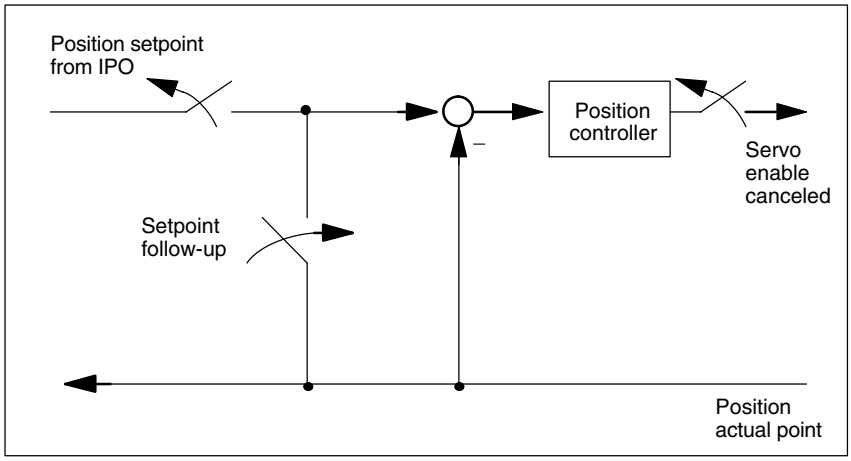


Fig. 2-1 Results when “Servo enable” and “Follow-up mode” are canceled

© Siemens AG, 2003. All rights reserved

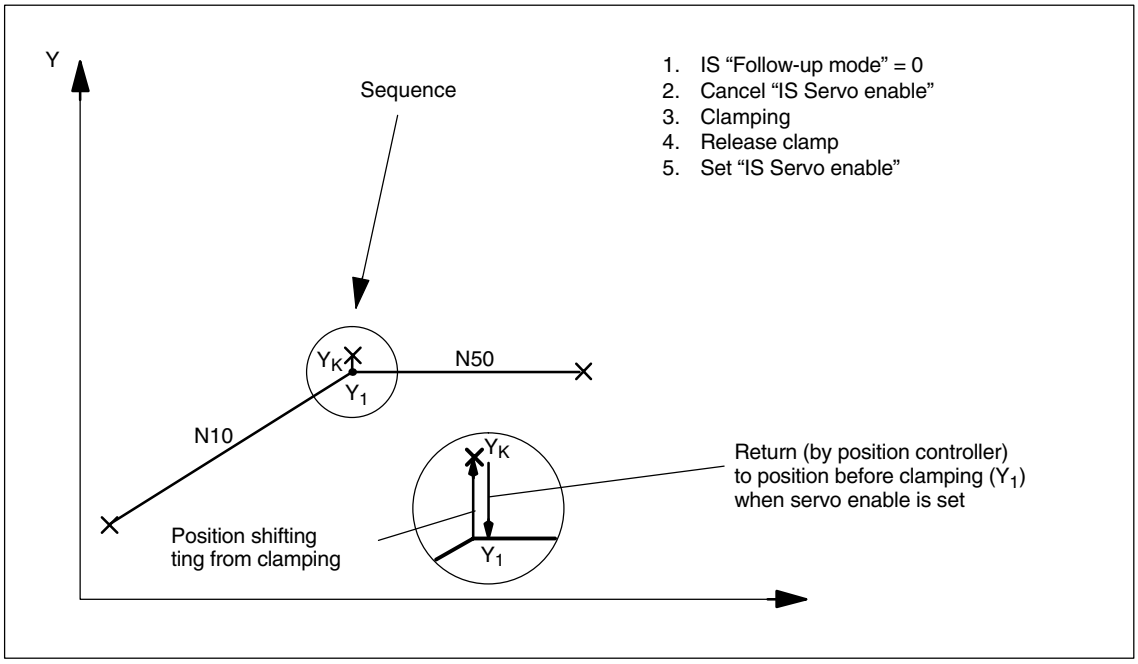


Fig. 2-2 Path followed with clamping **without** “Follow-up” (’hold’)

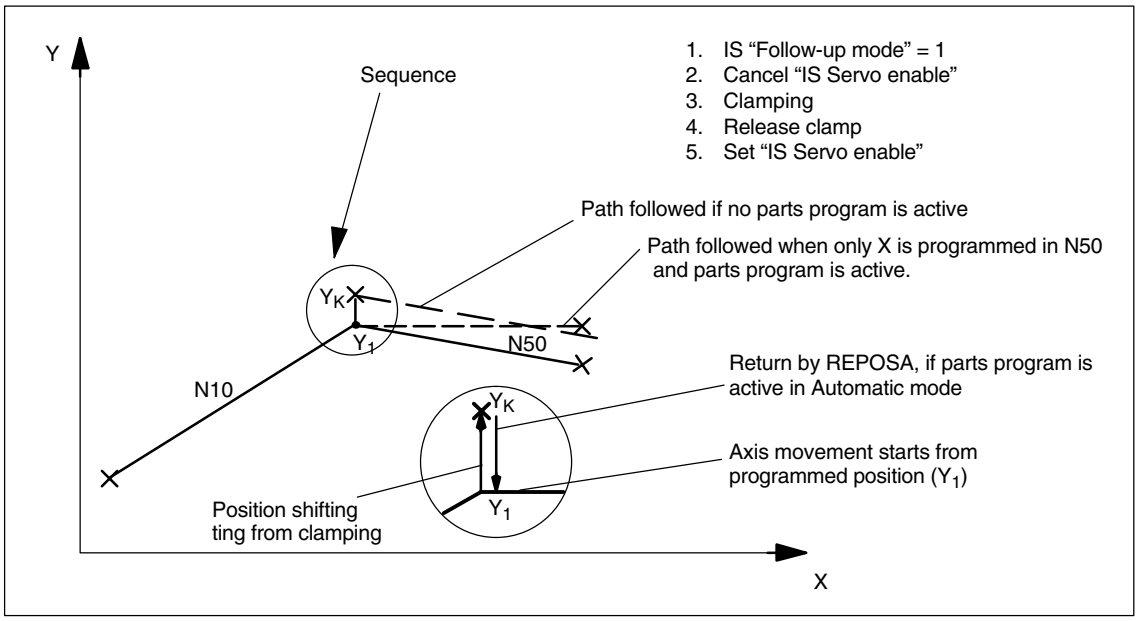


Fig. 2-3 Path followed with clamping **with** “Follow-up”

© Siemens AG, 2003. All rights reserved

Application option for analog drive

On analog drives, it is possible to periodically traverse the axis using an external setpoint. If IS “Follow-up mode” has been set for this axis, actual value acquisition of the axis continues so that subsequent referencing is not necessary.

The following procedure is recommended:

1. Canel IS “Servo enable” (DB31, ... DBX2.1) at the interface while setting IS “Follow-up mode” to “1” beforehand or in the same PLC cycle.
   ⇒ Axis/spindle is operating in follow-up mode
2. Apply external servo enable and external speed setpoint
   ⇒ Axis/spindle uses external setpoint value
   ⇒ NC continues to acquire the actual position value and corrects the setpoint to actual value position
3. Cancel external servo enable and external speed setpoint
   ⇒ Axis/spindle stops
4. Set “IS Servo enable” on interface
   ⇒ NC synchronizes to the actual value position. The next traversing movement begins at this position.

Note

IS “Follow-up mode” does not have to be canceled because it only has an effect in combination with IS “Servo enable”.

Canceling follow-up mode

When Servo enable has been canceled, no new actual value synchronization of the axis (reference point approach) is necessary if the maximum permissible limit frequency of the axis measuring system has not been exceeded during follow-up mode. The control system automatically detects this error by setting interface signal “Referenced/synchronized” (DB31, ... DBX60.4 and 60.5) to “0” and activating the alarm 21610 “Encoder frequency exceeded” for the relevant axis.

Note

If an axis follow-up is deactivated within an active transformation (e.g. TRANSMIT), REPOS movements can be initiated on the other axis or axes involved in the transformation.

Monitoring

If an axis/spindle is in follow-up mode (IS “Follow-up mode” active = 1), the following monitoring types are no longer active:

- Zero speed monitoring
- Clamping monitoring
- Positioning monitoring.

In addition, interface signals “Position reached with exact stop fine” (DB31, ... DBX60.7) and “Position reached with exact stop coarse” (DB31, ... DBX61.7) are set to “0”.

© Siemens AG, 2003. All rights reserved

Position measuring system 1 (DB31, ... DBX1.5)
Position measuring system 2 (DB31, ... DBX1.6)

Up to 2 measuring systems can be connected per axis/spindle. The active measuring system (see Fig. 2-4) is always used for position control, absolute value calculation and position display.

By switching interface signals “Position measuring system 1/2” and “Servo enable” (DB31, ... DBX2.1) appropriately, the following can be achieved:

- Activation of position measuring system 1 or 2
- Switchover between position measuring systems 1 and 2
- Parking position; i.e. both measuring systems are inactive and IS “Servo enable” is set to “0” signal.
- Spindle without position measuring system (speedcontrolled spindle); i.e. both measuring systems are inactive and IS “Servo enable” is “1” signal.

The following table shows the switching possibilities for the interface signals.

Meaning	IS “PMS 1” (DB31, ... DBX1.5)	IS “PMS 2” (DB31, ... DBX1.6)	IS “Servo enable” (DB31, ... DBX2.1)
Position measuring system 1 active	1	0 (or 1)	1
Position measuring system 2 active	0	1	1
Parking position	0	0	0
Spindle without position measuring system (speed-controlled)	0	0	1
Switchover from PMS1 to PMS2	1 ⇒ 0	0 ⇒ 1	1
Switchover from PMS2 to PMS1	0 ⇒ 1	1 ⇒ 0	1

Application

With some axes it is practical to connect a direct position measuring system (directly on the machining table) and an indirect measuring system (at the drive). The direct measuring system is preferred for extremely precise machining, otherwise the indirect measuring system is usually used. The measuring system is selected by the settings made in interface signals “Position measuring system 1” and “Position measuring system 2”.

Switch over measuring system

See Subsection 5.4.1

Note

References: /FB/, G2, “Velocities, Setpoint/Actual Value Systems, Closed-Loop Control”

© Siemens AG, 2003. All rights reserved

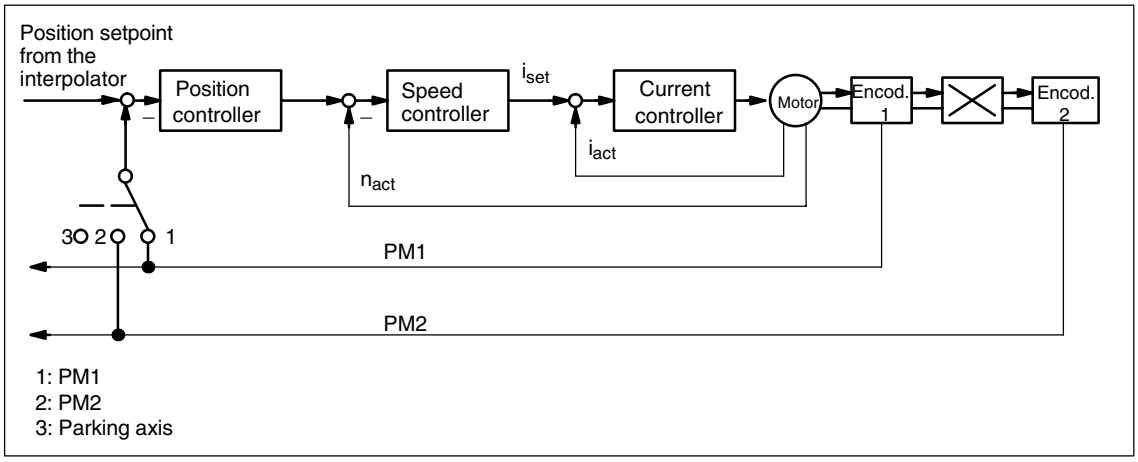


Fig. 2-4 Position measuring system 1 and 2

Servo enable (DB31, ... DBX2.1)

When the servo enable is activated for the drive, the position control loop of the axis/spindle is closed. The axis/spindle is then subject to position control.

When servo enable is removed the position control loop and with a delay the speed control loop of the axis/spindle are opened.

Activation methods

The servo enable for the drive can be set and removed from the following places:

1. From the PLC user program with interface signal “Servo enable” (in normal cases)

   Application: Removal of servo enable before clamping an axis/spindle.

2. Servo enable is canceled automatically by the control when certain faults occur in the machine, the drive, the position measuring system or the control (when faults occur)

   Application: The traversing axes must be brought to a standstill by a rapid stop due to a fault.

   For details of which faults (alarms) cause internal cancellation of servo enable, please refer to:
   References: /DA/, “Diagnostics Guide”

3. By the control if the following events occur:

   - “EMERGENCY OFF” (DB10, DBX56.1) is active at the PLC interface.
   - The pulse enable for the drive has been canceled (e.g. with IS “Pulse enable” (DB31, ... DBX21.7).
   - The drive is not ready (IS “Drive Ready” (DB31, ... DBX93.5) = 0).

© Siemens AG, 2003. All rights reserved

Cancellation of servo enable

The control-internal sequence of operations if the servo enable is canceled depends on whether the axis is moving or stationary at the time.

Possible procedures are described below (see Fig. 2-5).

... when axis/spindle is stationary

Sequence of operations if the servo enable is canceled for a stationary axis/spindle:

- Axis position control loop opened (Fig. 2-5)

⇒ IS “Position controller active” (DB31, ... DBX61.5) is set to 0 (checkback signal)

- The axis/spindle is switched to the “Hold” or “Follow-up” state depending on IS “Follow-up mode” (DB31, ... DBX1.4).

... when axis/spindle is in motion

Sequence of operations if the servo enable is canceled for a moving axis/spindle:

- The axis/spindle is braked down to zero speed according to the setting in MD 36610: AX_EMERGENCY_STOP_TIME (duration of the braking ramp in error states) with rapid stop. Then alarm 21612 “Servo enable reset during movement” is output (see Fig. 2-5).
- The position control loop of the axis/spindle is opened. Checkback signal to PLC with IS “Position controller active” (DB31, ... DBX61.5) = 0 state. The timer for the servo enable delay time (MD 36620: SERVO_DISABLE_DELAY_TIME (cutout delay servo enable)) is also triggered.
- As soon as the actual speed has reached the zero speed range, the drive servo enable is removed. Checkback signal to PLC with IS “Speed controller active” (DB31, ... DBX61.5) = 0 signal. The servo enable for the drive is canceled at the latest after the time set in MD 36620: SERVO_DISABLE_DELAY_TIME has elapsed.

  Caution: If the cutout delay is set too small, servo enable is removed even though the axis/spindle is still moving. The axis/spindle is then stopped abruptly with setpoint 0.
- The position actual value of the axis/spindle continues to be acquired by the control.
- At the end of the braking process, the axis/spindle is switched to “Follow-up mode” (regardless of the state of IS “Follow-up mode” (DB31, ... DBX1.4). Zero speed monitoring and clamping monitoring therefore have no effect.

This axis/spindle state cannot be changed until after “Reset”. With “Reset” the axis/spindle (if servo enable has not yet been activated) is brought into the “Hold” or “Follow-up” state depending on the setting in IS “Follow-up mode”.

© Siemens AG, 2003. All rights reserved

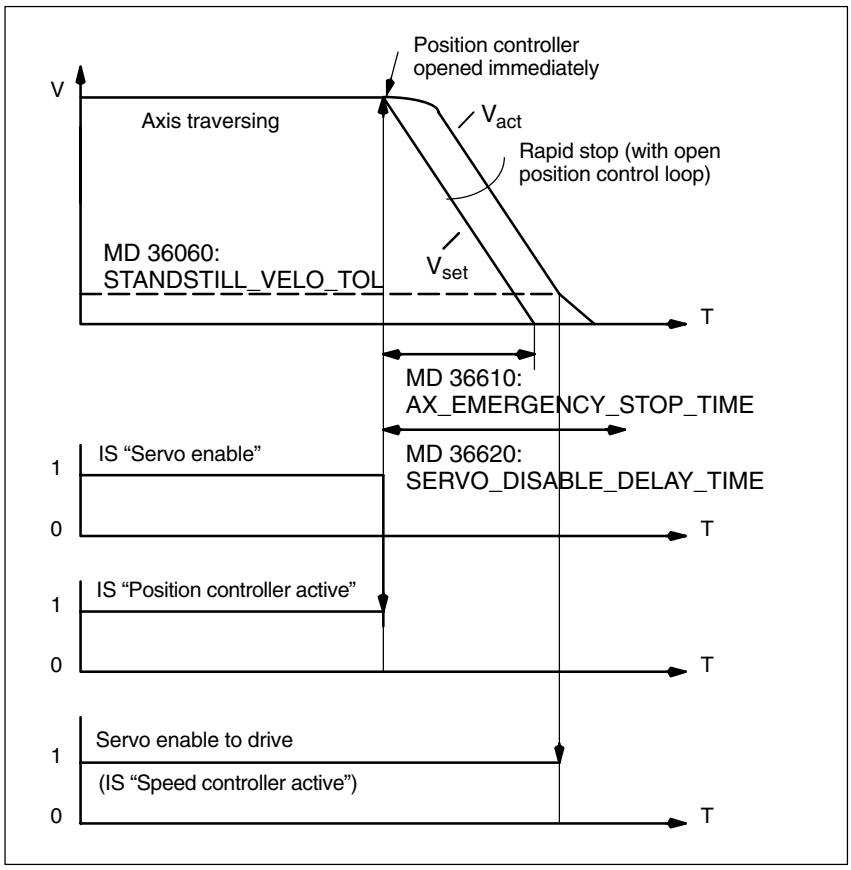


Fig. 2-5 “Remove servo enable” from interface when axis/spindle is moving

Interpolatory axis grouping

All the axes traversing within the interpolatory axis grouping are stopped as soon as the servo enable signal is canceled for one of the axes.

The axes are brought to a standstill as described above. All axes in the geometry grouping are brought to a standstill with rapid stop. Alarm 21612 “Servo enable reset during movement” is also triggered. Continued processing of the NC program after this event is no longer possible.

Connection with IS “Follow-up mode”

The positioning response of the axis/spindle when IS Servo enable (DB31, ... DBX2.1) is determined by the state of IS “Follow-up mode active”. In IS “Follow-up mode active” (DB31, ... DBX61.3) = 1 the standstill and terminal monitoring functions are inactive.

Actual point synchronization

After servo enable has been issued, renewed actual value synchronization of the axis (reference point approach) is **not** necessary if the maximum permissible limit frequency of the axis measuring system has not been exceeded during this time.

© Siemens AG, 2003. All rights reserved

Delete distance-to-go/ spindle reset (axis-specific/ spindle-specific) (DB31, ... DBX2.2)

The IS “Delete distance-to-go (axis-specific)” is active in the AUTOMATIC and MDA modes only for those axes that are not assigned to a geometry grouping (positioning axes).

The axis is brought to a standstill with ramp stop with the rising edge of the interface signals and its distance-to-go deleted. The next program block is then started.

For more information please refer to Subsection 5.4.1.

Spindle reset

References: /FB/, S1, “Spindles”

Controller parameter selection DB31, ... DBX9.0, 1, 2

The PLC user program stores a binary coded NC for a controller parameter set in bits 0 – 2. This is activated for the axis by the NC.

Up to six parameter sets are available. Each parameter set includes the following machine data:

MD number	MD identifier
36200	$MA_AX_VELO_LIMIT
32200	$MA_POSCTRL_GAIN
32800	$MA_EQUIV_CURRCTRL_TIME
32810	$MA_EQUIV_SPEEDCTRL_TIME
32910	$MA_DYN_MATCH_TIME
31050	$MA_DRIVE_AX_RATIO_DENOM
31060	$MA_DRIVE_AX_RATIO_NUMERA

Parameter sets can be assigned by:

- Parts programs
- Operator actions.

The assignment is as follows:

Parameter set	Index or binary value for selection
1	0
2	1
3	2
4	3
5	4
6	5 ... 7

Examples of controller parameter set switchover can be found in Chapter 6.

The NC switches parameters only if switchover is basically activated by axis-specific MD 35590: PARAMSET_CHANGE_ENABLE = 1 or = 2.

© Siemens AG, 2003. All rights reserved

Restriction

An MD 35590 set for spindles: PARAMSET_CHANGE_ENABLE = 1 has no effect.

Priority

For MD 35590: PARAMSET_CHANGE_ENABLE = 1
During the operations:

- Rigid tapping
- Thread cutting

the parameter sets used internally by the NC are not switched over as a result of a request for another parameter set from the PLC.

Please note that for MD 35590: PARAMSET_CHANGE_ENABLE = 2:

- No restrictions, no priority.

Note

For further information on the parameter sets, please refer to:

References: /FB1/, S1, “Spindles”, under “Spindle modes, axis mode” and “Gear stage change”

Switchover request while axis is moving

The response to the switchover depends on the modification to the servo gain MD 32200: $MA_POSCTRL_GAIN between old and new parameter set:

- Identical K_v factors or no position control:
  The NC responds immediately to the PLC’s switchover request. The parameter set is also changed during the movement.
- Unequal K_v factors and position control active:
  To achieve a smooth switchover, the switchover waits while the velocity of the axis or the path is reduced until the “Axis/spindle stationary” signal is detected. The switchover velocity value can be adapted in $MD_STANDSTILL_VELO_TOL (MD 36060) according to the application.

Note

Should additional measures be required for a specific application, then these must be implemented in the PLC program to suit the application concerned.

Switchover request information to PLC

The PLC application program can receive a switchover request directly from a parts program in the form of an auxiliary function output to the PLC. Auxiliary function outputs are described in:

References: /FB/, H2, “Output of Auxiliary Functions”

© Siemens AG, 2003. All rights reserved

The same auxiliary function value for the controller parameter switchover must be set in both the parts program and the user PLC program.

Note

When the controller parameter set switchover function is implemented using an auxiliary function output, the parts program is processed on the basis of auxiliary function value interpretation and processing by an associated PLC program.

Disable parameter set switchover commands from NC ... DBX9.3

The NC must not initiate any parameter set switchover when the interface bit is set, irrespective of which parameter set number is stored in interface bits DBX9.0 to DBX9.2.

2.8.2 Interface signals to from axis/spindle

Note

For further information about the interface signals listed below, please refer to Subsection 5.4.2.

Drive test travel request (DB31, ... DBX61.0)

If all traversing conditions are met by the relevant axes, the NC sends the following ready to travel signal to the PLC:

IS “Drive test travel request” (DB31, ... DBX61.0) = 1

Follow-up mode active (DB31, ... DBX61.3)

The axis/spindle is operating in follow-up mode. The setpoint position is continuously corrected to the actual value position.
The clamping and zero-speed monitoring functions are not active.

Axis/spindle stationary (DB31, ... DBX61.4)

The “Axis/spindle stationary” signal is enabled by the controller if
– no more new setpoints are output by the NC **and**
– the NC detects that the axis is within the defined velocity tolerance.

This tolerance is set in MD 36060: STANDSTILL_VELO_TOL (maximum velocity/speed for signal “Axis/spindle stationary”).

Position controller active (DB31, ... DBX61.5)

The position control loop for the axis/spindle is closed; the position control function is active.

© Siemens AG, 2003. All rights reserved

Speed control loop active (DB31, ... DBX61.6)

The speed control loop for the axis/spindle is closed; the speed control function is active.

Current control loop active (DB31, ... DBX61.7)

The current control loop for the axis/spindle is closed; the current control function is active.

Controller parameter set checkback signal DB31, ... DBX69.0, 1, 2

The signal corresponds to the request to switch over the parameter set (DB31, ... DBX9.0, 1, 2). It indicates a binary coded index of the parameter set which has been selected on the machine axis.

Parameter set	Index or binary value at interface
1	0
2	1
3	2
4	3
5	4
6	5

Successful switch-over is confirmed when the checkback signal contains the same value as the request signal. The index of the active controller parameter set is always specified.

Lubrication pulse (DB31, ... DBX76.0)

The signal state is 0 (FALSE) after POWER ON/Reset of the control.
The IS “Lubrication pulse” is **inverted** by the NCK as soon as the axis/spindle has traveled a greater distance than that set in MD 33050: LUBRICATION_DIST (travel path for lubrication from PLC).
Each 0/1 or 1/0 change indicates that the distance has been traversed again.

© Siemens AG, 2003. All rights reserved

2.9 Interface signals for digital drives (DB31, ...)

Note

For further information about the interface signals listed below, please refer to Subsection 5.4.1.

2.9.1 Signals to axis/spindle

Ramp function generator rapid stop (DB31, ... DBX20.1)

The PLC user program requests a rapid stop for the drive. The drive is then stopped without a ramp function (with speed setpoint 0). Servo enable is maintained.

Not on 810D

Torque limit 2 (DB31, ... DBX20.2)

The PLC user program requests torque limit 2 for the axis/spindle. The limit value is set using drive parameters.

Not on 810D

Speed setpoint smoothing (DB31, ... DBX20.3)

The PLC user program requests a speed setpoint smoothing filter for the axis/spindle. The smoothing is activated in the drive module only under certain conditions.

Not on 810D

Drive parameter set selection A, B, C (DB31, ... DBX21.0, 1, 2)

Using bit combination A, B, C, the PLC user program can select up to 8 different drive parameter sets on the SIMODRIVE 611D.

Motor selection A, B (DB31, ... DBX21.3, 4)

The PLC user program can select between 4 motor types or motor operating modes for an MSD drive module. In SW 6.3 and lower, it is possible to select star connection mode (A = 0, B = 0) and delta connection mode (A = 1, B = 0) for the main spindle drive.

Not on 810D

As of SW 6.4, it is permissible to combine (A = 0, B = 1) and (A =1, B = 1) on conjunction with the 611D Performance2 controller module.

All of the four specified A/B combinations may be used on 611U drives.

© Siemens AG, 2003. All rights reserved

Motor selection (DB31, ... DBX21.5)

The PLC user programs sends this signal to the drive to indicate successful motor selection. For example, in the case of star/delta switchover on the SIMODRIVE 611D or 611 universal, a message or signal must be provided when the motor contactor has switched. The drive then enables the pulses.

Not on 810D

Speed controller integrator disabled (DB31, ... DBX21.6)

The PLC user program inhibits the integrator of the speed controller for the drive. The speed controller is thus switched from PI to P controller.

Not on 810D

Pulse enable (DB31, ... DBX21.7)

The PLC user program enables the pulses for the axis/spindle. However, the pulse enable is only activated for the drive module if all the enable signals are present.

2.9.2 Signals from axis/spindle

Note

For further information about the interface signals listed below, please refer to Subsection 5.4.2.

Setup mode active (DB31, ... DBX92.0)

Setup mode is activated for the drive. Setup mode is selected via the terminals on the infeed/regenerative feedback module.

Ramp function generator rapid stop active (DB31, ... DBX92.1)

The drive signals back to the PLC that ramp function generator rapid stop is active. The drive is thus brought to a standstill without the ramp function (with speed setpoint 0).

Not on 810D

Torque limit 2 active (DB31, ... DBX92.2)

The drive signals back to the PLC that torque limit 2 is active for the axis/spindle. The torque limit value is defined using drive parameters.

Not on 810D

Speed setpoint smoothing active (DB31, ... DBX92.3)

The PLC user program requests a speed setpoint smoothing filter for the axis/spindle. The smoothing is activated in the drive module only under certain conditions.

Not on 810D

© Siemens AG, 2003. All rights reserved

Active drive parameter set A, B, C (DB31, ... DBX93.0, 1, 2)

The drive module sends this checkback to the PLC to indicate which drive parameter set is currently active. With bit combination A, B, C, eight different drive parameter sets can be selected by the PLC for the SIMODRIVE 611D.

Active motor A, B (DB31, ... DBX93.3, 4)

The drive module (MSD) sends this checkback to the PLC to indicate which of the 4 motor types or motor operating modes is active. In SW 6.3 and lower, it is possible to select star connection mode (A = 0, B = 0) and delta connection mode (A = 1, B = 0) for the main spindle drive. In SW 6.4 and higher, it is permissible to combine A = 0, B = 1 and A =1, B = 1 on conjunction with the 611D Performance2 controller module.

DRIVE ready (DB31, ... DBX93.5)

Checkback signal indicating that the drive is ready. The conditions required for traversing the axis/spindle are fulfilled.

Speed controller integration disabled (DB31, ... DBX93.6)

The speed controller integrator is disabled. The speed controller has thus been switched from PI to P controller.

Not on 810D

Impulse is enabled (DB31, ... DBX93.7)

Drive pulse enable for drive module available. The axis/spindle can now be traversed.

Motor temperature warning (DB31, ... DBX94.0)

The drive module signals to the PLC that the motor temperature has exceeded the warning threshold. If the motor temperature remains at this level the drive will be stopped after a defined time (drive MD) and the pulse enable removed.

Heat-sink temperature prewarning (DB31, ... DBX94.1)

The drive modules signals to the PLC that the heat-sink temperature has exceeded the warning threshold. The pulse enable will be removed for the drive module in question after 20 seconds.

Ramp-up function complete (DB31, ... DBX94.2)

This signal confirms that the speed actual value has reached the new setpoint allowing for the tolerance band set in drive MD 1426: SPEED_DES_EQ_ACT_TOL. The acceleration procedure is thus completed. Any subsequent speed fluctuations due to load changes will not affect the interface signal.

$|M_d| < M_{dx}$ (DB31, ... DBX94.3)

This signal indicates that the actual torque $|M_d|$ is lower than the threshold torque M_{dx} set in drive MD 1428: TORQUE_THRESHOLD_X. The threshold torque is entered as a percentage of the current speeddependent torque limitation.

© Siemens AG, 2003. All rights reserved

$|n_{act}| < n_{min}$ (DB31, ... DBX94.4)

This signal indicates that the actual speed $|n_{act}|$ is lower than the set minimum speed n_{min} (drive MD 1418: SPEED_THRESHOLD_MIN).

$|n_{act}| < n_x$ (DB31, ... DBX94.5)

This signal indicates that the actual speed $|n_{act}|$ is lower than the set threshold speed n_x (drive MD 1417: SPEED_THRESHOLD_X).

$n_{act} = n_{set}$ (DB31, ... DBX94.6)

The PLC receives this signal confirming that the actual speed n_{act} has reached the new setpoint allowing for the tolerance band set in drive MD 1426: SPEED_DES_EQ_ACT_TOL and continues to remain within the tolerance band.

Variable signal function (DB31, ... DBX94.7)

The variable signal function allows the SIMODRIVE 611D to monitor any quantity in any axis for violation of a programmable threshold and to send an appropriate checkback interface signal to the PLC.
The quantity to be monitored is set by means of 611D machine data.

Not on 810D

U_{DClink} < warning threshold (DB31, ... DBX95.0)

The drive signals to the PLC that the DC link voltage V_{DClink} has dropped below the lower DC link voltage threshold (drive MD 1604: LINK_VOLTAGE_WARN_LIMIT).

2.10 Screen settings

Contrast, monitor type, foreground language and display resolution to take effect after system start-up can be set in the operator panel machine data.

Contrast

MMC MD 9000: LCD_CONTRAST can be set to adjust the contrast (brightness) after system power-up for slimline operator panels with a monochrome LCD display.

The contrast can be varied in 16 steps (0 to 15).

Monitor type

The monitor type must be specified with MMC MD 9001: DISPLAY_TYPE to ensure optimum color adaptation.

Foreground language

On the SINUMERIK 840D/810D and FMNC systems, two languages are available simultaneously and can be set alternately in online mode by means of soft key selection.

MMC MD 9003: FIRST_LANGUAGE defines which language is selected after each system start-up.

© Siemens AG, 2003. All rights reserved

Display resolution

MMC MD 9004: DISPLAY_RESOLUTION and MD 9010: SPIND_DISPLAY_RESOLUTION (for spindles) are set to define the number of places after the decimal point (number of digits after the decimal point between 0 and 5) for the position display.

Three decimal places are displayed per default ⇒ display resolution = 10^{-3} [mm] or [degrees].

REFRESH suppression

In some parts programs, the main run has to wait until the preprocessor prepares new blocks. In the NC, the preprocessor and display refresh compete for computer time. With MD 10131: SUPPRESS_SCREEN_REFRESH, the user can modify the type of refresh suppression in favor of the preprocessor:

Value 0:	Refreshing of values is suppressed **in all channels**
Value 1:	Refreshing of values is suppressed **in all time-critical** channels.
Value 2:	Refreshing of values is **never** suppressed.

© Siemens AG, 2003. All rights reserved

2.11 General functions

2.11.1 Settings for involute interpolation (SW 6 and higher)

Introduction

The involute of a circle is a curve that is traced by the end point of a taut thread unwound from another circle. Involute interpolation allows path curves to be traced along an involute.

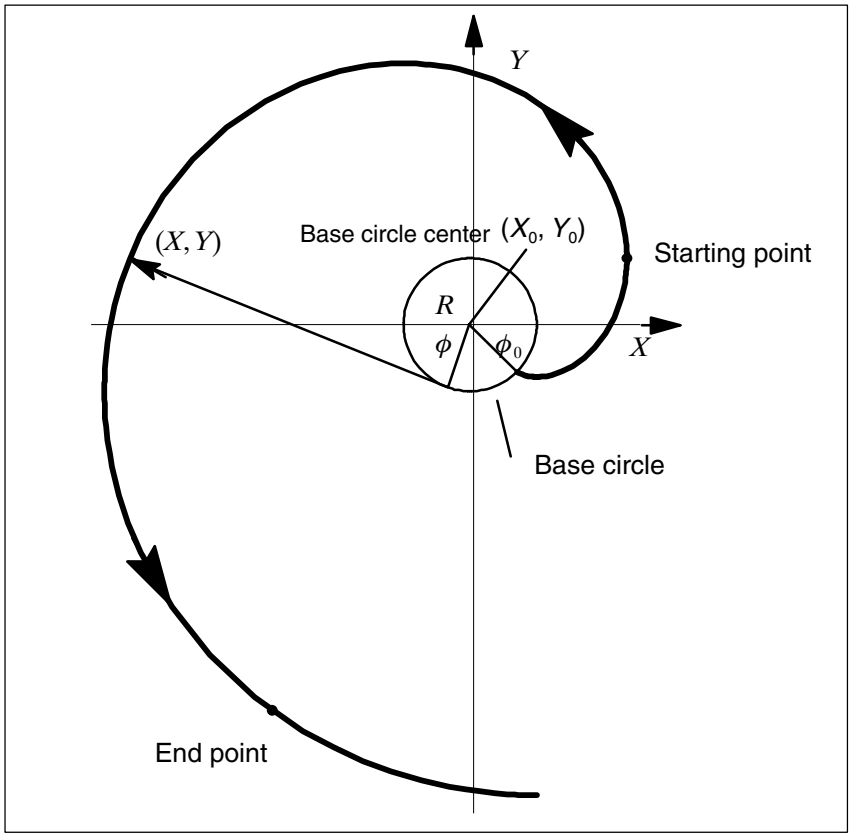


Fig. 2-6 Involute (unwound from base circle)

Programming

A general description of how to program involute interpolation can be found in:

References: /PG/, Programming Guide, Fundamentals, Section 4.7

In addition to the programmed parameters, machine data are relevant in two instances of involute interpolation; these data may need to be set by the machine manufacturer/end user.

© Siemens AG, 2003. All rights reserved

Accuracy

If the programmed end point does not lie exactly on the involute defined by the starting point, interpolation takes place between the two involutes defined by the starting and end points (see illustration below). The maximum deviation of the end point is specified by the setting in MD 21015: INVOLUTE_RADIUS_DELTA. If the deviation of the programmed end point in the radial direction is greater than the value set in this MD, alarm 14038 is generated and program execution aborted.

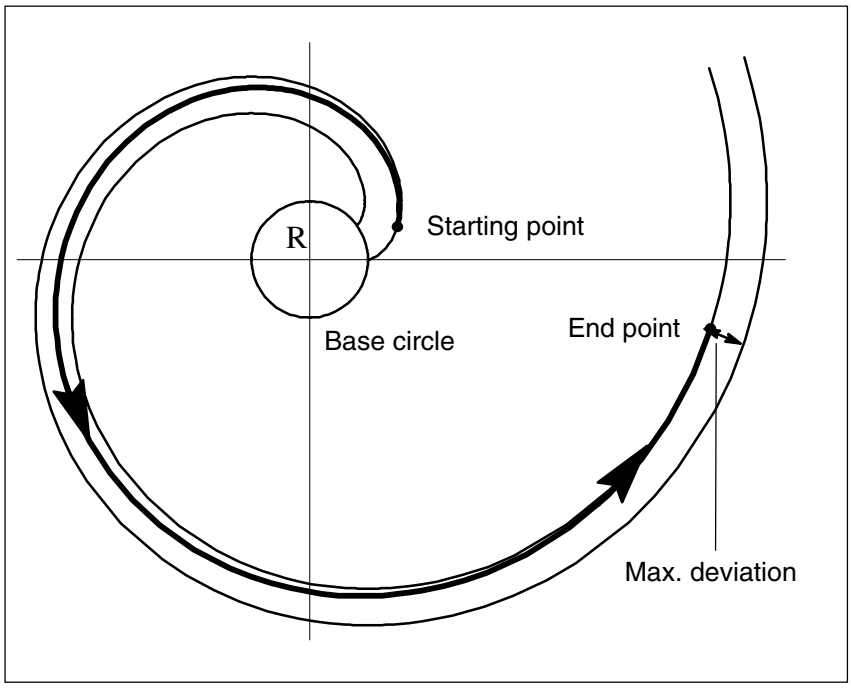


Fig. 2-7 MD 21015 specifies the max. permissible deviation

Limit angle

If AR is used to program an involute leading to the base circle with an angle of rotation that is greater than the maximum possible value, an alarm is output and program execution aborted.

© Siemens AG, 2003. All rights reserved

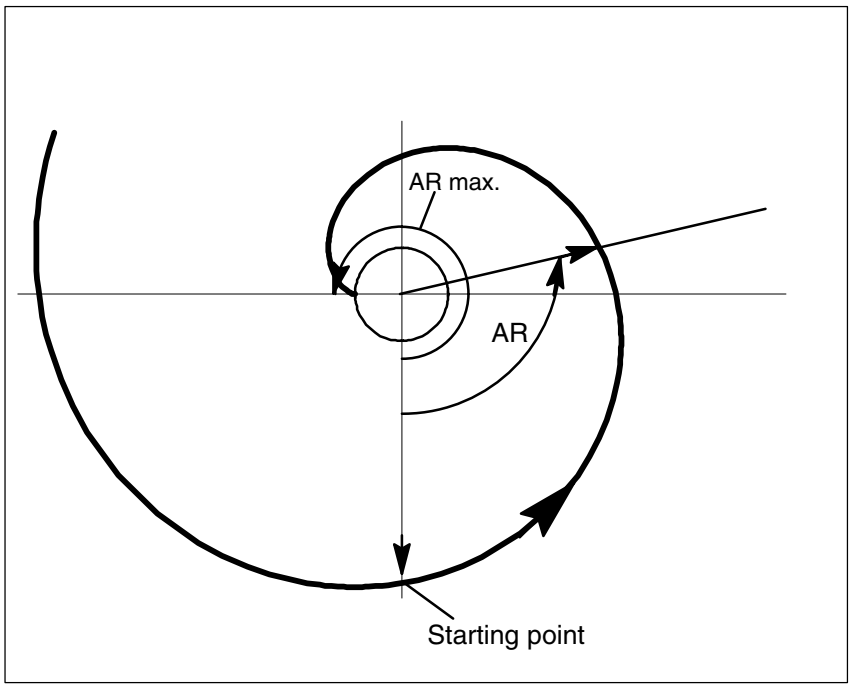


Fig. 2-8 Limited angle of rotation towards base circle

Output of alarm 14034 can be suppressed by setting MD 21016: INVOLUTE_AUTO_ANGLE_LIMIT = TRUE (default setting is FALSE).

The programmed angle of rotation is then also limited automatically and the interpolated path ends at the point at which the involute meets the base circle. This, for example, makes it easier to program an involute which starts at a point outside the base circle and ends directly on it.

Tool radius compensation

2 1/2 D tool radius compensation is the only tool radius compensation function permitted for involutes. If the 3D compensation function is active (both circumferential and face milling) and an involute is programmed, the machining operation is aborted with alarm 10782.

With 2 1/2 D tool radius compensation, the plane of the involute must lie in the compensation plane or else alarm 10781 will be generated. It is however permissible to program an additional helical component for an involute in the compensation plane.

Dynamic performance

Involutes that begin or end on the base circle have an infinite curvature at this point. To ensure that the velocity is adequately limited at this point when tool compensation is active, without reducing it too far at other points, the “Velocity limitation profiles function” must be activated, i.e. machine data MD 28530: MM_PATH_VELO_SEGMENTS must be set to a value higher than 1. A setting of 5 is recommended. This setting need not be made if only involute sections are used which have radii of curvature that change over a relatively small area.

© Siemens AG, 2003. All rights reserved

2.11.2 Activate DEFAULT memory

GUD start values

Using NC commands DEF... / REDEF... you can assign default settings to global user data (GUD). These default settings must be permanently stored in the system if they are to be available after certain system states (e.g. RESET).

The memory space needed is part of the memory area parameterized via MD 18150: MM_GUD_VALUES_MEM. The setting for activating the stored default values is made in MD 11270: DEFAULT_VALUES_MEM_MASK.

References: 2/FB/S7 Memory configuration
/PGA/, Chapter 3 Programming Guide, Advanced

© Siemens AG, 2003. All rights reserved

2.12 Reading and writing PLC variables (SW 4 and higher)

High-speed data channel

For highspeed exchange of information between the PLC and NC, a memory area is reserved in the interface memory of these modules (dualport RAM). Variables of any type (I/O, DB, DW, flags) may be exchanged within this memory area.

The PLC accesses this memory using 'Function Calls' (FC) while the NCK uses '$ variables'.

Organization of memory area

The user's programming engineer (NCK and PLC) is responsible for organizing (structuring) this memory area. Every storage position in the memory can be addressed provided that the limit is selected according to the appropriate data format (i.e. a DWORD for a 4-byte limit, a WORD for a 2-byte limit, etc.).

The memory is accessed via the data type and the position offset within the memory area.

Access from NC

To allow the NC to access PLC variables (from a parts program) quickly, $ variables are provided in the NCK. The PLC uses a function call (FC) to read and write $ variables. Data are transferred to and from the NCK immediately.

$ variables can be accessed (by the NCK) during preprocessing and in synchronized actions.

Data type information is determined by the $ variable data type, the position index is specified as an array index (in bytes).

The following '$ variables' are available:

```
$A_DBB      // Data byte (8 bits)
$A_DBW      // Data word (16 bits)
$A_DBD      // Data double word (32 bits)
$A_DBR      // Real data (32 bits)
```

Ranges of values

```
$A_DBB(n) 0 <= x <= 255
$A_DBW(n) –32768 <= x <= 3276
$A_DBD(n)  –2147483648 <= x <= 2147483647
```

Access from PLC

The PLC uses 'Function Calls' (FC) to access the memory. These FCs ensure that data are read and written in the DPR immediately, i.e. not just at the beginning of the PLC cycle. FCs receive data type information and the position offset as parameters.

© Siemens AG, 2003. All rights reserved

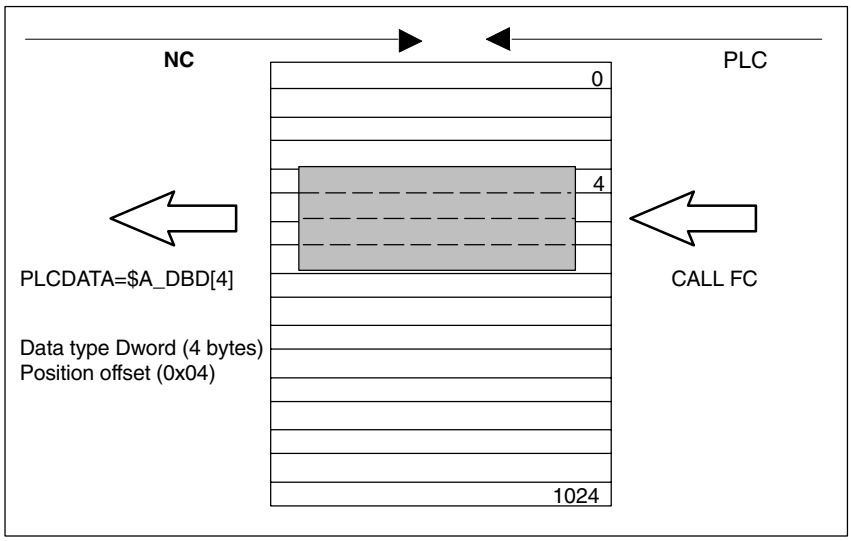


Fig. 2-9 Communications buffer (DPR) for NCKPLC data exchange

Supplementary conditions

1. The user's programming engineer (NCK and PLC) is responsible for organizing the DPR memory area. No checks are made for inconsistencies in the configuration.
2. A total of 1024 bytes are available in the input and output directions.
3. Single-bit operations are not supported and must be linked back to byte operations by the user (programming engineer).
4. Since the contents of variables are manipulated directly in the communications buffer, the user must remember that intermediate changes in values occur as a result of multiple access operations where a variable is evaluated several times or when variables are linked (i.e. it may be necessary to store values temporarily in local variables or R parameters or to set up a semaphore).
5. The user's programming engineer is responsible for coordinating access operations to the communications buffer from different channels.
6. Data consistency can be guaranteed only for access operations up to 16 bits (byte and word). The user's programming engineer is responsible for ensuring consistent transmission of 32-bit variables (double and real). A simple semaphore mechanism is provided in the PLC for this purpose.
7. The PLC stores data in 'Little Endian' format in the DPR.
8. Values transferred with $A_DBR are subject to a data conversion and hence to a loss of accuracy. The data format for floating-point numbers is DOUBLE (64 bits) on the NCK, but only FLOAT (32 bits) on the PLC. The format used for storage in the dual-port RAM is FLOAT. Conversion takes place respectively before/after storage in the dual-port RAM.

   If a read/write access is made from the NCK to a variable in the dualport RAM, the conversion is performed twice. It is impossible to prevent differences between read and written values because the data are stored in both formats.

© Siemens AG, 2003. All rights reserved

Possible remedy for the problem (example):
Compare with 'EPSILON' (small deviation)

```
N10   DEF REAL DBR
N12   DEF REAL EPSILON = 0.00001
N20   $A_DBR[0]=145.145
N30   G4 F2
N40   STOPRE
N50   DBR=$A_DBR[0]
N60   IF (ABS(DBR/145.145–1.0) < EPSILON) GOTOF ENDE
N70   MSG ("Error")
N80   M0
N90   ENDE:
N99   M30
```

Activation

The maximum number of output variables that can be written simultaneously (also applicable to block search) is set via memory-configuring machine data MD 28150 MM_NUM_VDIVAR_ELEMENTS. The default for this machine data is '0'.

Example

A WORD is to be transferred from the PLC to the NC.
The position offset within the NCK input (PLC output area) must be the 4th byte.
The position offset must be an integer multiple of the data width.

- Write from the PLC:

```
....
CALL FC21 (
Enable :=M10.0,      ;If TRUE, then FC21 active
Funct :=B#16#4,
S7Var :=P#M 104.0 WORD1,
IVAR1 :=04,
IVAR2 :=–1,
Error :=M10.1,
ErrCode :=MW12);
....
)
```

- Read in parts program:

```
...
PLCDATA = $A_DBW[4];                    // Read a word
...
```

Response to POWER ON, Block search

The DPR communications buffer is initialized during 'POWER ON'.

During a block search, the PLC variable outputs are collected and transferred to the DPR communications buffer with the approach block (analogous to writing of analog and digital outputs).

Other status transitions have no effect in this respect.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Supplementary Conditions 3

Supplementary conditions 840Di with drive 611U

Various interface signals (signals from/to axis/spindle) are not available when using the 840Di control system with drive 611U. The validity of the interface signals is indicated explicitly for each signal.

Overview of inapplicable interface signals

For SINUMERIK 840Di, various interface signals from/to axis/spindle do not apply. All interface signals that do **not apply** are listed in the table below.

Note

SINUMERIK 840Di in conjunction with SIMODRIVE 611 universal does not have the functionality Safety Integrated. All Safety Integrated interface signals thus do not apply.

Inapplicable interface signals

Axis/spindlespecific		Signals from PLC to axis/spindle	
DB number	**Byte, bit**	**Name**	**Refer to Doc.**
31, ...	20.0	Acceleration switch V/Hz operation	
31, ...	20.2	Torque limit 2	
Safety Integrated signals from PLC to axis/spindle			
31, ...	22.0	Deselection of safe velocity and zero speed (deselection of SBH/SG)	
31, ...	22.1	Deselection of safe operational stop (deselection of SBH)	
31, ...	22.3	Velocity limit, bit value 0 (SG selection)	
31, ...	22.4	Velocity limit, bit value 1	
31, ...	23.0–23.2	Ration selection, bit value 0 to bit value 2	
31, ...	23.5	Enable limit switch pair 2	
31, ...	23.7	Enable test stop	
Signals from axis/spindle to PLC			
31, ...	92.0	Setup mode active	
31, ...	92.2	Torque limit 2 active	
Safety Integrated signals from axis/spindle to PLC			
31, ...	108.0	Safe velocity or zero speed (SBH/SG active)	
31, ...	108.2	Clear status pulses	
31, ...	108.7	Axis referenced safely	
31, ...	109.0–109.7	Cam signals of plus and minus cams (SN1+/1– to SN4+/4–)	
31, ...	110.1	Safe operational stop active (SBH active)	

© Siemens AG, 2003. All rights reserved

DB number	Byte, bit	Name	Refer to Doc.
31, ...	110.3–110.4	Safe velocity active, bit value 0 to bit value 1	
31, ...	110.5	$n < n_x$	
31, ...	111.1	Safe operational stop active (SBH active)	
31, ...	111.4–11.7	Stop A/B to Stop E active	

■

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

4.1 Machine data for operator panel

Note

As of SW 6.1: (MMC 100 is the same as HMI Embedded) and
(MMC 103 is the same as HMI Advanced)

9000 MD number	**LCD_CONTRAST** Contrast		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 15	
Changes effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The contrast of the slimline operator panel with **monochrome** (monitor type 0) LCD which is to take effect after system start-up can be set in this MD. 16 contrast (brightness) settings are available. Setting | Effect 15 : Display brighter ⇑ 7 : Default setting ⇓ 0 : Display darker The contrast can also be altered with the soft key “LCD dark/bright” in the Diagnosis area. However, after POWER ON the value set in MD: LCD_CONTRAST is active.		
MD irrelevant for	Slimline operator panel with color LCD (MD 9001: DISPLAY_TYPE = <> 0)		
Related to	MD 9001: DISPLAY_TYPE (monitor type)		
References	/BA/, “Operator’s Guide”		

9001 MD number	**DISPLAY_TYPE** Monitor type	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2
Changes effective after POWER ON	Protection level: 3/4	Unit: –
Data type: BYTE	Applies from SW 1.1	

© Siemens AG, 2003. All rights reserved

9001 MD number	**DISPLAY_TYPE** Monitor type
Significance:	For optimum color adaptation the monitor type must be specified The following assignment applies: Monitor — Type Slimline op. panel OP031 LCD display monochr. — 0 Slimline operator panel OP031 LCD display color — 1 OP032 color monitor — 2
Related to	

© Siemens AG, 2003. All rights reserved

9002 MD number	**DISPLAY_MODE** (MMC 100 only) External monitor		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1		
Significance:	The external monitor type connected to the MMC must be specified to allow optimum color adjustment. The following assignment applies: Monitor: Type No monitor connected: 0 Monochrome monitor: 1 Color monitor: 2		
Related to			

9003 MD number	**FIRST_LANGUAGE** (MMC 100 only) Foreground language		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	With SINUMERIK 840D/840Di/810D and FMNC 2 languages are available simultaneously. The language which is to be automatically active after each system start-up is set in the machine data. It is possible to temporarily switch to a second language using a soft key in the Diagnosis area. After POWER ON, however, the language set in MD 9003: FIRST_LANGUAGE is again active.		
References	/BA/, “Operator’s Guide”, IM3 Installation and Start-up MMC 103		

9004 MD number	**DISPLAY_RESOLUTION** Display resolution		
Default setting: 3	Minimum input limit: 0	Maximum input limit: 5	
Changes effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The number of places after the decimal point for the position display is defined in this MD. The position display consists of max. 12 characters including sign and decimal point. The number of digits after the decimal point can be set to between 0 and 5. Three decimal places are displayed per default ⇒ Display resolution = 10^{-3} [mm] or [degrees]. **Note:** (SW 6.1 and higher) This functionality is available with HMI Embedded and HMI Advanced.		
Related to	The display resolution must be set to correspond meaningfully with the computational resolution (MD 10200: INT_INCR_PER_MM or MD: INT_INCR_PER_DEG).		

9005 MD number	**PRG_DEFAULT_DIR** (for MMC100) Basic setting for Program directory		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 5	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	This machine data defines the basic setting for the Program directory. **Note:** The Program directory basic setting can be selected with HMI Embedded only.		
Related to			

© Siemens AG, 2003. All rights reserved

9006 MD number	**DISPLAY_BLACK_TIME** (for MMC100) Time setting for screen saver		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 60	
Changes effective after POWER ON	Protection level: 3/4	Unit: minutes	
Data type: BYTE	Applies from SW 2.1		
Significance:	If no key is pressed on the keyboard for the duration defined in this machine data, the screen saver is automatically started. The screen saver is automatically deactivated if the value zero is set. **Note:** The automatic screen saver function is available only with HMI Embedded and is active only if IS “Screen dark” = “0”.		
Related to	IS “Screen dark” (DB19, ... DBX0.1)		

9007 MD number	**TABULATOR_SIZE** (for MMC100) Tabulator length		
Default setting: 4	Minimum input limit: 0	Maximum input limit: 30	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 2		
Significance:	This machine data defines the tabulator length. **Note:** The tabulator length can only be changed with HMI Embedded.		
Special cases, errors, ...			
Related to			

9008 MD number	**KEYBOARD_TYPE** (for MMC100) Keyboard type		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 3/4	Unit: minutes	
Data type: BYTE	Applies from SW 3.6		
Significance:	This machine data defines the keyboard type. Basic configuration for keyboard type 0: OP 1: MFII/QWERTY **Note:** (SW 6.1 and higher) This keyboard type can be set with HMI Embedded and HMI Advanced.		
Special cases, errors, ...			
Related to			

9009 MD number	**KEYBOARD_STATE** (for MMC100) Shift behavior of keyboard during booting		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 3.6		
Significance:	This machine data defines the shift behavior of the keyboard. Basic configuration for shift behavior of keyboard 0: Single, 1: Perm., 2: CAPSLOCK **Note:** (SW 6.1 and higher) The basic configuration for the shift behavior of the keyboard can be changed with HMI Embedded and HMI Advanced.		
Special cases, errors, ...			
Related to			

© Siemens AG, 2003. All rights reserved

9010 MD number	**SPIND_DISPLAY_RESOLUTION** Display resolution for spindle values		
Default setting: 3	Minimum input limit: 0	Maximum input limit: 5	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 4		
Significance:	The number of places after the decimal point for the position display is defined in the machine data display resolution for spindle values. The position display consists of max. 12 characters including sign and decimal point. The number of digits after the decimal point can be set to between 0 and 5. Three decimal places are displayed per default ⇒ Display resolution = 10^{-3} [mm] or [degrees]. **Note:** (SW 6.1 and higher) This functionality is available only with HMI Advanced.		
Related to	The display resolution must be set to correspond meaningfully with the computational resolution (MD 10200: INT_INCR_PER_MM or MD 10210: INT_INCR_PER_DEG).		

9011 MD number	**DISPLAY_RESOLUTION_INCH** Display resolution for INCH system of measurement		
Default setting: 3	Minimum input limit: 0	Maximum input limit: 5	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		
Significance:	The number of places after the decimal point for the position display is defined in the machine data display resolution for the INCH system of measurement. The position display consists of max. 12 characters including sign and decimal point. The number of digits after the decimal point can be set to between 0 and 6. Four decimal places are displayed per default ⇒ Display resolution = 10^{-4} [inch] . 3 decimal places are shown for rotary axes.		
Related to	The display resolution must be set to correspond meaningfully with the computational resolution (MD 10200: INT_INCR_PER_MM or MD 10210: INT_INCR_PER_DEG).		

9012 MD number	**ACTION_LOG_MODE** Set action mode for action log		
Default setting: 255	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2/2	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	This machine data can be parameterized to activate/deactivate the action log and to select the data to be logged. **Note:** (SW 6.1 and higher) This functionality can be set with HMI Embedded and HMI Advanced.		
References	/IAM/, “Operator’s Guide”, IM3 Installation and Start-up MMC 103		

9020 MD number	**TECHNOLOGY** Setting the computer mode (turning, milling)		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 2	
Change effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW	4.3 (MMC 100.2) 5.1 (MMC 103)	
Significance:	Basic configuration for simulation and free contour programming: 0: No specific assignment 1: Turning machine configuration 2: Milling machine configuration		
Related to			

© Siemens AG, 2003. All rights reserved

9030 MD number	**EXPONENT_LIMIT** (HMI Embedded only) Number of places to be displayed without exponent		
Default setting: 6	Minimum input limit: 0	Maximum input limit: 20	
Changes effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		
Significance:	This machine data defines the number of places to be displayed without an exponent.		

9031 MD number	**EXPONENT_SCIENCE** (HMI Embedded only) Technical exponent representation in three steps		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		
Significance:	This machine data defines the representation of exponents in three steps.		

9180–9181 MD number	**USER_CLASS_ ....** Protection level for access to tool carrier offsets...		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 6.1		
9180 MD number	**USER_CLASS_READ_TCARR** (HMI Embedded only) ... read general		
9181 MD number	**USER_CLASS_WRITE_TCARR** (HMI Embedded only) ... write general		

9200–9208 MD number	**USER_CLASS_ ....** Protection levels for access to ...		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1.1		
9200 MD number	**USER_CLASS_READ_TOA** ... read all tool offsets general		
9201 MD number	**USER_CLASS_WRITE_TOA_GEO** ... write tool geometry data		
9202 MD number	**USER_CLASS_WRITE_TOA_WEAR** ... write tool wear data		
9203 MD number	**USER_CLASS_WRITE_FINE** ... protection level write fine in MD 9450: MM_WRITE_TOA_FINE_LIMIT and write fine in MD 9451: MM_WRITE_ZOA_FINE_LIMIT		
9204 MD number	**USER_CLASS_WRITE_TOA_SC** (HMI Advanced only) ... change protection level total tool offsets (SW 5 and higher).		
9205 MD number	**USER_CLASS_WRITE_TOA_EC** (HMI Advanced only) ... change protection level tool setup offsets (SW 5 and higher).		
9206 MD number	**USER_CLASS_WRITE_TOA_SUPVIS** (HMI Advanced only) ... change protection level tool monitoring limit values (SW 5 and higher). Authorization applies to all the limit values: Workpiece count, service life, wear and mode of monitoring.		
9207 MD number	**USER_CLASS_WRITE_TOA_ASSDNO** (HMI Advanced only) ... change D No. assigned to a tool edge (SW 5 and higher).		
9208 MD number	**USER_CLASS_WRITE_MAG_WGROUP** (HMI Advanced only) ... change wear group magazine location/mag.		
Default setting: 3	Minimum input limit: 0	Maximum input limit: 5	

© Siemens AG, 2003. All rights reserved

9209–9224 MD number	**USER_CLASS_ ....** Protection levels for access to ...		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1.1		
9209 MD number	**USER_CLASS_WRITE_TOA_ADAPT** ... Write tool adapter geometry data (SW 5 and higher).		
9210 MD number	**USER_CLASS_WRITE_ZOA** ... Write settable zero offsets		
9211 MD number	**USER_CLASS_READ_GUD_LUD** (for HMI Embedded and HMI Advanced) ... read user variables (SW 6.1 and higher).		
9212 MD number	**USER_CLASS_WRITE_GUD_LUD** (for HMI Embedded and HMI Advanced) ... write user variables (SW 6.1 and higher).		
9213 MD number	**USER_CLASS_OVERSTORE_HIGH** ... Extended overstore		
9214 MD number	**USER_CLASS_WRITE_PRG_CONDIT** ... Enable program		
9215 MD number	**USER_CLASS_WRITE_SEA** ... Write setting data		
9216 MD number	**USER_CLASS_READ_PROGRAM** (MMC 100 only) ... Read parts program		
9217 MD number	**USER_CLASS_WRITE_PROGRAM** (MMC 100 only) ... write parts program		
9218 MD number	**USER_CLASS_SELECT_PROGRAM** ... select parts program		
9219 MD number	**USER_CLASS_TEACH_IN** ... Teach In		
9220 MD number	**USER_CLASS_PRESET** ... Preset		
9221 MD number	**USER_CLASS_CLEAR_RPA** ... Delete all R parameters		
9222 MD number	**USER_CLASS_WRITE_RPA** ... Write R parameters		
9223 MD number	**USER_CLASS_SET_V24** (MMC 100 only) ... User parameters via V24 (RS232)		
9224 MD number	**USER_CLASS_READ_IN** (MMC 100) only ... Read in data		

9225 MD number	**USER_CLASS_READ_CST** (MMC 100 only) Protection level for access to standard cycles		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 2		

9226 MD number	**USER_CLASS_READ_CUS** (MMC 100 only) Protection level for access to user cycles		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 2		

© Siemens AG, 2003. All rights reserved

9227 MD number	**USER_CLASS_SHOW_SBL2** Hide protection level for access to single block 2 (SBL2)		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 3.5 (MMC 100)		
Significance:	The SBL2 function is made available on the operator interface only if the currently active protection level provides equal or higher access authorization than set in this MD.		
Special cases, errors, ...	If SBL2 is selected and the access protection is then set to a level that does not display SBL2, SBL2 remains selected. It is then possible to switch over to SBL1, causing SBL2 to be deselected automatically.		
References	/BA/ Operator's Guide 840D/810D/840Di		

9228–9230 MD number	**USER_CLASS_ ....** Access level for...		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective after SW UPD	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 4.2		
9228 MD number	**USER_CLASS_READ_SYF** ... Select directory SYF		
9229 MD number	**USER_CLASS_READ_DEF** ... Select directory DEF		
9230 MD number	**USER_CLASS_READ_BD** ... Select directory BD		

Note MD 9231 to MD 9241 (for HMI Advanced only)

9231 MD number	**USER_CLASS_WRITE_RPA_1** (for HMI Advanced) Write protection for first RPA area		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9232 MD number	**USER_BEGIN_WRITE_RPA_1** (for HMI Advanced) Start of the first RPA area		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9233 MD number	**USER_END_WRITE_RPA_1** (for HMI Advanced) End of the first RPA area		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

© Siemens AG, 2003. All rights reserved

9234 MD number	**USER_CLASS_WRITE_RPA_2** (for HMI Advanced) Write protection for second RPA area		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9235 MD number	**USER_BEGIN_WRITE_RPA_2** (for HMI Advanced) Start of the second RPA area		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9236 MD number	**USER_END_WRITE_RPA_2** (for HMI Advanced) End of the second RPA area		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9237 MD number	**USER_CLASS_WRITE_RPA_3** (for HMI Advanced) Write protection for third RPA area		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9238 MD number	**USER_BEGIN_WRITE_RPA_3** (for HMI Advanced) Start of the third RPA area		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9239 MD number	**USER_END_WRITE_RPA_3** (for HMI Advanced) End of the third RPA area		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		

9240–9241 MD number	**USER_CLASS_ ....** Change...		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5		
9240 MD number	**USER_CLASS_WRITE_TOA_NAME** (for HMI Advanced) ... tool designation and Duplo.		
9241 MD number	**USER_CLASS_WRITE_TOA_TYPE** (for HMI Advanced) ... tool type.		

© Siemens AG, 2003. All rights reserved

Note MD 9460, MD 9461 and MD 9500 (for HMI Embedded only)

9460 MD number	**PROGRAMM_SETTINGS** (MMC 100 only) Reset-proof data storage for settings in the PROGRAM operating area		
Default setting: 5	Minimum input limit: 0	Maximum input limit: FFFF	
Changes effective immediately	Protection level: 3/4	Unit: Hex	
Data type: LONG	Applies from SW 5.1 (bits 2 and 3 from SW 5.3)		
Significance:	Data storage for settings in the PROGRAM area. The settings are retained after a RESET. Bit 0: Turning technology, used internally only Bit 1: Turning technology, used internally only Bit 2 = 1 Automatic enabling for programs Bit 2 = 0 No automatic enabling for programs, default for SW 5.1 Bit 3 = 1 The soft keys for the elements in the contour calculator (contour programming) are displayed as symbols, SW 5.3 and higher Bit 3 = 0 The soft keys for the elements in the contour calculator (contour programming) are labeled with texts, default in SW 5.3 and higher Bit 4 = 1 The LF symbol is hidden in the editor, SW 6.1 and higher Bit 4 = 0 The LF symbol is not hidden in the editor, SW 6.1 and higher		
Related to	–		

9461 MD number	**CONTOUR_END_TEXT** (MMC 100 only) String to be added at the end of the contour on completion of an input		
Default setting: None	Minimum input limit: 0 characters	Maximum input limit: 80 characters	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: STRING	Applies from SW 5.1		
Significance:	This string is appended to the end of the contour after completion of a contour input.		
Application example(s)	M30		

9500 MD number	**NC_PROPERTIES** (MMC 100 only) NC properties		
Default setting: 1111 1111	Minimum input limit: 0000 0000	Maximum input limit: 1111 1111	
Changes effective immediately	Protection level: 3/4	Unit: Bit field	
Data type: BYTE	Applies from SW 2		
Significance:	Basic configuration of NC properties: Bit 0 = 1 Digital drives Bit 1 = 1 Software start-up switch Bit 2...4: Reserved		
Related to	–		

9510–9513 MD number	**USER_CLASS_ ....** Protection levels for PROGRAM operating area for access to ...		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 6.1		
9510 MD number	**USER_CLASS_DIRECTORY1_P** (for HMI Advanced only) ... Network drive 1 (SW 6.1 and higher).		
9511 MD number	**USER_CLASS_DIRECTORY2_P** (for HMI Advanced only) ... Network drive 2 (SW 6.1 and higher).		
9512 MD number	**USER_CLASS_DIRECTORY3_P** (for HMI Advanced only) ... Network drive 3 (SW 6.1 and higher).		
9513 MD number	**USER_CLASS_DIRECTORY4_P** (for HMI Advanced only) ... Network drive 4 (SW 6.1 and higher).		

© Siemens AG, 2003. All rights reserved

9516–9519 MD number	**USER_CLASS_ ....** Protection levels for MACHINE operating area for access to ...		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 6.1		
9516 MD number	**USER_CLASS_DIRECTORY1_M** (for HMI Advanced only) ... Network drive 1 (SW 6.1 and higher).		
9517 MD number	**USER_CLASS_DIRECTORY2_M** (for HMI Advanced only) ... Network drive 2 (SW 6.1 and higher).		
9518 MD number	**USER_CLASS_DIRECTORY3_M** (for HMI Advanced only) ... Network drive 3 (SW 6.1 and higher).		
9519 MD number	**USER_CLASS_DIRECTORY4_M** (for HMI Advanced only) ... Network drive 4 (SW 6.1 and higher).		

© Siemens AG, 2003. All rights reserved

4.2 General machine data

10131 MD number	**SUPPRESS_SCREEN_REFRESH** Behavior of display refresh on overload		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 5		
Significance:	In some parts programs, the main run has to wait until the preprocessor prepares new blocks. In this case, the preprocessor and display refresh compete for computer time. The MD defines how the NC should react if the preprocessor is too slow. 0: Refreshing of the display is suppressed **in all channels** 1: Refreshing of the display is suppressed **in timecritical channels** 2: Refreshing of the display is basically **never** suppressed.		

10361 MD number	**FASTIO_DIG_SHORT_CIRCUIT** Short-circuits of digital inputs and outputs		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.2		
Significance:	Defines short circuits between digital output and input signals of highspeed NCK I/Os. The short circuits are implemented internally in the NCK through logic operations between defined output signals and signal states read in from highspeed NCK I/Os or from the PLC (via VDI interface). If several output bits are specified for an input bit in overwrite mode, the last assignment defined in the list determines the result. When non-existent or inactive inputs/outputs (MD 10350: FASTIO_DIG_NUM_INPUTS, MD 10360: FASTIO_DIG_NUM_OUTPUTS) are defined, they are ignored without generation of an alarm. Bits 0 ...7 (LSBs): Number of input byte (1 ...5) to be written Bits 8 ...15: Bit number within input byte (1 ...8) When A0 (hexadecimal) is added to the input bit number, the external input status is not overwritten, but ANDed with the specified output. When B0 (hexadecimal) is added to the input bit number, the external iinput status is not overwritten, but ORed with the specified output.		
Application example(s)	FASTIO_DIG_SHORT_CIRCUIT[0] = H04010302 Input: 3rd bit of 2nd byte Output: 4th bit of 1st byte (= 4th onboard NCU output) The input status is overwritten by the specified output. FASTIO_DIG_SHORT_CIRCUIT[1] = H0705A201 Input: 2nd bit of 1st byte (= 2nd onboard NCU input) Output: 7th bit of 5th byte The input status is ANDed with the specified output. FASTIO_DIG_SHORT_CIRCUIT[2] = H0103B502 Input: 5th bit of 2nd byte Output: 1st bit of 3rd byte The input status is ORed with the specified output.		

10706 MD number	**SLASH_MASK** Activation of block skip		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Change effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 5		

© Siemens AG, 2003. All rights reserved

10706 MD number	**SLASH_MASK** Activation of block skip
Significance:	When SLASH_MASK = 1 it is also possible to activate block skipping during program execution. **Caution:** After activation of block skipping the axes are stopped for the duration of reorganization process.
Special cases, errors, ...	–
Related to	–

11120 MD number	**MN_LUD_EXTENDED_SCOPE** Activate program-global variables		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOL	Applies from SW 4.4		
Significance:	MD=0: The user data of the main program level are only active at this level MD=1: The user data of the main program level are also visible in the subprogram level (program-global variable PUD)		
Application example(s)	See /PGA/		

© Siemens AG, 2003. All rights reserved

11230 MD number	**MD_FILE_STYLE** Structure of MD files for data backup		
Default setting: 3	Minimum input limit: –	Maximum input limit: –	
Changes effective immediately	Protection level: 3/7	Unit: –	
Data type: BYTE	Applies from SW 3		
Significance:	Appearance of a machine data file for 'upload' Bit 0 (LSB): Line checksum is generated Bit 1: MD numbers are generated Bit 2: Channel axis names as array index for axisMD in TEA file Bit 3: With NCU link the MDs of the LINK axes are also output. Bit 4: All local axes are output.		
Special cases, errors, ...	–		
Related to	–		

11270 MD number	**DEFAULT_VALUES_MEM_MASK** Activation of function "Memory for initialization values of NC data elements"		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Bit 0 = 1 GUD default settings Bit 1 = 0 The default settings specified when GUD are defined are not stored Bit 1 = 1 The default settings specified when GUD are defined are permanently stored. The memory reserved via MD 18150: MM_GUD_VALUES_MEM is used to store them. Stored defaults can then be restored again as configured with REDEF.		
Related to	MD 18150: MM_GUD_VALUES_MEM		
References	PGA Chapter 3		

4.3 Channel-specific machine data

28150 MD number	**MM_NUM_VDIVAR_ELEMENTS** Number of write elements for PLC variables		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.2		
Significance:	This MD defines the number of elements available to the user for writing PLC variables ($A_DBx= ...). This number also applies to block searches, but **not** to synchronized actions. Each element requires approximately 24 bytes of memory. One element is required to perform each write operation for highspeed sequential writing of PLC variables. However, if the variables are accessed for writing at separate times (block has already been transported), the number of elements can be reduced. The number of read access operations (var=$A_DBx) is not limited.		
Application example(s)			

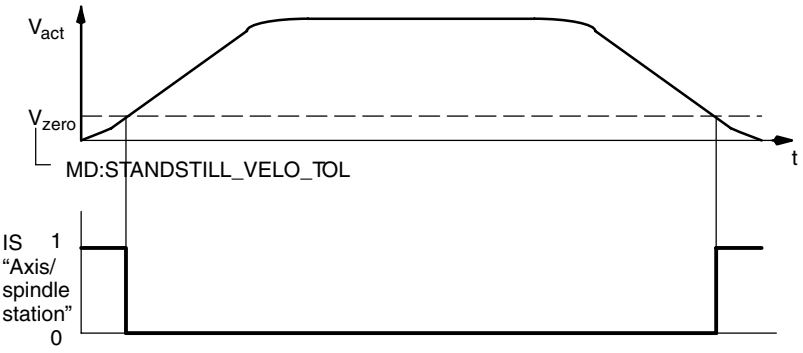
© Siemens AG, 2003. All rights reserved

4.4 Axis/spindle-specific NCK machine data

33050 MD number	**LUBRICATION_DIST** Traversing distance for lubrication from PLC		
Default setting: 100 000 000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	After the distance defined in the MD has been covered, the state of the axial interface signal “Lubrication pulse” is inverted with which an automatic lubrication device can be activated. The travel path is summated after POWER ON. The “Lubrication pulse” can be used with axes and spindles.		
Application example(s)	The machine bed can be lubricated as a function of the distance traversed in each case.		
Special cases, errors, ...	Note: Entering 0 sets the IS “Lubrication pulse” (DB31, ... DBX76.0) for each cycle.		
Related to	IS “Lubrication pulse” (DB31, ... DBX76.0)		

35590 MD number	**PARAMSET_CHANGE_ENABLE** Parameter set can be changed		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 4.1		
Significance:	0: Parameter set changes cannot be controlled. The first parameter set or the internal parameter set selection is always active. 1: The parameter set applied in the servo is specified via the VDI interface. VDI interface defined. The parameter sets 1 to 6 can be selected. Sets are selected by DB31 ff, DBB9, bits 0..2 in the binary-coded value range 0 ... 5. Values 6 and 7 select parameter set no. 6. Internal parameter set selection takes priority over a selection made by the PLC. 2: Same as 1:, however the internal parameter set selection is deactivated. The parameter set contains the following axial machine data: $MA_AX_VELO_LIMIT $MA_POSCTRL_GAIN $MA_EQUIV_CURRCTRL_TIME $MA_EQUIV_SPEEDCTRL_TIME $MA_DYN_MATCH_TIME $MA_DRIVE_AX_RATIO_DENOM $MA_DRIVE_AX_RATIO_NUMERA		
Application example(s)	see Chapter 6		
Related to	Interface signals DB31, ..., DBX9.0, 1, 2 and DBX69.0, 1, 2		
References	/FB/, H2, “Output of Auxiliary Functions to PLC”.		

© Siemens AG, 2003. All rights reserved

36060 MD number	**STANDSTILL_VELO_TOL** Maximum velocity/speed “Axis/spindle stationary”		
Default setting: 5	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: Linear axis: mm/min Rotary axis: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The zero speed range for the axis velocity and spindle speed is set in this machine data. If the current actual velocity of the axis or the actual speed of the spindle is less than the value I entered, the IS “Axis/spindle stationary” (DB 31 ...., DBX61.4) is set.		
			
Application example(s)	The pulse enable should not be removed until the axis/spindle is stationary to ensure that the axis/spindle is stopped in a controlled manner. Otherwise the axis will coast to rest.		
Special cases, errors, ...	**Note:** When the SAFETY function is used with active actualvalue coupling, a travel command for the following axis may be generated by the ELG grouping when the axis/spindle is test stopped. This command activates alarm 21612: VDI signal “Servo enable” reset during motion. Activation of this alarm can be prevented by setting a value higher than 5 mm in MD 36060: STANDSTILL_VELO_TOL.		
Related to	IS “Axis/spindle stationary” (DB31, ... DBX61.4)		

© Siemens AG, 2003. All rights reserved

36620 MD number	**SERVO_DISABLE_DELAY_TIME** Cutout delay servo enable		
Default setting: 0.1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Maximum delay time for removal of “Servo enable” after a fault. If the axis/spindle is still moving the speed enable (servo enable) of the drive is removed by the control at the latest after the delay time set. The entered delay time is activated in the following events: • With faults which cause the axes to stop immediately. • If IS “Servo enable” is removed by the PLC As soon as the actual speed has reached the zero speed range (MD 36060: STANDSTILL_VELO_ TOL) “Servo enable” is removed for the drive. The time set must be long enough to allow the axis/spindle to be brought to zero speed from maximum traversing velocity/speed. If the axis/spindle is stationary “Servo enable” is removed for the drive immediately. See also Figure . Machine data only takes effect in speed mode. This is not achieved with stepper motor without external encoder, because the internal position control mode is never left.		
Application example(s)	The drive speed control should be maintained long enough for the axis/spindle to come to a standstill from maximum traversing speed. Removal of “Servo enable” for a moving axis/ spindle must be delayed until this has happened.		
Special cases, errors, ...	**Caution:** If the setting for the servo enable cutout delay is too small the servo enable will be removed even though the axis/spindle is still moving. The axis/spindle is then stopped abruptly with setpoint 0. Therefore, the SERVO_DISABLE_DELAY_TIME should be greater than the braking ramp time in error situations (MD: AX_EMERGENCY_STOP_TIME).		
Related to	IS “Servo enable” (DB31, ... DBX2.1) MD: AX_EMERGENCY_STOP_TIME (Time for braking ramp when an error occurs)		
Related to	MD 1404: PULSE_SUPPRESSION_DELAY		

© Siemens AG, 2003. All rights reserved

4.5 Machine data for involute (SW 6 and later)

21015 MD number	**INVOLUTE_RADIUS_DELTA** End point monitoring for involute		
Default setting: 0.01	Minimum input limit: 0	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: mm	
Data type: DOUBLE	Applies from SW 6.3		
Significance:	Permissible absolute difference in radius with involute interpolation [mm]. With involute interpolation the radius of the base circle determined by the end point can differ from the programmed radius. This data limits the maximum permissible difference between the start and end radii.		
References	/PG/, Programming Guide: Fundamentals		

21016 MD number	**INVOLUTE_AUTO_ANGLE_LIMIT** Automatic angle limitation for involute interpolation		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.3		
Significance:	If the angle of rotation is programmed (AR=angle) for an involute, the maximum angle of rotation is limited if the involute is moving in the direction of the base circle (AR < 0). The maximum angle of rotation is reached when the involute meets the base circle. If an angle larger than the maximum angle is programmed, an alarm is normally output and the NC program aborted. If this MD is set to 1, then any angle is accepted for programming without alarm; if necessary, this may be limited automatically. This, for example, makes it easier to program an involute which starts at a point outside the base circle and ends directly on it.		
References	/PG/, Programming Guide: Fundamentals		

© Siemens AG, 2003. All rights reserved

4.6 System variables (SW 4 and later)

Name	**$P_FUMB**			
Meaning	Unassigned parts program memory (**F**ree **U**ser **M**emory **B**uffer)			
Data type	DWORD			
Value range				
Access	Read in parts program	Write in parts program	Read in Synchronized action	Write in Synchronized action
	yes	–	–	–
Implicit preprocessing stop	–	–	–	–

Name	**$A_DBB[n]**			
Meaning	Data on PLC (data type BYTE)			
Data type	INT			
Value range	0 – 255			
Indices	Meaning			Value range
	Position offset within PLC NC communications input area			0 – MD_MAX-NUM_VDI_VAR
Access	Read in parts program	Write in parts program	Read in Synchronized action	Write in Synchronized action
	yes	yes	yes	
Implicit preprocessing stop	yes			

Name	**$A_DBW[n]**			
Meaning	Data on PLC (data type WORD)			
Data type	INT			
Value range	0 – 65535			
Indices	Meaning			Value range
	Position offset within PLC NC communications input area			0 – MD_MAX-NUM_VDI_VAR
Access	Read in parts program	Write in parts program	Read in Synchronized action	Write in Synchronized action
	yes	yes	yes	
Implicit preprocessing stop	yes			

© Siemens AG, 2003. All rights reserved

Name	**$A_DBD[n]**			
Meaning	Data on PLC (data type DWORD)			
Data type	INT			
Value range	0 – 4294967295			
Indices	Meaning			Value range
	Position offset within PLC NC communications input area			0 – MD_MAX-NUM_VDI_VAR
Access	Read in parts program	Write in parts program	Read in Synchronized action	Write in Synchronized action
	yes	yes	yes	
Implicit preprocessing stop	yes			

Name	**$A_DBR[n]**			
Meaning	Data on PLC (data type REAL)			
Data type	DOUBLE			
Value range	1491298E–45 – 3402823E+38			
Indices	Meaning			Value range
	Position offset within PLC NC communications input area			0 – MD_MAX-NUM_VDI_VAR
Access	Read in parts program	Write in parts program	Read in Synchronized action	Write in Synchronized action
	yes	yes	yes	
Implicit preprocessing stop	yes			

■

© Siemens AG, 2003. All rights reserved

Signal Descriptions

5

The signal overviews list only the signals that are described below. For a complete list of signals, please see

References: /LIS/, “Lists”

5.1 NC-specific signals

5.1.1 Signals from PLC to NC (DB10)

DB 10 **DBX56.4 .. 7**	**Keyswitch setting 0 to 3**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Significance of signal	Access to certain data types can be disabled by the keyswitch position. Input, alteration, deletion of data as well as certain operations on the operator panel can be disabled for certain user groups in this way. Keyswitch position 0 offers the fewest access rights and position 3 the most. The signals “Key switch positions 1 to 3” can either be defined directly from the key switch on the machine control panel or from the PLC user program. Only one interface signal must be set in each case. If more than one interface signal is set at any one time, they are no longer valid and keyswitch position 3 is automatically set by the control. The following signal combinations apply:

	DB10, DBB56			
Keyswitch position	Bit 7	Bit 6	Bit 5	Bit 4
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	1	0	0	0

	The disabled data areas are assigned to the keyswitch positions in the MMC machine data for protection levels (see Subsection 2.5.3).
Application example(s)	Depending on the rights given to the operator, programmer or installation engineer, certain functions will be disabled by the keyswitch. Unintentional changes to data (e.g. zero offsets) or activation of program conditions (e.g. selecting dry run feedrate) by the operator are therefore ruled out.
Related to	Disabling with password (see Subsection 2.5.3).

© Siemens AG, 2003. All rights reserved

5.1.2 Signals from NC to PLC (DB10)

DB 10 DBX103.0	**MMC alarm is active**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ⟶ 1	At least 1 MMC alarm (alarm numbers 100 000 to 105 999) is active.	
Signal state 0 or signal transition 1 ⟶ 0	No MMC alarm is active.	

DB 10 DBX104.7	**NCK CPU Ready**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ⟶ 1	The NCK CPU is ready and sending ready signals cyclically to the PLC. After a proper initial start and the first complete OB1 cycle (initial setting cycle) the PLC and NCK continuously exchange sign of life signals. The PLC basic program sets IS “NCK CPU Ready” to 1.	
Signal state 0 or signal transition 1 ⟶ 0	The NCK CPU is not ready. If a sign of life is not received from the NCK, the PLC/NCK interface is neutralized by the PLC basic program and IS “NCK CPU Ready” is set to 0. The following measures are introduced by the PLC basic program: • Status signals from NCK to PLC (user interface) are reset • Modification signals for auxiliary functions are deleted • Cyclic processing of user PLC to NCK user interface is terminated.	
Application example(s)	Individual PLC outputs can, for example, be set to a defined state from the PLC user program.	
References	/DA/, “Diagnostics Guide” /FB/, P3, “Basic PLC Program”	

DB 10 DBX108.1 Data block	**MMC CPU2 Ready (to OPI or MPI)** Signal(s)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 3.1
Signal state 1 or signal transition 0 ⟶ 1	The MMC CPU is ready and signals cyclically to the NCK.	
Signal state 0 or signal transition 1 ⟶ 0	The MMC is not ready.	

DB 10 DBX108.2 DBX108.3	**MMC CPU1 Ready (MMC to MPI) MMC CPU1 Ready (MMC to OPI, standard link)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ⟶ 1	The MMCCPU is ready and sending signals cyclically to the NCK.	
Signal state 0 or signal transition 1 ⟶ 0	The MMC is not ready.	
Application example(s)	The necessary measures can be introduced by the PLC user program if “MMC-CPU1-Ready” = 0.	
References	/DA/, “Diagnostics Guide”	

© Siemens AG, 2003. All rights reserved

DB 10 DBX108.6	**611D Ready**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	All existing drives signal the status drive ready (summary of axial interface signals “DRIVE ready”).	
Signal state 0 or signal transition 1 ——> 0	As soon as the status drive not ready is signaled from a drive (i.e. IS “DRIVE ready” = 0).	
Signal irrelevant for ...	SINUMERIK FM-NC and 840Di	
Related to	IS “DRIVE ready” (DB31, ... DBX93.5)	

DB 10 DBX108.7	**NC Ready**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The control system is ready. This interface signal is an image of the relay contact “NC-ready”. This signal is set if: • Relay contact “NC ready” is closed • All the voltages in the control have been established • The control is in cyclic mode	
Signal state 0 or signal transition 1 ——> 0	The control is not ready. The relay contact “NC Ready” is open. The following faults will cause NC Ready to be canceled: • Undervoltage and overvoltage monitoring has responded • Individual components are not ready (NCK CPU Ready) • NC CPU watchdog If the signal “NC Ready” becomes 0 the following measures are introduced by the control if still possible: • Removal of servo enables (which stops the drives) • The following measures are introduced by the PLC basic program: – Status signals from NCK to PLC (user interface) are reset – Modification signals for auxiliary functions are deleted – Cyclic processing of the PLC to NCK user interface is terminated For further information see References! The control is not ready until after POWER ON.	
Related to	Relay contact “NC Ready”	
References	/DA/, “Diagnostics Guide” /FB/, P3, “Basic PLC Program”	

DB 10 DBX109.0	**NCK alarm is active**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	At least one NCK alarm is present. This is a group signal for the “Channel-specific NCK alarm is active” interface signals (DB21, ... DBX36.6) of all existing channels.	
Signal state 0 or signal transition 1 ——> 0	No NCK alarm is active.	
Related to	IS “Channel-specific NC alarm is active” (DB21, ... DBX36.6) IS “NCK alarm with processing stop is active” (DB21, ...DBX36.7)	
References	/DA/, “Diagnostics Guide”	

© Siemens AG, 2003. All rights reserved

DB 10 **DBX109.6**	**Air temperature alarm**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The ambient temperature or fan monitoring function has responded. This could be due to the following: • The temperature monitoring has registered ambient temperatures that are too high (approx. 60 °C). Alarm 2110 “NCK temperature alarm” is triggered. • The speed monitoring of the 24 V DC fan used to cool the module has responded. Alarm 2120 “NCK fan alarm” is triggered. Measures: Replace the fan and/or provide for additional ventilation. When a temperature or fan error responds, a relay contact (terminal 5.1, 5.2 or 5.1, 5.3) in the infeed/regenerative feedback unit is activated which can be evaluated by the customer.	
Signal state 0 or signal transition 1 ——> 0	Neither the temperature monitoring nor the fan monitoring has responded.	
Signal irrelevant for ...	SINUMERIK FM-NC	
Application example(s)	Appropriate measures can be introduced by the PLC user program if the temperature or fan monitoring is activated.	
Related to	When a temperature and fan error responds, a relay contact (terminal 5.1, 5,2 or 5.1, 5.3) in the infeed/regenerative feedback unit is activated. This can be evaluated.	
References	/DA/, “Diagnostics Guide”	

DB 10 **DBX109.7**	**NCK battery alarm**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The NCK battery voltage alarm has responded. This could be due to the following: • The battery voltage is in the prewarning limit range (approx. 2.7 to 2.9V). Alarm 2100 “NCK battery warning threshold reached” has been triggered. See References for effects and measures. • The battery voltage is below the prewarning limit (≦ 2.6V). Alarm 2101 “NCK battery alarm” is triggered cyclically. Effects: A supply voltage failure – e.g. when the control is switched off – would result in the loss of battery-backed data (e.g. parts program memory, variables, machine data ....). Measures: See References. • As the control boots, the battery voltage is detected to be below the prewarning limit (≦ 2.6 V). Alarm 2102 “NCK battery alarm” is triggered; NC Ready and Mode Group Ready are not output. Effects: Some of the batterybacked data may already have been lost! Measures: See References.	
Signal state 0 or signal transition 1 ——> 0	The battery voltage is above the lower limit value (normal conditions).	
Special cases, errors, ...	The NCK batteries should only be replaced while the NC is switched on to avoid data loss because of insufficient memory backup.	
References	/DA/, “Diagnostics Guide” /IAD/, SINUMERIK 840D, Installation and StartUp Guide /IBI/, SINUMERIK 840Di, Installation and StartUp Guide /IAG/, SINUMERIK 810D, Installation and StartUp Guide /IAF/, SINUMERIK FM-NC, Installation and StartUp Guide	

© Siemens AG, 2003. All rights reserved

5.2 Signals to/from operator panel (DB19)

5.2.1 Signals to operator panel (PLC → MMC)

DB 19 DBX0.0	**Screen bright**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 4.1
Signal state 1 or signal transition 0 ——> 1	The screen darkening is canceled.	
Signal state 0 or signal transition 1 ——> 0	The screen darkening remains in effect.	
Related to	DB19, DBX0.1	

DB 19 DBX0.1	**Darken screen**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	The screen is switched to “dark” by the PLC user program. The automatic bright/dark function for the screen is therefore inactive: i. e. the screen does not automatically become bright when a key on the keyboard is pressed.	
Signal state 0 or signal transition 1 ——> 0	The screen is switched to “bright” by the PLC user program. With this signal status, screen darkening/brightening by the control can be diverted automatically via the keyboard. The screen is darkened if the keyboard is not touched for a time period corresponding to the setting in MD 9006: DISPLAY_BLACK_TIME (screen darkening delay). The screen is switched to bright again the next time a key on the operator panel is pressed.	
Application example(s)	Screen saver	
Special cases, errors, ...	**Caution**: When IS “Darken screen” (DB19, DBX0.1) is set to 1, then the keyboard on the operator panel remains operational! It is therefore advisable to disable it at the same time by means of IS “Key disable” (DB19, DBX0.2).	
Related to	IS “Key disable” (DB19, DBX0.2) MD 9006: DISPLAY_BLACK_TIME (screen darkening delay time)	

DB 19 DBX0.2	**Key disable**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	The key is disabled for the user.	
Signal state 0 or signal transition 1 ——> 0	The key is enabled for the user.	
Application example(s)	If the screen is darkened by means of IS “Darken screen” (DB19, DBX0.1), the keyboard should be inhibited at the same time by means of IS “Key disable” (DB19, DBX0. 2) to prevent the operator from entering keyboard commands unintentionally.	
Related to	IS “Darken screen” (DB19, DBX0.1)	

© Siemens AG, 2003. All rights reserved

DB 19 DBX0.3	**Clear cancel alarms**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Clear error key on the machine control panel is depressed. The master control receives the bits and transmits the Cancel key to the alarm server, which then acknowledges all Cancel alarms of the NCK and MMC. The PLC application acknowledges the PLC alarms itself. POWER ON and Reset alarms remain active on the NCK until the cause of the error has been rectified.	
Signal state 0 or signal transition 1 ——> 0	Clear error key on the machine control panel is not depressed.	
Functionality	Only applicable to MMC 103 or PCU 50.	
Related to	DB19.DBX20.3	

DB 19 DBX0.4	**Clear recall alarms**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Clear error key on the machine control panel is depressed. The master control receives the bits and transmits the Cancel key to the alarm server, which then acknowledges all Cancel alarms of the NCK and MMC. The PLC application acknowledges the PLC alarms itself. POWER ON and Reset alarms remain active on the NCK until the cause of the error has been rectified.	
Signal state 0 or signal transition 1 ——> 0	Clear error key on the machine control panel is not depressed.	
Application example(s)	valid only for MMC 103	
Related to	DB19.DBX20.4	

DB 19 DBX0.7	**Actual value in WCS**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	The PLC selects the display of actual values in the workpiece coordinate system (WCS). When the machine area is selected, the WCS display is always activated, i.e. the “Position” window displays the machine and special axes as well as their actual positions and distances-to-go in the WCS. Since the interface signal is evaluated only when the machine basic display is called, the operator can switch over between the two coordinate systems within the machine area as often as required by means of soft keys “Actual values MCS” and “Actual values WCS”.	
Signal state 0 or signal transition 1 ——> 0	When the Machine area is selected the coordinate system previously selected (WCS or MCS) is reactivated and displayed.	
Application example(s)	IS “Actual value in WCS” (DB19, DBX0.7) = 1 can be applied to ensure that the workpiece coordinate system display (WCS) frequently required by the operator is selected every time he enters the machine area.	
Related to	IS “Switch over MCS/WCS” (DB19, DBX20.7)	
References	/BA/, “Operator’s Guide”	

© Siemens AG, 2003. All rights reserved

DB 19 DBW2	Higraph first error display	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1

DB 19 DBW4	Higraph first error display	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1

DB 19 DBX6.0 .. 7	Analog spindle 1, utilization in percent	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1

DB 19 DBX7.0 .. 7	Analog spindle 2, utilization in percent	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1

DB 19 DBX8.0 .. 7	Channel number of machine control panel on MMC	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1

DB 19 DBX10.0	Programming area selection	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	Program area selection active	
Signal state 0 or signal transition 1 ——> 0	Program area selection inactive	

DB 19 DBX10.1	Alarm area selection	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	Alarm area selection active	
Signal state 0 or signal transition 1 ——> 0	Alarm area selection inactive	

DB 19 DBX10.2	Tool offset selection	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	Tool offset selection active	
Signal state 0 or signal transition 1 ——> 0	Tool offset selection inactive	

© Siemens AG, 2003. All rights reserved

DB 19 DBX10.7	**ShopMill control signal**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1

DB 19 DBX12.2	**COM2**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	COM2 active	
Signal state 0 or signal transition 1 ——> 0	COM2 inactive	
Application example(s)	Valid for MMC 103; A file transfer can be initiated via RS232C.	

DB 19 DBX12.3	**COM1**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	COM1 active	
Signal state 0 or signal transition 1 ——> 0	COM1 inactive	
Application example(s)	Valid for MMC 103; A file transfer can be initiated via RS232C.	

DB 19 DBX12.4	**RS-232-C stop**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C stop active	
Signal state 0 or signal transition 1 ——> 0	RS232C Stop inactive	
Application example(s)	Valid for MMC 103; A file transfer can be initiated via RS232C.	

DB 19 DBX12.5	**RS-232-C external**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C external active	
Signal state 0 or signal transition 1 ——> 0	RS232C external inactive	
Application example(s)	Valid for MMC 103; A file transfer can be initiated via RS232C.	

DB 19 DBX12.6	**RS-232-C OFF**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C OFF active	
Signal state 0 or signal transition 1 ——> 0	RS232C OFF inactive	
Application example(s)	Valid for MMC 103; A file transfer can be initiated via RS232C.	

© Siemens AG, 2003. All rights reserved

DB 19 DBX12.7	**RS232C ON**		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C ON active		
Signal state 0 or signal transition 1 ——> 0	RS232C ON inactive		
Application example(s)	Valid for MMC 103; A file transfer can be initiated via RS232C.		

DB 19 DBX13.5	**Unload**		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	Unload active		
Signal state 0 or signal transition 1 ——> 0	Unloading inactive		
Application example(s)	Valid for MMC 103; A file transfer can be initiated using the hard disk.		

DB 19 DBX13.6	**Load parts program**		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Load active		
Signal state 0 or signal transition 1 ——> 0	Loading inactive		
Application example(s)	Valid for MMC 103; A file transfer can be initiated using the hard disk.		

DB 19 DBX13.7	**Selection**		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Selection active		
Signal state 0 or signal transition 1 ——> 0	Selection inactive		
Application example(s)	Valid for MMC 103; A file transfer can be initiated using the hard disk.		

DB 19 DBX14.0 .. 6	**PLC index**		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Description	This byte for controlling the RS232C defines the PLC index which specifies the axis, channel or TO no. for the standard control file. This file is handled from PLC–>MMC in accordance with the task stored in DB19.DBB12.		
Application example(s)	Valid for MMC 100, with reference to DB19.DBB12 Dependent on DB19.DBX14.7 =0 –> Act. FS: PLC index specifying the axis, channel or TO no. =1 –> Pas. FS: PLC index for the user control file		

© Siemens AG, 2003. All rights reserved

DB 19 DBX14.7	**Active or passive file system**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Passive file system	
Signal state 0 or signal transition 1 ——> 0	Active file system	
Application example(s)	Valid for MMC 100, with reference to DB19.DBB12	

DB 19 DBX15.0 .. 7	**PLC line offset**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Description	This byte for controlling file transfer via the RS232C defines the line of the default or user control file on which the control file to be transferred is specified.	
Application example(s)	Valid for MMC 100, with reference to DB19.DBB12 Dependent on DB19.DBX14.7 =0 –> Act. FS: PLC line offset in a standard control file =1 –> Pas. FS: PLC line offset in a user control file	

DB 19 DBX16.0 .. 6	**PLC index for the user control file**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Description	This byte for controlling file transfer via hard disk defines the index for the control file (job list). This file is handled from PLC–>MMC in accordance with the task stored in DB19.DBB13.	
Application example(s)	Valid for MMC 103, with reference to DB19.DBB13 Dependent on DB19.DBX14.7 =0 –> Act. FS: PLC index for the standard control file =1 –> Pas. FS: PLC index for the user control file	

DB 19 DBX16.7	**Active or passive file system**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Passive file system	
Signal state 0 or signal transition 1 ——> 0	Active file system	
Application example(s)	for MMC 103 always 1	

DB 19 DBX17.0 .. 7	**PLC line offset in the user control file**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Description	This byte for controlling file transfer via hard disk defines the line of the user control file on which the control file to be transferred is specified.	
Application example(s)	Valid for MMC 103, with reference to DB19.DBB13 Dependent on DB19.DBX14.7 =0 –> Act. FS: PLC line offset in a standard control file =1 –> Pas. FS: PLC line offset in a user control file	

© Siemens AG, 2003. All rights reserved

DB 19 DBX44.0	**Mode change disable**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Mode change disable active	
Signal state 0 or signal transition 1 ——> 0	BA group switchover disabled	

DB 19 DBX45.0	**FC9 Out: Active**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1

DB 19 DBX45.1	**FC9 Out: Done**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1

DB 19 DBX45.2	**FC9 Out: Error**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1

DB 19 DBX45.3	**FC9 Out: StartError**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1

5.2.2 Signals from operator panel (MMC → PLC)

DB 19 DBX20.1	**Screen is dark**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	The screen is darkened.	
Signal state 0 or signal transition 1 ——> 0	The screen is not darkened.	
Application example(s)	This IS allows the PLC to determine whether the screen is darkened via IS “Darken screen” (DB19, DBX0.1) or by means of MD 9006: DISPLAY_BLACK_TIME (screen darkening delay).	
Related to	MD 9006: DISPLAY_BLACK_TIME (screen darkening delay time)	

DB 19 DBX20.3	**Cancel alarm cleared**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Cancel alarm deleted active	
Signal state 0 or signal transition 1 ——> 0	Cancel alarm deleted inactive	

© Siemens AG, 2003. All rights reserved

DB 19 DBX20.4	**Recall alarm cleared**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Recall alarm deleted inactive	
Signal state 0 or signal transition 1 ——> 0	Recall alarm deleted inactive	

DB 19 DBX20.6	**Simulation selected**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	On entry to simulation = 1	
Signal state 0 or signal transition 1 ——> 0	On exit from simulation = 0	
Application example(s)	Can be evaluated by machine manufacturer in order to activate the test on NC start. Drive machine data 1012, bit 2 must be = 0. The status “ext. pulse disable active, terminal 663 open” is then not transmitted to the NC.	
Related to	Drive machine data 1012, bit 2	

DB 19 DBX20.7	**Switch over MCS/WCS**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	The coordinate system is switched over from workpiece coordinate system (WCS) to machine coordinate system (MCS) or from MCS to WCS. After it has been set, the signal is active for 1 PLC cycle.	
Signal state 0 or signal transition 1 ——> 0	No effect	
Application example(s)	IS “Switch over MCS/WCS” (DB19, DBX20.7) must be transferred to the IS “Actual value in WCS” (DB19, DBX0.7) to make the switchover effective.	
Related to	IS “Actual value in WCS” (DB19, DBX0.7)	

DB 19 DBX22.0 .. 7	**Displayed channel number of MMC**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1

DB 19 DBX24.0	**Error (Status RS-232-C from PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Error status active	
Signal state 0 or signal transition 1 ——> 0	Error status inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24.	
Related to	Valid for MMC 100	

© Siemens AG, 2003. All rights reserved

DB 19 DBX24.1	**O.K. (Status RS-232-C from PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	O.K. status active	
Signal state 0 or signal transition 1 ——> 0	O.K. status inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24.	
Related to	Valid for MMC 100	

DB 19 DBX24.2	**COM2 (Status RS-232-C from PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	COM2 active	
Signal state 0 or signal transition 1 ——> 0	COM2 inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24.	
Related to	Valid for MMC 100	

DB 19 DBX24.3	**COM1 (Status RS-232-C from PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	COM1 active	
Signal state 0 or signal transition 1 ——> 0	COM1 inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24.	
Related to	valid for MMC 100i	

DB 19 DBX24.4	**RS-232-C stop (Status RS-232-C from PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C stop active	
Signal state 0 or signal transition 1 ——> 0	RS232C Stop inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24.	
Related to	Valid for MMC 100	

DB 19 DBX24.5	**RS232C external (Status RS232C of PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C external active	
Signal state 0 or signal transition 1 ——> 0	RS232C external inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24. The “RS232C external” bit is delayed until transfer of the external parts program has started and the selection has been made. Only then is an “NC Start” possible.	
Related to	Valid for MMC 100	

© Siemens AG, 2003. All rights reserved

DB 19 DBX24.6	**RS232C OFF (Status RS232C from PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C OFF active	
Signal state 0 or signal transition 1 ——> 0	RS232C OFF inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24.	
Related to	Valid for MMC 100	

DB 19 DBX24.7	**RS232C ON (Status RS232C from PLC)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	RS232C ON active	
Signal state 0 or signal transition 1 ——> 0	RS232C ON inactive	
Application example(s)	The current file management status is transmitted to the PLC in acknowledgment byte DB19.DBB24.	
Related to	Valid for MMC 100	

DB 19 DBB25.0 ...7	**Error RS232C**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Description	0 = No error 1 = Invalid number for control file (value in DB19.DBB14<127 or invalid) 2 = Unable to read DB19.DBB15 3 = Control file /BD.DIR/PLC_IN_OUT_xxx.TEA not found (value in DB19.DBB14 invalid) 4 = Invalid index in control file. (value in DB19.DBB15 wrong) 5 = Unable to open selected job list in control file (only MMC 103) 6 = Error in job list (job list interpreter returns error) (only MMC 103) 7 = Job list interpreter returns empty job list (only MMC 103) 8 = Error during RS232C transmission. The error text is entered in SERVICES LOG. 9 = Error while executing job list (only MMC 103)	
Related to	Valid for MMC 100	

DB 19 DBX26.1	**O.K. (job list selection from PLC, status)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Transfer terminated correctly	
Signal state 0 or signal transition 1 ——> 0	Transfer terminated with error	
Related to	Valid for MMC 103	

© Siemens AG, 2003. All rights reserved

DB 19 **DBX26.2**	**Error (job list selection from PLC, status)** (formerly bit 0)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Transfer terminated with error	
Signal state 0 or signal transition 1 ——> 0	Transfer terminated correctly	
Related to	Valid for MMC 103	

DB 19 **DBX26.3**	**Active (job list selection from PLC, status)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Job in progress	
Signal state 0 or signal transition 1 ——> 0	No job in progress	
Related to	Valid for MMC 103	

DB 19 **DBX26.5**	**Unload (job list selection from PLC, status)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Unload active	
Signal state 0 or signal transition 1 ——> 0	Unloading inactive	
Related to	Valid for MMC 103	

DB 19 **DBX26.6**	**Load (job list selection from PLC, status)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Load active	
Signal state 0 or signal transition 1 ——> 0	Loading inactive	
Related to	Valid for MMC 103	

DB 19 **DBX26.7**	**Selection (job list selection from PLC, status)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ——> 1	Selection active	
Signal state 0 or signal transition 1 ——> 0	Selection inactive	
Related to	Valid for MMC 103	

© Siemens AG, 2003. All rights reserved

DB 19 **DBB27.0 ...7**	**Data transfer error**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Meaning	0 = No error 1 = Invalid number for control file (value in DB19.DBB16<127 or invalid) 2 = Unable to read DB19.DBB17 3 = Control file /BD.DIR/PLC_IN_OUT_xxx.TEA not found (value in DB19.DBB14 invalid) 4 = Invalid index in control file. (value in DB19.DBB17 wrong) 5 = Unable to open selected job list in control file 6 = Error in job list. (job list interpreter returns error) 7 = Job list interpreter returns empty job list 8 = Error during RS232C transmission. The error text is entered in SERVICES LOG. 9 = Error while executing job list	
Related to	Valid for MMC 103	

DB 19 **DBX40.0 .. 7**	**Mode group number**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Meaning	This byte contains the mode group number.	

DB 19 **DBX41.0 .. 7**	**Channel number (FC9: ChanNo)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Meaning	This byte contains the channel number (FC9: ChanNo).	

DB 19 **DBX42.0**	**FC9: Start (measuring in Jog mode)**	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.1
Signal state 1 or signal transition 0 ⟶ 1	Start active	
Signal state 0 or signal transition 1 ⟶ 0	Start inactive	

© Siemens AG, 2003. All rights reserved

5.3 Channel-specific signals

5.3.1 Signals to channel

DB21, ... **DBX6.2** Data block	**Delete distance-to-go (channel-specific)** Signal(s) to channel (PLC → NC)		
Edge evaluation: yes		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Delete distance-to-go (channel-specific): IS “Delete distance-to-go (channel-specific)” is only active in AUTOMATIC mode for path axes. The rising edge of the interface signal only affects the axes involved in the geometry grouping. They are stopped with ramp stop and their distance-to-go deleted. Any remaining following error is not deleted. The next program block is then started. IS “Delete distance-to-go (channel-specific)” is therefore ignored by positioning axes. Note: IS “Delete distance-to-go” does not affect the dwell time in a program block with dwell time.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Signal irrelevant for ...	Positioning axes		
Application example(s)	To terminate the travel movement because of an external signal (e.g. probe)		
Special cases, errors, ...	“Delete distance-to-go (channel-specific)” When the axes have been stopped with IS “Delete distance-to-go” the next program block is prepared with the new positions. The geometry axes thus follow a different contour to the one originally defined in the parts program after a “Delete distance-to-go”. If G90 is programmed in the block after “Delete distance-to-go” it is at least possible to approach the programmed absolute position, whereas with G91 the position originally defined in the parts program is not reached.		
Related to	IS “Delete distance-to-go (axis-specific)” (DB31, ... DBX2.2)		

© Siemens AG, 2003. All rights reserved

5.3.2 Signals from channel

DB21, ... DBX36.6	**Channel-specific NCK alarm is active**	
Data block	Signal(s) from channel (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	At least one NCK alarm is present for this channel. The group interface signal “NCK alarm is present” is consequently also set (DB10, DBX103.0). The PLC user program can find out whether processing for the channel in question has been interrupted because of an NCK channel (IS “NCK alarm with processing stop present”).	
Signal state 0 or signal transition 1 ——> 0	No NCK alarm is active for this channel.	
Related to	IS “NCK alarm with processing stop present” (DB21, ...DBX36.7) IS “NCK alarm present” (DB10, DBX103.0)	
References	/DA/, “Diagnostics Guide”	

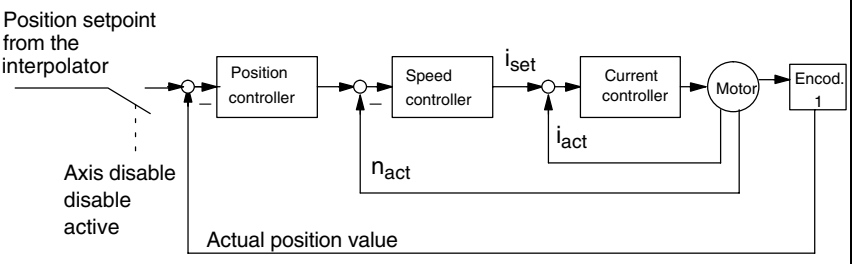
DB21, ... DBX36.7	**NCK alarm with processing stop present**	
Data block	Signal(s) from channel (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	At least one NCK alarm which is causing a processing stop of the parts program running in this channel is active.	
Signal state 0 or signal transition 1 ——> 0	There is no alarm active in this channel that is causing a processing stop.	
Application example(s)	With this alarm a program interruption because of an NCK alarm can be recognized immediately by the PLC user program and the necessary steps introduced.	
Related to	IS “Channel-specific NC alarm is present” (DB21, ... DBX36.6)	
References	/DA/, “Diagnostics Guide”	

DB21, ... DBX318.7	**Overstore active**	
Data block	Signal(s) from channel (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.1
Signal state 1 or signal transition 0 ——> 1	The function “Overstore” is enabled (with the channel-specific PI service “_N_OST_ON”). If the PI service is denied, the signal will not change.	
Signal state 0 or signal transition 1 ——> 0	The function “Overstore” is disabled (with the channel-specific PI service “_N_OST_OFF”). If the PI service is denied, the signal will not change.	
Application example(s)		
Related to		
References		

© Siemens AG, 2003. All rights reserved

5.4 Axis/spindle-specific signals

5.4.1 Signals to axis/spindle

DB 31, ... **DBX1.3** Data block	**Axis/spindle disable** Signal(s) to axis/spindle (PLC → NC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ⟶ 1 – Axis disable	(Test conditions) If IS “Axis disable” is output, no more position partial setpoints are output for this axis to the position controller (see Fig. 5–1); the axis travel is therefore disabled. The position control loop remains closed and the remaining following error is reduced to zero. If an axis is traversed with axis disable the actual value position display shows the setpoint position and the velocity actual value display shows the setpoint velocity even though the machine axis is not actually traversing. A RESET sets the position actual value display to the real actual value of the machine. Travel commands continue to be output to the PLC for this axis. If the interface signal is canceled the associated axis can again traverse normally. If the interface signal “Axis disable” is set for a traversing axis, the axis is stopped with a ramp stop.	
– Spindle disable	If IS “Spindle disable” is set, as for axis disable, no more speed setpoints are output to the speed controller in openloop control mode and no more position partial setpoints are output to the position controller in positioning mode. The movement of the spindle is thus disabled. The speed actual value display displays the speed setpoint value. Spindle disable can only be canceled per “Reset” or with M2 followed by a program restart. If interface signal “Spindle disable” is set while a spindle is turning, the spindle is stopped according to its acceleration characteristic.	
Signal state 0 or signal transition 1 ⟶ 0	(Normal conditions) The position setpoint values are transferred to the position controller cyclically. The speed setpoint values are transferred to the speed controller cyclically. Cancellation of the “Axis/spindle disable” signal (edge change 1 → 0) does not take effect until the axis/spindle is stationary (i.e. no interpolation setpoint is applied).	
Fig. 5–1		
Application example(s)	The interface signal “Axis disable” and “Spindle disable” is used when a new NC program is being tested and the machine axes and spindles are not to carry out any traversing or rotational movements.	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX1.3** Data block	**Axis/spindle disable** Signal(s) to axis/spindle (PLC → NC)
Special cases, errors, ...	If the signal “Axis/spindle disable” is active for an axis/spindle, interface signals “Controller enable”, “Feedrate/spindle hold” and where applicable “Hardware limit switch” do not affect axis/spindle braking. The axis/spindle can however be brought into the “follow-up” or “hold” state (see interface signal IS “Follow-up mode”). **Notes:** • This signal inhibits setpoint output to the drive. • A brief pulse can bring a traversing axis to a standstill. The axis will not move again in this block, but only when the next block is reached. • Resynchronization takes place automatically on the next traversing command for this axis, i.e. the axis traverses the remaining distance-to-go. • Example: G0 X0 Y0 , G1 F1000 X100 ; At X20 the “Disable axis” signal is applied briefly, ; X axis stops, NC keeps interpolating Y100 ; X200 ; X traverses from approx. 20 to 200 ... **The following behavior applies in association with synchronous mode (SW 4 and higher):** **References:** /FB/, S3, “Synchronous Spindle” • This signal is no longer effective when the coupling for FS/FA is activated. → No. 6 • If the signal for the LS/LA is set, it also applies to the following spindle(s)/axis(es) → No. 7 • A workpiece clamped between two spindles (workpiece transfer from front to rearside machining) cannot be destroyed. No. |enabled: 1, disabled: 0 | LS/LA | FS/FA |coupling | |behavior 1 | 0 0 | off |setpoints of axes are output 2 | 0 1 | off |no setpoint output for FS/FA 3 | 1 0 | off |no setpoint output for LS/LA 4 | 1 1 | off |no setpoint output for LS/LA and FS/FA 5 | 0 0 | on |setpoints of axes are output 6 | 0 1 | on |Disable not effective for FS/FA 7 | 1 0 | on |Disable also effective for FS/FA 8 | 1 1 | on |no setpoint output for LS/LA and FS/FA
Related to	IS “Program test active” (DB21, ... DBX33.7)

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX1.4** Data block	**Follow-up mode** Signal(s) to axis/spindle (PLC → NC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1	Follow-up mode is selected for the axis/spindle by the PLC. The position setpoint is thus continuously corrected to the actual value if servo enable has been canceled for the drive. As soon as follow-up mode is active, IS “Follow-up mode active” (DB31, ... DBX61.3) are set to 1. The actual value continues to be acquired and updated. If the axis/spindle is moved from its current position by external influences the zero speed and clamping monitoring do not issue an alarm. When the closed-loop control system is reactivated, a control-internal repositioning operation is performed (REPOSA: linear approach with all axes) to the last programmed position if a parts program is active. For more information please refer to Subsection 2.8.1.	
Signal state 0	Follow-up mode is not selected (hold). When “servo enable” is removed the previous position setpoint is maintained automatically. If the axis/spindle is moved out of position during this time a following error between the position setpoint and the position actual value occurs. This position difference is reduced to zero immediately by issuing “Servo enable” so that the previous setpoint position is restored. Then, all the other axis movements start from the setpoint position valid before “Servo enable” was removed. When the position control is switched on again a setpoint jump of the axis position might take place. Zero speed monitoring and clamping monitoring is still active. In order to switch off the zero speed monitoring, IS “Clamping active” should be set when an axis is clamped. IS “Follow-up mode active” is set to 0 in the “hold” condition.	
Application example(s)	1) In analog operation (FMNC only): The axis is to be traversed periodically with an external speed setpoint without the control losing the axis actual value. Subsequent referencing of the axis is then not necessary (see Subsection 2.8.1). 2) Spindle (if position controlled): When IS “Servo enable” is set for the spindle it is not usual for the previous setpoint position to be approached again. The IS “Follow-up mode” must be set to 1 signal for spindles. A setpoint jump when IS “Servo enable” is set is thus avoided.	
Special cases, errors, ...	If servo enable for the drive is canceled internally because of a fault the following must be noted: After the triggered alarms have been successfully canceled (i.e. servo enable is issued again internally), “Hold” is activated (IS “Follow-up mode” = 0) before NC Start. Otherwise the travel path of the previous NC block is not executed on NC Start with follow-up mode selected because of the internal Delete distance-to-go. **Caution:** When changing over from the “Follow-up” state to the “Hold” state and in the servo mode (with release of the servo enable signal), a Delete distance-to-go command is activated in the control. This has the effect, for example, of an NC block in which only this access is to travel being ended directly.	
Related to	IS “Servo enable” (DB31, ... DBX2.1) IS ”Clamping active” (DB31, ... DBX2.3) IS “Follow-up mode active” (DB31, ... DBX61.3)	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX1.5 and 1.6** Data block	**Position measuring system 1 (PMS1)** **Position measuring system 2 (PMS2)** Signal(s) to axis/spindle (PLC → NC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
PMS1: Signal state 1 PMS2: Signal state 0	Position measuring system 1 is used for the axis/spindle (e.g. for position control, absolute value calculation, display). If a 2nd position measuring system is installed (MD 30200: NUM_ENCS = 2), this actual value is also acquired.	
PMS1: Signal state 0 PMS2: Signal state 1	Position measuring system 2 is used for the axis/spindle (e.g. for position control, absolute value calculation, display). If a position measuring system 1 also exists, this actual value is also acquired.	
PMS1: Signal state 1 PMS2: Signal state 1	As it is not possible to use both position measuring systems simultaneously for the position control of an axis/spindle, the control automatically selects position measuring system 1. If a position measuring system 2 also exists, this actual value is also acquired.	
PMS1: Signal state 0 PMS2: Signal state 0	1) The axis is in the PARK position. The following applies: • Both position measuring systems are inactive. • There is no actual value acquisition. • The position measuring system monitoring is switched off (e.g. cable connection of encoder). • The reference point has no effect (interface signal “Referenced/synchronized” has the signal condition 0). As soon as an axis is parked, IS “Position controller active” (DB31, ... IS “Speed controller active” (DB31, ... DBX61.6) and “Current control loop active” (DB31, ...DBX61.7) are set to 0. After parking the axis must be rereferenced (reference point approach). If interface signals PMS1 and PMS2 are set to 0 while the axis is moving, the axis is stopped with ramp stop without removing servo enable internally. This is advisable in the following cases: • Spindle encoder stops outputting above a certain speed (supplies no further pulses) • Spindle encoder is decoupled mechanically because it would not be able to cope with the speed. The spindle can then continue to run in speed-controlled mode. In order to really bring the axis/spindle to a stop, the servo enable must always be removed additionally by the PLC. 2) The spindle does not have a position measuring system and is only speed controlled. In this case IS “Servo enable” must be set to 1.	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX1.5 and 1.6** Data block	**Position measuring system 1 (PMS1)** **Position measuring system 2 (PMS2)** Signal(s) to axis/spindle (PLC → NC)
Application example(s)	1) Switching from position measuring system 1 to position measuring system 2 (and vice versa): If the axis in both position measuring systems was referenced and the limit frequency of the motor sensor has not been exceed in the meantime (i. e. IS “Referenced/ synchronized 1 and 2” (DB31, ... DBX60.4 and 60.5) has signal status 1), another reference point approach is not necessary after switchover. On switchover the actual difference between position measuring system 1 and 2 is traversed immediately. MD 36500: ENC_CHANGE_TOL (maximum tolerance on position actual value switchover) in which the difference between the two actual values existing on switchover can lie. If the actual value difference is greater than the tolerance, a switchover between the two systems does not take place and alarm 25100 “Measuring system switchover” not possible is triggered. For further information see Description of Functions G2, Velocities, Setpoint/ActualValue Systems, Closed-Loop Control. 2) Parking axis (i.e. no PMS is active): If the encoder has to be removed – e.g. if a rotary table has to be dismantled from the machine – the position measuring system monitoring is switched off in the parking position. The mounted axis/spindle encoder turns so quickly in certain applications that it can no longer maintain its electrical characteristics (edge gradient, etc.). 3) Switching off the measuring system: When measuring system 1 or 2 is switched off the associated interface signal “Referenced/synchronized 1 (or 2)” is reset. 4) Reference point approach: Reference point approach of the axis is executed with the selected position measuring system. Each PMS must be referenced separately.
Special cases, errors, ...	If the state “Parking axis is active”, the interface signal “Referenced/synchronized” for this axis is ignored on NC Start.
Related to	IS “Referenced/synchronized” 1 (DB31, ... DBX60.4) IS “Referenced/synchronized” 2 (DB31, ... DBX60.5) IS “Speed controller active” (DB31, ... DBX61.6) IS “Servo enable” (DB31, ... DBX2.1) MD 36500: ENC_CHANGE_TOL (Max. tolerance on position actual value switchover) MD 30200: NUM_ENCS (Number of encoders)
References	/FB/, G2, “Velocities, Setpoint/ActualValue Systems, Closed-Loop Control”

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX2.1** Data block	**Servo enable** Signal(s) to axis/spindle (PLC → NC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1	The position control loop of the axis/spindle is closed; the axis/spindle is in closed-loop control.		
Signal state 1 or signal transition 0 ——> 1	When “Servo enable” is set by the PLC user program: • Position control loop of axis is closed. • Position actual value is no longer switched to the position setpoint. • The servo enable of the drive is output. • IS “Position controller active” is set to 1. When “Servo enable” has been signaled no new actual value synchronization of the axis (reference point approach) of the axis is necessary if the maximum permissible limit frequency of the axis measuring system has not been exceeded during follow-up mode. As a function of the the start of IS “Follow-up mode” (DB31, ... As a function of the state of IS Follow-up mode (DB31, ... DBX1.4), it is possible to select whether or not the axis first traverses back to the earlier setpoint position (i.e. the positional deviation caused by the clamping process is eliminated again) (see IS “Follow-up mode”).		
Signal transition 1 ——> 0 signal state 0	“Servo enable” is removed: The procedure for removing “Servo enable” depends on whether the axis/spindle or an axis of the geometry grouping is stationary or traversing at this point in time (Fig. 2-5). • Axis/spindle stationary: – Position control loop of axis is opened. – When IS Follow-up mode = 1 the position actual value is switched to the position setpoint (i.e. the setpoint position is corrected to the actual value position). The position actual value of the axis/spindle continues to be acquired by the control. – The servo enable of the drive is removed. – The interface signals “Position controller active”, “Speed controller active” and “Current controller active” are set to 0. • Axis/spindle traverses: – The axis is stopped with rapid stop. – Alarm 21612 “VDI signal servo enable reset during movement” is triggered. – The position control loop of the axis/spindle is opened. – At the end of braking the position actual value is switched to the position setpoint independently of IS “Follow-up mode” (i.e. the setpoint position is corrected to the actual value position). The position actual value of the axis/spindle continues to be acquired by the control. IS “Follow-up mode” is set. – The interface signals “Position controller active”, “Speed controller active” and “Current controller active” are set to 0. The axis status cannot be changed again until after RESET.		
Application example(s)	• Using servo enable when clamping the axis: The axis is positioned to the clamping position. As soon as it has stopped it is clamped and then servo enable is removed. Servo enable is removed because the axis could be mechanically pressed out of position slightly by clamping and the position controller would continuously have to work against the clamping. When clamping is to be stopped, first servo enable is set again and then the axis is freed from clamping.		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX2.1** Data block	**Servo enable** Signal(s) to axis/spindle (PLC → NC)
Special cases, errors, ...	• If an attempt is made to traverse the axis/spindle without servo enable, the axis/spindle remains stationary but sends a travel command to the PLC (axis only). The travel command is maintained and is executed when the servo enable is activated. • If the servo enable of a traversing geometry axis is removed the programmed contour cannot be maintained. • Servo control is removed internally if certain faults occur on the machine, the position measuring system or in the control. (For more information see Subsection 2.8.1).
Related to	IS “Follow-up mode active” (DB31, ... DBX61.3) IS “Follow-up mode” (DB31, ... DBX1.4) IS “Position controller active” (DB31, ... DBX61.5) IS “Speed controller active” (DB31, ... DBX61.6) IS “Current control loop active” (DB31, ... DBX61.7) MD 36620: SERVO_DISABLE_DELAY_TIME (Switchoff delay servo enable) MD 36610: AX_EMERGENCY_STOP_TIME (Time for braking ramp when an error occurs)

DB 31, ... **DBX2.2** Data block	**Delete distance-to-go (axis-specific)/Spindle reset** Signal(s) to axis/spindle (PLC → NC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Delete distance-to-go (axis-specific): Depending on the operating mode one of the following happens on “Delete distance-to-go axial”: • in JOG mode: If the interface signal is set for one axis (signal transition 0—>1), this axis is stopped with ramp stop and its distance-to-go deleted. Any remaining following error is not deleted. • AUTOMATIC and MDA: The rising edge of the interface signals only affects the axes which are not in the geometry grouping. These are stopped by ramp stop and the distance-to-go deleted. After this the next program block can be started. IS “Delete distance-to-go axial” is ignored by geometry axes. Note: IS “Delete distance-to-go” does not affect the dwell time in a program block with dwell time.	
Signal state 0 or signal transition 1 ——> 0	No effect	
Application example(s)	To terminate the travel movement because of an external signal (e.g. probe)	
Special cases, errors, ...	“Delete distance-to-go (axial)” • After the axes have been stopped with “Delete distance-to-go” the next program block is prepared with the new positions. The axes thus follow a different contour to the one originally defined in the parts program after a “Delete distance-to-go”. If G90 is programmed in the block after “Delete distance-to-go” it is at least possible to approach the programmed absolute position, whereas with G91 the position originally defined in the parts program is not reached.	
Related to	IS “Delete distance-to-go (channel-specific)” (DB21, ... DBX6.2)	
References	/FB/, S1, “Spindles” for spindle reset	

DB 31, ... **DBX9.0, 1, 2** Data block	**Controller parameter set switchover (request)** Requested parameter set Signal(s) to axis/spindle	
Edge evaluation: no	Signal(s) updated: On request	Signal(s) valid from SW: 4.1
Signal state 1 or signal transition 0 ——> 1	–	
Signal state 0 or signal transition 1 ——> 0	–	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX9.0, 1, 2** Data block	**Controller parameter set switchover (request)** Requested parameter set Signal(s) to axis/spindle
Signal irrelevant for ...	MD 35590: $MA_PARAMSET_CHANGE_ENABLE = 0
Application example(s)	The binary coded index of the parameter set to be activated is stored in the 3 bits. 0 corresponds to 1st parameter set 1 corresponds to 2nd parameter set, etc. (max. 6 parameter sets possible)
Special cases, errors, ...	Indices 6–7 are imaged on parameter set 6.
Related to	DB31, ...DBX69.0, 1, 2

DB 31, ... **DBX9.3** Data block	**Disable parameter set switchover commands from NC** Disable Signal(s) to axis/spindle
Signal state 1 or signal transition 0 ——> 1	NC must not initiate any parameter set switchovers
Signal state 0 or signal transition 1 ——> 0	Parameter set switchover by the NC is enabled.
Related to	DB 31, ... DBX9.0 1, 2

DB 31, ... **DBX20.0** Data block	**Ramp-up times** Signal(s) to axis/spindle (PLC → 611D)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	V/Hz mode is activated in MD 1014: U/F_MODE_ENABLE. The time set in MD 1126: U/F_MODE_RAMP_TIME_2 is effective.	
Signal state 0 or signal transition 1 ——> 0	V/Hz mode is activated in MD 1014: U/F_MODE_ENABLE. The time set in MD 1125: U/F_MODE_RAMP_TIME_1 is effective.	
Signal irrelevant for ...	SINUMERIK FM-NC and SINUMERIK 840Di	
Related to	MD 1014: U/F_MODE_ENABLE (activate V/Hz mode) MD 1125: UF_MODE_RAMP_TIME_1 (ramp-up time 1 for V/Hz mode) MD 1126: UF_MODE_RAMP_TIME_2 (ramp-up time 2 for V/Hz mode)	
References	/FBA/, SIMODRIVE 611D Description of Functions, Drives	

DB 31, ... **DBX20.1** Data block	**Ramp function generator rapid stop** Signal(s) to axis/spindle (PLC → 611D)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	A rapid stop is triggered by the PLC for the drive (611D). Speed setpoint 0 is defined. The drive is stopped without a ramp function (regenerative braking). As soon as the rapid stop is recognized by the drive module, IS “Ramp function generator rapid stop active” (DB31, ... DBX92.1) is returned to the PLC.	
Signal state 0 or signal transition 1 ——> 0	No rapid stop is requested by the PLC for the drive.	
Signal irrelevant for ...	SINUMERIK FM-NC and SINUMERIK 840Di	
Related to	IS “Ramp function generator rapid stop active” (DB31, ... DBX92.1)	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX20.2** Data block	**Torque limit 2** Signal(s) to axis/spindle (PLC → 611D)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Torque limit 2 is requested by the PLC for the axis/spindle. With the 611D two torque limit values can be set for each axis/spindle, where torque limit 2 refers to torque limit 1 (reduction factor). Torque limit 2 is selected via the interface. The limit value for each is defined with the drive parameters. As soon as “torque limit 2 is active” for a drive, the drive outputs IS “Torque limit 2 active” (DB31, ... DBX92.2).		
Signal state 0 or signal transition 1 ——> 0	Only torque limit 1 has been selected by the PLC.		
Signal irrelevant for ...	SINUMERIK FM-NC and SINUMERIKI 840Di		
Application example(s)	In order to reduce the strain on the mechanics and workpiece, the current torque limit can be reduced with torque limit 2 for certain machining operations.		
Related to	IS “Torque limit 2 active” (DB31, ... DBX92.2)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D		

DB 31, ... **DBX20.3** Data block	**Speed setpoint smoothing** Signal(s) to axis/spindle (PLC → 611D), or (PLC → 611U) on 840Di		
11Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The PLC requests a filter to smooth the speed setpoint value. In the drive module the interface signal only has an effect under the following conditions: • Speed setpoint filter 1 is active in the drive • and speed setpoint filter 1 has been configured as a lowpass filter (i.e. not as a bandstop filter). As soon as these conditions exist, whereby the smoothing of the speed setpoint is active, the 611D or 611U signals the IS “Speed setpoint smoothing active” to the PLC (DB31, ... DBX92.3).		
Signal state 0 or signal transition 1 ——> 0	No smoothing of the speed setpoint value is requested by the PLC.		
Signal irrelevant for ...	SINUMERIK FM-NC		
Application example(s)	With this interface signal, speed setpoint smoothing can be activated from the PLC user program for a spindle during speed control to achieve smooth torque coupling. Speed setpoint smoothing can be deactivated when the spindle is in positioning mode.		
Special cases, errors, ...			
Related to	IS “Speed setpoint smoothing active” (DB31, ... DBX92.3)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX21.0 to 21.2** Data block	**Drive parameter set selection A, B, C** Signal(s) to axis/spindle		
Edge evaluation: no	Signal(s) updated: cyclically		Signal(s) valid from SW: 1.1
Meaning	With the bit combinations A, B and C it is possible to select 8 different drive parameter sets for the digital drives SIMODRIVE 611D/611U/SINUMERIK 810D. The following assignment applies Drive parameter set C B A 1 0 0 0 2 0 0 1 3 0 1 0 4 0 1 1 5 1 0 0 6 1 0 1 7 1 1 0 8 1 1 1 The switchable drive parameters are organized as follows: • Current setpoint filter (lowpass, bandstop); for adaptation to the mechanics • Motor speed normalization • Speed controller parameters • Speed setpoint filter • Speed monitoring data As soon as the new drive parameter set is active, the drive informs the PLC by sending interface signals “Active drive parameter set” (DB31, ... DBX93.0,1 and 2).		
Signal irrelevant for ...	SINUMERIK FM-NC		
Application example(s)	Drive parameter switchover can be used for the following: • Gear switchover • Measuring circuit switchover		
Special cases, errors, ...	In principle it is possible to switch over drive parameter sets at any time. However, as torque jumps can occur when switching over speed controller parameters and motor speed normalization, parameters should only be switched over when the axis/spindle is stationary (especially when the axis is stationary).		
Related to	IS “Active drive parameter set” (DB31, ... DBX93.0,1 and 2)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAC/, SINUMERIK 810D Installation and StartUp Guide		

DB 31, ... **DBX21.3 and 21.4** Data block	**Motor selection A, B** Signal(s) to axis/spindle (PLC → 611D)		
Edge evaluation: no	Signal(s) updated: cyclically		Signal(s) valid from SW: 1.1
Meaning	The PLC can switch between 4 different motors or motor mode types with motor selection. The following assignment applies **Motor selection** **Application** **B** **A** Motor 1 Mode 1 0 0 Motor 2 Mode 2 0 1 Motor 3 Res. up to SW 6.3, then for 611D Performance 2 or 611U usable for mode 3 or 4 1 0 Motor 4 1 1 As soon as a new motor selection is detected, the drive cancels the pulse enable (IS checkback signal “active motor”). Using the motor selection, it is possible, for example, to choose mode 1 as star-connected operation and mode 2 as delta-connection operation for the main spindle drive (MSD). The drive returns the currently selected motor to the PLC via interface signals “active motor” (DB31, ... DBX93.3 and 4).		
Signal irrelevant for	SINUMERIK FM-NC, 810D		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX21.3 and 21.4** Data block	**Motor selection A, B** Signal(s) to axis/spindle (PLC → 611D)
Application example(s)	Timing for stardelta switchover
Special cases, errors, ...	Note: IS “Motor selection in progress” must be set to 0 before a new motor is selected!
Related to	IS “Active motor” (DB31, ... DBX93.3 and 4) IS “Motor selection in progress” (DB31, ... DBX21.5)
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D

DB 31, ... **DBX21.5** Data block	**Motor selection in progress** Signal(s) to axis/spindle (PLC → 611D)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The PLC transmits IS “Motor selection in progress” to the 611D to confirm that the external contactor has been switched over to the new motor (e.g. that motor contactor 2 has been switched over as part of the star/delta switchover process). The pulses are then enabled by the drive.	
Signal state 0 or signal transition 1 ——> 0	IS “Motor selection in progress” must be reset to 0 by the PLC user program before a new motor is selected! Otherwise the pulses from the drive might be enabled too early.	
Signal irrelevant for ...	SINUMERIK FM-NC, 810D	
Related to	IS “Active motor” (DB31, ... DBX93.3 and 4) IS “Motor selection in progress A, B” DBX21.3 and 4)	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D	

DB 31, ... **DBX21.6** Data block	**Speed controller integrator disable** Signal(s) to axis/spindle (PLC → 611D) or (PLC → 611U)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The interface signal is used by the 611D/611U to disable the integrator of the speed controller. The speed controller is thus switched from PI to P controller. Note: If the speed controller integrator disable is activated equalization process might occur in certain applications (e.g. if the integrator was already holding a load while stationary). The 611D/611U acknowledges the integrator disable to the PLC with the IS “Speed controller integrator disabled” (DB31, ... DBX93.6).	
Signal state 0 or signal transition 1 ——> 0	The integrator of the speed controller is enabled.	
Signal irrelevant for ...	SINUMERIK FM-NC, 810D	
Related to	IS “Speed controller integrator disabled” (DB31, ... DBX93.6)	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D	

DB 31, ... **DBX21.7** Data block	**Pulse enable** Signal(s) to axis/spindle (PLC → 611D) or (PLC → 611U)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Pulse enable is signaled by the PLC for this drive (axis/spindle). The pulses for the drive modules are enabled only if all enable signals (hardware and software) are available (see Fig. for DB31, ... DBX93.5). In this case, interface signal “Pulses enabled” (DB31, ... DBX93.7) set to 1 is transferred to the PLC. For further information, see IS “Pulses enabled” (DB31, ... DBX93.7) or References.	
Signal state 0 or signal transition 1 ——> 0	The pulses are disabled by the PLC for this drive. If pulse enable is canceled for a moving axis/spindle the axis/spindle is not longer braked under control. The axis coasts to rest.	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX21.7** Data block	**Pulse enable** Signal(s) to axis/spindle (PLC → 611D) or (PLC → 611U)
Signal irrelevant for ...	SINUMERIK FM-NC
Application example(s)	Signal relevant to safety
Special cases, errors, ...	If pulse enable is canceled for a moving axis/spindle on EMERGENCY STOP, the axis/spindle only coasts to rest.
Related to	IS “Pulses enabled” (DB31, ... DBX93.7)
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAC/, SINUMERIK 810D Installation and StartUp Guide

DB 31, ... **DBX1.0** Data block	**Drive test travel enable** Signal(s) from axis/spindle (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Safety handshake when function generator is started by the NC. If an axis is to be moved without further operator intervention, the NC outputs IS “Drive test travel request” (DB31, ... DBX61.0) = 1 signal. to request a travel enable from the PLC. If all axis travel conditions are fulfilled, the PLC acknowledges the ready state with IS “Drive test travel enable” (DB31, ... DBX1.0) = 1 signal The PLC always has priority in deciding whether an axis can be moved.	
Signal state 0 or signal transition 1 ——> 0	The NC does not request permission to move axis from the PLC with IS “Drive test travel enable” (DB31, ... DBX1.0) = 0 signal.	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAC/, SINUMERIK 810D Installation and StartUp Guide	

© Siemens AG, 2003. All rights reserved

5.4.2 Signals from axis/spindle

DB 31, ... DBX61.0	**Drive test travel request**	
Data block	Signal(s) from axis/spindle (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The control signals that all travel conditions for the axis are fulfilled. The following conditions must be fulfilled: 1. The mechanical brake of the appropriate axis has already been released and all other axis travel conditions are fulfilled. With IS “Drive test travel request” (DB31, ... DBX61.0) = 1 signal the according axes can be moved. 2. The axis disabling IS “Axis/spindle disable” = 1 signal is not active.	
Signal state 0 or signal transition 1 ——> 0	The control signals that the axes cannot be moved. Axes cannot be moved when: IS “Drive test travel request” (DB31, ... DBX61.0) = 0 signal or internally in the control with faults. The necessary conditions mentioned above are not fulfilled.	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAC/, SINUMERIK 810D Installation and StartUp Guide	

DB 31, ... DBX61.3	**Follow-up mode active**	
Data block	Signal(s) from axis/spindle (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The control signals that follow-up mode is active for the axis/spindle. The following conditions must be fulfilled: 1. Servo enable has been canceled for the drive (either by the PLC with IS “Servo enable” = 0 or internally in the case of a fault, see References) **and** 2. Follow-up mode is selected (either by the PLC with IS “Follow-up mode = 1 or internally e.g. when the servo enable for a traversing axis is canceled). The position setpoint is constantly corrected to the actual value while follow-up mode is active. Zero speed and clamping monitoring **are not active**.	
Signal state 0 or signal transition 1 ——> 0	The control signals that follow-up mode for the axis/spindle is not active. Zero speed and clamping monitoring **are active**. The necessary conditions mentioned above are not fulfilled. IS “Follow-up mode” active is reset during the “Hold” state.	
Special cases, errors, ...	Note: A delete distance-to-go is triggered internally on transition from “Follow up” to “Hold” (IS “Follow-up mode” = 0) or in control mode (IS “Servo enable” = 1).	
Related to	IS “Servo enable” (DB31, ... DBX2.1) IS “Follow-up mode” (DB31, ... DBX1.4)	
References	/DA/, “Diagnostics Guide”	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX61.4** Data block	**Axis/spindle stationary** Signal(s) from axis/spindle (NCK → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The current velocity of the axis or actual speed of the spindle is below the limit defined in MD 36060: STANDSTILL_VELO_TOL (max. velocity/speed for the signal “Axis/spindle stationary”).		
Signal state 0 or signal transition 1 ——> 0	The current velocity of the axis or actual speed of the spindle is greater than the value specified in the MD (standstill range). If a traversing command is active, e.g. for a spindle, then the signal is always 0, even if the current speed is lower than the setting in MD 36060: STANDSTILL_VELO_TOL. If the IS “Axis/spindle stationary” is signaled and no position control is active for the spindle, zero is displayed for the actual speed on the MMC and zero is read with the system variable $AA_S[n].		
Application example(s)	• Enable signal for opening a protective device (e.g. open door). • Open the workpiece chuck or the tool clamping device only when the spindle is stationary. • Oscillation mode can be switched on during gear stage change once the spindle has been stopped. • The tool clamping device must have been closed before the spindle can be accelerated up to maximum speed.		
Related to	MD 36060: STANDSTILL_VELO_TOL (maximum velocity/speed for signal “Axis/spindle stationary”)		

DB 31, ... **DBX61.5** Data block	**Position controller active** Signal(s) from axis/spindle (NCK → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The control signals that the position controller for the axis or spindle is closed.		
Signal state 0 or signal transition 1 ——> 0	The control signals that the position controller for the axis or spindle is open. If “Servo enable” is canceled because of a fault or by the PLC user program the position controller is opened and signal “Position controller active” reset to 0. Spindle without position control: Signal “Position controller active” is always “0”. See References for other effects.		
Application example(s)	• If position control is active the axis/spindle is kept in position by the position controller. Any brakes or clamps can thus be opened. • IS “Position controller active” can be used as a return signal for IS “Servo enable”. • The holding brake of a vertical axis must be activated as soon as the position control is no longer active. • Provided a spindle has the appropriate technical capability, it can be switched over to position-controlled mode as an axis or spindle in the parts program (with SPCON or M70). In the above cases interface signal “Position controller active” is set.		
Special cases, errors, ...	Special case for simulation axes (MD 30350: SIMU_AX_VDI_OUTPUT = “1”: Interface signal “Position controller active” is also set for simultaneous axes as soon as MD = “1”.		
Related to	IS “Servo enable” (DB31, ... DBX2.1) IS “Follow-up mode” (DB31, ... DBX1.4) IS “Position measurement system 1” and “Position measurement system 2” (DB31, ... DBX1.5 and 6)		
References	/DA/, “Diagnostics Guide“		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX61.6** Data block	**Speed controller active** Signal(s) from axis/spindle (NCK → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The control signals that the speed controller is closed for the axis or spindle.		
Signal state 0 or signal transition 1 ——> 0	The control signals that the speed controller is open for the axis or spindle. The speed controller output is cleared.		
Application example(s)	If the spindle is not under position control the interface signal can be used as a return signal for IS “Servo enable”.		
Special cases, errors, ...	Special case for simulation axes (MD 30350: SIMU_AX_VDI_OUTPUT = “1”): Interface signal “Speed controller active” is also set for simulation axes as soon as MD 30350: SIMU_AX_VDI_OUTPUT (output of axis signals for simulation axes) = “1”.		
Related to	IS “Position controller active” (DB31, ... DBX61.5)		

DB 31, ... **DBX61.7** Data block	**Current controller active** Signal(s) from axis/spindle (NCK → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The control signals that the current controller is closed for the axis or spindle.		
Signal state 0 or signal transition 1 ——> 0	The control signals that the current controller is open for the axis or spindle. The current controller output (including the feedforward quantities on the manipulated variable for the voltage) is cleared.		
Related to	IS “Position controller active” (DB31, ... DBX61.5) IS “Speed controller active” (DB31, ... DBX61.6)		

DB 31, ... **DBX69.0, 1, 2** Data block	**Controller parameter set switchover (checkback signal)** Active parameter set Signal(s) from axis/spindle (NCK → PLC)		
Edge evaluation: no		Signal(s) updated: After Switchover	Signal(s) valid from SW: 4.1
Signal state 1 or signal transition 0 ——> 1	–		
Signal state 0 or signal transition 1 ——> 0	–		
Signal irrelevant for ...	MD 35590: $MA_PARAMSET_CHANGE_ENABLE = 0		
Application example(s)	The binary coded index of the activated parameter set is stored in the 3 bits. 0 corresponds to 1st parameter set 1 corresponds to 2nd parameter set, etc. (max. 6 parameter sets possible)		
Special cases, errors, ...	Index 0 is returned is the switchover function is deactivated in MD 35590: PARASET_CHANGE_ENABLE = 0. In this case, the 1st parameter set is always active.		
Related to	DB31, ...DBX9.0, 1, 2		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX76.0** Data block	**Lubrication pulse** Signal(s) from axis/spindle (NCK → PLC)		
Edge evaluation: yes		Signal(s) updated: when change	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	If the axis has traversed a greater distance than set in MD 33050: LUBRICATION_DIST (travel distance for lubrication from PLC) the IS “Lubrication pulse” is set for one PLC cycle.		
Signal state 0 or signal transition 1 ——> 0	IS “Lubrication pulse” is automatically reset to 0 after one PLC cycle.		
Application example(s)	The lubrication pump for the axis can be activated with IS “Lubrication pulse”. Bed lubrication therefore always depends on the path traveled.		
Related to	MD 33050: LUBRICATION_DIST (lubrication pulse distance)		

DB 31, ... **DBX92.0** Data block	**Setup mode active** Signal(s) from axis/spindle (611D → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Setup mode is active for the drive (611D). Setup mode is selected via the terminals on the infeed/regenerative feedback module. Setup mode is required for optimizing the machining process. The following adaptations and additional functions are possible: • Drive: Reduction of speed setpoint limits Reduction of current setpoint limits • I/R: Switching off DC link voltage control		
Signal state 0 or signal transition 1 ——> 0	Normal operation is active for the drive: The following thus applies: • The maximum limit values for speed and current setpoint are active • DC link voltage control is active		
Signal irrelevant for ...	SINUMERIK FM-NC, 840Di, 810D		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D		

DB 31, ... **DBX92.1** Data block	**Ramp function generator rapid stop active** Signal(s) from axis/spindle (611D → PLC) or (611U → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	PLC receives the signal that the ramp function generator rapid stop is active. The function has been activated by IS “Ramp function generator rapid stop” (DB31, ... DBX20.1). The drive is stopped without a ramp function with speed setpoint = 0 and without pulse suppression.		
Signal state 0 or signal transition 1 ——> 0	Ramp function generator rapid stop is not active for the drive.		
Signal irrelevant for ...	SINUMERIK FM-NC, 810D		
Application example(s)	Avoiding the use of the ramp function generator on the servo side		
Related to	IS “Ramp function generator rapid stop” (DB31, ... DBX20.1)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX92.2** Data block	**Torque limit 2 active** Signal(s) from axis/spindle (611D → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically		Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Drive (611D) sends acknowledgment to the PLC that torque limit 2 is active in addition to torque limit 1. The limit value for each is defined with the drive parameters.		
Signal state 0 or signal transition 1 ——> 0	Only torque limit 1 is active.		
Signal irrelevant for ...	SINUMERIK FM-NC, 840Di, 810D		
Related to	IS “Torque limit 2” (DB31, ... DBX20.2)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D		

DB 31, ... **DBX92.3** Data block	**Speed setpoint smoothing active** Signal(s) from axis/spindle (611D → PLC) or (611U → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically		Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Speed setpoint smoothing requested by the PLC with IS “Speed setpoint smoothing” (DB31, ... DBX20.3) is active.		
Signal state 0 or signal transition 1 ——> 0	No speed setpoint smoothing is active.		
Signal irrelevant for ...	SINUMERIK FM-NC, 810D		
Related to	IS “Speed setpoint smoothing” (DB31, ... DBX20.3)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D		

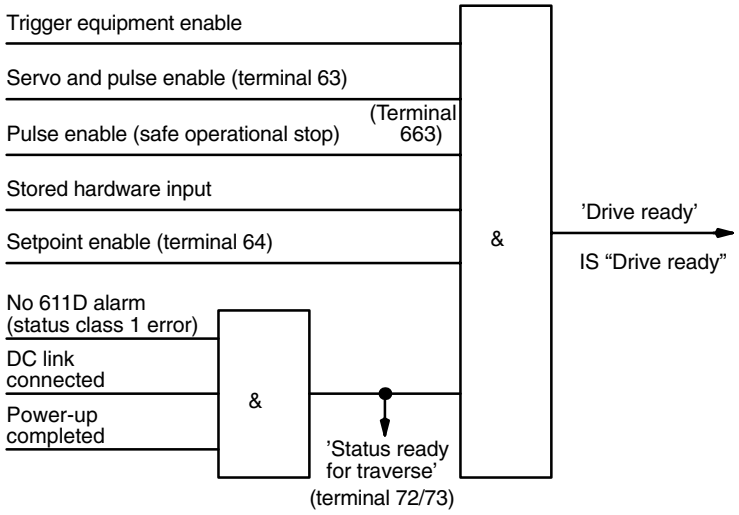
DB 31, ... **DBX93.0 93.2** Data block	**Active drive parameter set A, B, C** Signal(s) from axis/spindle (611D → PLC) or (611U → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically		Signal(s) valid from SW: 1.1
Meaning	The drive module (611D/611U) signals back to the PLC which drive parameter set is currently active. With bit combinations A, B and C, 8 different drive parameter sets can be selected for the 611D. The following assignment applies		
Signal irrelevant for ...	SINUMERIK FM-NC		
Application example(s)	Drive parameter switchover can be used for the following: • Gear switchover • Measuring circuit switchover		
Related to	IS “Drive parameter set selection” (DB31, ... DBX21.0,1 and 2)		
References	IS “Drive parameter set selection” (DB31, ... DBX21.0,1 and 2) or /IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide		

Active drive parameter set	C	B	A
1	0	0	0
2	0	0	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX93.3 and 93.4** Data block	**Active motor A, B** Signal(s) from axis/spindle (611D → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Meaning	The drive module (611D) returns signals to PLC stating which motor selection is currently active. The following assignment applies Active motor / Application / B / A Motor 1 / MSD: Star mode / 0 / 0 Motor 2 / MSD: Delta mode / 0 / 1 Motor 3 / reserved / 1 / 0 Motor 4 / reserved / 1 / 1 Motor selection can be used with a main spindle drive (MSD) to switch between star and delta operation and so reduce the starting current.		
Signal irrelevant for ...	SINUMERIK FM-NC, 810D		
Related to	IS “Motor selection” (DB31, ... DBX21.3 and 4) IS “Motor selection in progress” (DB31, ... DBX21.5)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D		

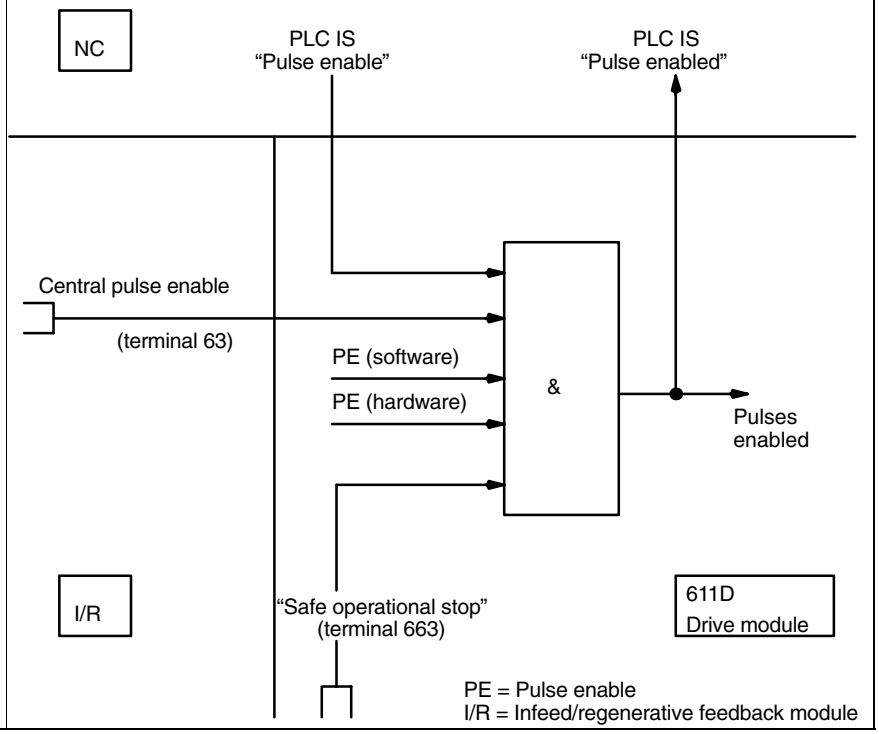
© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX93.5** Data block	**Drive Ready** Signal(s) from axis/spindle (611D → PLC) or (611U → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	Acknowledgment from the drive to the PLC that the drive is ready.		
Signal state 0 or signal transition 1 ——> 0	The drive is not ready. The drive might be disabled for the following reasons (see Fig. 5-3): • Enable terminals missing (e.g. terminal 63 “Servo and pulse enable”; terminal 663 “Safe operational stop”, terminal 64 “Setpoint enable”) • Drive alarm active (e.g. motor temperature has reached switchoff threshold) • DC link voltage is too low • Drive has not yet reached cyclic operation • Hardware fault • No position measuring system is active (“parking axis” state) • I/R is not switched on As soon as the drive is ready for operation it is stopped (depending on fault either with pulse disable or rapid stop) or pulse disable is maintained during ramp up. The interface signals “611D Ready” (DB10, DBX108.6), “Current controller active” and “Speed controller active” are similarly removed. “611D Ready” is not available on the 840Di in conjunction with the drive 611U.		
Signal irrelevant for ...	SINUMERIK FM-NC		
Fig. 5–2			
Related to	IS “611D Ready” (DB10, DBX108.6) IS “Current controller active” (DB31, ... DBX61.7) IS “Speed controller active” (DB31, ... DBX61. 6)		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide		

DB 31, ... **DBX93.6** Data block	**Speed controller integrator disabled** Signal(s) from axis/spindle (611D → PLC) or (611U → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	Deactivation of the integrator by the speed controller requested by the PLC with IS “Speed controller integrator disabled” (DB31, ... DBX21.6) is active for the drive module. The speed controller has therefore switched from a PI to a P controller.		
Signal state 0 or signal transition 1 ——> 0	The integrator of the speed controller is enabled. The speed controller functions as a PI controller.		
Signal irrelevant for ...	SINUMERIK FM-NC, 810D		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX93.6** Data block	**Speed controller integrator disabled** Signal(s) from axis/spindle (611D → PLC) or (611U → PLC)
Related to	IS "Speed controller integrator disabled" (DB31, ... DBX21.6)
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D

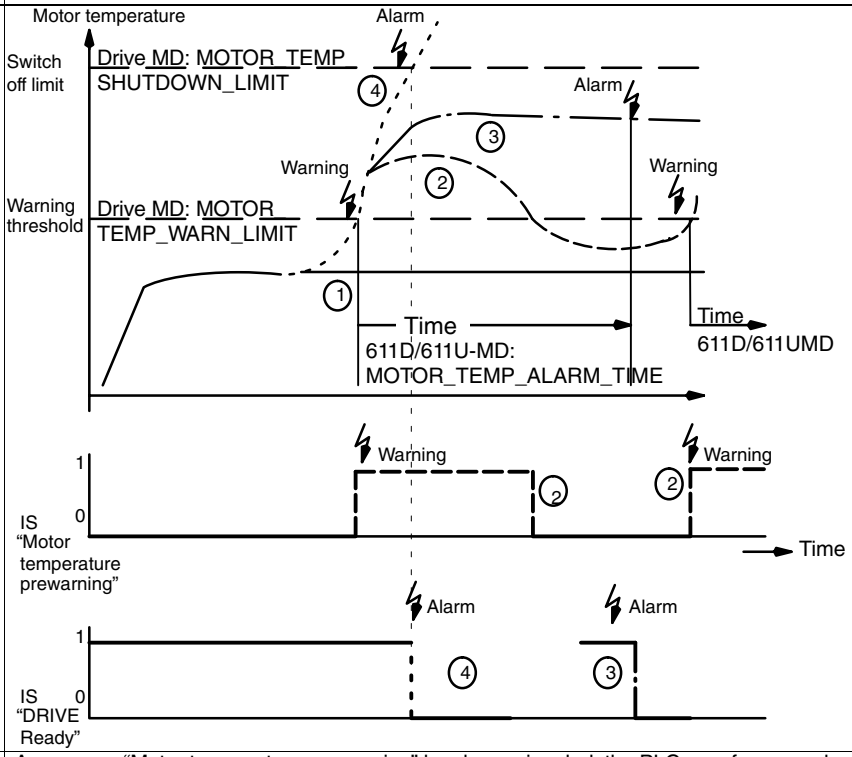
DB 31, ... **DBX93.7** Data block	**Pulses enabled** Signal(s) from axis/spindle (drive → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Drive pulse enable for drive module available. The axis/spindle can now be traversed.	
Signal state 0 or signal transition 1 ——> 0	The drive module pulses are suppressed. The axis/drive cannot therefore be traversed. The pulses are suppressed as soon as there is no enable signal (see Fig. 5-4). Also, if the "Servo enable of drive" is removed from the servo, the drive is stopped with setpoint 0 (regenerative braking). In this case, a timer is started in the drive module followed by pulse disabling when the programmed time period expires (MD 1404: PULSE_SUPPRESSION_DELAY (timer pulse suppression)). If the actual speed reaches creep speed within this time period (MD 1403: PULSE_SUPPRESSION_SPEED (creep speed pulse suppression)) pulse suppression is triggered instantaneously. If the speed is lower/equal to the speed threshold (MD 1403: PULSE_SUPRES-SION_SPEED) and the drive servo enable signal canceled, pulse suppression is triggered immediately. Pulse suppression is also triggered if the servo reports that there is no position measuring system (Parking axis state). Pulse suppression is also triggered if the servo reports that there is no position measuring system (Parking axis state). As soon as the pulses are suppressed, interface signals "Current controller active" and "Speed controller active" are also reset.	
Signal irrelevant for ...	SINUMERIK FM-NC	
Fig. 5-3 Pulse enable for 611D drive module		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX93.7** Data block	**Pulses enabled** Signal(s) from axis/spindle (drive → PLC)
Related to	IS “Pulse enable” (DB31, ... DBX21.7) MD 1404: PULSE_SUPPRESSION_DELAY MD 1403: PULSE_SUPPRESSION_SPEED
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide

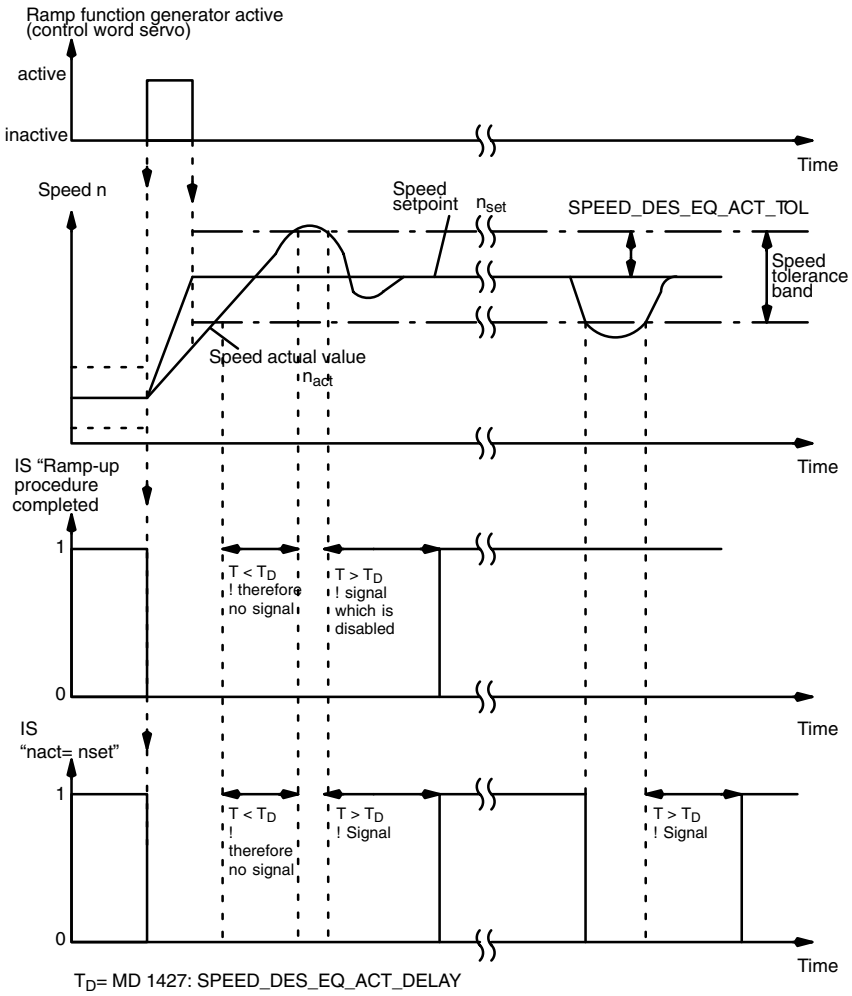
DB 31, ... **DBX94.0** Data block	**Motor temperature prewarning** Signal(s) from axis/spindle (drive → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The drive module sends “Motor temperature warning” to the PLC. In this case, the motor temperature has exceeded the defined prewarning threshold (MD 1602: MOTOR_TEMP_WARN_LIMIT (maximum motor temperature); default setting 120 °C) exceeded (see Fig. 5–5).[2]). If the motor temperature remains at this level the drive will be braked in generator mode after a defined time (drive MD 1603: MOTOR_TEMP_ALARM_TIME (timer motor temperature alarm; default 240s) and the pulses suppressed (see Fig. 5–5 [3]). In addition, alarm 300614 is activated and IS “Drive Ready” canceled. If the motor temperature continues to rise and the defined cutout threshold (MD 1607: MOTOR_TEMP_SHUTDOWN_LIMIT (cutout threshold motor temperature); default setting 155 °C) is reached, the drive is shut down immediately (see Fig. 5–5 [4]). An alarm is also output and IS “Drive Ready” canceled. However, if the motor temperature drops back down to the warning threshold early enough, the IS is reset to 0 (see Fig. 5–5 [2]). Special case: If no temperature sensor signal is measured this is interpreted as a fault in the motor PTC thermistor. In such a case IS “Motor temperature prewarning” is also set. Procedure continues as above.	
Signal state 0 or signal transition 1 ——> 0	The motor temperature is below the warning threshold. The current motor temperature is displayed in the axis/spindle service display in the operating area Diagnosis. The display corresponds to MD 1702: MOTOR_TEMPERATURE (motor temperature).	
Signal irrelevant for ...	SINUMERIK FM-NC	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX94.0** Data block	**Motor temperature prewarning** Signal(s) from axis/spindle (drive → PLC)
Fig. 5–4	
Application example(s)	As soon as "Motor temperature prewarning" has been signaled, the PLC can, for example, initiate controlled shutdown of the drives.
Related to	IS "DRIVE ready" (DB31, ... DBX93.5) MD 1602: MOTOR_TEMP_WARN_LIMIT MD 1603: MOTOR_TEMP_ALARM_TIME MD 1607: MOTOR_TEMP_SHUTDOWN_LIMIT
References	/DA/, "Diagnostics Guide" /IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide

DB 31, ... **DBX94.1** Data block	**Heatsink temperature prewarning** Signal(s) from axis/spindle (drive → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The drive module sends the warning "Heatsink temperature prewarning" to the PLC. This triggers the following: • Terminal 5 on the infeed/regenerative feedback module is activated immediately. • The drive module is switched off after 20 seconds. The drives are stopped when the impulse enable is removed. Then alarm 300515 is triggered.	
Signal state 0 or signal transition 1 ——> 0	The drive module heatsink temperature prewarning has not responded.	
Signal irrelevant for ...	SINUMERIK FM-NC	
Application example(s)	As soon as "Heatsink temperature warning" has been signaled, the PLC can, for example, initiate controlled shutdown of the drives.	
References	/DA/, "Diagnostics Guide"	

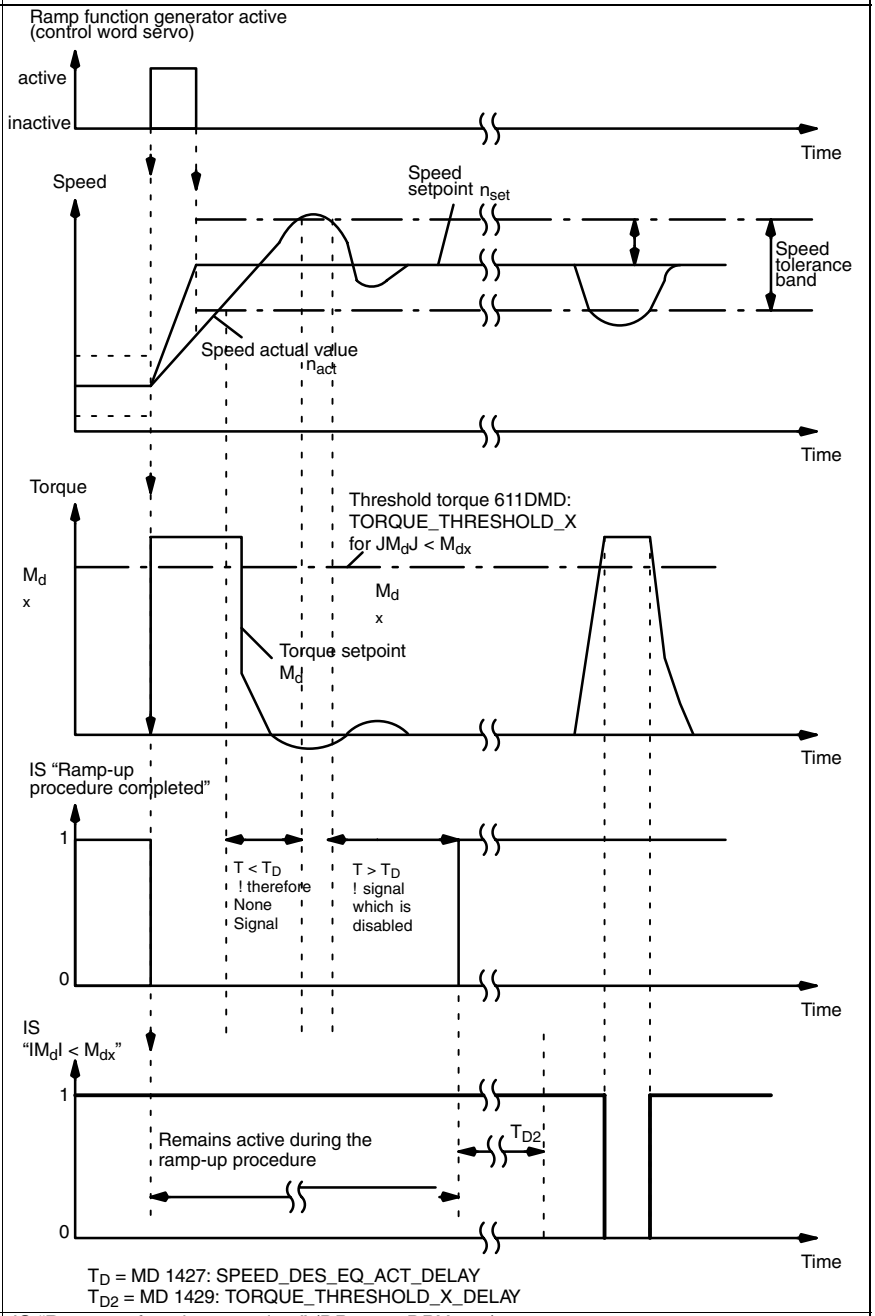
© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX94.2** Data block	**Ramp up function complete** Signal(s) from axis/spindle (drive → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The PLC receives confirmation that the actual speed value has reached the tolerance band (MD 1426: SPEED_DES_EQ_ACT_TOL (tolerance band for $n_{set} = n_{act}$ signal)) and has remained within this tolerance band for a time period corresponding to the setting in MD 1427: SPEED_DES_EQ_ACT_DELAY (delay time $n_{set} = n_{act}$ signal) (see Fig. 5–6). Even if the speed actual value leaves the tolerance band (because of speed fluctuations resulting from changes in load) the signal "Ramp up function complete" remains (1 signal).		
Signal state 0 or signal transition 1 ——> 0	The conditions described above have not yet been fulfilled. The ramp up function has therefore not yet been completed.		
Signal irrelevant for ...	SINUMERIK FM-NC		
Fig. 5–5			
Related to	IS "$n_{act} = n_{set}$" (DB31, ... DBX94.6) IS "$|M_D| = M_{dx}$" (DB31, ... DBX94.3) MD 1426: SPEED_DES_EQ_ACT_TOL MD 1427: SPEED_DES_EQ_ACT_DELAY		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX94.3** Data block	**$\|M_d\| < M_{dx}$** Signal(s) from axis/spindle (drive → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	611D reports to the PLC that the torque setpoint $\|M_d\|$ does not exceed the threshold torque M_{dx} in the stationary condition (i.e. rampup function completed) (see Fig. 5–7). The threshold torque is set in MD 1428: TORQUE_THRESHOLD_X (threshold torque) as a percentage of the current torque limit value. The torque threshold is speed-dependent. During ramp up, IS $\|M_d\| < M_{dx}$ remains at 1. The signal $\|M_d\| < M_{dx}$ becomes active as soon as the ramp up function is complete (IS "Rampup function complete"= 1) and the signal disable time for the torque threshold (MD 1429: TORQUE_THRESHOLD_X_DELAY (delay time $n_d < n_{dx}$ signal) has expired.	
Signal state 0 or signal transition 1 ——> 0	The torque setpoint $\|M_d\|$ is larger than the threshold torque M_{dx}. If necessary, the PLC user program starts a reaction.	
Signal irrelevant for ...	SINUMERIK FM-NC	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX94.3** Data block	**$|M_d| < M_{dx}$** Signal(s) from axis/spindle (drive → PLC)
Fig. 5–6	
Related to	IS "Ramp-up function complete" (DB31, ... DBX94.2) MD 1428: TORQUE_THRESHOLD_X MD 1429: TORQUE_THRESHOLD_X_DELAY MD 1427: SPEED_DES_EQ_ACT_DELAY
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide

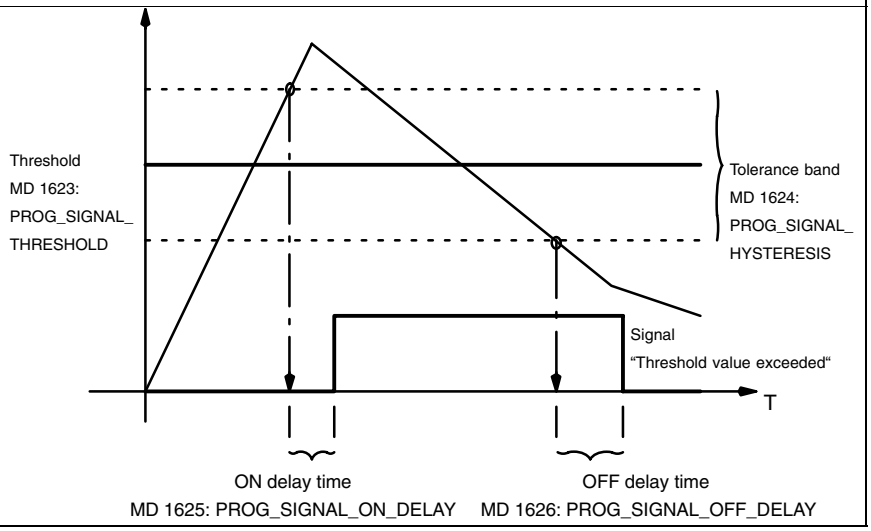
© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX94.4** Data block	**$\|n_{act}\| < n_{min}$** Signal(s) from axis/spindle (drive → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	It is signaled from SIMODRIVE 611D/ 611U to the PLC that the actual speed value n_{act} is lower than minimum speed value (n_{min}). The minimum speed is defined in MD 1418: SPEED_TESHOLD_MIN.		
Signal state 0 or signal transition 1 ——> 0	The speed actual value is higher than the minimum speed.		
Signal irrelevant for ...	SINUMERIK FM-NC		
Related to	MD 1418: SPEED_THRESHOLD_MIN (minimum speed value (n_{min} for $n_{act} < n_{min}$))		
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide		

DB 31, ... **DBX94.5** Data block	**$\|n_{act}\| < n_x$** Signal(s) from axis/spindle (drive → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	It is signaled from the 611D/ 611U to the PLC that the actual speed value n_{act} is lower than the threshold speed value (n_x). The threshold speed is defined in MD 1417: SPEED_ THRESHOLD_X.	
Signal state 0 or signal transition 1 ——> 0	The speed actual value is higher than the threshold speed.	
Signal irrelevant for ...	SINUMERIK FM-NC	
Related to	MD 1417: SPEED_THRESHOLD_MIN (minimum speed value (n_x for $n_{act} < n_x$))	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide	

DB 31, ... **DBX94.6** Data block	**$n_{act} = n_{set}$** Signal(s) from axis/spindle (drive → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	It is signaled from SIMODRIVE 611D/611U to the PLC that following a new speed value setting the actual speed value n_{act} has reached the speed tolerance band (MD 1426: SPEED_DES_EQ_ACT_TOL (tolerance band for $n_{set} = n_{act}$ signal) and has remained within it for at least the time period set in MD 1427: SPEED_DES_EQ_ACT_DELAY (delay time $n_{set} = n_{act}$ signal) (see Fig. 5-6). If the speed actual value then leaves the tolerance band, then IS "$n_{act} = n_{set}$" is reset to "0" instead of "Rampup function complete".	
Signal state 0 or signal transition 1 ——> 0	The conditions described above have not yet been fulfilled. The speed actual value is outside the speed tolerance band.	
Signal irrelevant for ...	SINUMERIK FM-NC	
see Fig. 56		
Related to	IS "Ramp-up function complete" (DB31, ... DBX94.2) MD 1426: SPEED_DES_EQ_ACT_TOL MD 1427: SPEED_DES_EQ_ACT_DELAY	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX94.7** Data block	**Variable signaling function** Signal(s) from axis/spindle (drive → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1	It is signaled from SIMODRIVE 611D/611U to the PLC that the threshold value has exceeded the value being monitored. With the variable signaling function it is possible to monitor the exceeding of a defined threshold for any programmable value of SIMODRIVE 611D/611U for each axis and to send a corresponding interface signal to the PLC. The parameters for the variables being monitored are set in the following machine data: • PROG_SIGNAL_FLAGS (bits variable signal function) • PROG_SIGNAL_NR (signal number variable signal function) • PROG_SIGNAL_ADDRESS (address variable signal function) • PROG_SIGNAL_THRESHOLD (threshold variable signal function) • PROG_SIGNAL_HYSTERESIS (hysteresis variable signal function) • PROG_SIGNAL_ON_DELAY (ON delay variable signal function) • PROG_SIGNAL_OFF_DELAY (OFF delay variable signal function) Monitoring: The parameterized variable is monitored to check whether it exceeds a defined threshold. In addition, a tolerance band (hysteresis) can be defined which is considered when scanning for violation of the threshold value. The signal 'Threshold exceeded' can be also be combined with an ON delay and OFF delay time (see Fig. 5–8). Selection: The variable to be monitored can be selected by entering a signal number or by entering a symbolic address. The MD 1620: PROG_SIGNAL_FLAGS (bits variable signal function). It is also possible to determine whether the threshold value comparison is to be signed or unsigned. For further information see References.		
Signal state 0	SIMODRIVE 611D reports to the PLC that the threshold value of the variable being monitored has not been exceeded or that the conditions defined in the above 611DMD have not been fulfilled. If the variable signaling function is deactivated (MD 1620: PROG_SIGNAL_FLAGS), the signal status "0" is output to the PLC.		
Signal irrelevant for ...	SINUMERIK FM-NC		
Fig. 5–7			

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX94.7** Data block	**Variable signaling function** Signal(s) from axis/spindle (drive → PLC)	
Application example(s)	With the variable signal function the machine tool manufacturer can monitor one additional threshold value for specific applications for each axis/spindle and evaluate the result in the PLC user program. Example: IS “Variable signal function” is to be set to 1 when the motor torque exceeds 50 % of the rated torque.	
Related to	MD 1620: PROG_SIGNAL_FLAGS MD 1621: PROG_SIGNAL_NR MD 1622: PROG_SIGNAL_ADDRESS MD 1623: PROG_SIGNAL_THRESHOLD MD 1624: PROG_SIGNAL_HYSTERESIS MD 1625: PROG_SIGNAL_ON_DELAY MD 1626: PROG_SIGNAL_OFF_DELAY	(bits variable signal function) (signal number variable signal function) (address variable signal function) (threshold variable signal function) (hysteresis variable signal function) (ON delay variable signal function) (OFF delay variable signal function)
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide	

DB 31, ... **DBX95.0** Data block	**U_{DClink} < warning threshold** Signal(s) from axis/spindle (drive → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The drive signals to the PLC that the DC link voltage U_{DC} link has dropped below the DC link undervoltage warning threshold. The DC link undervoltage warning threshold is defined in MD:1604 LINK_VOLTAGE_WARN_LIMIT. The DC link undervoltage warning threshold should be defined as a value which is greater than 400V depending on the application. If the DC link voltage drops below 280V, it is switched off by the hardware.	
Signal state 0 or signal transition 1 ——> 0	The DC link voltage U_{DClink} is greater than the DC link undervoltage warning threshold.	
Signal irrelevant for ...	SINUMERIK FM-NC	
Application example(s)	If a warning signal is given, measures can be taken by the PLC user program, for example, to stop machining (e.g. start tool retraction) or to back up the DC link voltage.	
Related to	MD 1604: LINK_VOLTAGE_WARN_LIMIT (DC link undervoltage warning threshold)	
References	/IAD/, SINUMERIK 840D Installation and StartUp Guide, Section SIMODRIVE 611D or /IAG/, SINUMERIK 810D Installation and StartUp Guide	

■

© Siemens AG, 2003. All rights reserved

6 Example

6.1 Controller parameter set switchover

Example parameter sets

The position controller gain (servo gain factor) of machine axis AX1 must be switched over. The following settings have been made:

```
$MA_POSCTRL_GAIN [0, AX1] = 4.0          //Servo gain setting for par. set 1
$MA_POSCTRL_GAIN [1, AX1] = 2.0          //Servo gain setting for par. set 2
$MA_POSCTRL_GAIN [2, AX1] = 1.0          //Servo gain setting for par. set 3
$MA_POSCTRL_GAIN [3, AX1] = 0.5          //Servo gain setting for par. set 4
$MA_POSCTRL_GAIN [4, AX1] = 0.25         //Servo gain setting for par. set 5
$MA_POSCTRL_GAIN [5, AX1] = 0.125        //Servo gain setting for par. set 6

$MA_DRIVE_AX_RATIO_DENOM [0, AX1] = 3  //Denominator load gearbox for par. set 1
$MA_DRIVE_AX_RATIO_DENOM [1, AX1] = 3  //Denominator load gearbox for par. set 2
$MA_DRIVE_AX_RATIO_DENOM [2, AX1] = 3  //Denominator load gearbox for par. set 3
$MA_DRIVE_AX_RATIO_DENOM [3, AX1] = 3  //Denominator load gearbox for par. set 4
$MA_DRIVE_AX_RATIO_DENOM [4, AX1] = 3  //Denominator load gearbox for par. set 5
$MA_DRIVE_AX_RATIO_DENOM [5, AX1] = 3  //Denominator load gearbox for par. set 6

$MA_DRIVE_AX_RATIO_NUMERA [0, AX1] = 5  //Numerator load gearbox for par. set 1
$MA_DRIVE_AX_RATIO_NUMERA [1, AX1] = 5  //Numerator load gearbox for par. set 2
$MA_DRIVE_AX_RATIO_NUMERA [2, AX1] = 5  //Numerator load gearbox for par. set 3
$MA_DRIVE_AX_RATIO_NUMERA [3, AX1] = 5  //Numerator load gearbox for par. set 4
$MA_DRIVE_AX_RATIO_NUMERA [4, AX1] = 5  //Numerator load gearbox for par. set 5
$MA_DRIVE_AX_RATIO_NUMERA [5, AX1] = 5  //Numerator load gearbox for par. set 6

$MA_AX_VELO_LIMIT [0 ...5, AX1]          //Setting for each parameter set*)
$MA_EQUIV_CURRCTRL_TIME [0..5, AX1]//Setting for each parameter set*)
$MA_EQUIV_SPEEDCTRL_TIME [0..5, AX1]  //Setting for each parameter set*)
$MA_DYN_MATCH_TIME [0 ...5, AX1]         //Setting for each parameter set*)
```

*) The appropriate line must be specified separately for each parameter set according to the applicable syntax rules.

Parameter set 1 is selected (DB31, DBB9, Bit0 ...2 = 0). The machine data with index “0” belonging to the parameter set are active.
The effective servo gain is 4.0 /s:

```
$MA_POSCTRL_GAIN [0, AX1] = 4.0
$MA_DRIVE_AX_RATIO_DENOM [0, AX1] = 3
$MA_DRIVE_AX_RATIO_NUMERA [0, AX1] = 5
$MA_AX_VELO_LIMIT [0, AX1] = ...
$MA_EQUIV_CURRCTRL_TIME [0, AX1] = ...
$MA_EQUIV_SPEEDCTRL_TIME [0, AX1] = ...
$MA_DYN_MATCH_TIME [0, AX1] = ...
```

Switchover

Parameter set switchover is enabled: $MA_PARAMSET_CHANGE_ENABLE [AX1]= 1 or 2. The fourth parameter set is now selected by the PLC.

© Siemens AG, 2003. All rights reserved

Interface: Request

DB31, DBB9 (bits0 ...2) => 3: The fourth parameter set is selected for the machine axis AX1.

After a delay time, the parameter set is switched over. The machine data with index “3” belonging to the parameter set are active.

```
$MA_POSCTRL_GAIN [3, AX1] = 0.5                 //The effective servo gain is now 0.5 /s
$MA_DRIVE_AX_RATIO_DENOM [3, AX1] = 3           //Caution! Data that must not be
                                                //altered must be
$MA_DRIVE_AX_RATIO_NUMERA [3, AX1] = 5          //identical for all indices.
$MA_AX_VELO_LIMIT [3, AX1] = ...
$MA_EQUIV_CURRCTRL_TIME [3, AX1] = ...
$MA_EQUIV_SPEEDCTRL_TIME [3, AX1] = ...
$MA_DYN_MATCH_TIME [3, AX1] = ...
```

Interface: Feedback

DB31, DBB69 (bits0 ...2) => 3: The fourth parameter set is selected for the feedback for machine axis AX1.

After a delay time, the parameter set is switched over. The machine data with index “3” belonging to the parameter set are active.

■

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

7

7.1 Interface signals

DB number	Bit, byte	Name	Refer to Doc.
General		**Signals from NC to PLC**	
10	56.4–56.7	Keyswitch setting 0 to 3	
		Signals from NC to PLC	
10	103.0	MMC alarm is active	
10	104.7	NCK CPU Ready	
10	108.1	MMC CPU2 Ready (MMC to OPI or MPI)	
10	108.2	MMC CPU1 Ready (MMC to MPI)	
10	108.3	MMC CPU1 Ready (MMC to OPI, standard link)	
10	108.6	611D Ready	
10	108.7	NC Ready	
10	109.0	NCK alarm is active	
10	109.6	Air temperature alarm	
10	109.7	NCK battery alarm	

Operatorpanelspecific		**Signals to operator panel (PLC → MMC)**	
19	0.0	Screen bright	
19	0.1	Darken screen	
19	0.2	Key disable	
19	0.3	Delete Cancel alarms (MMC 103 only)	
19	0.4	Delete Recall alarms (MMC 103 only)	
19	0.7	Istwert in WKS	
19	10.0	Programming area selection	
19	10.1	Alarm area selection	
19	10.2	Tool offset selection	
19	10.7	ShopMill control signal	
19	12.2	COM2 active (job byte of PLC)	
19	12.3	COM1 active (job byte of PLC)	
19	12.4	RS232C stop (job byte of PLC)	
19	12.5	RS232C external (job byte of PLC)	
19	12.6	RS232C OFF (job byte of PLC)	
19	12.7	RS232C on (job byte of PLC)	
19	13.5	Unload parts program	

© Siemens AG, 2003. All rights reserved

DB number	Bit, byte	Name	Doc. Refer to
19	13.6	Load parts program	
19	13.7	Parts program selection	
19	14.7	File system active/passive (for MMC 100.2, always passive)	
19	16.7	File system active/passive (for MMC 103, always passive)	
19	44.0	Mode change disable	
Signals from operator panel (MMC → PLC)			
19	20.1	Screen is dark	
19	20.7	Switch over MCS/WCS	
19	24.0	Error (Acknowledgment byte for current RS232C status)	
19	24.1	O.K. (Acknowledgment byte for current RS232C status)	
19	24.2	COM2 active (Acknowledgment byte for current RS232C status)	
19	24.3	COM1 active (Acknowledgment byte for current RS232C status)	
19	24.4	RS232 stop (Acknowledgment byte for current RS232C status)	
19	24.5	RS232 External (Acknowledgment byte for current RS232C status)	
19	24.6	RS232 OFF (Acknowledgment byte for current RS232C status)	
19	24.7	RS232 ON (Acknowledgment byte for current RS232C status)	
19	26.0	Error (Parts program handling status)	
19	26.1	O.K. (Parts program handling status)	
19	26.3	Active (Parts program handling status)	
19	26.5	Unload (Parts program handling status)	
19	26.6	Load (Parts program handling status)	
19	26.7	Select (Parts program handling status)	
19	42.0	FC9: Measure in JOG mode	
19	45.0	FC9 Out: Active	
19	45.1	FC9 Out: Done	
19	45.2	FC9 Out: Error	
19	45.3	FC9 Out: StartErr	

Channel-specific		**Signals from PLC to NCK channel**	
21, ...	6.2	Delete distance-to-go (channel-specific)	
Signals from NCK channel to PLC			
21, ...	36.6	Channel-specific NCK alarm is active	
21, ...	36.7	NCK alarm with processing stop present	
21, ...	318.7	Overstore active	

Axis/spindle-specific		**Signals from PLC to axis/spindle**	
31, ...	1.3	Axis/spindle disable	
31, ...	1.4	Follow-up mode	
31, ...	1.5	Position measuring system 1	
31, ...	1.6	Position measuring system 2	
31, ...	2.1	Servo enable	

© Siemens AG, 2003. All rights reserved

31, ...	2.2	Delete distance-to-go (axis-specific)/Spindle reset	
31, ...	9.0, 1, 2	Parameter set switchover (request)	

DB number	Bit, byte	Name	Refer to Doc.
31, ...	20.0	Ramp-up times	
31, ...	20.1	Ramp function generator rapid stop	
31, ...	20.2	Torque limit 2	
31, ...	20.3	Speed setpoint smoothing	
31, ...	21.0–21.2	Drive parameter set selection A, B, C	
31, ...	21.3–21.4	Motor selection A, B	
31, ...	21.5	Motor selection to follow	
31, ...	21.6	Speed controller integrator disable	
31, ...	21.7	Pulse enable	
Signals from axis/spindle to PLC			
31, ...	61.3	Follow-up mode active	
31, ...	61.4	Axis/spindle stops	
31, ...	61.5	Position controller active	
31, ...	61.6	Speed controller active	
31, ...	61.7	Current controller active	
31, ...	69.0, 1, 2	Parameter set switchover (feedback)	
31, ...	76.0	Lubrication pulse	
31, ...	92.0	Setup mode active	
31, ...	92.1	Ramp function generator rapid stop active	
31, ...	92.2	Torque limit 2 active	
31, ...	92.3	Speed setpoint smoothing active	
31, ...	93.0–93.2	Active drive parameter set A, B, C	
31, ...	93.3–93.4	Active motor A, B	
31, ...	93.5	Drive Ready	
31, ...	93.6	Speed controller integrator disabled	
31, ...	93.7	Pulses enabled	
31, ...	94.0	Motor temperature prewarning	
31, ...	94.1	Heatsink temperature prewarning	
31, ...	94.2	Ramp-up function complete	
31, ...	94.3	$\|M_d\| < M_{dx}$	
31, ...	94.4	$\|n_{act}\| < n_{min}$	
31, ...	94.5	$\|n_{act}\| < n_x$	
31, ...	94.6	$n_{act} = n_{set}$	
31, ...	94.7	Variable signaling function	
31, ...	95.0	$U_{DC\ link}$<warning threshold	

© Siemens AG, 2003. All rights reserved

7.2 Machine data

Number		Identifier	Name	Refer to Doc.
MMC machine data ($M_ ...)				
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBEDDED (SW 6.1 and higher)	
9000	9000	LCD_CONTRAST	Contrast	IM2/IM4
9001	9001	DISPLAY_TYPE	Monitor type	IM2/IM4
	9002	DISPLAY_MODE	External monitor (1: black and white, 2: color)	IM2
	9003	FIRST_LANGUAGE	Foreground language	IM2
9004	9004	DISPLAY_RESOLUTION	Display resolution	IM2/IM4
	9005	PRG_DEFAULT_DIR	Basic setting Program directory	IM2
	9006	DISPLAY_BLACK_TIME	Time setting for screen saver	IM2
	9007	TABULATOR_SIZE	Tabulator length	IM2
9008	9008	KEYBOARD_TYPE	Keyboard type (0: OP, 1: MFII/QWERTY)	IM2/IM4
9009	9009	KEYBOARD_STATE	Shift behavior of keyboard during booting	IM2/IM4
9010		SPIND_DISPLAY_RESOLUTION	Display resolution for spindle values	IM4
9011	9011	DISPLAY_RESOLUTION_INCH	Display resolution for INCH_system of units	IM2/IM4
9012	9012	ACTION_LOG_MODE	Set action mode for action log	IM2/IM4
	9013	SYS_CLOCK_SYNC_TIME	System clock synchronizing time	IM2
9020	9020	TECHNOLOGY	Basic configuration for simulation and free contour programming	IM2/IM4
	9030	EXPONENT_LIMIT	Number of places to be displayed without exponent	IM2
	9031	EXPONENT_SCIENCE	Technical exponent representation in three steps	IM2
	9180	USER_CLASS_READ_TCARR	Protection level read tool carrier offsets general	IM2
	9181	USER_CLASS_WRITE_TCARR	Protection level write tool carrier offsets general	IM2
9200	9200	USER_CLASS_READ_TOA	Protection level read tool offsets general	IM2/IM4
9201	9201	USER_CLASS_WRITE_TOA_GEO	Protection level write tool geometry	IM2/IM4
9202	9202	USER_CLASS_WRITE_TOA_WEAR	Protection level write tool wear data	IM2/IM4
9203	9203	USER_CLASS_WRITE_FINE	Protection level write fine	IM2/IM4
9204		USER_CLASS_WRITE_TOA_SC	Protection level change total tool offsets	IM4
9205		USER_CLASS_WRITE_TOA_EC	Protection level change tool setup offsets	IM4
9206		USER_CLASS_WRITE_TOA_SUPVIS	Protection level change tool monitoring limit values	IM4
9207		USER_CLASS_WRITE_TOA_ASSDNO	Change D No. assigned to a tool edge	IM4
9208		USER_CLASS_WRITE_MAG_WGROUP	Change wear group magazine location/mag.	IM4
9209	9209	USER_CLASS_WRITE_TOA_ADAPT	Protection level write tool adapter data	IM2/IM4
9210	9210	USER_CLASS_WRITE_ZOA	Protection level write settable zero offset	IM2/IM4

© Siemens AG, 2003. All rights reserved

Number		Identifier	Name	Refer to Doc.
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBEDDED (SW 6.1 and higher)	
9211	9211	USER_CLASS_READ_GUD_LUD	Protection level read user variables	IM2/IM4
9212	9212	USER_CLASS_WRITE_GUD_LUD	Protection level write user variables	IM2/IM4
9213	9213	USER_CLASS_OVERSTORE_HIGH	Protection level extended overstore	IM2/IM4
9214	9214	USER_CLASS_WRITE_PRG_CONDIT	Protection level program control	IM2/IM4
9215	9215	USER_CLASS_WRITE_SEA	Protection level write setting data	IM2/IM4
	9216	USER_CLASS_READ_PROGRAM	Protection level read parts program	IM2
	9217	USER_CLASS_WRITE_PROGRAM	Protection level enter parts program	IM2
9218	9218	USER_CLASS_SELECT_PROGRAM	Protection level parts program selection	IM2/IM4
9219	9219	USER_CLASS_TEACH_IN	Protection level TEACH IN	IM2/IM4
9220	9220	USER_CLASS_PRESET	Protection level PRESET	IM2/IM4
9221	9221	USER_CLASS_CLEAR_RPA	Protection level delete R parameters	IM2/IM4
9222	9222	USER_CLASS_WRITE_RPA	Protection level write R parameters	IM2/IM4
	9223	USER_CLASS_SET_V24	Protection level for RS232C/interface parameterization	IM2
	9224	USER_CLASS_READ_IN	Protection level for import data	IM2
	9225	USER_CLASS_READ_CST	Protection level standard cycles	IM2
	9226	USER_CLASS_READ_CUS	Protection level user cycles	IM2
	9227	USER_CLASS_SHOW_SBL2	Skip single block2 (SBL2)	IM2
	9228	USER_CLASS_READ_SYF	Access level select directory SYF	IM2
	9229	USER_CLASS_READ_DEF	Access level select directory DEF	IM2
	9230	USER_CLASS_READ_BD	Access level select directory BD	IM2
9231		USER_CLASS_WRITE_RPA_1	Write protection for first RPA area	IM4
9232		USER_BEGIN_WRITE_RPA_1	Start of the first RPA area	IM4
9233		USER_END_WRITE_RPA_1	End of the first RPA area	IM4
9234		USER_CLASS_WRITE_RPA_2	Write protection for second RPA area	IM4
9235		USER_BEGIN_WRITE_RPA_2	Start of the second RPA area	IM4
9236		USER_END_WRITE_RPA_2	End of the second RPA area	IM4
9237		USER_CLASS_WRITE_RPA_3	Write protection for third RPA area	IM4
9238		USER_BEGIN_WRITE_RPA_3	Start of the third RPA area	IM4
9239		USER_END_WRITE_RPA_3	End of the third RPA area	IM4
9240		USER_CLASS_WRITE_TOA_NAME	Change tool designation and duplo	IM4
9241		USER_CLASS_WRITE_TOA_Type	Change tool type	IM4
	9460	PROGRAMM_SETTINGS	Reset-proof data storage for settings in the PROGRAM operating area	IM2
	9461	CONTOUR_END_TEXT	String to be added to end of contour on completion of input	IM2
	9478	TO_OPTION_MASK	Variants of tool offsets	IM2
	9500	NC_PROPERTIES	NC_properties: Bit 0: Digital drives, Bit 1: Software start-up switch	IM2

© Siemens AG, 2003. All rights reserved

Number		Identifier	Name	Refer to Doc.
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBRDDED (SW 6.1 and higher)	
9510		USER_CLASS_DIRECTORY1_P	Protection level for network drive 1 for PROGRAM operating area	IM4
9511		USER_CLASS_DIRECTORY2_P	Protection level for network drive 2 for PROGRAM operating area	IM4
9512		USER_CLASS_DIRECTORY3_P	Protection level for network drive 3 for PROGRAM operating area	IM4
9513		USER_CLASS_DIRECTORY4_P	Protection level for network drive 4 for PROGRAM operating area	IM4
9516		USER_CLASS_DIRECTORY1_M	Protection level for network drive 1 for MACHINE operating area	IM4
9517		USER_CLASS_DIRECTORY2_M	Protection level for network drive 2 for MACHINE operating area	IM4
9518		USER_CLASS_DIRECTORY3_M	Protection level for network drive 3 for MACHINE operating area	IM4
9519		USER_CLASS_DIRECTORY4_M	Protection level for network drive 4 for MACHINE operating area	IM4

Number	Identifier	Name	Refer to Doc.
General ($MN_ ...)			
10350	FASTIO_DIG_NUM_INPUTS	Number of active digital NCK input bytes	A4
10360	FASTIO_DIG_NUM_OUTPUTS	Number of active digital NCK output bytes	A4
10361	FASTIO_DIG_SHORT_CIRCUIT	Shorting of digital inputs and outputs	
11120	LUD_EXTENDED_SCOPE	Activate program-global variables (PUD)	PGA
11270	DEFAULT_VALUES_MEM_MSK	Activ. function: Save DEFAULT values of GUD.	
18150	MM_GUD_VALUES_MEM	Reserve memory space for GUD	S7
Channel-specific ($MC_ ...)			
21015	INVOLUTE_RADIUS_DELTA	End point monitoring for involute	
21016	INVOLUTE_AUTO_ANGLE_LIMIT	Automatic angle limitation for involute interpolation	
27800	TECHNOLOGY_MODE	Technology in channel	
28150	MM_NUM_VDIVAR_ELEMENTS	Number of write elements for PLC variables	
28530	MM_PATH_VELO_SEGMENTS	Number of storage elements for limiting path velocity in block	K1
Axis/channel-specific ($MA_ ...)			
30350	SIMU_AX_VDI_OUTPUT	Output of axis signals for simulation axes	G2
33050	LUBRICATION_DIST	Lubrication pulse distance	
35590	PARAMSET_CHANGE_ENABLE	Parameter set definition possible from PLC	
36060	STANDSTILL_VELO_TOL	Maximum velocity/speed when axis/spindle stationary	
36610	AX_EMERGENCY_STOP_TIME	Length of the braking ramp for error states	A3
36620	SERVO_DISABLE_DELAY_TIME	Cutout delay servo enable	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer to Doc.
Drive machine data ($MD_ ...)			
1403	PULSE_SUPPRESSION_SPEED	Shutoff speed for pulse suppression	/IAD/, /IAG/
1404	PULSE_SUPPRESSION_DELAY	Time for pulse suppression	/IAD/, /IAG/
1417	SPEED_THRESHOLD_X	n_x for $n_{act} < n_{set}$ signal	/IAD/, /IAG/
1418	SPEED_THRESHOLD_MIN	n_{min} for $n_{act} < n_{set}$ signal	/IAD/, /IAG/
1426	SPEED_DES_EQ_ACT_TOL	Tolerance band for $n_{set} = n_{act}$ signal	/IAD/, /IAG/
1427	SPEED_DES_EQ_ACT_DELAY	Delay time $n_{set} = n_{act}$ signal	/IAD/, /IAG/
1428	TORQUE_THRESHOLD_X	Threshold torque	/IAD/, /IAG/
1429	TORQUE_THRESHOLD_X_DELAY	Delay time $n_d < n_{dx}$ signal	/IAD/, /IAG/
1602	MOTOR_TEMP_WARN_LIMIT	Maximum motor temperature	/IAD/, /IAG/
1603	MOTOR_TEMP_ALARM_TIME	Time for motor temperature alarm	/IAD/, /IAG/
1604	LINK_VOLTAGE_WARN_LIMIT	DC link under voltage warning threshold	/IAD/, /IAG/

7.3 System variables

Identifier	Name	Refer to Doc.
$P_FUMB	Unassigned parts program memory (**F**ree **U**ser **M**emory **B**uffer)	
$A_DBB[n]	Data on PLC (data type BYTE)	
$A_DBW[n]	Data on PLC (data type WORD)	
$A_DBD[n]	Data on PLC (data type DWORD)	
$A_DBR[n]	Data on PLC (data type REAL)	

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, Diagnostics Guide
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Axis Monitoring, Protection Zones (A3)

1 Brief Description 1/A3/1-3

1.1 Axis monitoring functions 1/A3/1-3

1.2 Protection zones 1/A3/1-3

2 Detailed Description 1/A3/2-5

2.1 Motion monitoring functions 1/A3/2-5
2.1.1 Contour monitoring 1/A3/2-5
2.1.2 Positioning monitoring 1/A3/2-7
2.1.3 Zero speed monitoring 1/A3/2-10
2.1.4 Clamping operations and monitoring 1/A3/2-11
2.1.5 Speed setpoint monitoring 1/A3/2-17
2.1.6 Actual velocity monitoring 1/A3/2-19

2.2 Encoder monitoring functions 1/A3/2-21
2.2.1 Encoder limit frequency monitoring 1/A3/2-21
2.2.2 Zero mark monitoring 1/A3/2-22
2.2.3 Hardware faults and contamination signal 1/A3/2-23

2.3 Monitoring of static limits 1/A3/2-24
2.3.1 Limit switch monitoring 1/A3/2-24
2.3.2 Working area limitation 1/A3/2-28

2.4 Protection zones 1/A3/2-32
2.4.1 General 1/A3/2-32
2.4.2 Types of protection zone 1/A3/2-32
2.4.3 Coordinate system, orientation 1/A3/2-36
2.4.4 Definition of protection zones 1/A3/2-36
2.4.5 Activation and deactivation of protection zones 1/A3/2-41
2.4.6 Protection zone violation and temporary enabling of individual protection zones 1/A3/2-45
2.4.7 Limits of protection zones 1/A3/2-52

3 Supplementary Conditions 1/A3/3-55

3.1 Axis monitoring functions 1/A3/3-55

3.2 Protection zones 1/A3/3-55

4 Data Descriptions (MD, SD) 1/A3/4-57

4.1 General machine data 1/A3/4-57

© Siemens AG, 2003. All rights reserved

4.2 Channel-specific machine data 1/A3/4-57

4.3 Axis/Spindle-specific machine data 1/A3/4-58
4.3.1 Axis monitoring functions 1/A3/4-58
4.3.2 Protection zones 1/A3/4-65

4.4 Axis/spindle-specific setting data 1/A3/4-66
4.4.1 Axis monitoring functions 1/A3/4-66
4.4.2 Protection zones 1/A3/4-67

5 Signal Descriptions 1/A3/5-69

5.1 Axis/spindle-specific signals 1/A3/5-69
5.1.1 Axis monitoring functions 1/A3/5-69
5.1.2 Protection zones 1/A3/5-71

5.2 Channel-specific signals 1/A3/5-72
5.2.1 Axis monitoring functions 1/A3/5-72
5.2.2 Protection zones 1/A3/5-72

6 Examples 1/A3/6-77

6.1 Axis monitoring functions 1/A3/6-77

6.2 Protection zones 1/A3/6-77
6.2.1 Definition and activation of protection zones 1/A3/6-77

7 Data Fields, Lists 1/A3/7-81

7.1 Axis/spindle-specific interface signals 1/A3/7-81
7.1.1 Channel-specific signals 1/A3/7-81
7.1.2 Axis monitoring functions 1/A3/7-81
7.1.3 Protection zones 1/A3/7-81

7.2 Channel-specific interface signals 1/A3/7-82
7.2.1 Axis monitoring functions 1/A3/7-82
7.2.2 Protection zones 1/A3/7-82

7.3 General machine data 1/A3/7-83
7.3.1 Axis monitoring functions 1/A3/7-83
7.3.2 Protection zones 1/A3/7-83

7.4 Axis/spindle-specific machine data 1/A3/7-83
7.4.1 Axis monitoring functions 1/A3/7-83
7.4.2 Protection zones 1/A3/7-84

7.5 Channel-specific machine data 1/A3/7-85
7.5.1 Axis monitoring functions 1/A3/7-85
7.5.2 Protection zones 1/A3/7-85

7.6 Axis/spindle-specific setting data 1/A3/7-85
7.6.1 Axis monitoring functions 1/A3/7-85
7.6.2 Protection zones 1/A3/7-85

7.7 Alarms 1/A3/7-86

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

1.1 Axis monitoring functions

Axis monitoring

A comprehensive monitoring mechanism must be provided in order to protect the operator and machinery.

The monitoring functions available include the following:

- Motion monitoring functions
  - Contour monitoring
  - Positioning monitoring
  - Zero speed monitoring
  - Clamping monitoring
  - Speed setpoint monitoring
  - Actual velocity monitoring
  - Encoder monitoring functions
- Monitoring of static limits
  - Limit switch monitoring
  - Working area limitation

1.2 Protection zones

With protection zones, various elements of the machine and its equipment (e.g. spindle chuck, tool changer, toolholder, tailstock, probe) and the workpiece being machined are protected from incorrect movements (see Chapter 3).

Number of Protection zones

NCU 570, NCU 571, NCU 571.2, CCU1:
A maximum of 4 machinerelated protection zones
NCU 572, 573.2, CCU2:
A maximum of 10 machinerelated and channelspecific protection zones, total: **10**

The protection zones are activated and deactivated in the NC parts program.

At the beginning of a machining block, the control calculates the path and checks whether protection zones will be crossed.

It is possible to travel into a protection zone as a result of an operator action. The deactivation applies until the protection zone has again been left.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 Motion monitoring functions

2.1.1 Contour monitoring

Contour error

Contour errors are caused by signal distortions in the position control loop. A distinction is made between:

- Linear signal distortion. This is caused by:
  - Speed and position controller not being set optimally
  - Different servo gain factor settings for the feed axes involved in creating the path

    If the servo gain factor of two axes involved in linear interpolation is equal, the actual point follows the setpoint on the same contour but with a delay.
    If the servo gain factors are not equal, a parallel offset between the set contour and actual contour occurs.
  - Unequal dynamic response of the feed drives

    Unequal drive dynamic responses lead to path deviations especially on contour changes. Circles are distorted into ellipses by unequal dynamic responses of the two feed drives.
- Non-linear signal distortions. This is caused by:
  - Activation of the current limitation within the machining area
  - Activation of the limitation of the set speed
  - Backlash within and/or outside the position control loop.

    Contour errors arise because of backlash and friction while traveling a circular contour.
    While traveling straight lines, a contour error occurs because of the backlash outside the position control loop, e.g. due to a tilting milling spindle. This causes a parallel offset between the actual and the set contour. The shallower the gradient of the straight line, the larger the offset.
  - Nonlinear friction behavior of slide guides.

© Siemens AG, 2003. All rights reserved

Following error monitoring

Function

The principle on which the following error monitoring function works is constant comparison of the following error measured and that calculated in advance from the NC position setpoint. For the precalculation of the following error a model is used that simulates the dynamics of the position control including feedforward control.

So that the monitoring function does not respond incorrectly on slight speed fluctuations (caused by changes of load) a tolerance band is allowed for the max. contour deviation.

If the admissible actual value deviation entered in the MD 36400: CONTOUR_TOL (tolerance band contour monitoring) is exceeded, an alarm is output and the axes are stopped.

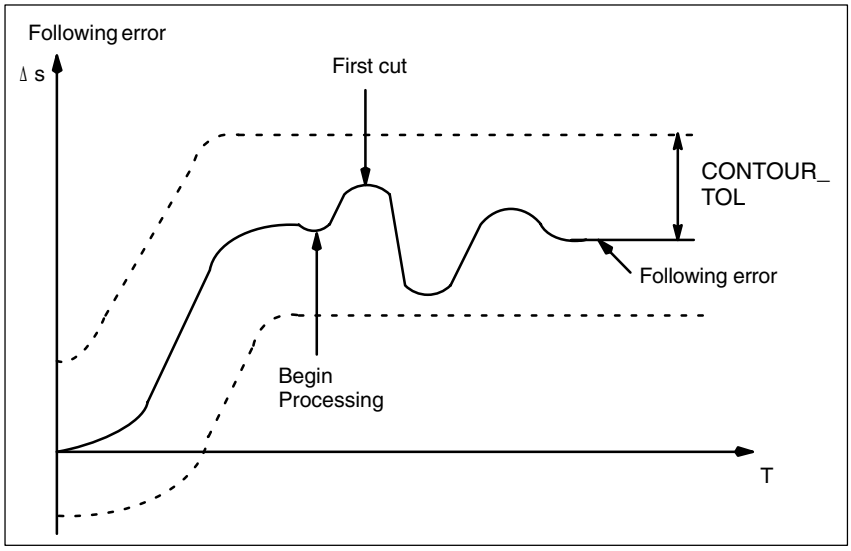


Fig. 2-1 Following error monitoring

Activation

The following error monitoring function is active in position control mode with:

- Linear and rotary axes
- Position controlled spindles
- With or without feedforward control
- Acceleration and deceleration processes
- Constant velocity phases
- Continuous and discontinuous contours (e.g. circular path)

Following error monitoring is not active in follow-up mode.

Substitution

This function is always active in position control mode.

© Siemens AG, 2003. All rights reserved

Effect

If the following error deviation is too large, this has the following effect:

- The alarm 25050 “Contour monitoring” is output
- The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp.
  The braking ramp time is set in MD 36610: AX_EMERGENCY_STOP_TIME (duration of the braking ramp on error states).
- If the axis/spindle is involved in interpolation with other axes/spindles, these are brought to a standstill with rapid stop with following error reduction (position setpoint = constant).

Remedy

- Increase the tolerance band
- The actual “servo gain factor” must correspond to the desired servo gain factor set in MD 32200: POSCTRL_GAIN[n] (servo gain factor).
  On SINUMERIK FM-NC:
  MD 32260: RATED_VELO (rated motor speed) and
  MD 32250: RATED_OUTVAL (rated output voltage) must be checked.
- Check optimization of the speed controller
- Check smooth running of the axes
- Check machine data for traversing movements
  (feedrate override, acceleration, max. velocities, ...)
- Operation with feedforward control:
  MD 32810: EQUIV_SPEEDCTRL_TIME (equivalent time constant speed control loop for feedforward control) or
  MD 32800: EQUIV_CURRCTRL_TIME (equivalent time constant current control loop for feedforward control) must be checked.
  If the machine data are set too inexactly, then the setting in
  MD 36400: CONTOUR_TOL must be increased.

2.1.2 Positioning monitoring

Function

In order to ensure that an axis reaches the required position within a specified time period, the timer that can be programmed in MD 36020: POSITIONING_TIME (time delay exact stop fine) is started at the end of each motion block (partial position setpoint = 0 at end of motion) and, when the timer runs out, a check made to ascertain whether the following error has dropped below the limit value for MD 36010: STOP_LIMIT_FINE (exact stop fine).

“Exact stop coarse and fine” see:
References: /FB/, B1, “Continuous Path Mode, Exact Stop and Look Ahead”

The following diagram shows the interaction between the positioning, zero speed and clamping monitoring functions.

© Siemens AG, 2003. All rights reserved

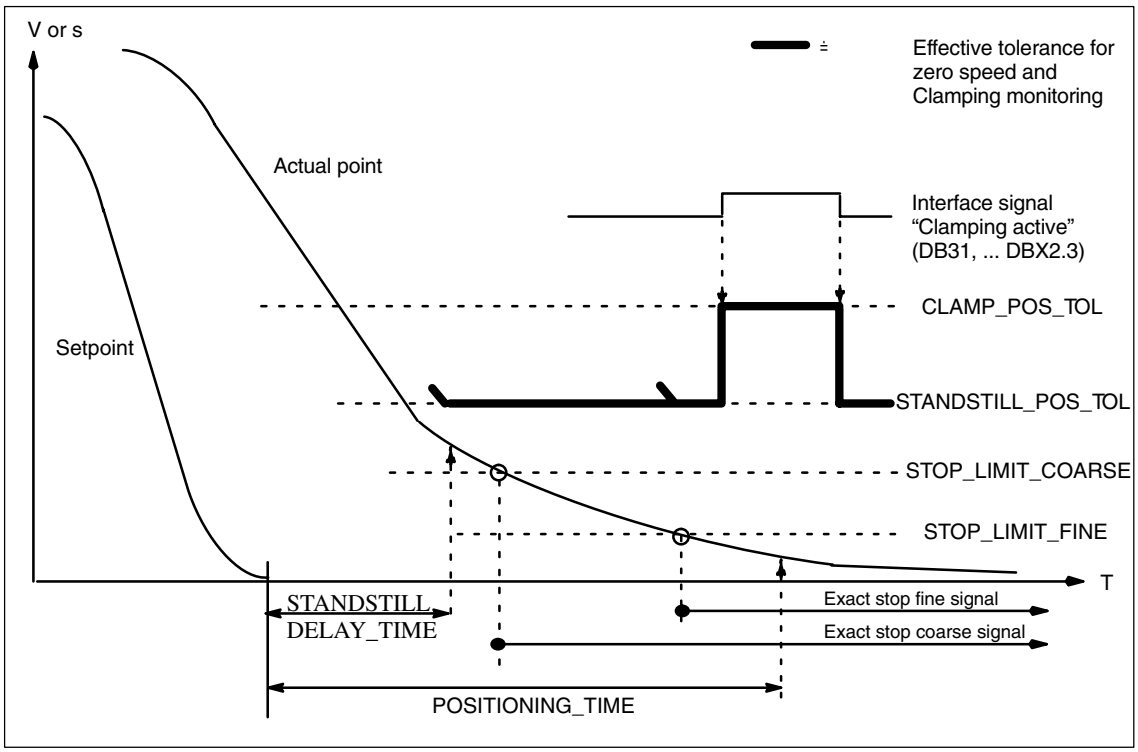


Fig. 2-2 Positioning, zero speed and clamping monitoring

Activation

Positioning monitoring is always activated after termination of motion blocks "according to the setpoint".

This function is activated with:

- Linear and rotary axes
- Position controlled spindles

In follow-up mode the positioning monitoring function is not active.

Deactivation

When the programmed "exact stop limit fine" has been reached or a new partial position setpoint <> 0 output (e.g. for positioning according to "Exact stop coarse" following by block change), the positioning monitor is switched off and replaced by the zero speed monitor.

© Siemens AG, 2003. All rights reserved

Effect

If the limit value for “exact stop fine” is not yet reached, the following action is performed:

- Output of alarm 25080 “Positioning monitoring”
- The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp.
  The braking ramp time is set in MD 36610:
  AX_EMERGENCY_STOP_TIME (duration of the braking ramp on error states).
- If the axis/spindle is involved in interpolation with other axes/spindles, these are stopped by rapid stop with following error reduction (default for position partial setpoint = 0).

Cause of error/ remedy

- Position control gain too low —> alter machine data for position control gain
  MD 32200: POSCTRL_GAIN[n] (servo gain factor), MD 32910:
  DYN_MATCH_TIME[n] (time constant for dynamic response matching)
- Positioning window (Exact stop fine), positioning monitoring time and position control gain have not been coordinated
  —> alter machine data: MD 36010: STOP_LIMIT_FINE (exact stop fine),
  MD 36020: POSITIONING_TIME (delay exact stop fine),
  MD 32200: POSCTRL_GAIN[n] (servo gain factor)
- Drift too high (only for analog drives) —> compensate drift
  References: /FB/, K3, “Compensations”

Rule of thumb

- Positioning window large —> max. positioning monitoring time can be set to a relatively short value
- Positioning window small —> max. positioning monitoring time must be selected to be relatively long
- Position controller gain low —> max. positioning monitoring time must be selected to be relatively long
- Position control gain high —> max. positioning monitoring time must be selected to be relatively short

---

Note

The size of the positioning window affects the block change time. The smaller these tolerances are selected the longer the positioning task will take, which in turn means a longer time before the next command can be executed.

© Siemens AG, 2003. All rights reserved

2.1.3 Zero speed monitoring

Function

The zero speed monitoring function has the following functionality:

- At the end of a motion block
  (partial position setpoint=0 at end of motion), the following error is monitored to ascertain whether it reaches the limit value for MD 36060: STANDSTILL_POS_TOL (zero speed tolerance) within the time programmed in MD 36040: STANDSTILL_DELAY_TIME (delay zero speed monitoring).
- At the end of a positioning operation
  (exact stop fine reached), the positioning monitor is replaced by the zero speed monitor. In this case, the axis is monitored to ascertain whether it has moved out of position by more than the distance set in MD 36030: STANDSTILL_POS_TOL (zero speed tolerance).
  Zero speed monitoring is activated when
  “Exact stop fine” is reached and the
  “Delay zero speed monitoring” set in MD: 36040
  STANDSTILL_DELAY_TIME is still running.
- Zero speed control, SW 4.4 and higher
  Zero speed control is no longer activated with “Exact stop fine”. Zero speed monitoring is always active after termination of the “Delay zero speed monitoring” parameterized in MD 36040 STANDSTILL_DELAY_TIME or on reaching Exact stop fine, as long as no new traverse command has been given.

See Fig. 2–2

Activation

Zero speed monitoring is always active after termination of the “Delay zero speed monitoring” or on reaching “Exact stop fine”, as long as no new traverse command has been given.

This function is activated with:

- Linear and rotary axes
- Position controlled spindles

In follow-up mode the zero speed control monitoring function is not active.

Effect

When the monitoring function responds it has the effects:

- Output of alarm 25040 “Zero speed monitoring”
- The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp.
  The braking ramp time is set in MD 36610: AX_EMERGENCY_STOP_TIME (duration of the braking ramp on error states).
- If the axis/spindle is involved in interpolation with other axes/spindles, these are stopped by rapid stop with following error reduction (default for position partial setpoint = 0).

© Siemens AG, 2003. All rights reserved

Cause of error/ remedy

- Position controller gain too high (control loop oscillation) —> alter machine data for controller gain
  MD 32200: POSCTRL_GAIN[n] (servo gain factor)
- Zero speed window too small —> alter machine data
  MD 36030: STANDSTILL_POS_TOL (zero speed tolerance)
- Axis is mechanically “pushed” out of position —> eliminate cause

2.1.4 Clamping operations and monitoring

Clamping monitoring function

If the axis must be clamped once it has been positioned, the axis monitoring function can be activated via interface signal “Clamping in progress” (DB31, ... DBX2.3).

This might be necessary because during the clamping process the axis can be pushed further out of the set position than the zero speed tolerance. The amount by which the axis may travel from the setpoint position is set in MD 36050: CLAMP_POS_TOL (clamping tolerance for interface signal clamping active).

See Fig. 2-2

Activation

Clamping monitoring is activated by the interface signal “Clamping active”. It replaces zero speed monitoring during clamping.

This function is activated with:

- Linear and rotary axes
- Position controlled spindles

The clamping monitoring is not active in follow-up mode.

Effect

If the axis is pushed out of position beyond the clamping tolerance during clamping the following occurs:

- Output of alarm 26000 “Clamping monitoring”
- The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp.
  The braking ramp time is set in MD 36610: AX_EMERGENCY_STOP_TIME (duration of the braking ramp on error states).
- If the axis/spindle is assigned to an interpolatory grouping with other axes/spindles, then these are also braked by rapid stop with following error reduction (default for partial position setpoint = 0).

Cause of error

- Axis is mechanically “pushed” out of position.

© Siemens AG, 2003. All rights reserved

Optimized axis clamping with SW6.4 and higher

Special support is given to axis clamping:

- Simple programming
- Simultaneous motion of path axes during clamping and release.

Simplified programming

Releasing a clamping axis takes time. In continuous-path mode, therefore, you must program a stop (G09 or G60 or auxiliary function output) to ensure that the program stops before it reaches the block in which the clamped axis needs to be moved.

By setting MD 36052: STOP_ON_CLAMPING[..] = 'H01', you can program the control to initiate the **stop automatically** in continuous-path mode. This is based on the assumption that the PLC will behave in a particular way, i.e.

- that the PLC will always release the axis from the clamp if an appropriate **motion command** is applied.
- that the servo enable signal is reset for the clamped axis. Setting the servo enable again therefore indicates to the NCK that the axis has been released from the clamp.
- the NCK therefore inserts a stop prior to the block in which a new axis motion is programmed. This stop is automatically canceled if the servo enable is inserted immediately before the beginning of the block.
- if the servo enable signal is not set, the NCK stops the path motion at the beginning of the block.

If bit 0 is not set in MD 36052: STOP_ON_CLAMPING[..], the NCK behaves as in the earlier software version.

© Siemens AG, 2003. All rights reserved

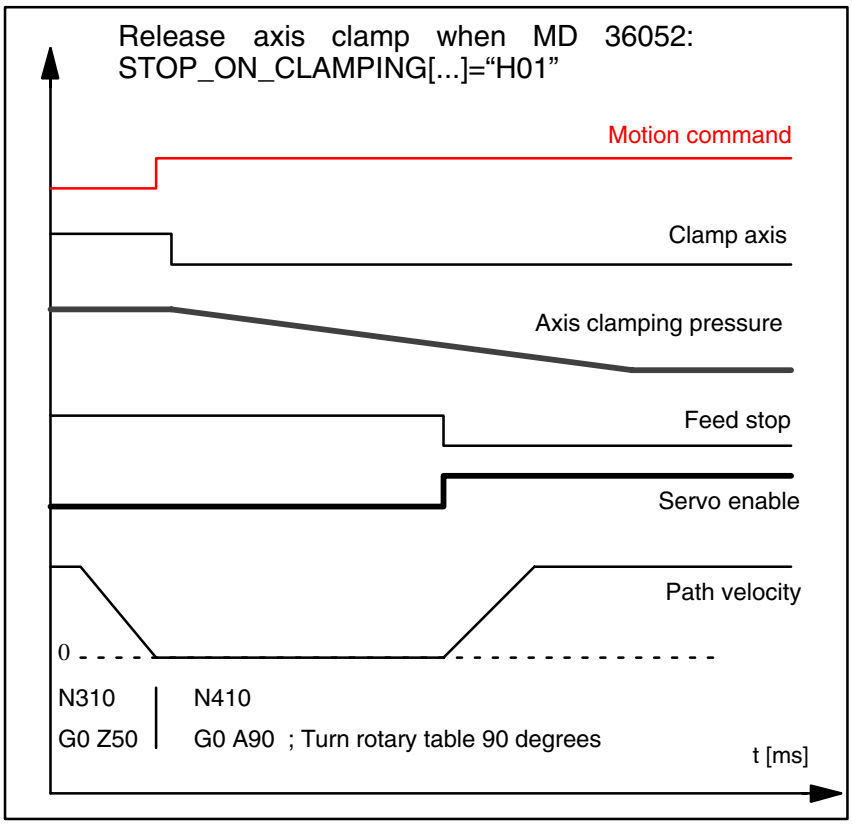


Fig. 2-3 Example of interface signals of a rotary table to be clamped, states on releasing axis from clamp. (The blocks in the diagram refer to the schematic example under Boundary conditions).

Look Ahead for axis clamp release

By setting MD 36052: STOP_ON_CLAMPING[..] is set to ’H03’, the NCK optimizes the clamping function, thereby increasing productivity through **simultaneous execution of operations**. This method of optimization can be utilized only if the PLC behaves according to the following rules:

- The PLC will always release the axis from the clamp if an appropriate **motion command** is applied.
- The axis may be released from the clamp for positioning purposes only. Programming of the **G0** indicates whether axes are being programmed.
- As a result, the NCK sets the motion command for the clamped axis on the Look Ahead principle if G0 blocks have been programmed for path axes, causing the PLC to cancel the axis clamping instruction. – This motion command is set a maximum of 2 G0 parts program blocks (incl. intermediate blocks generated by the NCK) in advance in order to maintain a clear correlation between the command and the block which generated it on the operator panel.

Bit 1 in MD 36052: STOP_ON_CLAMPING[..] will work with the Look Ahead solution described only if bit 0 is set as well.

© Siemens AG, 2003. All rights reserved

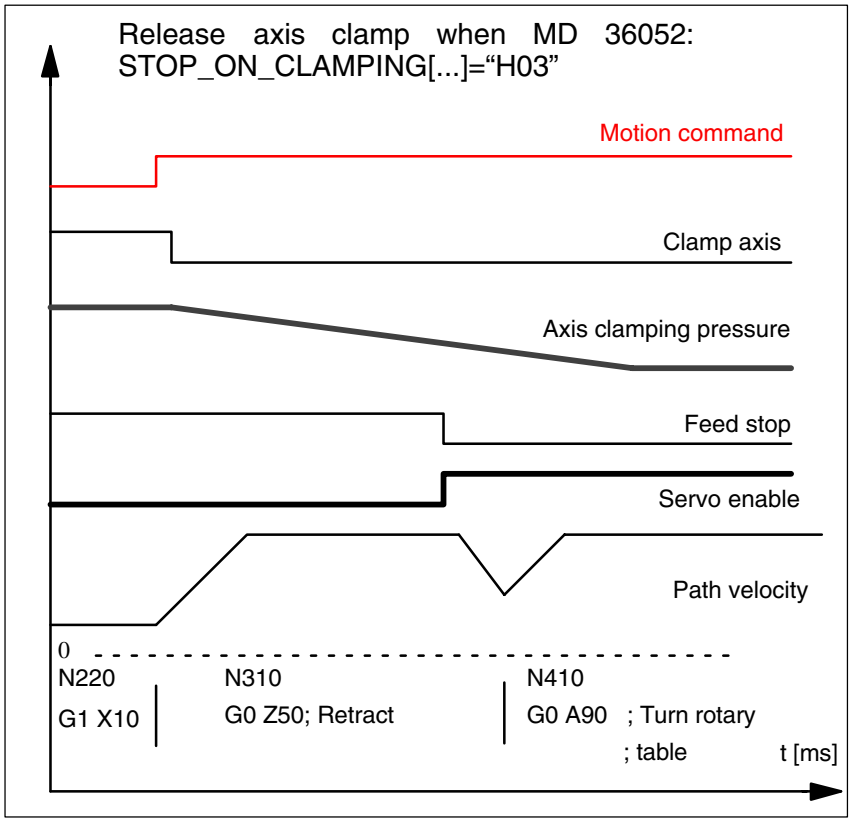


Fig. 2-4 Interface signals of the rotary table axis, states on releasing axis from clamp with Look Ahead. (The blocks in the diagram refer to the schematic example under Boundary conditions).

Optimized setting of clamping

Clamping an axis takes time. In continuous-path mode, therefore, you must program a stop (G09 or G60 or auxiliary function output) to ensure that the axis is securely clamped before you start machining the part.

By setting $MA_STOP_ON_CLAMPING[..] = 'H04', you can control the NCK to stop the axis automatically in continuous-path mode. This is based on the assumption that the PLC will behave in a particular way, i.e.

© Siemens AG, 2003. All rights reserved

- that the PLC will always clamp the axis if no **motion command** is applied.
- that the axis need not be clamped while the other axes are being positioned. Programming of the **G0** indicates whether axes are being programmed. – The stop command is not therefore set immediately at the beginning of the block containing the axis, but at the beginning of the next machining block. – A **machining block** contains, e.g. preparatory function G1, but certainly **not G0**.
- the final assumption is that the axis is clamped if the **feedrate override** for **machining blocks is set to a value other than 0**. If the axis is clamped before the next machining block, i.e. the feedrate override is other than 0 again, no further stop is generated and the axis is allowed to continue moving.

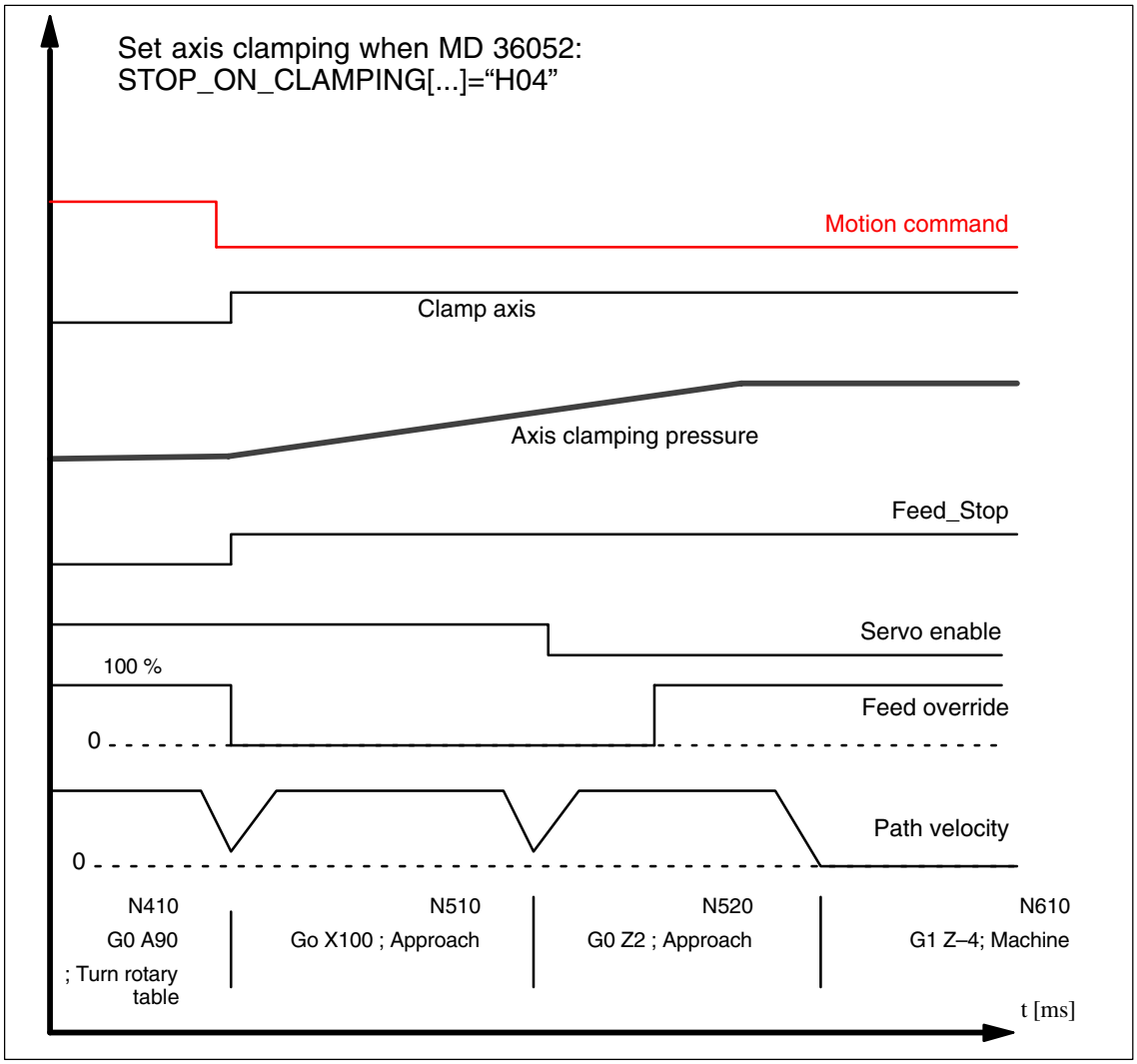


Fig. 2-5 Interface signals of the rotary table axis, states on setting axis clamping with Look Ahead. (The blocks in the diagram refer to the schematic example under Boundary conditions).

© Siemens AG, 2003. All rights reserved

If bit 2 in MD 36052: STOP_ON_CLAMPING[..] is not set, the NCK behaves as in SW 6.3 and lower with respect to clamping.

Supplementary conditions

For bits 0 – 2 in MD 36052: STOP_ON_CLAMPING[..], the requirements specified above of the PLC and the machine must be met exactly. The Look Ahead option must also be used.

Parts program blocks without path motion (e.g. M82/M83) interrupt continuous-path mode and thus also the Look Ahead function. If, for example, the parts program above is modified in the following way:

```
N100        G0 X0 Y0 Z0 A0 G90 G54 F500
N101        G641 ADIS=.1 ADISPOS=5

N210        G1 X10          ; Machine
N220        G1 X5 Y20       ;   "

N310        G0 Z50          ; Retract
N320        M82             ; <<<< No path motion
N410        G0 A90          ; Turn rotary table
N420        M83             ; <<<< No path motion
N510        G0 X100         ; Approach
N520        G0 Z2           ;   "

N610        G1 Z–4          ; Machine
N620        G1 X0 Y–20      ;   "

                M30
```

then the following applies:

- MD 36052: STOP_ON_CLAMPING[..] = 'H03' is no longer operative. The motion command is set in advance only for blocks with active continuous-path mode. M82 generates a stop and thus interrupts continuous-path mode. The Look Ahead stop at N410 is invalid as well because the axis is stopped anyway.
- MD 36052: STOP_ON_CLAMPING[..] = 'H04' however generates a stop irrespective of M83 which is executed as a function of "Feedrate override 0%". The axis is thus always stopped before the first machining block.

MD 36052: STOP_ON_CLAMPING[..] = 'H01' and MD 36052: STOP_ON_CLAMPING[..] = 'H04' work independently of axis clamping within the NCK and can therefore be used for other types of application.

- MD 36052: STOP_ON_CLAMPING[..] = 'H01' generates a Look Ahead stop if no servo enable signal is active for the relevant axis.
- MD 36052: STOP_ON_CLAMPING[..] = 'H04' generates a Look Ahead stop if the feedrate override is 0% on the "Rapid traverse –> Not rapid traverse" transition.

Both functions ensure that the axis stops moving at the beginning of a block with continuous-path mode and not after the block has started running.

Mode group RESET POWER ON

If the setting in MD 36052: STOP_ON_CLAMPING[..] is modified via the parts program, it will remain valid after mode group RESET and POWER ON.

© Siemens AG, 2003. All rights reserved

Block advance based on clamping tolerance

If the clamping monitor is activated via interface signal “Clamping in progress”, the clamping tolerance is sufficient to ensure block advance in AUTOMATIC MODE (MD 36050: CLAMP_POS_TOL[..]). The exact stop condition can be selected more finely than the clamping tolerance.

If the clamping operation has moved the axis out of its setpoint position, the position controller will readjust it as soon as the axial servo enable signal is set again.

If the axis has been moved a long way from its exact stop position, the PLC can switch it to follow-up mode to ensure that it is returned to its setpoint position by interpolation when the servo enable signal is set again. The exact stop signals (exact stop fine, exact stop coarse) can be applied as triggers for switchover to follow-up mode.

2.1.5 Speed setpoint monitoring

Function

Speed setpoint monitoring is used to check whether the physical limits of the drives (10V maximum voltage for speed setpoint for analog drives or maximum permissible motor speed for digital drives) have been exceeded.

The maximum speed setpoint can also be reduced and monitored for test mode.

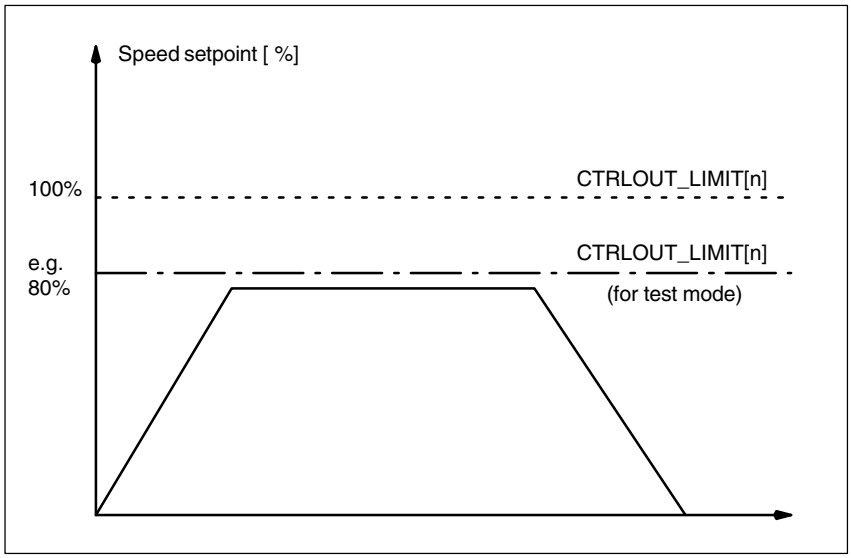


Fig. 2-6 Speed setpoint monitoring

The setting in MD 36210: CTRLOUT_LIMIT[n] (maximum set speed) is also monitored for violation.

© Siemens AG, 2003. All rights reserved

The MD 36220: CTRLOUT_LIMIT_TIME[n] (delay for speed setpoint monitoring) is set to define how long the set speed can remain at the limitation before the speed setpoint monitoring function responds.

The speed setpoint is composed of the speed setpoint of the position control, the forward compensation quantity (if feedforward control is active) and the drift compensation (only for analog drives).

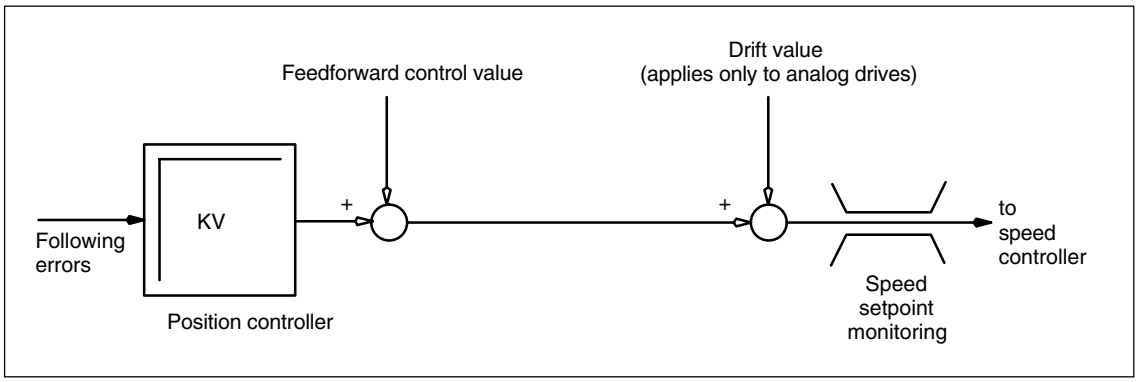


Fig. 2-7 Speed setpoint calculation

Activation

Speed setpoint monitoring is always active.

This function is activated with:

- Linear and rotary axes
- Open loop controlled and position controlled spindles.

Effect

When the maximum speed setpoint is exceeded and the delay time has expired (MD 36220: CTRLOUT_LIMIT_TIME[n]) the following occurs:

- Output of alarm 25060 “Speed setpoint limitation”
- The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp.
  The braking ramp time is set in MD 36610:
  AX_EMERGENCY_STOP_TIME (duration of the braking ramp on error states).
- If the axis/spindle is involved in interpolation with other axes/spindles, these are brought to a standstill with rapid stop with following error reduction (default for position partial setpoint = 0).

© Siemens AG, 2003. All rights reserved

Causes of error/ Remedy

- Tachogenerator compensation (on SINUMERIK FM-NC) has not been performed correctly or a measuring circuit error or drive fault has occurred.
- Setpoint inputs are to high (accelerations, velocities, reduction factors)
- Obstacle in machining room (e.g. machining table) —> remove obstacle.

Note

The application of the set speed limitation makes the control loop non-linear. This generally causes contour deviations if an axis remains in the set speed limitation.

2.1.6 Actual velocity monitoring

Function

This function monitors the actual velocity for exceeding a permissible limit value programmed in MD 36200: AX_VELO_LIMIT[n] (threshold value for velocity monitoring).

Activation

The actual velocity monitor is operative whenever the active measuring circuit selected via IS “Position measuring system 1 or 2” (DB31, ... DBX1.5, DBX1.6) is supplying actual values, i.e. still operating below the limit frequency.

This function is activated with:

- Linear and rotary axes
- Open loop controlled and position controlled spindles.

Effect

If the “Threshold for velocity monitoring” is exceeded the following occurs:

- Output of alarm 25030 “Actual velocity alarm limit”
- The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp.
  The braking ramp time is set in MD 36610: AX_EMERGENCY_STOP_TIME (duration of the braking ramp on error states).
- If the axis/spindle is involved in interpolation with other axes/spindles, these are brought to a standstill with rapid stop with following error reduction (default for position partial setpoint = 0).

© Siemens AG, 2003. All rights reserved

Cause of error/remedy

- Check speed setpoint cable
- Check actual values
- Check position control direction
- MD 36200: AX_VELO_LIMIT[n] (threshold value for velocity monitoring).

© Siemens AG, 2003. All rights reserved

2.2 Encoder monitoring functions

2.2.1 Encoder limit frequency monitoring

Function

If the permissible limit frequency of a measuring system entered in MD 36300: ENC_FREQ_LIMIT[n] (encoder limit frequency) is exceeded, the positional synchronization (reference point) between the machine and control system will be lost. Correct position control is no longer possible.
This state is signaled to the PLC.

Activation

The encoder limit frequency monitoring function is always active when the encoder is switched on.

This function is activated with:

- Linear and rotary axes
- Open loop controlled and position controlled spindles.

Effect

When the limit frequency of an encoder is exceeded the following occurs:

- The IS “Encoder limit frequency exceeded 1 / 2” (DB31, ... DBX60.2 or 60.3) is set.
- Spindles continuing to run withspeed control.
  If the spindle speed is reduced to a value that causes the encoder frequency to drop below the setting in MD 36302: ENC_FREQ_LIMIT_LOW, the spindle is automatically resynchronized with the reference system of the encoder.
- If the limit frequency is exceeded while the measuring system of a position controlled axis is active, alarm 21610 “Frequency is exceeded” is signaled.
- The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp.
  The braking ramp time is set in MD 36610: AX_EMERGENCY_STOP_TIME (duration of the braking ramp on error states).
- If the axis/spindle is involved in interpolation with other axes/spindles, these are brought to a standstill with rapid stop with following error reduction (default for position partial setpoint = 0).

Cause of error/remedy

- After the axes have stopped, the position control is automatically resumed.

Note

The axis affected must be rereferenced.

© Siemens AG, 2003. All rights reserved

2.2.2 Zero mark monitoring

Function

The zero mark monitoring is used to monitor the constant number of pulses between two zero marks of the position actual value encoder for any changes that may have occurred. It checks that the encoder always outputs the same number of pulses between two zero marks. In the event of a discrepancy, an alarm is output.

Activation

The machine data MD 36310: ENC_ZERO_MONITORING can be used to activate zero mark monitoring. Its meaning depends on the type of encoder used:

MD 36310 = 0 No zero mark monitoring
MD 36310 = 100 No zero mark monitoring and deactivation of all encoder alarms

MD 36310 > 0

MD 30240: ENC_TYPE = 0 no zero mark monitoring
MD 30240: ENC_TYPE = 3 no zero mark monitoring

MD 30240: ENC_TYPE = 1
MD 30240: ENC_TYPE = 2
Number of recognized changes at which an alarm should be output. Every time the encoder is switched on, counting starts at “0”.

MD 30240: ENC_TYPE = 4
Admissible deviation between absolut and incremental encoder signal in 0.5 encoder lines.
An alarm will be generated if this value is exceeded.
It is sufficient to enter one 0.5 encoder lines.

MD 30240: ENC_TYPE = 5 no zero mark monitoring
Encoder alarms are indicated by resetting alarm 20510. POWER ON alarm 25000 is not used.

Effect

On encoders used for position control, zero mark monitoring is activated and triggers alarm 25020.

The affected axis/spindle is brought to a standstill with rapid stop (with open position control loop) along a speed setpoint ramp. The braking ramp time is set in MD 36610: AX_EMERGENCY_STOP_TIME.

If the axis/spindle is assigned to an interpolatory grouping with other axes/ spindles, then these are also brought to a standstill with rapid stop with following error reduction (specified partial position setpoint = 0).

On passive encoders, when zero mark monitoring is activated, alarm 25021 is output. This alarm is only used for display purposes.

For more information about synchronizing the PLC with various position measuring systems on machine axes or spindles, see:
References: /FB1/, R1, “Reference Point Approach” 1

Error causes

- MD 36300: ENC_FREQ_LIMIT (encoder limit frequency) set too high.
- Encoder cable damaged or encoder/encoder electronics faulty

© Siemens AG, 2003. All rights reserved

2.2.3 Hardware faults and contamination signal

Function

The measuring circuit monitoring functions for alarms 25000, 25001 “Hardware fault” and alarms 25010, 25011 “Contamination of measuring” system are only active when the measuring system is active. This makes it possible to change an encoder on the non-active measuring system without having to restart the control. If one of these two errors is still present when the switchover to the other measuring system is made, the appropriate alarm is triggered.

Note

If a hardware fault occurs in the control loop, the IS “Referenced/synchronized 1/2” (DB31, ... DBX60.4 or DBX60.5) is removed; i.e. the axis must be rereferenced.

References: /DA/, “Diagnostics Guide”

© Siemens AG, 2003. All rights reserved

2.3 Monitoring of static limits

2.3.1 Limit switch monitoring

Overview of possible limit switch monitoring functions:

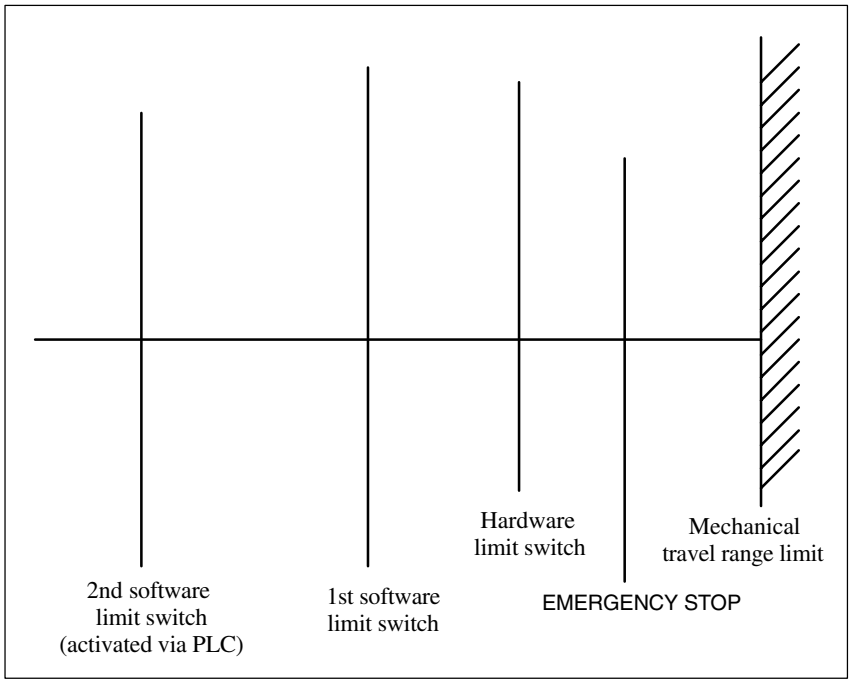


Fig. 2-8 Travel limits

Hardware limit switches

Function

For every axis there is a hardware limit switch for each direction of travel to prevent the slide from being moved out of the slide bed.
If the hardware limit switch is crossed, the PLC signals this to the NC in IS “Hardware limit switch plus/minus” (DB31–48, DBX12.1 or DBX12.0) and the motion of all axes is stopped.
The method of braking can be selected in MD 36600: BRAKE_MODE_CHOICE (deceleration behavior on hardware limit switch).

Activation

HW limit switch monitoring is active after the control has started up in all modes.

© Siemens AG, 2003. All rights reserved

Effect

- When a hardware limit switch is crossed in either direction, alarm 21614 “Hardware limit switch + or –” is triggered.
- The axis is brought to a standstill according to the setting in MD 36600: BRAKE_MODE_CHOICE (deceleration behavior on hardware limit switch).
- If the axis is assigned to an interpolatory grouping with other axes, then these are also braked according to the method selected in MD 36600: BRAKE_MODE_CHOICE (deceleration behavior on hardware limit switch).
- The direction keys in the approach direction are disabled.

Remedy

- Perform mode group reset
- Move in the opposite direction (in JOG mode)
- Correct the program.

Software limit switches

Function

These are used to delimit the maximum traverse range of each axis.
Each machine axis has 2 software limit switch pairs that are defined via the following machine data in the machine axis system:
MD 36110: POS_LIMIT_PLUS (1st software limit switch plus)
MD 36100: POS_LIMIT_MINUS (1st software limit switch minus)
MD 36130: POS_LIMIT_PLUS (2nd software limit switch plus)
MD 36120: POS_LIMIT_MINUS (2nd software limit switch minus)

© Siemens AG, 2003. All rights reserved

Activation

- Software limit switch monitoring is activated after reference point approach in all modes.
- The position of the software limit switch can be approached.
- The monitoring function is no longer valid after a PRESET. It only becomes active again after reference point approach.
- The 2nd software limit switch can be activated by the interface signal “2nd software limit switch plus/minus” (DB31–48, DBX12.3 or 12.2) from the PLC in order to reduce the working area, for example, if a tailstock has been swung into position. The change becomes active immediately. The The 1st software limit switch plus/minus is then deactivated.
- The SW limit switch monitor does not operate for endlessly turning rotary axes, i.e. if MD 30310: ROT_IS_MODULO = 1.
  (Modulo conversion for rotary axis and spindle).

  In SW 5.3 and higher, the SW limit switches can be activated through programming with newconf.

© Siemens AG, 2003. All rights reserved

Effect

Depending on the mode, different responses to software limit switch violation are possible:

Case 1 Automatic modes without transformation, without overlaid motion, unchanged software limit switches

- The block that would violate the software limits switches is not started. The previous block is terminated properly.

Case 2 Automatic modes with transformation

- Response as in Case 1 or
- The block that violates the software limit switch is started. The axis stops at the limit switch.
- The other axes that are involved in the contour are decelerated and the program contour is exited.

Case 3 Automatic modes with overlaid motion

- The block that violates the software limit switches is started. Axes that are traveling with overlaid motion or have traveled with overlaid motion stop at the limit switch in question.
- The other axes that are involved in the contour are decelerated and the program contour is exited.

Case 4 JOG mode without transformation

- The axis stops at the software limit switch position.

Case 5 JOG mode with transformation

- The axis stops at the limit switch.
- The other axes involved in the transformation are decelerated and the specified contour is exited.

Case 6 Change of the valid software limit switch

- If the current position lies behind the new software limit switch when the software limit switch is switched over, the axis is decelerated with the maximum permissible axial acceleration.
- If the current position is in front of the software limit switch after the switchover, the procedure is as in Case 2.

© Siemens AG, 2003. All rights reserved

Response

The following responses are possible in each mode:

- If during the preparation of a block it is found that the axis position to be approached is greater or less than the positive/negative software limit switch, one of the following alarms is signaled:

  10720 “Software limit switch + or –”; Case 1
  10620 “Axis reached software limit switch + or –”; Cases 2, 3, 6

- If the position of a software limit switch is reached in JOG mode and further travel in this direction is programmed, alarm 10621 “Axis at software limit switch + or –” is signaled; Case 4
- If while traversing geometry axes in JOG mode it is found that the position of the geometry axis to be approached is greater than/less than the positive/negative software limit switch, alarm 10620 “Axis reached software limit switch + or –” is signaled; Case 5.
- If the monitoring function responds, the axes are decelerated.
  If an axis is involved in interpolation with other axes, these are also decelerated.
  A contour violation may occur; Cases 2, 3, 5
- Program processing is aborted; Cases 1, 2, 3
- The direction keys are disabled in the approach direction;
  Cases 1, 2, 3, 4, 5.

Remedy

- Perform mode group reset
- Move in the opposite direction (in JOG mode)
- Correct the program.
- Cancel DRF offset.

2.3.2 Working area limitation

Function

Working area limitations describe the area in which machining is possible. They enable the user to limit the traverse range of the axes in addition to the limit switches.
The limitations are referred to the base coordinate system, please see

References: /PG/, “Programming Guide: Fundamentals”

A check is made to see whether the tool tip P is within the protected working area. The value entered for the working area limitation is the final permissible permission for the axis.

The MD 21020: WORKAREA_WITH_TOOL_RADIUS (allowance for tool radius with working area limitation) can be set to determine whether the monitor is to take account of the tool radius.

One pair of values (minus/plus) is possible for each axis to describe the protected working area.

© Siemens AG, 2003. All rights reserved

Working area limitation

The working area limitation can be defined or changed in two ways:

– Via the operator panel in the “Parameters” operating area using the following setting data:
SD 43430: WORKAREA_LIMIT_MINUS (working area limitation minus)
SD 43420: WORKAREA_LIMIT_PLUS (working area limitation plus)
Changes are only possible in the reset state in AUTOMATIC mode and become active immediately.
In JOG mode, changes are always possible but they only become active when a new movement is started.

– In a program with G25/G26. Changes become active immediately.
A programmed motion has priority, it overwrites the value entered in the setting data and is retained after RESET and End of Program.

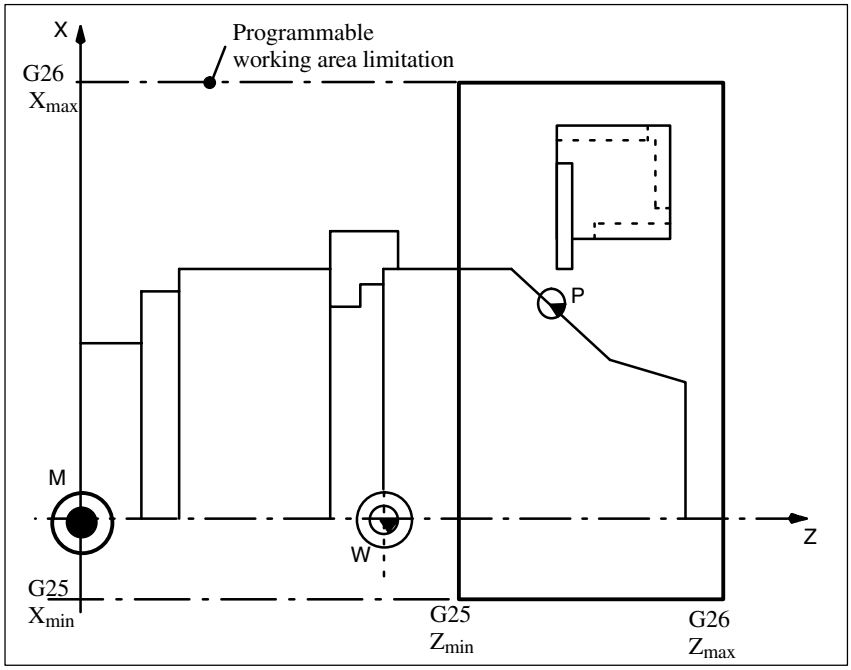


Fig. 2-9 Working area limitation

© Siemens AG, 2003. All rights reserved

Activation

- Via the SD: WORKAREA_MINUS_ENABLE, WORKAREA_PLUS_ENABLE (working area limitation in the negative or positive direction active), the working area limitation can be activated. Scanning begins after reference point approach.
- The monitoring function is no longer valid after a PRESET. The working area limits only become active again after reference point approach.
- During program execution, the working area limitation can be activated with the modal G code “WALIMON” and deactivated with “WALIMOF”. The initial setting for the G code (working area limitation activated or deactivated on NC start) can be set in MD 20150: GCODE_RESET_VALUES[n] (initial setting of G groups).
- The working area limitation is not active with endlessly turning axes, i.e. when MD 30310: ROT_IS_MODULO = 1.
(Modulo conversion for rotary axis and spindle).

Effect

Depending on the mode, the response to a violation of the working area limitation can be different:

Case 1 Automatic modes with/without transformation

– The block that violates the working area limitation is not started. The previous block is terminated properly.

Case 2 Automatic modes with overlaid motion

– The block that violates the working area limitation is started. Axes that are traversed with overlaid motion stop at the working area limitation.

Case 3 JOG mode with/without transformation

– The axis is positioned at the working area limitation and then stopped.

© Siemens AG, 2003. All rights reserved

Responses

The following responses are possible in each mode:

- If it is found during block preparation that the position the axis is to approach is greater than/less than the positive/negative working area limitation, one of the following alarms is signaled:

  10730 “Working area limitation + or –”; Case 1
  10630 “Axis reached working area limitation + or –”; Case 2

- If in JOG mode, the position of a working area limitation is reached and the intention is to continue traversing in the same direction, alarm 10631 “Axis at working area limitation + or –” is signaled; Case 3
- If the monitoring function responds, the axes are decelerated.
  If an axis is involved in interpolation with other axes, these are also decelerated.
  A contour violation may occur; Case 2
- Program processing is aborted; Cases 1, 2
- The direction keys are disabled in the approach direction;
  Cases 1, 2, 3

Remedy

- Perform mode group reset
- Check the working area limitation in the parts program (G25/G26) or in the setting data.
- Move the axis in question away from the working area limitation (JOG mode).

© Siemens AG, 2003. All rights reserved

2.4 Protection zones

2.4.1 General

What can be protected?

Various elements of the machine and its equipment and the workpiece being machined can be protected from incorrect movements with the protection zone function.

The following elements can be protected:

- Permanent parts of the machine and attachments (e.g. toolholding magazine, swiveling probe).

  Only the elements that can be reached by possible axis constellations are relevant.
- Moving parts belonging to the tool (e.g. tool, toolholder)
- Moving parts belonging to the workpiece (e.g. parts of the workpiece, clamping table, clamping shoe, spindle chuck, tailstock).

How is protection implemented?

Two and three dimensional protection zones are defined in the parts program or in system variables for the elements to be protected. These protection zones can be activated or deactivated in the parts program.

Note

During machining in operating modes JOG, MDA or AUTOMATIC, the control checks **whether the tool (and its protection zones) violate the protection zones of the workpiece.**

Protection zone monitoring is channel-specific, i.e. all the active protection zones of a channel monitor one another.

2.4.2 Types of protection zone

Workpiece/tool related protection zones

Protection zones are basically classed as workpiece-related or tool-related.

Depending on the machine construction, permanent machine parts to be protected are defined either as tool-related or workpiece-related protection zones.

© Siemens AG, 2003. All rights reserved

2- and 3-dimensional protection zones

It is possible to define 2-dimensional or 3-dimensional protection zones as polygons with a maximum of 10 corner points. The protection zones can also contain arc contour elements.

Polygons are defined in a previously defined plane.

Expansion in the third dimension can be limited between $-\infty$ and $+\infty$. The following definitions are possible:

- Third dimension of protection zone from $-\infty$ to $+\infty$
- Third dimension of protection zone from $-\infty$ to upper limit
- Third dimension of protection zone from lower limit to $+\infty$
- Third dimension of protection zone from lower limit to upper limit.

Protection zones relating to machine or channel

Protection zones can be defined as valid for one channel or for the whole machine.

Whether the protection zone is classified as machine-related or channel-related does not influence the way it operates, but only affects the area (one or all channels) for which it is programmed.

Application: Doubleslide turning machine

- The tool-related protection zones are assigned to channel 1 or 2.
- The workpiece-related protection zones are assigned to the machine.
- The coordinate system must be identical for both channels.

The maximum number of machine and channel-related protection zones can be defined in the following machine data:

- MD 18190: MM_NUM_PROTECT_AREA_NCK
  (number of files for machine-related protection zones)
- MD 28200: MM_NUM_PROTECT_AREA_CHAN
  (number of files for channel-specific protection zones)

Note

A maximum of 10 machine-specific and 10 channel-specific protection zones per channel can be defined.

The default setting is 0 protection zones.

A change in the default setting affects the memory allocation of the NC.

© Siemens AG, 2003. All rights reserved

Absolute and relative protection zones

The coordinates of a protection zone must always be programmed as absolute values with respect to the reference point of the protection zone. When the protection zone is activated via the NC parts program it is possible to offset the reference point of the protection zone.

Examples

Examples for

- tool-related,
- workpiece-related,
- two and three dimensional,
- absolute and relative

protection zones are shown below.

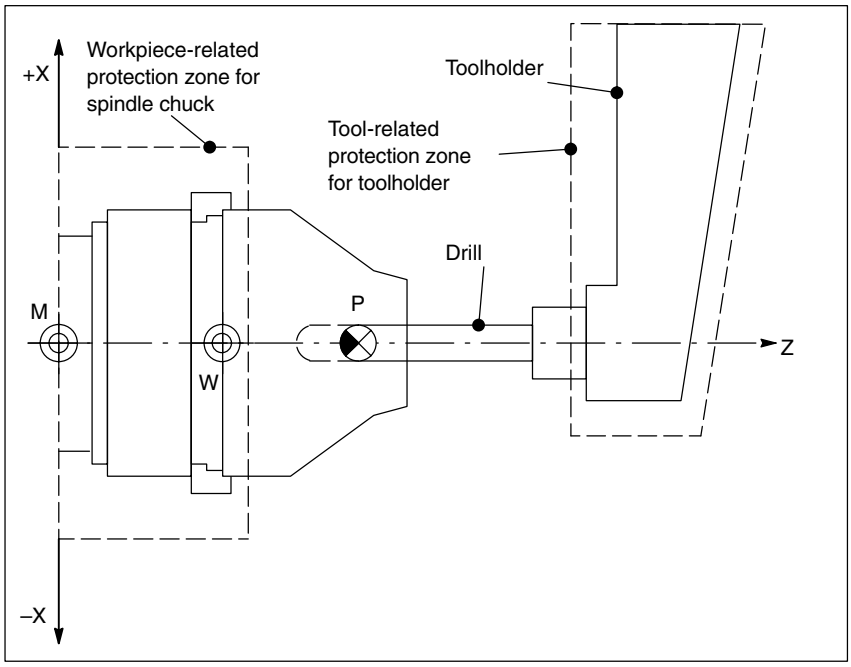


Fig. 2-10 Example of application on turning machine

© Siemens AG, 2003. All rights reserved

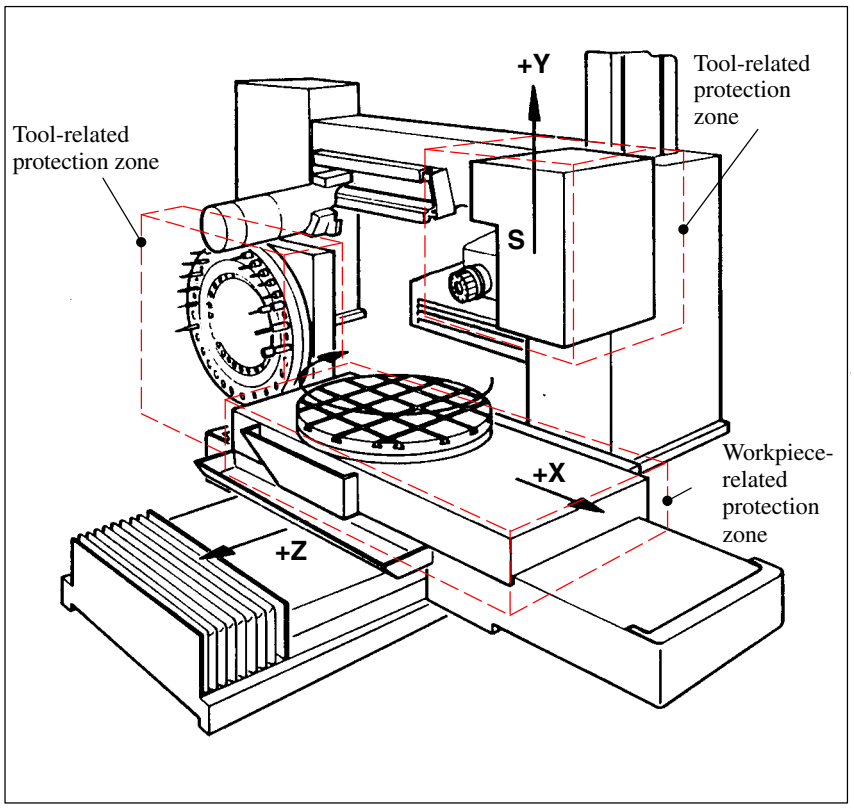


Fig. 2-11 Example of application on milling machine

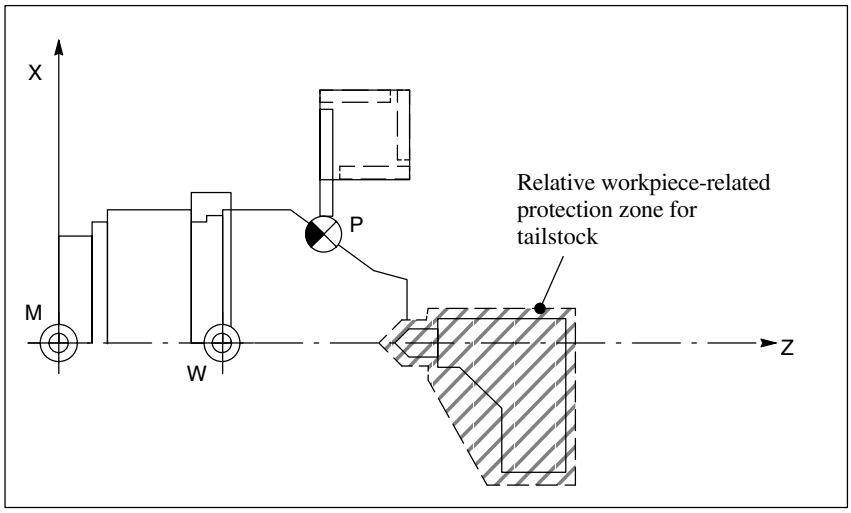


Fig. 2-12 Example of application on turning machine with relative protection zone for tailstock

© Siemens AG, 2003. All rights reserved

2.4.3 Coordinate system, orientation

Coordinate system, axis types

The space described by the protection zones must be in the Cartesian coordinate system. The protection zones are therefore described with geometry axes in the right-angled basic coordinate system (BCS) and thus in the coordinate system used for programming (with the exception of frames).

Coordinates for tool-related protection zones must be given as absolute values referred to the tool carrier reference point F.

Coordinates for workpiece-related protection zones must be given as absolute values referred to the zero point of the basic coordinate system.

Orientation

The orientation of the protection zones is determined by the plane definition (abscissa/ordinate) in which the contour is described and the axis perpendicular to the contour (vertical axis).

The orientation of the protection zones must be the same for the tool and workpiece-related protection zones.

2.4.4 Definition of protection zones

General

Protection zones can be defined in the parts program or with system variables.

A protection zone definition must contain the following information:

- Workpiece or tool-related protection zone
- Orientation of protection zone
- Type of limitation in the third dimension
- Upper and lower limit of the protection zone in the third dimension
- Protection zone active immediately (in the case of machine-related protection zones only possible via system variable)
- Up to 10 contour elements.

If defined in the parts program, the data for the protection zone are stored in the relevant system variables.

© Siemens AG, 2003. All rights reserved

Protection zone definition in the parts program

Protection zones are defined in an NC program as follows:

Definition beginning

Description of the protection zones starts with the subprogram:

CPROTDEF(n,t,applim,appplus,appminus)
for channel-specific protection zones

NPROTDEF(n,t,applim,appminus,appplus)
for machine-specific protection zones

The following is included:

CPROTDEF	Channel-specific protection area definition
NPROTDEF	NCK-specific protection area definition
n (int)	Number of the defined protection zone
t (Boolean)	TRUE: Tool-oriented protection zone FALSE: Workpiece-oriented protection zone
applim (int)	Type of limitation in the third dimension
0:	No limitation in the third dimension
1:	Limitation in the plus direction of the third dimension
2:	Limitation in the minus direction of the third dimension
3:	Limitation in the minus and plus direction of the third dimension
appminus (Real)	Value of limitation in the minus direction of the third dimension
appplus (Real)	Value of limitation in the plus direction of the third dimension

Note

appplus must be greater than appminus.

Orientation

The orientation of the protection zone is determined by the plane selection (G17, G18, G19). It may be necessary to store the plane selection before defining the protection zone and then to update it again.

Description of the contour

The contour description contains:

- Definition of the path type (straight line, circle in clockwise/counterclockwise direction)
- Definition of the target point (two values for the plane)
- For circle: definition of the interpolation parameters (2 values for the plane)

The protection zone contour is defined in the same way as a travel movement. The valid protection zone is **the zone left of the contour**. The contour of an internal protection zone must therefore described in the counterclockwise direction.

© Siemens AG, 2003. All rights reserved

The following are permissible:

- G0, G1 for straight contour elements
- G2 for circular segments in the clockwise direction
  (only permissible for workpiece-related protection zones because tool-related protection zones can only be convex).
- G3 for circular segments in the counterclockwise direction.

External protection zones (only possible for workpiece-related protection zones) must be defined in the clockwise direction.

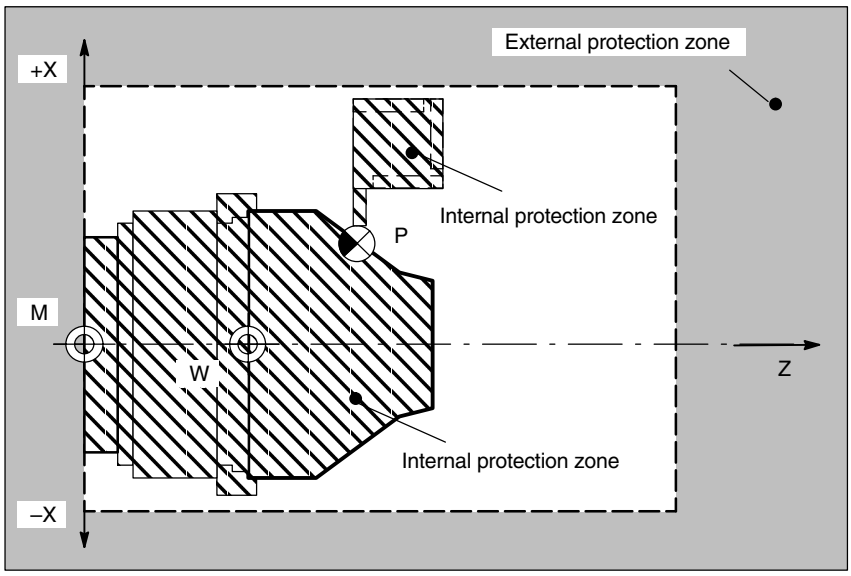


Fig. 2-13 External and internal protection zone

The last point in the contour definition (max. 11th point) must coincide with the first point of the contour definition.

In the case of a rotation-symmetrical contour (e.g. spindle chuck) the whole contour must be described (not merely the contour to the turning center).

Tool-related protection zones must be convex. If a concave protection zone is required, the protection zone must be divided up into several convex protection zones.

Note

- The NCU571 and FM-NC have no channel-specific zones (only 1 channel), but only NCK protection zones.
- A maximum of four contour elements (traversing motions) are available in the FM-NC, each of which can define one protection zone (max. 4 protection zones).

© Siemens AG, 2003. All rights reserved

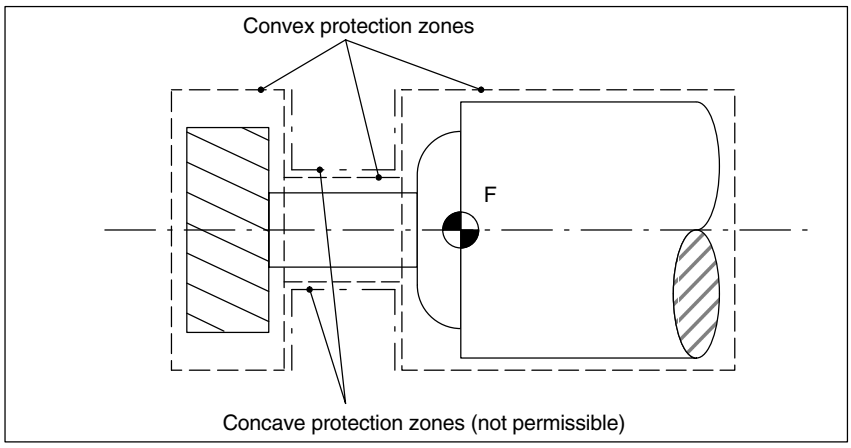


Fig. 2-14 Example of convex and concave tool-related protection zones

End of definition

The definition of a machine or channel-specific protection zone is completed with the subprogram **EXECUTE(n).**

n (int) Variable (of no significance for protection zones)

All the blocks between the commands CPROTDEF or NPROTDEF and EXECUTE are non-executable blocks.

Note

The following applies to contour definition

- The contour definition between CPROTDEF or NPROTDEF and EXECUTE is an independent geometry definition. It has no relation to previous and following geometry definitions.
- Programmable frames (TRANS, ROT, SCALE, MIRROR) and settable frames (G54 and G57) have no effect.
- Inch/metric switchovers with G70/G71 or G700/G710 are effective.
- Reference point approach G74, travel to fixed stop G75, dwell time G4, block search stop STOPRE and the M functions M0, M1, M2, M17, M30 are not permitted in the contour definition.
- G40 (no tool radius compensation) must be active.

Protection zone definition with system variables

Protection zones can also be defined via the following system variables.

The same conditions apply for this type of contour definition as apply for contour definition in the parts program.

© Siemens AG, 2003. All rights reserved

System variables	Meaning	Type
$SN_PA_ACTIV_IMMED[n] $SC_PA_ACTIV_IMMED[n]	**Identifier for "Active immediately after referencing"** The protection zone is active immediately after power up of the control and referencing of the axes. FALSE: Protection zone is not immediately active TRUE: Protection zone is immediately active	BOOL
$SN_PA_T_W[n] $SC_PA_T_W[n]	**Identifier for "Workpiece or tool-related protection zone"** 0: Workpiece-related protection zone 1: Reserved 2: Reserved 3: Tool-related protection zone	INT
$SN_PA_ORI[n] $SC_PA_ORI[n]	**Identifier for protection zone orientation** 0: Polygon definition in plane of the 1st and 2nd geometry axes 1: Polygon def. in the plane of the 2nd and 1st geometry axes 2: Polygon def. in the plane of the 2nd and 3rd geometry axes	INT
$SN_PA_LIM_3DIM[n] $SC_PA_LIM_3DIM[n]	**Identifier for limitation of the protection zone** in the axis perpendicular to the polygon definition 0: No limitation ($-\infty < x < +\infty$) 1: Limitation in pos. direction ($-\infty < x <$ + limit) 2: Limitation in neg. direction (– limit $< x < +\infty$) 3: Limitation in both directions (– limitation $< x <$ + limitation)	INT
$SN_PA_PLUS_LIM[n] $SC_PA_PLUS_LIM[n]	**Plus limitation of protection zone** in the axis perpendicular to the polygon definition	REAL
$SN_PA_MINUS_LIM[n] $SC_PA_MINUS_LIM[n]	**Minus limitation of protection zone** in the axis perpendicular to the polygon definition	REAL
$SN_PA_CONT_NUM[n] $SC_PA_CONT_NUM[n]	Number of valid contour elements	INT
$SN_PA_CONT_TYP[n,i] $SC_PA_CONT_TYP[n,i]	**Contour type[i]** Contour type (G1, G2, G3) of the ith contour element	INT
$SN_PA_CONT_ABS[n,i] $SC_PA_CONT_ORD[n,i]	**End point of the contour[i]** Ordinate value	REAL
$SN_PA_CONT_ORD[n,i] $SC_PA_CONT_ABS[n,i]	**End point of the contour[i]** Abscissa value	REAL
$SN_PA_CENT_ABS[n,i] $SC_PA_CENT_ORD[n,i]	**Center point of the circular contour[i]** Absolute ordinate value	REAL
$SN_PA_CENT_ORD[n,i] $SC_PA_CENT_ABS[n,i]	**Center point of the circular contour[i]** Absolute abscissa value	REAL

$SN_ ... are system variables for NCK-specific protection zones.
$SC_ ... are system variables for channel-specific protection zones.

The index "n" corresponds to the number of the protection zone and starts with "0" (≙ 1st protection zone)

The index "i" corresponds to the number of the contour element. The index starts with "0" (≙ 1st contour element). The contour elements must be defined in ascending order.

Note

The system variables are not restored again during reorganization. This also applies if the data have been written with CPROTDEF, EXECUTE.

© Siemens AG, 2003. All rights reserved

Data storage

The data blocks containing the protection zone definitions are **stored** in the following files:

File	Blocks
_N_NCK_PRO	Data block for NCK-specific protection zones
_N_CHAN_PRO	Data block for channel-specific protection zones in channel 1
_N_CHAN_PRO	Data block for channel-specific protection zones in channel 2

The data blocks are also **backed up** in the following files:

File	Blocks
_N_INITIAL_INI	All data blocks of the protection zones
_N_COMPLETE_PRO	All data blocks of the protection zones
_N_CHAN_PRO	All data blocks of the channel-specific protection zones

2.4.5 Activation and deactivation of protection zones

General

Protection zones can be “preactivated”, “activated” and “deactivated”. A protection zone is monitored for violation only when it is activated.

Protection zones are usually **activated** in the parts program. Protection zones can also be activated on power up and by the PLC user program. If activated via the PLC, the protection zone must first be preactivated in the NC parts program.

Protection zones that are activated in the parts program are always active and do not require a PLC signal to become effective.

Preactivation, activation and deactivation of protection zones is performed channel-specifically, even for machine-related protection zones.

(Pre)activated machine-related and channel-specific protection zones are only active in the channel in which they are activated.
Machine-related protection zones activated during power up become active in all channels.

A protection zone can also be active in several channels at once (e.g. quill with two parallel slides → doubleslide single-spindle machine).

Protection zones are not taken into account until all the geometry axes involved have been referenced.

The status of the protection zones (deactivated, preactivated, activated) is maintained after program end or reset.

Protection zones can only be **deactivated** in the parts program.

Setting active protection zones

Individual basic frames can be deleted with MD 28210:
MM_NUM_PROTECT_AREA_ACTIVE the maximum number of protection zones active simultaneously in a channel is defined.
A maximum of ten channel-specific or machine-specific protection zones can be active at any one time.

© Siemens AG, 2003. All rights reserved

The default setting is 0 protection zones. A change in the default setting affects the memory allocation of the NC.

In SW 6.4 and higher, MD 28212: MM_NUM_PROTECT_AREA_CONTUR can be used to control the dynamic memory required for partoriented protection zones with a large number of inside contours. The default setting leaves no memory reserve. The value of this machine data must be increased if it is displayed by an alarm 14710 when further protection zones are being activated.

Error on activation Activation

When a protection zone is activated, the corresponding system data are checked and the protection zone is not activated if an error is detected. Possible errors are:

- Incomplete or contradicting contour definition.
- A tool-related protection zone that is not convex.
- Protection zone number does not exist.
- The maximum possible number of active protection zones has been exceeded.
- The protection zones have different orientations.

Deactivation, preactivation, activation via parts program

For this, the following predefined subprogram calls are used:

CPROT (n,state,xMov,yMov,zMov)
for channel-specific protection zones

NPROT (n,state,xMov,yMov,zMov)
for machine-specific protection zones

The following is included:

CPROT	Channel-specific protection area on
NPROT	NCK-specific protection area on
n (int)	Number of the protection zone
state (int)	Status activated
0:	Deactivate protection zone
1:	Preactivate protection zone
2:	Activate protection zone
xMov, yMov, zMov (Real)	Offset of protection zone already defined in the geometry axes

(Pre)activation is possible with or without offset of the protection zone in the 1st, 2nd or 3rd dimension.

The offset is defined with reference to the toolholder reference point F (for tool-specific protection zones) or machine zero (for workpiece-related protection zones).

Note

A protection zone cannot be activated in a single channel with different offsets simultaneously.

© Siemens AG, 2003. All rights reserved

Activation via PLC user program

Preactivated protection zones can be made operative by the PLC user program via the PLC interface. For example, the relevant protection zone is activated and monitored before a tool sensor is moved into the working range.

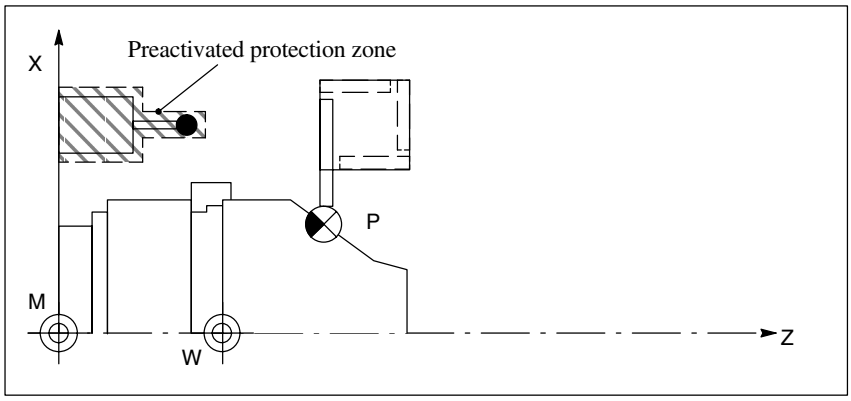


Fig. 2-15 Example for lathe: Preactivated protection zone for tool probe

Protection zones are preactivated and a position offset defined in the parts program.

The channel-specific IS “Machine-related protection zone 1 (...10) preactivated” (DB21, ... DBX272.0 – 273.1) or “Channel-specific protection zone 1 (...10) preactivated” (DB21, ... DBX274.0 – 275.1) are sent to the PLC to confirm which protection zones are preactivated in the current block of the parts program.

The PLC is also told which

- activated protection zones in the current block are violated,
- preactivated protection zones would be violated in the current block if the preactivated protection zone were to be activated by the PLC.

The information is transferred via the channel-specific IS “machine-related protection zone 1 (...10) violated” (DB21, ... DBX276.0 to DBX277.1) or “channel-specific protection zone 1 (...10) violated” (DB21, ... DBX278.0 bis DBX279.1).

It is thus possible, for example, before moving parts into the working range to check whether the tool (or workpiece) is located in the area into which the part is to be moved.

Protection zones preactivated by the NC can be made active or inactive via the PLC. When they are made “active”, it is not possible to specify the absolute position of the protection zone reference point. This is done when the protection zone is preactivated.

The PLC uses channel-specific IS “Activate machine-related protection zone 1 (...10) ” (DB21, ... DBX8.0 to DBX9.1) or “Activate channel-specific protection zone 1 (...10)” (DB21, ... DBX10.0 to DBX11.1) to activate or deactivate protection zones.

Protection zones activated from the parts program cannot be deactivated by the PLC program.

© Siemens AG, 2003. All rights reserved

Note

It therefore follows that protection zone numbers have to be programmed in the PLC user program which are then activated via the PLC. Preactivation in the parts program is only useful for these protection zones. It only makes sense to activate protection zones in the parts program which are introduced by the NC programmer and not entered in the PLC program.

Activation on control system power up

System variables
$SN_PA_ACTIV_IMMED[n] or **$SC_PA_ACTIV_IMMED[n]** can be programmed to (pre)activate machine-related and channel-specific protection zones immediately after power up and subsequent referencing.

This activates machine-related protection zones in all channels.

A relative offset cannot be programmed.

Like the other protection zones, these protection zones, too, can be preactivated, activated and deactivated in the parts program.

Note

If no tool-related protection zone is active, the tool path is checked against the workpiece-related protection zones.

If no workpiece-related protection zone is active, the protection zone is not checked.

Block search

In block search with calculation, the last programmed state of a protection zone (deactivated, preactivated, activated) is taken into account.

Program test

(Pre)activated protection zones are monitored during program testing (in Automatic mode).

© Siemens AG, 2003. All rights reserved

2.4.6 Protection zone violation and temporary enabling of individual protection zones

General behavior

Workpiece and tool-related protection zones that are activated or deactivated are monitored for collision. If a protection zone violation is detected, behavior in the individual operating modes is as follows.

End of the temporary enabling

Temporary enabling of a protection zone is terminated

- NC-RESET
- if an end of block lies outside the protection zone in AUTOMATIC and MDA modes
- if an end of movement lies outside the protection zone in the manual modes
- another protection zone has been activated and included.

On RESET, all the enabled protection zones become active again. If the program is started again or jog is introduced, the protection zones must be reenabled. If the current position lies within a protection zone that becomes active again after RESET, this protection zone must be enabled again on the first path movement.

Preactivated protection zones

Protection zones can be preactivated by the NC program. To make them fully operative, they must also be set to the “operative” state by the PLC.

In contrast to the AUTOMATIC mode, a change in the PLC interface signals “Make preactivated protection zones operative” only has an effect on stationary axes in the geometry system. This means: If an inoperative protection zone is made “operative” once a motion has been started, it is not evaluated until the axes have stopped, possibly resulting in the output of an alarm.

However, the warning signal “Protection zone monitoring not guaranteed” and the PLC interface signal “Protection zone monitoring not guaranteed” are output if a preactivated protection zone is made “operative” while a traversing motion is in progress.

Deactivation of tool-related protection zones

Tool-related protection zones can be deactivated only in the parts program or, if they have been preactivated, by being rendered “inoperative” by the PLC.

© Siemens AG, 2003. All rights reserved

Geometry axis replacement

Machine data can be used to configure protection zones and working area limitations for geometry axis replacement.

- In SW 6.1 and higher, protection zones can be activated with:
  MD 10618: PROTAREA_GEOAX_CHANGE_MODE, bit 1 = 1
- In SW 6.3 and higher, working area limitations can be activated with:
  MD 10604: WALIM_GEOAX_CHANGE_MODE, bit 0 = 1

If functions such as protection zone or working area limitation are deactivated during geometry axis replacement, the bits should be set to zero.

Frames when changing over geometry axes – see Frames
References: /FB1/, K2 “Axes, Coordinate Syst., Frames”.

Transformation changeover

Protection zones can also be activated during transformation changeover using machine data MD 10618: PROTAREA_GEOAX_CHANGE_MODE, bit 0 = 1. Bit 0 = 0 deactivates this function.

Monitoring overlaid motion

Note

Axes that have been assigned to another channel are not taken into account. The last position to be approached is taken to be the end position. It is not taken into account whether the axis has traversed after changing channels.

© Siemens AG, 2003. All rights reserved

Behavior in operating modes

Automatic mode and MDA

Protection zones are not violated in Automatic modes:

- If the movement in a block is from outside into the protection zone (N30), deceleration is performed by the end of the previous block (N20) and the movement is stopped (see Fig. below).
  - If the protection zone is preactivated but not activated by the PLC, machining is continued (case 1).
  - If the protection zone is activated or preactivated and activated by the PLC, machining is stopped (case 2).
- If the starting point of the block is inside the protection zone, the movement is not started.

If the protection zone is violated, the alarm
10700 “NCK protection zone violated in Automatic or MDA” or
10701 “Channel-specific protection zone violated in Automatic or MDI” is triggered for the workpiece-related protection zone.

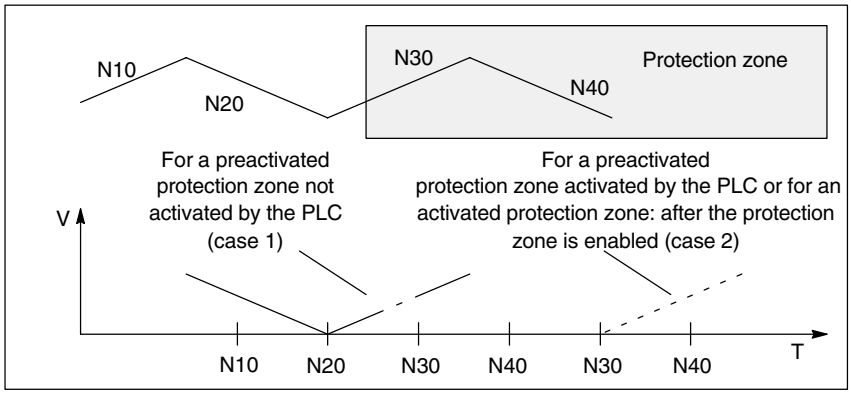


Fig. 2-16 Behavior of the path velocity when entering a protection zone

Overlaying several axis motions

Overlaid motions of external ZO (zero offsets) or DRF are taken into account by protection zones if they are executed at a sufficiently early point in time.

If an overlaid motion occurs while a protection zone is active or operative, an alarm is output as a warning. This alarm has no effect on the machining operation and resets itself if the transferred motion has been fully taken into account. The PLC signal “Protection zone monitoring not guaranteed” is set at the same time as the alarm.

Enabling of workpiece-related protection zones

When a workpiece-related protection zone has been violated, the operator can enable it temporarily with NC start in the AUTOMATIC and JOG modes so that it can be crossed. The alarm is reset and

- the protection zone is entered in AUTOMATIC and MDA modes.

© Siemens AG, 2003. All rights reserved

Only workpiece-related protection zones can be enabled temporarily with NC start and entered by all tool-related protection zones including the programmed path.

If, on NC start, the preactivated tool or workpiece-related protection zone is deactivated by the PLC after the alarm, machining is continued without the protection zone being enabled temporarily.

If a fully operative, preactivated protection zone causes a machining interruption and the output of an alarm owing to protection zone violation, machining can be resumed on NC start if the PLC makes the zone inoperative again.

If enabling of a protection zone is to be safeguarded better than with a simple NC start, NC start must be disabled or made dependent on other conditions in the PLC user program when this alarm is triggered.

If the user does not want to permit entry into the protection zone, he can terminate the traversing movement with NC RESET.

If several protection zones are violated at the same time by the movement, acknowledgment is required for each of these protection zones. With NC start the individual protection zones can then be enabled one after the other.

Application for temporary enabling:

- Drilling a turned part: The drill is allowed to enter the protection zone of the spindle chuck.

Monitoring of overlaid motion

On preparation of the NC blocks, part of the offsets of geometry axes resulting from the overlaid motions are taken into account.

If further offsets occur that could not be taken into account on preparation of the blocks, the channel-specific PLC interface signal “Protection zone monitoring not guaranteed” is set. This signal is set while offsets are active that cannot be taken into account. The signal can be set and reset within a block.

Simultaneously to the PLC interface signal, a self-canceling alarm “Protection zones not guaranteed” is output.

The following overlaid motions of geometry axes are taken into account in the preparation of blocks:

1. DRF offsets
2. Work offsets external
3. Fine tool offsets
4. Rapid retraction
5. Offsets generated by compile cycles
6. Oscillation
7. Concurrent positioning axes
8. Positioning axes

© Siemens AG, 2003. All rights reserved

The alarm is canceled or the PLC interface signal reset when the offsets from the overlaid motions are taken into account again or when the offsets are reduced to zero again.

Note

The end position for positioning axes is taken to be a position in the whole block. This means that the alarm “Protection zones not guaranteed” is output when the positioning axis starts to move.

The overlaid motions themselves are not limited, nor is there any intervention in processing of the program.

Behavior in JOG mode

Overlaying several axis motions

Simultaneous JOG traversing movements of several geometry axes are permitted.

Effective monitoring of the protection zones is no longer reliable. A warning “Protection zone monitoring not guaranteed” and the interface signal “Protection zone monitoring not guaranteed” are therefore set.

The motion of the axes is limited in all directions by the protection zones with the same effect as they had at the start point.

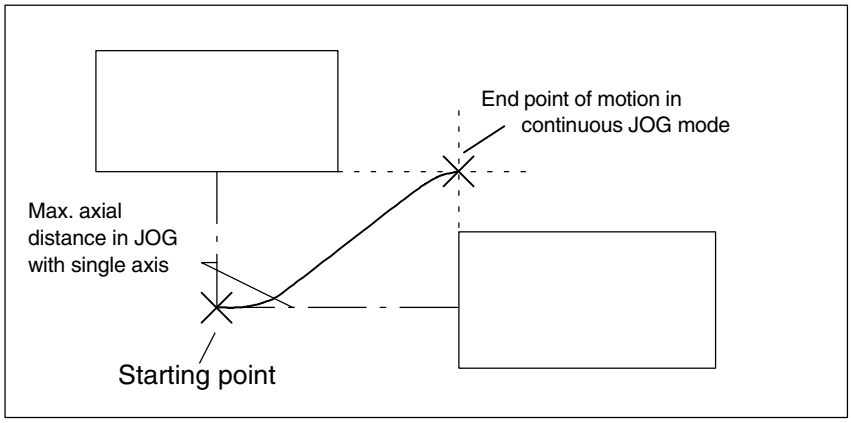


Fig. 2-17 Limitation of motion through protection zones in JOG mode with several axes

After the axes in the geometry system have completed their motions (end of interpolation), the warning is reset and the final position checked to see whether it is within one or several protection zones.

© Siemens AG, 2003. All rights reserved

There are three possible situations in this case:

1. If the position is **outside** all active **protection zones**, the next traversing motion can be started normally. The appropriate PLC interface signals “machine-specific or channel-specific protection zone violated” are set for the protection zones that are enabled or just preactivated, but not yet operative.
2. If the position is within an active protection zone, the alarm “Protection zone violated in JOG” is generated, thereby disabling any traversing motions. The appropriate PLC interface signals “Machine-specific or channel-specific protection zone violated” are also set. The alarm is reset by
   a) temporary enabling of the affected protection zones
   b) deactivation of the relevant protection zones if they are preactivated
   c) deactivation of the protection zone in MDA.
3. If the position is on the protection zone limitation (position is still valid), no alarm is generated.

Note

While any one axis in the geometry system is still oscillating, the status “Motions of axes in geometry system completed” cannot be reached. The warning remains active, the other axes in the geometry system can continue to traverse.

The alarm “Protection zone reached in JOG” is not output if the motion of the first axis to be started is terminated by the limitation determined prior to the motion.

Manual modes

(Pre)activated protection zones are also monitored in manual modes (JOG, INC, handwheel).

Limitation of traversing motion of an axis

Axis motions are limited in the JOG mode by means of software limit switches or the working area limitation. The protection zones are an additional limiting element on the traversing motion of the geometry axes.

If the traversing motion of an axis is limited because it has reached a protection zone, then a self-resetting alarm “Protection zone reached in JOG” is generated. The alarm text specifies the violated protection zone and the relevant axis. It is assured that no protection zone will be violated when **one** axis is traversing in JOG. (This response is analogous to approaching software limit switches or a working area limitation).

The alarm is reset

1. when an axis is traversed along a path that does not lead into the protection zone,
2. when the protection zone is enabled and
3. on NC Reset.

© Siemens AG, 2003. All rights reserved

If an axis starts to move towards a protection zone when it is at a protection zone limit, then a selfresetting alarm “Protection zone reached in JOG” is output and the motion is not started.

Enabling of workpiece-related protection zones

When a workpiece-related protection zone has been violated, the operator can also enable it temporarily in JOG mode so that it can be crossed. The alarm is reset and

- the movement started in the manual modes after a new travel command.

Temporary enabling of protection zones

Protection zones can be enabled in JOG mode

1. if the current position is within a protection zone (alarm active) or
2. if a motion is to be started on the protection zone limit (alarm active).

A protection zone is enabled if

- a positive signal edge arrives at PLC interfaces “Temporary enabling of protection zones”. This enabling signal resets the active alarm.
- if the axis then starts to move again into the same protection zone.

Initiation of the motion causes

- the protection zone to be enabled,
- the appropriate PLC interface signals “Machine-specific or channel-specific protection zone violated” to be set and
- the axis to start moving.

The enabling signal is canceled if a motion is started that does not lead into the enabled protection zone.

If the current position is located in other active protection zones or the limit for other protection zones must be crossed with the motion that has been started, then alarms 10702, 10703 or 10706, 10707 are output. The PLC interface signal “Temporary enabling of protection zones” can be set again to enable the protection zone for which an alarm is output.

The enabling signals for the individual protection zones are still valid on switchover to operating modes AUTOMATIC or MDA, and vice versa, the enabling signals of protection zones that were output in AUTOMATIC and MDA remain valid.

If the end position is located outside the relevant protection zone the next time the axes in the geometry system stop, then

- the enabling signals of the individual protection zones are canceled and
- the appropriate PLC interface signal “Machine-specific or channel-specific protection zone violated” is reset.

© Siemens AG, 2003. All rights reserved

2.4.7 Limits of protection zones

Area of application

Protection zones must be defined for geometry axes in the rectangular basic coordinate system. In addition, all protection zones must have the same orientation.

If these conditions are fulfilled, protection zone monitoring can be performed for example for machines with the following configurations:

- Three-axis milling machine
- Boring mill with constant drilling orientation
- Lathes without transformation
- The following is possible for lathes with Transmit and peripheral surface transformation:
  - Machine parts with protection zones defined in symmetry with rotation around the spindle axis (a DRF offset must not be active in this case).
  - Protection zones on the workpiece to be machined
- Perpendicular knee-type milling machine where the protection zone is defined around the knee table or the tool holder.

No tool protection zone monitoring is possible for machines with the following configurations:

- Milling machines with orientation axes
- On lathes with Transmit or peripheral surface transformation no fixed machine parts can be protected except for
  - Machine parts with protection zones defined in symmetry with rotation around the spindle axis (a DRF offset must not be active in this case).
- Laser machines with orientation axes
- If several tools are active simultaneously and these must monitor one another.

Positioning axes

Only the end point is checked for geometry axes that traverse as positioning axes, the alarm “Protection zone monitoring not guaranteed” is generated during the motion.

Axis exchange

If an axis is not currently active in a channel because of the axis exchange function, the position of the axis last approached in the channel is taken as the currently valid position. If this axis has not yet been traversed in the channel, axis position = 0 is taken as the position.

If is not possible to check the actual positions of the axis.

© Siemens AG, 2003. All rights reserved

Machine-related protection zones

Machine-related protection zones must also be defined in the BCS. If a machine-related protection zone is to be active is different channels, the coordinate system of the geometry axes must have the same position in the basic coordinate system in all these channels. If this is not the case, the protection zone has different physical positions on the machine in the different channels. This is of special relevance to the activation of machine-related protection zones in the power up phase because these protection zones are then active in all the channels.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Supplementary Conditions 3

3.1 Axis monitoring functions

To ensure that the monitoring functions respond correctly, it is important that the correct values are entered in the following machine data:

- MD 31030: LEADSCREW_PITCH (leadscrew pitch)
- Gear ratio (load gearbox, encoder)
  MD 31050: DRIVE_AX_RATIO_DENOM[n] (denominator load gearbox)
  MD 31060: DRIVE_AX_RATIO_NUMERA[n] (numerator load gearbox)
  MD 31070: DRIVE_ENC_RATIO_DENOM[n] (denominator measuring gearbox)
  MD 31080: DRIVE_ENC_RATIO_NUMERA[n] (numerator measuring gearbox)
- MD 32810: EQUIV_SPEEDCTRL_TIME[n]
  (equivalent time constant speed control loop for feedforward control)
- Relationship between output voltage/output speed
  (applies only to analog drives)
  MD 32260: RATED_VELO (rated motor speed)
  MD 32250: RATED_OUTVAL (rated output voltage)
- Encoder resolution

The associated machine data are described in:

References: /FB/, G2, “Velocities, Setpoint/Actual Value Systems, Closed-Loop Control”

3.2 Protection zones

Protection zone monitoring is operative only if the relevant axes are referenced. These are the axes of the geometry system in the case of the protection zones.

Availability

The function “Protection zones” is available with

- SINUMERIK 840D with NCU 571/572/573 with SW 2 and higher
- SINUMERIK 810D, SW 1 and higher
- SINUMERIK FMNC with NCU 570 with SW 2 and higher

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

4.1 General machine data

10604 MD number	**WALIM_GEOAX_CHANGE_MODE** Working area limitation when replacing geometry axes		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 6.3		
Significance:	This machine data specifies whether any active protection zones must be retained or deactivated when a geo axis is replaced. The machine data values have the following meanings: Bit 0 = 0: Working area limitation deactivated during geo axis replacement. Bit 0 = 1: Working area limitation activated during geo axis replacement.		

10618 MD number	**PROTAREA_GEOAX_CHANGE_MODE** Protection zone for switchover of geo axes		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 6.1		
Significance:	This machine data specifies whether any active protection zones must be retained or deactivated when a transformation is changed or a geo axis exchanged. The machine data is bitcoded with the following bit significance: Bit 0 = 0: Protection zones are deactivated when a transformation is changed. Bit 0 = 1: Active protection zones remain active after a transformation is changed. Bit 1 = 0: Protection zones are deactivated when a geo axis is exchanged. Bit 1 = 1: Active protection zones remain active when a geo axis is exchanged.		

4.2 Channel-specific machine data

21020 MD number	**WORKAREA_WITH_TOOL_RADIUS** Allowance for tool radius with working area limitation		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 3.1		
Significance:	= 1: The tool radius is taken into account with respect to the working area limitation. = 0: The tool radius is not taken into account.		

© Siemens AG, 2003. All rights reserved

4.3 Axis/Spindle-specific machine data

4.3.1 Axis monitoring functions

36020 MD number	**POSITIONING_TIME** Time delay exact stop fine		
Default setting: 5	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	This MD defines the time after which the following error must have reached the limit value for exact stop fine when approaching a position (position partial setpoint=0 at the end of the movement). If this does not happen, alarm 25080 “Positioning monitoring” is triggered and the axis involved is brought to a standstill. The value set in the MD should be large enough to prevent triggering the monitoring function in normal operation because the whole traversing operation (acceleration, constant travel, deceleration) is monitored extensively by other functions.		
Related to	MD 36010: STOP_LIMIT_FINE (exact stop fine)		

36030 MD number	**STANDSTILL_POS_TOL** Zero speed tolerance		
Default setting: 0.2	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	This MD provides the tolerance band for the following monitoring functions: • On completion of a motion block (partial position setpoint=0 at the end of the movement) it is monitored if the following error has reached the threshold for the STANDSTILL_POS_TOL (zero speed tolerance) after the programmable STANDSTILL_DELAY_TIME (delay time zero speed monitoring). • At the end of a positioning operation (exact stop fine reached), the positioning monitor is replaced by the zero speed monitor. In this case, the axis is monitored to ascertain whether it has moved out of position by more than the distance set in MD: STANDSTILL_POS_TOL (zero speed tolerance). If the setpoint position zero speed tolerance has been violated in any direction, alarm 25040 “Zero speed monitoring” is triggered and the axis is brought to a standstill.		
Special cases, errors, ...	The zero speed tolerance must be greater than the “Exact stop limit coarse”.		
Related to	MD 36040: STANDSTILL_DELAY_TIME (delay time zero speed monitoring)		

36040 MD number	**STANDSTILL_DELAY_TIME** Delay time zero speed monitoring		
Default setting: 0.2	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	See MD 36030: STANDSTILL_POS_TOL (zero speed tolerance)		
Related to	MD 36030: STANDSTILL_POS_TOL (zero speed tolerance)		

© Siemens AG, 2003. All rights reserved

36050 MD number	**CLAMP_POS_TOL** Clamping tolerance with interface signal “Clamping active”		
Default setting: 0.5	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: mm; degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	IS “Clamping active” (DB31, ... DBX2.3) activates terminal monitoring. If the monitored axis has been pushed out of the setpoint position (exact stop limit) by more than the clamping tolerance, alarm 26000 “Clamping monitoring” is triggered and the axis is stopped.		
Special cases, errors, ...	The clamping tolerance must be larger than the “Exact stop limit coarse”.		
Related to	IS “Clamping active” (DB31, ... DBX2.3)		

36052 MD number	**STOP_ON_CLAMPING** Special functions for clamped axis		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0x07	
Changes effective after NEW_CONF	Protection level: 1 / 2	Unit: –	
Data type: BYTE	Applies from SW 6.4		
Significance:	Specifies how a clamped axis is to be treated. Bit 0 = 0: Must a clamped axis be moved again in continuous-path mode, measures must be taken in the parts program to ensure that the path axes are stopped to allow time for the axis to be released from the clamp. Bit 0 = 1: Must a clamped axis be moved again in continuous-path mode, measures must function stops the path motion in advance when necessary until the clamped axis can be moved again by the position controller, i.e. until the servo enable signal is set again. Bit 1 = 0: Must a clamped axis be moved again in continuous-path mode, measures must released from the clamp without application of Look Ahead. Bit 1 = 1: Must a clamped axis be moved again in continuous-path mode, measures must for the relevant axis is issued in the G0 blocks immediately in front of it to ensure that the PLC releases the axis clamp. Bit 1 is relevant only if bit 0 is set. Bit 2 = 0: If an axis must be clamped in continuous-path mode, measures must be taken in the parts program to ensure that the path axes are stopped to leave enough time for setting the clamp. Bit 2 = 1: If an axis must be clamped in continuous-path mode, the Look Ahead function stops the path motion before the next “NonG0 block” if the axis has not been clamped by then, i.e. the PLC has set the feedrate override value to zero.		
Related to	MD 36050: CLAMP_POS_TOL		

© Siemens AG, 2003. All rights reserved

36100 MD number	**POS_LIMIT_MINUS** 1st software limit switch minus		
Default setting: – 100 000 000	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: mm; degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	As for 1st software limit switch plus but the traversing range limitation is in the negative direction. The MD becomes active after reference point approach if PLC interface signal 2nd software limit switch minus is not set.		
MD irrelevant for ...	If axis is not referenced.		
Related to	IS “2nd software limit switch minus” (DB31, ... DBX12.2)		

36110 MD number	**POS_LIMIT_PLUS** 1st software limit switch plus		
Default setting: 100 000 000	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: mm; degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	A software limit switch can be activated in addition to a hardware limit switch. The absolute position in the machine axis system of the positive range limit of each axis is entered. The MD is active after reference point approach if PLC interface signal “2nd software limit switch plus” has not been set.		
MD irrelevant for ...	If axis is not referenced.		
Related to	IS “2nd software limit switch plus” (DB31, ... DBX12.3)		

36120 MD number	**POS_LIMIT_MINUS** 2nd software limit switch minus		
Default setting: – 100 000 000	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: mm; degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	As for 2nd software limit switch plus but the traversing range limitation is in the negative direction. An interface signal from the PLC can select which of the two software limit switches 1 or 2 is to be active. e.g. DB31, DBB12 Bit 2 = 0 “1st software limit switch minus” active for 1st axis 12 Bit 2 = 1 “2nd software limit switch minus” active for 1st axis		
MD irrelevant for ...	If axis is not referenced.		
Related to	IS “2nd software limit switch minus” (DB31, ... DBX12.2)		

36130 MD number	**POS_LIMIT_PLUS** 2nd software limit switch plus		
Default setting: 100 000 000	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: mm; degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	With this machine data a 2nd software limit switch position in the positive direction can be defined. An interface signal from the PLC can select which of the two software limit switches 1 or 2 is to be active. e.g.: DB31, DBB12 Bit 3 = 0 “1st software limit switch plus” active for 1st axis 12 Bit 3 = 1 “2nd software limit switch plus” active for 1st axis		
MD irrelevant for ...	If axis is not referenced.		
Related to	IS “2nd software limit switch plus” (DB31, ... DBX12.3)		

© Siemens AG, 2003. All rights reserved

36200 MD number	**AX_VELO_LIMIT**[**n**] Threshold value for velocity monitoring		
Default setting: 11500	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: mm/min rev/min	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The threshold value for the actual velocity monitoring function is entered in this MD. If the axis has at least one active encoder and this is operating below its limit frequency, alarm 25030 “Actual velocity alarm limit” is activated and the axes brought to a standstill when the threshold value is exceeded. Settings: • For axes, a value that lies 1015% above MD: MAX_AX_VELO (maximum axis velocity) should be chosen. If temperature compensation, set in MD: TEMP_COMP_TYPE (temperature compensation) is active, the maximum axis velocity is increased by an additional factor which results from MD: COMP_ADD_VELO_FACTOR (velocity overshoot due to compensation). The following rule must therefore be applied when programming the velocity monitoring threshold value: MD: AX_VELO_LIMIT[n] > MD: MAX_AX_VELO * (1,1 ... 1,15 + MD: COMP_ADD_VELO_FACTOR) • For spindles a value should be selected for each gear stage which is 10–15% higher than MD: GEAR_STEP_MAX_VELO_LIMIT[n] (maximum speed of gear stage). Coding of machine data index [n]: [closed-loop control set of parameters set no.]: 0–5 Please refer to the following for the effect of control parameters sets: **References:** /FB/, G2, “Velocities, Setpoint/Actual Value Systems, Cycle Times”		

36220 MD number	**CTRLOUT_LIMIT_TIME**[**n**] Delay time for speed setpoint monitoring		
Default setting: 0.0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	This MD defines the period of time for which the speed setpoint may dwell at limit CTRLOUT_LIMIT[n] (max. speed setpoint) before the monitor responds. The monitoring function (and therefore also this machine data) is always active. When the limitation is reached, the position control loop becomes non-linear. Contour errors result if the axis subject to speed setpoint limitation is involved in generating the contour. Therefore the MD has the default value 0, i.e. monitoring responds when the speed setpoint reaches the limit. Codings of machine data index [n] : [setpoint branch]: 0		

36300 MD number	**ENC_FREQ_LIMIT**[**n**] Encoder limit frequency		
Default setting: 300000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: Hz	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The encoder limit frequency is entered in this MD. Two encoders can be used for each axis. The active encoder is determined by IS “Position measuring system 1/2” (DB31, ... DBX1.5 or DBX1.6). Coding of machine data index [n]: [encoder no.]: 0 oder 1		

© Siemens AG, 2003. All rights reserved

36302 MD number	**ENC_FREQ_LIMIT_LOW** Encoder limit frequency resynchronization		
Default setting: 99,9	Minimum input limit: 0	Maximum input limit: 100	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: %	
Data type: DOUBLE	Applies from SW 3		
Significance:	The encoder frequency monitoring function operates with a hysteresis. MD 36300: ENC_FREQ_LIMIT is the frequency at which the encoder is switched off, ENC_FREQ_LIMIT_LOW is the frequency at which the encoder is switched on again. ENC_FREQ_LIMIT is entered directly in hertz-while ENC_FREQ_LIMIT_LOW is a percentage of ENC_FREQ_LIMIT. The default setting in MA_ENC_FREQ_LIMIT_LOW is normally sufficient. However, the limit frequency of the absolute track on absolute value encoders with EnDat interface is significantly lower than the limit frequency of the incremental track. By setting a low value in ENC_FREQ_LIMIT, it is possible to ensure that the encoder is not switched on again until the frequency is lower than the limit frequency of the absolute track and is thus not referenced until permitted by the absolute track. This referencing is implemented automatically for spindles. Example: EQN 1325: Limit frequency of electronics of incremental track: 430 kHz ===> ENC_FREQ_LIMIT = 430000 Hz Limit frequency of absolute track: Approx. 2000 encoder rev/min with 2048 marks, i.e. limit frequency (2000/60) * 2048 Hz = 68 kHz ===> ENC_FREQ_LIMIT_LOW = 68/430 = 15%		
Further information	See Subsection 2.3.4		

36310 MD number	**ENC_ZERO_MONITORING** Zero mark monitoring		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 1		
Significance:	This MD can be used to activate zero mark monitoring. Zero mark monitoring checks that the encoder always outputs the same number of pulses between two zero marks. Its meaning depends on the type of encoder used: MD 36310 = 0: No zero mark monitoring MD 36310 = 100 No zero mark monitoring, deactivation of all encoder alarms **MD 36310 > 0:** MD 30240: ENC_TYPE = 0 No zero mark monitoring MD 30240: ENC_TYPE = 3 No zero mark monitoring MD 30240: ENC_TYPE = 1 MD 30240: ENC_TYPE = 2 Number of changes detected for which an alarm should be output. Every time the encoder is switched on, counting starts at “0”. MD 30240: ENC_TYPE = 4 Admissible deviation between absolut and incremental encoder signal in 0.5 encoder lines. An alarm will be generated if this value is exceeded. It is sufficient to enter one 0.5 encoder lines. MD 30240: ENC_TYPE = 5 No zero mark monitoring Encoder alarms are indicated by resetting alarm 20510. Poweron alarm 25000 is not used. On encoders used for position control, zero mark monitoring is activated and triggers alarm 25020. On passive encoders, when zero mark monitoring is activated, alarm 2051 is output. This alarm is only used for display purposes.		
Special cases, errors, ...	Zero mark monitoring is not used for SSI encoders. Laser beam encoders are used. Encoder fault messages can be output frequently if the beam is interrupted. For this encoder, MD 30240: ENC_TYPE = 5 “No zero mark monitoring” is to be applied.		
Further information	See Subsection 2.2.2		

© Siemens AG, 2003. All rights reserved

36400 MD number	**CONTOUR_TOL** Tolerance band contour monitoring		
Default setting: 1.0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: mm; degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Tolerance band for max. following error deviation. The permissible deviation between the real and the expected actual value is entered in this MD. A tolerance band is defined to avoid incorrect triggering of the following error monitoring function because of slight speed fluctuations which can result from normal control processes (e.g. when beginning cutting). This MD must be matched to the position controller gain, in the case of feedforward control to the precision of the control system model MD: EQUIV_SPEEDCTRL_TIME (equivalent time constant for speed loop feedforward control) and to the permissible acceleration rates and velocities.		
References	See Chapter 2		

36600 MD number	**BRAKE_MODE_CHOICE** Deceleration behavior on hardware limit switch		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	If a rising edge for the axisspecific hardware limit switch is recognized while an axis is traversing, the axis is decelerated immediately. The type of deceleration used is defined in the machine data: Value = 0: Controlled deceleration according to the acceleration ramp defined in MAX_AX_ACCEL (axis acceleration). Value = 1: Fast deceleration (defined by setpoint = 0) with decrease of the following		
Related to	IS “Hardware limit switch plus or minus” (DB31, ... DBX12.1 or DBX12.0)		

© Siemens AG, 2003. All rights reserved

36610 MD number	**AX_EMERGENCY_STOP_TIME** Length of the braking ramp for error states		
Default setting: 0.05	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		

Significance:

1. **Axis**: When the following monitoring functions respond, the affected axis is brought to a standstill with rapid stop (with open position control loop) along a braking ramp of the speed setpoint:

- EMERGENCY STOP
- Following error monitoring
- Positioning monitoring
- Zero speed monitoring
- Clamping monitoring
- Speed setpoint monitoring
- Actual velocity monitoring
- Encoder limit frequency monitoring (except on speed-controlled spindles)
- Zero mark monitoring
- Contour tunnel monitoring

When the encoder limit frequency is exceeded, the speed setpoint is displayed as an actual value in the NC basic display.
The period of time required to reduce the speed setpoint from its maximum value down to setpoint = 0 must be entered in MD 36610. The time taken to stop the axis depends on the actual speed setpoint at the time the monitoring responds.

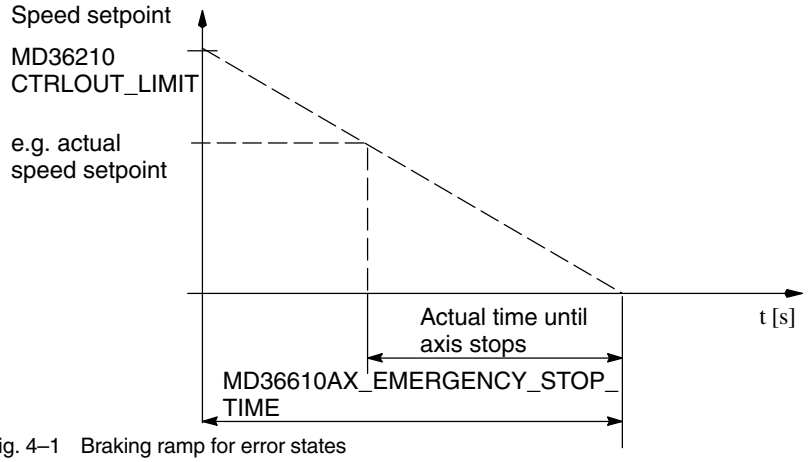


Fig. 4–1 Braking ramp for error states

2. **Spindle**: In the case of spindles without an active position control, the speed-controlled spindle may continue to rotate with no rapid stop braking response when the encoder frequency monitoring function responds (i.e. no valid actual value information available). The speed setpoint monitor is active, but not the actual velocity monitor (MD 36200) when the encoder is active. The spindle speed setpoint limit has only a limiting effect (no alarm generation), the setpoint is limited to the maximum chuck speed (MD 35100) and displayed in IS “Programmed speed too high”.
The current speed cannot be displayed because no valid actual value is currently available.

© Siemens AG, 2003. All rights reserved

36610 MD number	**AX_EMERGENCY_STOP_TIME** Length of the braking ramp for error states
Significance:	Where interpolating axes are involved, maintenance of the contour during the deceleration phase cannot be guaranteed. Notice: If the value for the duration of the braking ramp for error states has been set too high, servo enable is canceled even though the axis/spindle is still traversing. It is then stopped abruptly with speed setpoint 0. The time set in MD: AX_EMERGENCY_STOP_TIME should therefore be less than the time set in MD: SERVO_DISABLE_DELAY_TIME (cutout delay servo enable). Machine data only takes effect in speed mode. This is not achieved with stepper motor without external encoder, because the internal position control mode is never left.
Related to	MD 36620: SERVO_DISABLE_DELAY_TIME Cutout delay servo enable MD 36210: CTRLOUT_LIMIT Maximum speed setpoint
Related to	MD 1404: PULSE_SUPPRESSION_DELAY

4.3.2 Protection zones

30800 MD number	**WORK_AREA_CHECK_TYPE** Type of checking of working area limits		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after NEW_CONF	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 5.2		
Significance:	With this MD it is possible to determine whether the working area limits are checked only for traveling axes (FALSE) or whether axes which are at standstill are also checked within a motion block (TRUE). The value FALSE corresponds to the behavior of SW 5 and lower.		
Related to			

© Siemens AG, 2003. All rights reserved

4.4 Axis/spindle-specific setting data

4.4.1 Axis monitoring functions

43400 SD number	**WORKAREA_PLUS_ENABLE** Working area limitation active in positive direction		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after: immediate	Protection level: MMC-MD 9220	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	1: The working area limitation of the axis concerned active in the positive direction. 0: The working area limitation of the axis concerned is switched off in the positive direction. The setting data is parameterized via the operator panel in the operating area “Parameters” by activating/deactivating the working area limitation.		
SD irrelevant for ...	G code: WALIMOF		

43410 SD number	**WORKAREA_MINUS_ENABLE** Working area limitation active in negative direction		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after: immediate	Protection level: MMC-MD 9220	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	1: The working area limitation of the axis concerned is active in the negative direction. 0: The working area limitation of the axis concerned is switched off in the negative direction. The setting data is parameterized via the operator panel in the operating area “Parameters” by activating/deactivating the working area limitation.		
SD irrelevant for ...	G code: WALIMOF		

43420 SD number	**WORKAREA_LIMIT_PLUS** Working area limitation plus		
Default setting: 100 000 000	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after: immediate	Protection level: MMC-MD 9220	Unit: mm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The working range as defined in the basic coordinate system in the positive direction of the axis concerned can be limited with axial working area limitation. The setting data can be changed via the operator panel in the operating area “Parameters”. The positive working area limitation can be changed in the program with G26.		
SD irrelevant for ...	G code: WALIMOF		
Related to	SD 43400: WORKAREA_PLUS_ENABLE MD 10710: MN_PROG_SD_RESET_SAVE_TAB		

© Siemens AG, 2003. All rights reserved

43430 SD number	**WORKAREA_LIMIT_MINUS** Working area limitation minus		
Default setting: – 100 000 000	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after: immediate	Protection level: MMC-MD 9220	Unit: mm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The working range as defined in the basic coordinate system in the negative direction of the axis concerned can be limited with axial working area limitation. The setting data can be changed via the operator panel in the operating area “Parameters”. The negative working area limitation can be changed in the program with G25.		
SD irrelevant for ...	G code: WALIMOF		
Related to	SD 43410: WORKAREA_MINUS_ENABLE MD 10710: MN_PROG_SD_RESET_SAVE_TAB		

4.4.2 Protection zones

None

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Signal Descriptions

5

5.1 Axis/spindle-specific signals

5.1.1 Axis monitoring functions

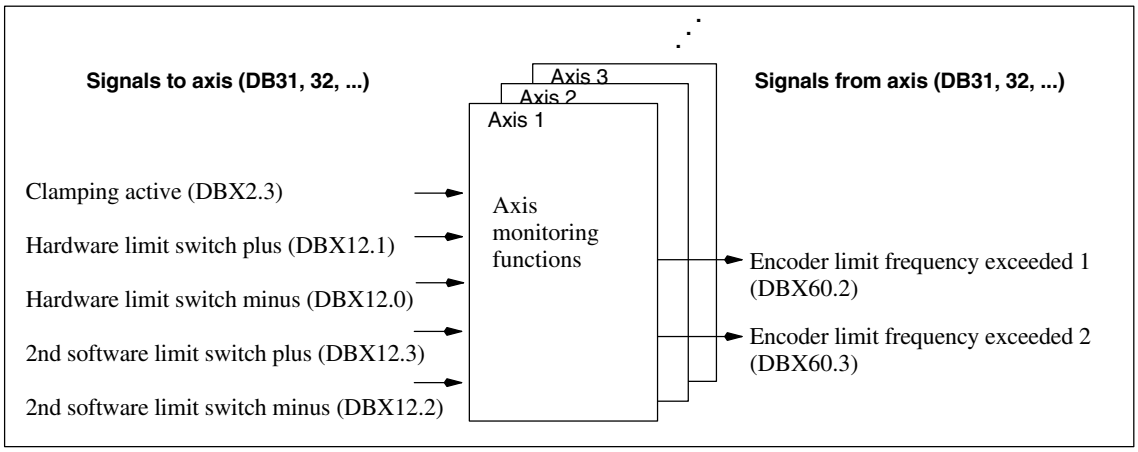


Fig. 5-1 PLC interface signals for axis monitoring functions

Signals to axis/spindle

DB 31, ... **DBX2.3** Data block	**Clamping in progress** Signal(s) to axis/spindle (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Clamping active. The clamping monitoring function is activated.	
Signal state 0 or signal transition 1 ——> 0	Clamping terminated. The zero speed monitoring function takes over from the clamping monitoring function.	
Related to	MD 36050: CLAMP_POS_TOL (clamping tolerance)	
References	Chapter 2	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX3.6** Data block	**Velocity/spindle speed limitation** Signal(s)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The NCK limits the velocity/spindle speed to the limit value set in MD 35160: SPIND_EXTERN_VELO_LIMIT.		
Signal state 0 or signal transition 1 ——> 0	No limitation active.		
Related to	MD 35100: SPIND_VELO_LIMIT (max. spindle speed) SD 43220: SPIND_MAX_VELO_G26 (progr. spindle speed limitation G26) SD 43230: SPIND_MAX_VELO_LIMIT (progr. spindle speed limitation G96)		

DB 31, ... **DBX12.1 and 12.0** Data block	**Hardware limit switches plus and minus** Signal(s) to axis/spindle (PLC ——> NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	A switch can be mounted at each end of the travel range of a machine axis which will cause a signal “Hardware limit switch plus or minus” to be reported to the NC via the PLC if it is approached. If the signal is recognized as set, alarm 021614 “Hardware limit switch + or –” is output and the axis is decelerated immediately. The braking mode is defined in MD 36600: BRAKE_MODE_CHOICE (deceleration behavior on hardware limit switch). If, in addition to the signal “Hardware limit switch”, servo enable is also canceled, the axis reacts as described in the Description of Functions, “Various Interface Signals A2”.		
Signal state 0 or signal transition 1 ——> 0	Normal condition, hardware limit switch not triggered.		
Related to	MD 36600: BRAKE_MODE_CHOICE (deceleration behavior on hardware limit switch)		

DB 31, ... **DBX12.3 and 12.2** Data block	**2nd software limit switch plus or minus** Signal(s) to axis/spindle (PLC ——> NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	2nd software limit switch for the plus or minus direction is active. 1st software limit switch for the plus or minus direction is not active. In addition to the 1st software limit switches (plus and minus), the 2nd software limit switches (plus and minus) can be activated with these interface signals. The position is defined by means of MD 36130: POS_LIMIT_PLUS2, MD 36120 POS_LIMIT_MINUS2 (2nd software limit switch plus, 2nd software limit switch minus).		
Signal state 0 or signal transition 1 ——> 0	1st software limit switch for the plus or minus direction is active. 2nd software limit switch for the plus or minus direction is not active.		
Related to	MD 36110: POS_LIMIT_PLUS, MD 36130: POS_LIMIT_PLUS, MD 36100: POS_LIMIT_MINUS, MD 36120: POS_LIMIT_MINUS2, (software limit switch plus, software limit switch minus)		

© Siemens AG, 2003. All rights reserved

Signals from axis/spindle

DB 31, ... **DBX60.2 and 60.3** Data block	**Encoder limit frequency exceeded 1** **Encoder limit frequency exceeded 2** Signal(s) from axis/spindle (NCK ⟶ PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ⟶ 1	The time set in MD 36300: ENC_FREQ_LIMIT(encoder limit frequency) has been exceeded. The reference point for the position measuring system concerned has been lost (IS: Referenced/synchronized is in signal state 0). Proper position closed-loop control is no longer possible. The spindle continues to turn with speed control loop. The axes are stopped with rapid stop (with open position control loop) along a speed setpoint ramp.		
Signal state 0 or signal transition 1 ⟶ 0	The time set in MD 36300: ENC_FREQ_LIMIT (limit frequency < ENC_FREQ_LIMIT_LOW). The encoder frequency must have dropped below the setting in MD 36302 ENC_FREQ_LIMIT_LOW to achieve signal transition 1 ⟶ 0.		

5.1.2 Protection zones

None

© Siemens AG, 2003. All rights reserved

5.2 Channel-specific signals

5.2.1 Axis monitoring functions

None

5.2.2 Protection zones

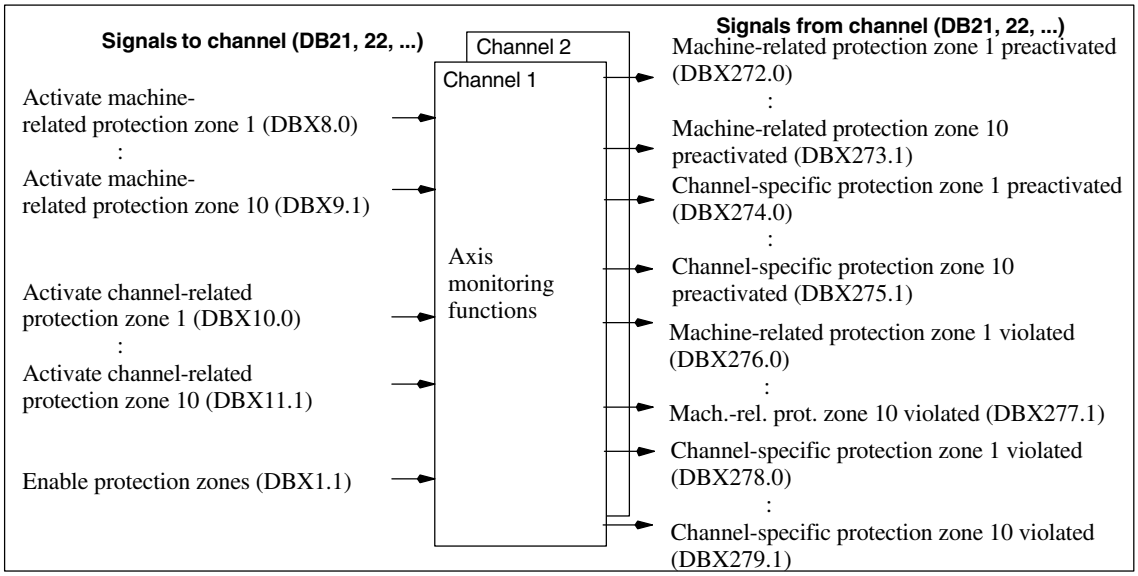


Fig. 5-2 Channelspecific PLC interface signals for “protection zones”

Signals to channel

DB21–28 **DBX1.1** Data block	**Enable protection zone e** Signal(s) to channel (PLC → NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 3.1
Signal state 1 or signal transition 0 ——> 1	When a positive edge of this signal appears, a protection zone is enabled and the active alarm reset. A motion into the same protection zone can then start. As the motion start command is issued, the protection zone is enabled, IS “Machine or channel-specific protection zone violated” set and the axis motion commenced. The enabling signal is invalidated if a motion is started that does not lead into the enabled protection zone.	
Signal state 0 or signal transition 1 ——> 0	No effect	
Application example(s)	These signals can be used to enable protection zones – if the current position is within the protection zone (alarm 2 is active) – if a motion is to be started on the protection zone limit (alarm 1 or 2 is present)	

© Siemens AG, 2003. All rights reserved

DB21, ... DBX8.0 – 9.1 Data block	**Activate machinerelated protection zone 1 (... 10)** Signal(s) to channel (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	The preactivated, machine-related protection zone 1 (...10) is activated by the PLC user program. The protection zone is immediately active. Only protection zones that have been preactivated in the parts program can be activated.	
Signal state 0 or signal transition 1 ——> 0	The preactivated, machine-related protection zone 1 (...10) is **deactivated** by the PLC user program. The protection zone is immediately deactivated. Only protection zones that have been activated via the PLC and have been preactivated in the NC parts program can be deactivated.	
Application example(s)	Before a sensor, for example, is moved into the working range, the relevant machinerelated protection zone can be activated.	

DB21, ... DBX10.0 – 11.1 Data block	**Activate channel-specific protection zone 1 (...10)** Signal(s) to channel (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.0
Signal state 1 or signal transition 0 ——> 1	The preactivated, channel-specific protection zone 1 (...10) is activated by the PLC user program. The protection zone is immediately active. Only protection zones that have been preactivated in the parts program can be activated.	
Signal state 0 or signal transition 1 ——> 0	The preactivated, channel-specific protection zone 1 (...10) is **deactivated** by the PLC user program. The protection zone is immediately deactivated. Only protection zones that have been activated via the PLC and have been preactivated in the NC parts program can be deactivated.	
Application example(s)	Before a synchronous spindle, for example, is moved into the working range, the relevant machine-related protection zone can be activated.	

Signals from channel

DB21, ... DBX272.0 – 273.1 Data block	**Machine-related protection zone 1 (...10) preactivated** Signal(s) from channel (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.0
Signal state 1 or signal transition 0 ——> 1	The machine-related protection zone 1 (...10) is preactivated in the current block. (Preactivated in the parts program). The protection zone can be activated or deactivated via IS “Activate machine-related protection zone 1 (...10)” (DB21, ... DBX8.0 – 9.1) in the PLC user program.	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX272.0 – 273.1** Data block	**Machine-related protection zone 1 (...10) preactivated** Signal(s) from channel (NCK → PLC)
Signal state 0 or signal transition 1 ——> 0	The machine-related protection zone 1 (...10) is deactivated in the current block. (Deactivated in the parts program). The protection zone **cannot** be activated or deactivated via IS “Activate machinerelated protection zone 1 (...10)” (DB21, ... DBX8.0 9.1) in the PLC user program.
Related to	IS “Activate machine-related protection zone 1 (...10)” (DB21, ... DBX8.0 – 9.1)

DB21, ... **DBX274.0 – 275.1** Data block	**Channels-pecific protection zone 1 (...10) preactivated** Signal(s) from channel (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.0
Signal state 1 or signal transition 0 ——> 1	The channel-specific protection zone 1 (...10) is preactivated in the current block. (Preactivated in the parts program). The protection zone can be activated or deactivated in the PLC user program via IS “Activate channel-specific protection zone 1 (...10)” (DB21, ... DBX10.0 bis 11.1).	
Signal state 0 or signal transition 1 ——> 0	The channel-specific protection zone 1 (...10) is deactivated in the current block. (Deactivated in the parts program). The protection zone cannot be activated or deactivated in the PLC user program via IS “Activate channel-specific protection zone 1 (...10)” (DB21, ... DBX10.0 bis 11.1).	
Related to	IS “Activate channel-specific protection zone 1 (...10)” (DB21, ... DB10.0 – 11.1)	

DB21, ... **DBX276.0 – 277.1** Data block	**Machine-related protection zone 1 (...10) violated** Signal(s) from channel (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.0
Signal state 1 or signal transition 0 ——> 1	• The activated, machine-related protection zone 1 (...10) is violated in the current block or in the current JOG movement. • The preactivated, machine-related protection zone 1 (...10) would be violated in the current block if it were activated by the PLC.	
Signal state 0 or signal transition 1 ——> 0	• The activated, machine-related protection zone 1 (...10) is **not** violated in the current block. • The preactivated, machine-related protection zone 1 (...10) would **not** be violated in the current block if it were activated by the PLC.	
Application example(s)	With this IS it is possible to check whether the tool or workpiece is located in the machine-related protection zone of a moving part before the latter is moved into the working range.	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX278.0 – 279.1** Data block	**Channel-specific protection zone 1 (...10) violated** Signal(s) from channel (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.0
Signal state 1 or signal transition 0 ——> 1	• The activated, channel-specific protection zone 1 (...10) is violated in the current block. • The preactivated, channel-specific protection zone 1 (...10) would be violated in the current block if it were activated by the PLC.	
Signal state 0 or signal transition 1 ——> 0	• The activated, channel-specific protection zone 1 (...10) is **not** violated in the current block. • The preactivated, channel-specific protection zone 1 (...10) would **not** be violated in the current block if it were activated by the PLC.	
Application example(s)	With this IS it is possible to check whether the tool or workpiece is located in the channel-specific protection zone of a moving part before the latter is moved into the working range.	

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

6 Examples

6.1 Axis monitoring functions

None

6.2 Protection zones

6.2.1 Definition and activation of protection zones

Request

The following internal protection zones are to be defined for a lathe:

- A machine and workpiece-related protection zone for the spindle chuck, without limitation in the third dimension.
- A channel-specific protection zone for the workpiece, without limitation in the third dimension.
- A channel-specific, tool-related protection zone for the toolholder, without limitation in the third dimension.

The workpiece zero is placed on the machine zero to define the protection zone for the workpiece.

When activated, the protection zone is then offset by 100mm in the Z axis in the positive direction.

© Siemens AG, 2003. All rights reserved

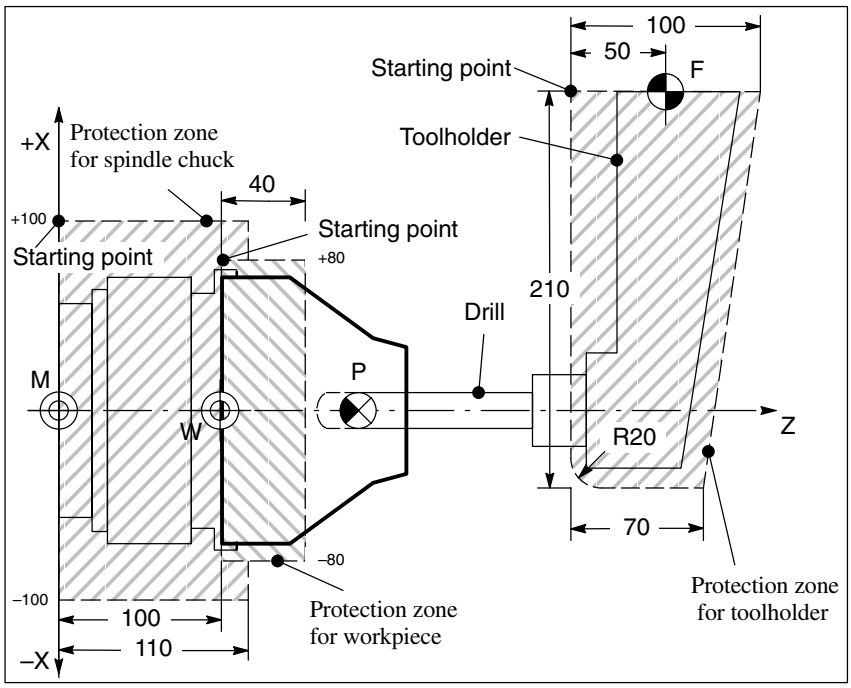


Fig. 6-1 Example of protection zones on a lathe

Protection zone definition in the parts program

Associated parts program:

```
                    :
DEF INT AB
G18
NPROTDEF(0,FALSE,0,0,0)

NPROTDEF(1,FALSE,0,0,0)        ; Protection zone for spindle chuck
G01 X100 Z0
G01 X–100 Z0
G01 X–100 Z110
G01 X100 Z110
G01 X100 Z0
EXECUTE(AB)

CPROTDEF(1,FALSE,0,0,0)        ; Protection zone for workpiece
G01 X80 Z0
G01 X–80 Z0
G01 X–80 Z40
G01 X80 Z40
G01 X80 Z0
EXECUTE(AB)

CPROTDEF(2,TRUE,0,0,0)         ; Protection zone for toolholder
G01 X0 Z–50
G01 X–190 Z–50
G03 X–210 Z–30 I20 J0
G01 X–210 Z20
G01 X0 Z50
G01 X0 Z–50
EXECUTE(AB)
```

© Siemens AG, 2003. All rights reserved

Protection zone definition with system variables

Associated system variable settings:

```
        :
$SN_PA_T_W[0]=0          ; Protection zone for spindle chuck
$SN_PA_ORI[0]=1
$SN_PA_LIM_3DIM[0]=0
$SN_PA_PLUS_LIM[0]=0
$SN_PA_MINUS_LIM[0]=0
$SN_PA_CONT_NUM[0]=5
$SN_PA_CONT_TYP[0.0]=1
$SN_PA_CONT_ORD[0.0]=100
$SN_PA_CONT_ABS[0.0]=0
$SN_PA_CONT_TYP[0.1]=1
$SN_PA_CONT_ORD[0.1]=–100
$SN_PA_CONT_ABS[0.1]=0
$SN_PA_CONT_TYP[0.2]=1
$SN_PA_CONT_ORD[0.2]=–100
$SN_PA_CONT_ABS[0.2]=110
$SN_PA_CONT_TYP[0.3]=1
$SN_PA_CONT_ORD[0.3]=100
$SN_PA_CONT_ABS[0.3]=110
$SN_PA_CONT_TYP[0,4]=1
$SN_PA_CONT_ORD[0,4]=100
$SN_PA_CONT_ABS[0,4]=0

$SC_PA_TW[0]=0           ; Protection zone for workpiece
$SC_PA_ORI[0]=1
$SC_PA_LIM_3DIM[0]=0
$SC_PA_PLUS_LIM[0]=0
$SC_PA_MINUS_LIM[0]=0
$SC_PA_CONT_NUM[0]=5
$SC_PA_CONT_TYP[0.0]=1
$SC_PA_CONT_ORD[0.0]=80
$SC_PA_CONT_ABS[0.0]=0
$SC_PA_CONT_TYP[0.1]=1
$SC_PA_CONT_ORD[0.1]=–80
$SC_PA_CONT_ABS[0.1]=0
$SC_PA_CONT_TYP[0.2]=1
$SC_PA_CONT_ORD[0.2]=–80
$SC_PA_CONT_ABS[0.2]=40
$SC_PA_CONT_TYP[0.3]=1
$SC_PA_CONT_ORD[0.3]=80
$SC_PA_CONT_ABS[0.3]=40
$SC_PA_CONT_TYP[0.4]=1
$SC_PA_CONT_ORD[0.4]=80
$SC_PA_CONT_ABS[0.4]=0
```

© Siemens AG, 2003. All rights reserved

```
$SC_PA_T_W[1]=3          ; Protection zone for toolholder
$SC_PA_ORI[1]=1
$SC_PA_LIM_3DIM[1]=0
$SC_PA_PLUS_LIM[1]=0
$SC_PA_MINUS_LIM[1]=0
$SC_PA_CONT_NUM[1]=6
$SC_PA_CONT_TYP[1.0]=1
$SC_PA_CONT_ORD[1.0]=0
$SC_PA_CONT_ABS[1.0]=–50
$SC_PA_CONT_TYP[1.1]=1
$SC_PA_CONT_ORD[1.1]=–190
$SC_PA_CONT_ABS[1.1]=–50
$SC_PA_CONT_TYP[1.2]=3
$SC_PA_CONT_ORD[1.2]=–210
$SC_PA_CONT_ABS[1.2]=–30
$SC_PA_CENT_ORD[1,2]=–190
$SC_PA_CENT_ABS[1,2]=–30
$SC_PA_CONT_TYP[1.3]=1
$SC_PA_CONT_ORD[1.3]=–210
$SC_PA_CONT_ABS[1.3]=20
$SC_PA_CONT_TYP[1.4]=1
$SC_PA_CONT_ORD[1.4]=0
$SC_PA_CONT_ABS[1.4]=50
$SC_PA_CONT_TYP[1,5]=1
$SC_PA_CONT_ORD[1,5]=0
$SC_PA_CONT_ABS[1,5]=–50
```

Activation

The three protection zones in a channel are activated in a parts program by means of the following subprograms:

```
NPROT(1,2,0,0,0)         ; Protection zone for spindle chuck
      :
CPROT(1,2,0,0,100)       ; Protection zone for workpiece
                         ; On activation the protection zone
                         ; is shifted 100mm along the Z axis.
      :
CPROT(2,2,0,0,0)         ; Protection zone for toolholder.
```

■

© Siemens AG, 2003. All rights reserved

7 Data Fields, Lists

7.1 Axis/spindle-specific interface signals

7.1.1 Channel-specific signals

DB number	Bit, byte	Name	Refer-ence
Channel-specific			
21, ...	4.0–4.7	Feed override	V1
21, ...	6.0	Feed disable	V1

7.1.2 Axis monitoring functions

DB number	Bit, byte	Name	Refer-ence
Axis/spindle-specific			
31, ...	1.4	Follow up mode	A2
31, ...	1.5 / 1.6	Position measurement system 1/Position measurement system 2	A2
31, ...	2.1	Servo enable	A2
31, ...	2.3	Clamping in progress	
31, ...	3.6	Velocity/spindle speed limitation	
31, ...	4.3	Feed stop	V1
31, ...	12.0 / 12.1	Hardware limit switch minus/Hardware limit switch plus	
31, ...	12.2 / 12.3	2nd software limit switch minus/2nd software limit switch plus	
31, ...	60.2 / 60.3	Encoder limit frequency exceeded 1/Encoder limit frequency exceeded 2	
31, ...	60.4 / 60.5	Referenced/synchronized 1/Referenced/synchronized 2	R1
31, ...	64.6 / 64.7	Motion command	H1

7.1.3 Protection zones

None

© Siemens AG, 2003. All rights reserved

7.2 Channel-specific interface signals

7.2.1 Axis monitoring functions

None

7.2.2 Protection zones

DB number	Bit, byte	Name	Refer-ence
Channel-specific		**Signals to channel**	
21, ...	1.1	Enable protection zones	
21, ...	8.0	Activate machine-related protection zone 1	
	:	:	
21, ...	8.7	Activate machine-related protection zone 8	
21, ...	9.0	Activate machine-related protection zone 9	
21, ...	9.1	Activate machine-related protection zone 10	
21, ...	10.0	Activate channel-specific protection zone 1	
	:	:	
21, ...	10.7	Activate channel-specific protection zone 8	
21, ...	11.0	Activate channel-specific protection zone 9	
21, ...	11.1	Activate channel-specific protection zone 10	
Channel-specific		**Signals from channel**	
21, ...	272.0	Machine-related protection zone 1 preactivated	
	:	:	
21, ...	272.7	Machine-related protection zone 8 preactivated	
21, ...	273.0	Machine-related protection zone 9 preactivated	
21, ...	273.1	Machine-related protection zone 10 preactivated	
21, ...	274.0	Channel-specific protection zone 1 preactivated	
	:	:	
21, ...	274.7	Channel-specific protection zone 8 preactivated	
21, ...	275.0	Channel-specific protection zone 9 preactivated	
21, ...	275.1	Channel-specific protection zone 10 preactivated	
21, ...	276.0	Machine-related protection zone 1 violated	
	:	:	
21, ...	276.7	Machine-related protection zone 8 violated	
21, ...	277.0	Machine-related protection zone 9 violated	
21, ...	277.1	Machine-related protection zone 10 violated	
21, ...	278.0	Channel-specific protection zone 1 violated	
	:	:	
21, ...	278.7	Channel-specific protection zone 8 violated	
21, ...	279.0	Channel-specific protection zone 9 violated	
21, ...	279.1	Channel-specific protection zone 10 violated	

© Siemens AG, 2003. All rights reserved

7.3 General machine data

7.3.1 Axis monitoring functions

None

7.3.2 Protection zones

Number	Identifier	Name	Reference
General ($MN_ ...)			
10604	WALIM_GEOAX_CHANGE_MODE	Working range limitation for geometry axis replacement (SW 6.3 and higher)	
10618	PROTAREA_GEOAX_CHANGE_MODE	Protection zone for switchover of geo axes. (SW 4.3 and higher)	
18190	MM_NUM_PROTECT_AREA_NCK	Number of files for machine-related protection zones	S7

7.4 Axis/spindle-specific machine data

7.4.1 Axis monitoring functions

Number	Identifier	Name	Reference
Axis/channel-specific($MA_ ...)			
30310	ROT_IS_MODULO	Modulo conversion for rotary axis and spindle	R2
32200	POSCTRL_GAIN [n]	Servo gain factor	G2
32250	RATED_OUTVAL	Rated output voltage	G2
32260	RATED_VELO	Rated motor speed	G2
32300	MAX_AX_ACCEL	Axis acceleration	B2
32800	EQUIV_CURRCTRL_TIME [n]	Equivalent time constant current control loop for feedforward control	K3
32810	EQUIV_SPEEDCTRL_TIME[n]	Equivalent time constant speed control loop for feedforward control	K3
32910	DYN_MATCH_TIME [n]	Time constant for dynamic matching	G2
35160	SPIND_EXTERN_VELO_LIMIT	Spindle speed limitation via PLC	S1
36000	STOP_LIMIT_COARSE	Exact stop coarse	B1
36010	STOP_LIMIT_FINE	Exact stop fine	B1
36020	POSITIONING_TIME	Time delay exact stop fine	
36030	STANDSTILL_POS_TOL	Zero speed tolerance	
36040	STANDSTILL_DELAY_TIME	Delay time zero speed monitoring	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Reference
36050	CLAMP_POS_TOL	Clamping tolerance with IS “Clamping active”	
36052	STOP_ON_CLAMPING	Special functions for clamped axis	
36060	STANDSTILL_VELO_TOL	Maximum velocity/speed “Axis/spindle stationary”	A2
36100	POS_LIMIT_MINUS	1st software limit switch minus	
36110	POS_LIMIT_PLUS	1st software limit switch plus	
36120	POS_LIMIT_MINUS	2nd software limit switch minus	
36130	POS_LIMIT_PLUS	2nd software limit switch plus	
36610	AX_EMERGENCY_STOP_TIME	Length of the braking ramp for error states	
36200	AX_VELO_LIMIT [n]	Threshold value for velocity monitoring	
36210	CTRLOUT_LIMIT[n]	Maximum speed setpoint	G2
36220	CTRLOUT_LIMIT_TIME[n]	Delay time for speed setpoint monitoring	
36300	ENC_FREQ_LIMIT[n]	Encoder limit frequency	
36302	ENC_FREQ_LIMIT_LOW	Encoder limit frequency resynchronization	
36310	ENC_ZERO_MONITORING [n]	Zero mark monitoring	
36400	CONTOUR_TOL	Tolerance band contour monitoring	
36500	ENC_CHANGE_TOL	Maximum tolerance for position actual value switchover	G2
36600	BRAKE_MODE_CHOICE	Deceleration behavior on hardware limit switch	
36620	SERVO_DISABLE_DELAY_TIME	Cutout delay servo enable	A2

7.4.2 Protection zones

Number	Identifier	Name	Reference
Axis/channel-specific($MA_ ...)			
30800	WORK_AREA_CHECK_TYPE	Type of checking of working area limits	

None

© Siemens AG, 2003. All rights reserved

7.5 Channel-specific machine data

7.5.1 Axis monitoring functions

Number	Identifier	Name	Reference
Channel-specific ($MC_ ...)			
20150	GCODE_RESET_VALUES[n]	Reset G groups	K1
21020	WORKAREA_WITH_TOOL_RADIUS	Allowance for tool radius with working area limitation	

7.5.2 Protection zones

Number	Identifier	Name	Reference
Channel-specific ($MC_ ...)			
28200	MM_NUM_PROTECT_AREA_CHAN	Number of files for channel-specific protection zones	S7
28210	MM_NUM_PROTECT_AREA_ACTIVE	Number of simultaneously active protection zones in one channel	S7
28212	MM_NUM_PROTECT_AREA_CONTUR	Elements for active protection zones (DRAM)	S7

7.6 Axis/spindle-specific setting data

7.6.1 Axis monitoring functions

Number	Identifier	Name	Reference
Axis/spindle-specific ($SA_ ...)			
43400	WORKAREA_PLUS_ENABLE	Working area limitation active in positive direction	
43410	WORKAREA_MINUS_ENABLE	Working area limitation active in negative direction	
43420	WORKAREA_LIMIT_PLUS	Working area limitation plus	
43430	WORKAREA_LIMIT_MINUS	Working area limitation minus	

7.6.2 Protection zones

None

© Siemens AG, 2003. All rights reserved

7.7 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Continuous Path Mode, Exact Stop and Look Ahead (B1)

1	**Brief Description**		**1/B1/1-3**
2	**Detailed Description**		**1/B1/2-5**
	2.1	General	1/B1/2-5
	2.1.1	Velocities	1/B1/2-7
	2.1.2	Initial setting for exact stop/continuous-path mode	1/B1/2-7
	2.1.3	Block change with positioning axes	1/B1/2-8
	2.1.4	Stop for synchronization	1/B1/2-9
	2.2	Exact stop	1/B1/2-9
	2.3	Continuous-path mode	1/B1/2-13
	2.3.1	Velocity reduction according to overload factor	1/B1/2-16
	2.3.2	Rounding according to path criterion	1/B1/2-18
	2.3.3	Jerk limitation on path through velocity reduction	1/B1/2-23
	2.3.4	Machine axis-specific jerk limitation (SW 5 and higher)	1/B1/2-25
	2.3.5	Smoothing of the path velocity (SW 5.3 and higher)	1/B1/2-30
	2.3.6	Dynamic response adaptation (SW 6.3)	1/B1/2-34
	2.4	Look Ahead	1/B1/2-42
	2.5	NC block compressor COMPON, COMPCURV, -CAD	1/B1/2-47
3	**Supplementary Conditions**		**1/B1/4-49**
	3.1	Smoothing of the path velocity (SW 5.3 and higher)	1/B1/4-49
4	**Data Descriptions (MD, SD)**		**1/B1/4-49**
	4.1	General machine data	1/B1/4-49
	4.2	Channel-specific machine data	1/B1/4-50
	4.3	Axis-specific machine data	1/B1/4-56
	4.4	Channel-specific setting data	1/B1/4-60
5	**Signal Descriptions**		**1/B1/5-61**
	5.1	Channel-specific signals	1/B1/5-61
	5.2	Axis-specific signals	1/B1/5-61
6	**Example**		**1/B1/7-63**
7	**Data Fields, Lists**		**1/B1/7-63**
	7.1	Interface signals	1/B1/7-63

© Siemens AG, 2003. All rights reserved

7.2 Machine data 1/B1/7-64
7.3 Setting data 1/B1/7-66
7.4 Alarms 1/B1/7-66

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

For continuous-path control, the CNC processes a parts program block by block. Only when the functions of the current block have been completed is the next block processed. Various requirements on the part, e.g. contour accuracy, processing time, workpiece surface, demand different block change criteria. There are two ways that the path axes can behave at block boundaries.

- The first way is called exact stop and means that all path axes must have reached the set target position depending on an exact stop criterion before the next block change is initiated. To be able to fulfill the criterion, the path axes must reduce the path velocity at every block change which, however, delays the block change.
- The second way is called continuous-path mode and it attempts to avoid deceleration of the path velocity at the block boundary in order to change to the next block with as little change of path velocity as possible.

In the following function description, the characteristics and options of the exact stop, continuous-path mode and Look Ahead functions are described.

Expansion in SW 5.3

Smoothing of path velocity
This function can be used to suppress short acceleration/deceleration processes under settable conditions if the resulting increase in the processing time does not exceed a given limit. The advantages are:
– Less excitations of machine resonance
– More constant cutting rate.

Expansion in SW 6.3

Adaptation of dynamic response

This function allows reduced limit values for acceleration and jerk to be applied under settable conditions for short acceleration/deceleration processes.
A slight increase in the machining time is the price to pay for
– less excitations of machine resonance.

The processes:

- smoothing of path velocity and
- adaptation of dynamic response

can be combined.

© Siemens AG, 2003. All rights reserved

NC block compressor

COMPON/COMPCURV/COMPCAD/COMPOF

CAD/CAM systems produce linear blocks which meet the configured accuracy specifications. In the case of complex contours, a large volume of data and short path sections can result. The short path sections restrict the processing rate. The compressor allows a certain number (max. 10) of short path sections to be combined in a single path section.

The modal G code **COMPON** or **COMPCURV** can be used to activate an "NC block compressor". This function collects a series of linear blocks during linear interpolation (the number is limited to 10) and approximates them within a tolerance specified in machine data via a 3rd-degree (COMPON) or 5th-degree (COMPCURV) polynomial. One traversing block is processed by the NC instead of a large number of small blocks.

The **COMPCAD** G code can be used to select a further compression which optimizes the **surface quality and velocity**. The interpolation accuracy can again be specified in machine data. COMPCAD is processor and memory-intensive. It should only be used if surface quality enhancement measures cannot be incorporated in the CAD/CAM program.

The programming is described in
References: /PGA/ Programming Guide, Advanced, Chapter 5.

■

© Siemens AG, 2003. All rights reserved

2 Detailed Description

2.1 General

The exact stop and continuous-path mode functions are implemented for path axes that are all assigned to the same channel. Machine axes that are related interpolatively must have the same dynamic response, i.e. the same following error at any given velocity.

Path axes

The term Path axes refers to all machining axes assigned to the channel which are controlled by the interpolator calculating the path points in such a manner that:

- All axes involved start at the same time
- All the axes involved travel with the correct velocity ratios
- All the axes reach the programmed target position at the same time.

The accelerations of the axes can be different depending on the path, e.g. circle.
Path axes can be geometry axes as well as special axes (e.g. synchronous axes, workpiece turning axes that are involved in the workpiece machining process).

Geometry axes

Geometry axes describe a two-dimensional or three-dimensional area in the workpiece coordinate system.

Orientation axis

This is the axis which aligns the tool in relation to the machining plane in 5-axis mode.

© Siemens AG, 2003. All rights reserved

Channel

A channel represents a separate CNC which, supported by a parts program, is capable of controlling the motions of axes and spindles as well as machine functions independently of other channels. The independence of the channels is guaranteed by:

- An active parts program per channel
- Channel-specific interface signals such as
  - NC Start
  - NC Stop
  - Reset
- One feedrate override per channel
- One rapid traverse override per channel
- Channel-specific evaluation and display of alarms
- Channel-specific display e.g. for
  - Axis positions
  - Active G functions
  - Active auxiliary functions
  - Current program block
- Channel-specific testing and control of parts programs
  - Block
  - Dry run feed
  - POWER ON, block search

For further descriptions of channel functions see:

References: /FB/, K1, “Mode Group, Channels, Program Operation Mode”.

© Siemens AG, 2003. All rights reserved

2.1.1 Velocities

The axis-specific velocity limits and acceleration limits apply to the path axes.

Feed

The programmed feedrate F corresponds to the path feedrate. This is modal and is programmed as a velocity e.g. in the units mm/min, inch/min or degrees/min. The feedrate is specified for the types of motion G01, G02 and G03. If path axes are programmed without a feedrate, the alarm 10860 “No feedrate programmed” is output.

Path feed

The path feed refers to axes that are assigned to the basic coordinate system. It represents the geometric sum of feedrates of axes involved in the interpolation.

Feedrate override

There is one common rapid traverse override for the path axes assigned to a channel.

Rapid traverse

Rapid traverse G00 allows an axis to be traversed at its fastest possible traversing velocity. If an axis is traversed at rapid traverse rate, the maximum axis velocity limits the rapid traverse velocities.

Rapid traverse override

There is one common feedrate override for the path axes assigned to a channel.

Velocity for zero cycle blocks

The term zero cycle is applied to blocks whose path length is shorter than the distance that can be traveled on the basis of the programmed set feedrate and the interpolator cycle. For reasons of precision the velocity is reduced until at least one interpolator cycle is required for the distance. The velocity is then equal to or less than the quotient path length of the block divided by the IPO cycle.

2.1.2 Initial setting for exact stop/continuous-path mode

Which of the functions Exact stop (including exact stop criterion) or continuous-path mode is active in the parts program without the function-specific program code is determined by the default setting made in MD20150: GCODE_ RESET_VALUES (initial setting of the G groups). With standard installation exact stop G60 with the block change criterion “Exact stop fine, G601” is the initial setting. For further information on making the initial settings see:

References: /FB/, K1, “Mode Group, Channels, Program Operation Mode”.

© Siemens AG, 2003. All rights reserved

2.1.3 Block change with positioning axes

Path axes can be programmed either on their own or with positioning axes in one NC block. Positioning axes do not interpolate with the path interpolator but with the axis interpolator and therefore reach their programmed destination at a different time than the path axes. For the block change it is therefore not only the criteria exact stop and continuous-path mode of the path axes which are relevant but also the type of positioning axis. Because there are two types of positioning axes, the following block change response are achieved:

- Positioning axis type POS
  A block change is executed when all path axes and positioning axes have reached their programmed end position. Because of this, continuous-path mode for path axes is only possible if the positioning axes reach their final position before the path axes.
- Positioning axis type POSA
  The block change is executed once the path axes have reached their programmed end position. Continuous-path mode is therefore possible without restriction.
- Positioning axes themselves do not differentiate between the functions continuous-path mode and exact stop because they always reach their final position with the criterion exact stop fine.

References: /FB/, P2, “Positioning Axes”.

© Siemens AG, 2003. All rights reserved

2.1.4 Stop for synchronization

Regardless of whether exact stop or continuous-path mode is selected, the block change can be delayed by synchronization processes which can stop the path axes. In exact stop mode, the path axes are stopped at the end of the current block. In continuous-path mode, the path axes are stopped at the next block end point at which they can be decelerated without violating their deceleration limits. The following synchronization processes cause axes to stop.

- PLC acknowledgment
  If acknowledgment by the PLC is required for an auxiliary function that is output before or after the end of motion, the axes stop at the end of the block.
- Missing following blocks
  If following blocks cannot be prepared fast enough (e.g. in DNC operation), the axes stop at the last approachable block boundary.
- Empty buffer
  If the NC parts program requests that the preprocessing run be synchronized with the main run (empty the buffer), this involves an implicit blockrelated velocity reduction or exact stop.

Stopping because of synchronization does not cause contour violations. However, stopping is undesirable, especially in continuous-path mode because it can cause backing off.

2.2 Exact stop

With the exact stop function, all the path axes must reach the program block end point. Only when all path axes have reached the exact stop criterion is the block change performed. The velocity at the block transition is practically zero.

This means:

- That the path axes at the block end point are decelerated almost to rest without overshoot.
- The delay for fulfilling the exact stop criterion prolongs the machining time.
- The delay for fulfilling the exact stop criterion can cause backing off.

The use of the exact stop function is suitable for precise traversing of contours.

© Siemens AG, 2003. All rights reserved

Exact stop is not suitable if

- exact traversing of the contour on the basis of the criterion (e.g. exact stop fine) can deviate from the programmed contour in order to achieve faster machining.
- an absolutely constant velocity is required.

Exact stop spindle number converter

The “Exact stop” function can be selected in the NC parts program by command G60 or G09. Before or on selection, the required exact stop criterion must be specified with the appropriate program code. G60 is modal, G09 is non modal. G09 is used if continuous-path mode is to be interrupted. Both exact stop functions only function with the selected exact stop criterion. The function exact stop is deselected with the function continuous path mode.

Exact stop criteria

- Exact stop fine
  This criterion is applied to monitor whether the actual/setpoint position deviation of the axis has remained within a specific distance. The permissible deviation distance is defined in MD 36010: STOP_ LIMIT_FINE (exact stop fine).
- Exact stop coarse
  Functionality as for exact stop fine, but the monitoring window for this function is defined in MD 36000: STOP_LIMIT_COARSE (exact stop coarse). To permit a faster block change than with the exact stop fine criterion, the exact stop coarse criterion is set to be larger than the exact stop fine criterion.

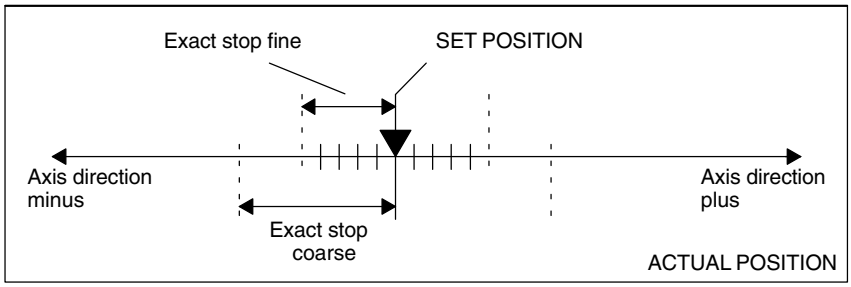


Fig. 2-1 Representation of exact stop coarse/fine

- Interpolator end
  When this criterion is applied, the block change is initiated when the interpolator has calculated the setpoint velocity of the axes from zero for an interpolation cycle. The block change is thus dependent on the dynamic response of the axes. The actual positions of the path axes are not monitored for the block change. With this criterion rounding of contours is possible at non-tangential block transitions because of the residual following error.
  With interpolator end, the auxiliary functions in the block are transferred to the PLC, if they are to be output at the end of motion, independently of continuous-path mode or of the active exact stop criterion of the exact stop function.

© Siemens AG, 2003. All rights reserved

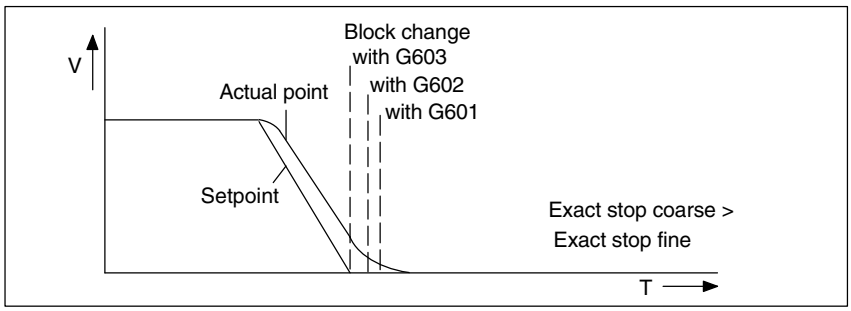


Fig. 2-2 Block change depending on exact stop criteria

Activating exact stop criteria

The exact stop criteria can be selected in every NC parts program block by the following G codes:

- G601 – Exact stop fine
- G602 – Exact stop coarse
- G603 – Interpolator end.

All three criteria can be activated globally by the programming their G function and are evaluated when the exact stop functions G60 or G09 are selected. An active criterion is deactivated by selecting another criterion.
Certain situations in continuous-path mode can necessitate an exact stop which is then performed according to one of the three criteria (see “Implicit exact stop” in Section 2.3).

Evaluation of standstill conditions

In **SW 5** and higher, the standstill conditions:
MD 36000: STOP_LIMIT_COARSE,
MD 36010: STOP_LIMIT_FINE,
MD 36030: STANDSTILL_POS_TOL
can be reevaluated depending on the parameter set.

MD 36012

These three machine data are evaluated with the factor kept in MD 36012: STOP_LIMIT_FACTOR. The factor is valid for all three magnitudes together. The ratio between the three values is retained. If a new value is set in MD 36012: STOP_LIMIT_FACTOR it becomes effective on NEW CONF. The default selection of the factor is 1.0.

Automatic switchover (SW 6.1)

The effectiveness of programmed exact stop criteria can be disabled for G commands of group 1 if an exact stop criterion is defined for the G command of this group channel-specifically via machine data.

© Siemens AG, 2003. All rights reserved

The G commands of this group 1 are:

- Rapid traverse:
  G0
- Machining G codes:
  G1, G2, G3, CIP, ASPLINE, BSPLINE, CSPLINE, POLY, G33, G34, G35, G331, G332, OEMIPO01, OEMIPO02, CT.
  References: /PG/, Programming Guide, Fundamentals, List of G functions

MD 20550: EXACT_POS_MODE allows separate specification for rapid traverse and machining G codes in total.

Table 2-1 Possible settings for MD 20550: EXACT_POS_MODE

Ten's place in MD: Setting for the **machining G codes** mentioned above	**Unit's** place in MD: Setting for G0: **Rapid traverse**
0: With the machining G codes, the **programmed** exact stop conditions become active in each case.	0: With G0, the **programmed** exact stop conditions become active in each case.
1: With the machining G codes, irrespective of the programmed exact stop condition, G601 (positioning error **fine**) becomes active.	1: With G0, irrespective of the programmed exact stop condition, G601 (positioning error **fine**) becomes active.
2: With the machining G codes, irrespective of the programmed exact stop condition, G602 (positioning error **coarse**) becomes active.	2: With G0, irrespective of the programmed exact stop condition, G602 (positioning error **coarse**) becomes active.
3: With the machining G codes, irrespective of the programmed exact stop condition, G603 (positioning error **Setpoint reached**) becomes active.	3: With G0, irrespective of the programmed exact stop condition, G603 (positioning error **Setpoint reached**) becomes active.

Enter the addition of ten's and unit's places true to the digits into the machine data as a value. This results in the range of values 0 – 33. (A leading 0 – ten's place – can be omitted. Any combination of unit's and ten's places as possible per table is permissible.

Example for automatic switchover

MD 20550: EXACT_POS_MODE = 2

The ten's place is 0: With the machining G codes, the programmed exact stop conditions become active in each case.

The unit's place is 2: With G0, irrespective of the programmed exact stop condition, G602 (positioning error **coarse**) becomes active. This has no influence on the response with the other G codes of the 1st group; i.e. in each case, the programmed exact stop condition is active.

© Siemens AG, 2003. All rights reserved

2.3 Continuous-path mode

In continuous-path mode, the path velocity is not decelerated for the block change in order to permit the fulfillment of an exact stop criterion. The objective of this mode is to avoid rapid deceleration of the path axes at the block change point so that the axis velocity remains as constant as possible when the program moves on to the next block. To achieve this objective, the function “LookAhead” is also activated when continuous-path mode is selected (see following Section).

Continuous-path mode causes the smoothing and tangential shaping of sudden block transitions by local changes in the programmed contour. The extent of the change relative to the programmed contour can be limited by specifying the overload factor or smoothing criteria.

Continuous-path operation causes:

- Rounding of the contour
- Shorter machining times through elimination of braking and acceleration processes that are required to comply with the exact stop criterion.
- Improved cutting conditions because of the more constant velocity.

Continuous-path mode operation is suitable if:

- A contour must be traversed as quickly as possible (e.g. with rapid traverse)
- The exact contour may deviate from the programmed contour within a specific tolerance for the purpose of obtaining a continuous contour.

Continuous-path mode operation is suitable if:

- A contour is to be traversed precisely
- An absolutely constant velocity is required.

© Siemens AG, 2003. All rights reserved

Implicit exact stop

In some cases, an exact stop needs to be generated in continuous-path mode to allow the execution of subsequent actions. In such situations, the path velocity is reduced to zero.

- If auxiliary functions are output before the traverse motion, the previous block is only terminated when the selected exact stop criterion is fulfilled.
- If auxiliary functions are to be output after the traverse motion, they are output after the interpolator end of the block.
- If an executable block (e.g. starting a positioning axis) contains no travel information for the path axes, the previous block is terminated on reaching the selected exact stop criterion.
- If a positioning axis is declared to be the geometry axis, the previous block is terminated at the interpolator end when the geometry axis is programmed.
- If a synchronous axis is programmed that was last programmed as a positioning axis or spindle (initial setting of the special axis is positioning axis), the previous block is ended at the interpolator end.
- If the transformation is changed, the block previously processed is terminated with the active exact stop criterion.
- A block is terminated on interpolator end if the following block contains the switchover of the acceleration profile BRISK/SOFT. For further information on BRISK and SOFT see:

  References: /FB/, B2, “Acceleration”
- If the function “Empty buffer” is programmed, the previous block is terminated when the selected exact stop criterion is reached. (See also Section 2.1).

Velocity = 0 in continuous-path mode

Regardless of the implicit exact stop response, the path motion is braked down to zero velocity at the end of the block in cases where:

- Positioning axes have been programmed with syntax POS (see Section 2.1) and have a traversing time that exceeds that of the path axes. Block change occurs when the exact stop fine of the positioning axes is reached.
- The time taken to position a spindle programmed with syntax SPOS is longer than the travel time of the path axes. The block change is carried out when exact stop fine of the positioning spindle is reached.
- The current block contains traversing commands for geometry axes and the following block traversing commands for synchronous axes or, alternatively, the current block contains traversing commands for synchronous axes and the subsequent block traversing commands for geometry axes.
- A synchronization process needs to be carried out (see Section 2.1).

© Siemens AG, 2003. All rights reserved

Auxiliary function output during traversal

In continuous-path mode with auxiliary function output during traversal and short traversing blocks, the path velocity is reduced even before the PLC has acknowledged the auxiliary functions.
In this way the axes can be brought to rest at the end of the block without violating the acceleration limits. At the end of the block acknowledgment is awaited before motion can continue.

Acknowledgment during velocity reduction
If an acknowledgment is received while the velocity is being reduced, the axis is accelerated up to the programmed path velocity again. To prevent this response in continuous-path mode, a time can be set in machine data MD10110: PLC_CYCLE_TIME_AVERAGE (maximum acknowledgment time) of the CNC within which the PLC of the CNC guarantees to acknowledge the auxiliary functions.

Acknowledgment outside travel time
If the travel time is shorter than the setting in MD 10110: PLC_CYCLE_TIME_AVERAGE due to the programmed path length and path velocity with auxiliary function output, the path velocity for the block is reduced with Look Ahead such that the block time equals the MD setting. If acknowledgment is not received within the time, the following prepared block cannot be processed and the axes are braked to rest with setpoint = 0 without considering the acceleration limits.

Acknowledgment not received by block end
If the acknowledgment is not received by the block end in long blocks in which the velocity has not needed to be reduced on account of the PLC acknowledgment time, the velocity is maintained until the block end and then reduced as described above.

Acknowledgment during braking
If the acknowledgment arrives while the axis is decelerating, the axis is not accelerated back up to the requested velocity.

SW 5.1 and higher

Machine data MD 10110: PLC_CYCLE_TIME_AVERAGE is no longer evaluated.

© Siemens AG, 2003. All rights reserved

2.3.1 Velocity reduction according to overload factor

The function lowers the path velocity in continuous-path mode until the nontangential block transition can be traversed in one interpolation cycle whilst respecting the deceleration limit and taking and overload factor into account. With the drop on velocity axial jumps in velocity are produced with a non-tangential contour at the block transition. These jumps in velocity are also performed by the coupled motion synchronous axes. The jump in velocity prevents the path velocity dropping to zero. This jump is performed if the axial velocity was reduced with the axial acceleration to a velocity from which the new setpoint can be reached with the jump. The magnitude of the setpoint jump can be limited using an overload factor. Because the magnitude of the jump is axial, the minimum jump of the path axes which are active during the block change is considered during block transition. With a practically tangential block transition, the path velocity is not dropped if the permissible axial accelerations are not exceeded. In this way, very small sudden changes in the contour can be overtravelled directly.

Overload factor

The overload factor restricts step changes in the machine axis velocity at block ends. So that the velocity jump does not exceed the maximum load on the axis, the jump is derived from the acceleration of the axis. The overload factor defines the amount by which the acceleration rate of the machine axis set in MD 32300: MAX_AX_ACCEL (axis acceleration) may be exceeded for one IPO cycle. The velocity jump is the product of the axial acceleration * (overload factor –1) * interpolator cycle. The factor is stored in machine data MD 32210: MAX_ACCEL_OVL_FACTOR (overload factor for axial velocity jumps).
Factor 1.0 means that only tangential transitions with finite velocity can be traversed. For all other transitions, the velocity is reduced to zero by changing the setpoint. This behavior is equivalent to the function “Exact stop with interpolator end”. This is undesirable for continuous-path mode, so the factor must be set to greater than 1.0.
During start-up it is important that the factor must be reduced when the machine is subject to vibrations on a knee-shaped block transition and rounding is not to be used.
MD 20490 IGNORE_OVL_FACTOR_FOR_ADIS can be set to ensure that block transitions are always smoothed irrespective of the set overload factor.

Selection and deselection of velocity reduction

Continuous-path mode with velocity reduction according to overload factor can be selected modally in every NC parts program block by means of program code G64.

Continuous-path mode G64 can be

- interrupted non modally when exact stop G09 is selected,
- deselected when exact stop G60 is selected,
- deselected when contour rounding G641 is selected.

© Siemens AG, 2003. All rights reserved

Implicit continuous-path mode

- With multiblock threads (series of several thread blocks), the block change between the thread blocks is automatically performed with continuous-path mode G64.
- If it is not possible to insert approximate positioning blocks due to the very short block path lengths (e.g. zero-clocked blocks) in continuous-path mode with approximate positioning G641, the mode is switched over to continuous-path mode G64.

The figure below shows how the function velocity drops according to an overload factor.

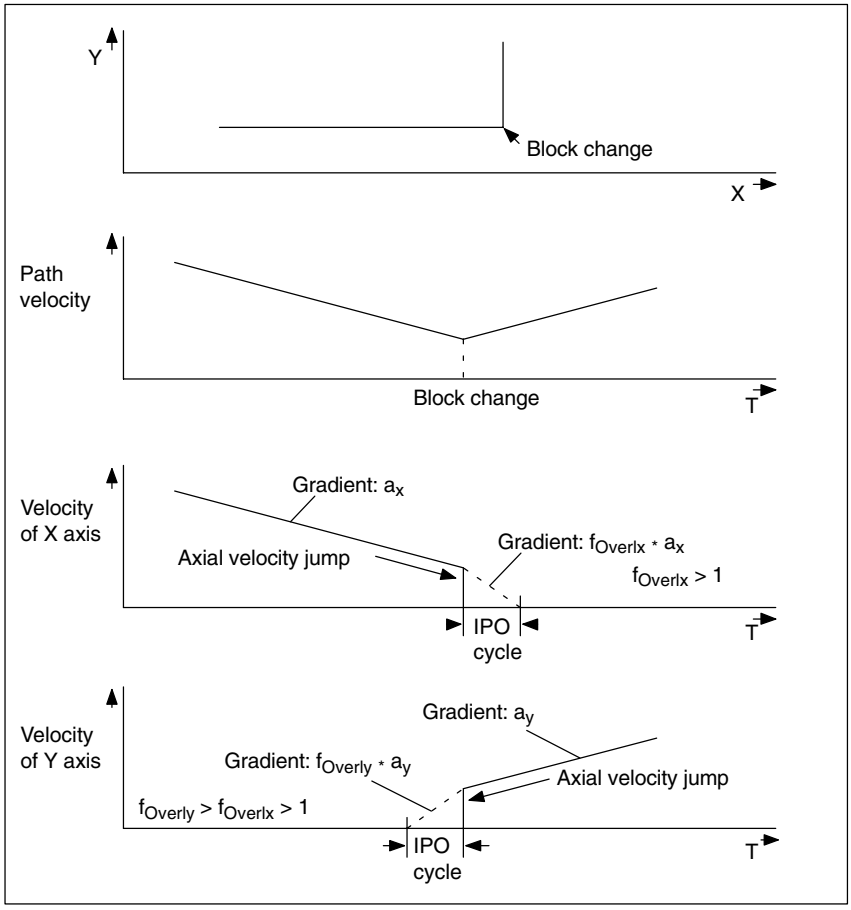


Fig. 2-3 Axial velocity change on block transition

© Siemens AG, 2003. All rights reserved

2.3.2 Rounding according to path criterion

Rounding means that a knee-shaped block transition is changed to a tangential block transition by a local change to the programmed feedrate. Rounding replaces the area in the vicinity of the original knee-shaped block transition (including transitions between blocks inserted by the CNC) by a continuous contour. In this case, it is not only the geometry axes that are taken into account, but all machine axes which are traversing synchronously. The rounding function therefore smoothes the traversing path of orientation axes as well as general velocity step changes in synchronous axes.

Note

Rounding cannot and should not be used as a substitute for the following defined smoothing functions: RND, RNDM, ASPLINE, BSPLINE, CSPLINE.

Rounding is initiated by shortening discontinuously adjoining blocks and inserting one or two intermediate blocks at this point. The original block boundary is removed and can no longer be used for synchronization conditions (e.g. auxiliary function output parallel to motion, stop at end of block). With rounding, all synchronization conditions are best referred to the end of the shortened first block and not to the end of the intermediate rounding block. The following block is thus not started and with a stop at end of block, the contour of the following block can still be changed.

Rounding is only performed if the block transition is to be traveled with finite velocity. The maximum path speed is influenced by the curvature. The maximum acceleration values of the axes are not exceeded. A block without traverse information for the path axes requires velocity zero and therefore no rounding.

Rounding is also used if the traversal of the block transition requires a velocity that lies below the permissible velocity at the end of the block according to G64 (see overload factor). This means that very small knees in the contour (e.g. 0.5 degrees) can be overtravelled directly.

No intermediate rounding blocks

An intermediate rounding block is not inserted in the following situations:

1		Movement **is stopped** between the two blocks. This occurs if ...
	1	The auxiliary function output is programmed before the movement in the following block
	2	The following block does not contain a path movement
	3	An axis which was previously a positioning axis traverses as a path axis for the first time in the following block
	4	An axis which was previously a path axis traverses as a positioning axis for the first time in the following block
	5	The previous block moves geometry axes and the following block does not
	6	The following block moves geometry axes and the previous block does not

© Siemens AG, 2003. All rights reserved

	7	Before thread cutting: The following block uses G33 as a preparatory function and the previous block does not
	8	A change is made between BRISK and SOFT
	9	Axes involved in a transformation are not completely assigned to the path movement (e.g. with oscillation, positioning axes)
2		The rounding block would **slow down** parts program processing. This occurs if ...
	1	When a rounding block is inserted between very short blocks. Since every block requires at least one interpolation cycle, the inserted block would double the processing time.
	2	A block transition is allowed to be crossed with G64 (continuous-path mode without rounding) without a reduction in velocity. Rounding would increase the processing time.
3		Rounding is **not parameterized**. This occurs if ...
	1	ADISPOS == 0 in G0 blocks. (default!) see below
	2	ADIS == 0 in nonG0 blocks. (default!) see below
	3	During the transition between G0 and non-G0 or non-G0 and G0 the smaller of the ADISPOS and ADIS values applies.

Synchronous axes

If a number of paths need to be synchronized (e.g. contour, special axis), then every path must always have its own rounding area. It is not practicable to take this into account precisely. Because of the special importance of the contour (geometry axis), the following procedure is followed:

Table 2-2 Rounding behavior with synchronous axes

Original path for		Result for
geometry axes	**orientation axis/ synchronous axis**	**rounding path**
Smooth	Smooth	Defined path is traversed exactly
Smooth	Knee-shaped curve	2 intermediate blocks, the geometry axes follow the path exactly, all orientation/synchronous axis paths are smoothed
Knee-shaped curve	Smooth	1 intermediate block, the geometry axes perform rounding, all orientation/synchronous axes are smoothed
Knee-shaped curve	Knee-shaped curve	1 intermediate block, the geometry axes perform rounding, all orientation/synchronous axes are smoothed

Path criterion

The size of the rounding range can be controlled by path criteria ADIS and ADISPOS. These are the precoincidences for the block change. ADIS and ADISPOS the distance which the rounding block may begin, at the earliest, before the end of the block or the distance after the end of block within which the rounding block must be terminated.

© Siemens AG, 2003. All rights reserved

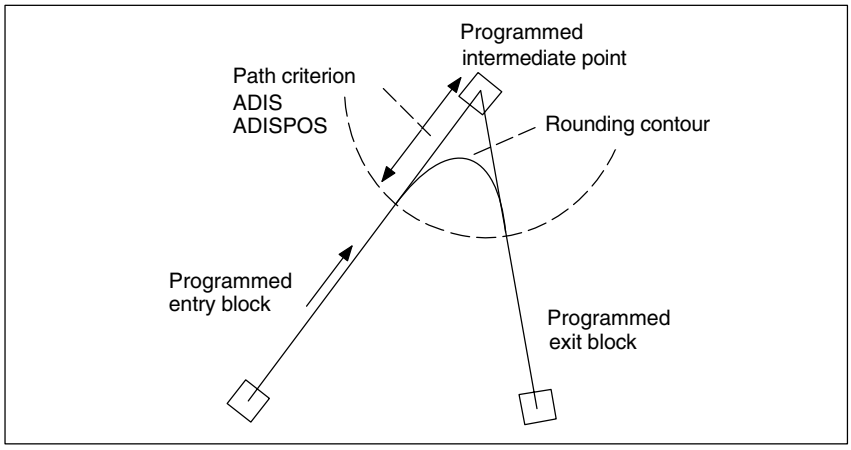


Fig. 2-4 Example for rounding an angular block transition

Acute angles produce rounding curves with a large degree of curvature and therefore cause a corresponding reduction in velocity.

Parameterization of the path criterion

- ADIS is specified in the standard unit of length (e.g. mm) and is automatically converted to the internal increments (unit) by the control and used in this form both for linear and rotary axes.
- ADISPOS is programmed in the same way as ADIS, but must be used specifically for movements in rapid traverse mode (G00).
- ADIS and ADISPOS are preset in the parts program.
  E.g. ADIS = 0.3 means a path criterion with a rounding distance of 0.3 mm, and the same goes for ADISPOS.

Scope of the path criterion

- ADIS and ADISPOS are global.
- ADIS or ADISPOS must be programmed. If the default is zero, G641 behaves like G64.
- If only one of the blocks involved is rapid traverse G00, the smaller rounding distance applies.
- If a very small value is used for ADIS, it is important that control makes sure that every interpolated block, even an intermediate rounding block, contains at least one interpolation point. The maximum tool path velocity is therefore limited to ADIS/interpolation cycle (see zero blocks, Section 2.1).
- Regardless of ADIS and ADISPOS, the rounding area is limited by the block length.

© Siemens AG, 2003. All rights reserved

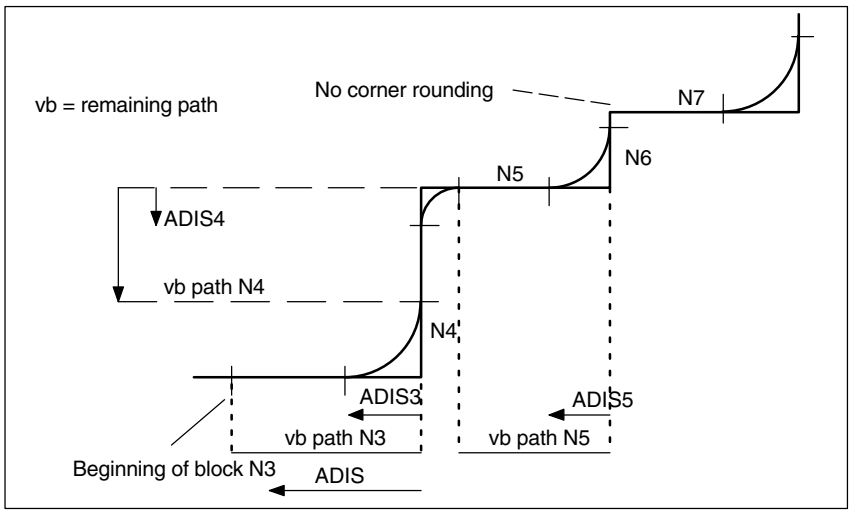


Fig. 2-5 Path with limitation of ADIS

Programming example

```
N1 G641Y50 F10 ADIS= 0.5
N2 X50
N3 X50.7
N4 Y50.7
N5 Y51.4
N6 Y51.0
N7 X52.1
```

In blocks with short distances (distance < 4 * ADIS and < 4 * ADISPOS respectively), the rounding distance is reduced so that a traversable part of the original block is retained. The remaining length depends on the axis path and is approximately 60% of the distance still to be traversed in the block. ADIS or ADISPOS are therefore reduced to the remaining 40% of the distance to be traversed. This algorithm prevents a rounding block being inserted for a very small change in contour. In this case, switchover to continuous-path mode G64 is automatic until rounding blocks can be inserted again.

Selection and deselection of Look Ahead

Program code G641 can be inserted in any NC parts program block to modally select rounding according to a path criterion. Before or on selection, the path criteria ADIS/ADISPOS must be specified. Continuous-path mode G641 can be

- interrupted non-modally when exact stop G09 is selected,
- deselected when exact stop G60 is selected,
- deselected when velocity drop G64 is selected.

Response in SW 4.3 and higher

In addition to rounding using the command G641, rounding with axial tolerances enabled modally using G642 is possible in SW 4.3 and higher.

In this case, the rounding does not take place within a defined ADIS range, but it ensures the axial tolerances defined in MD 33100: COMPRESS_POS_TOL are retained. In all other respects, the function is the same as G641. See also:

References: /PG/, Programming Guide, Fundamentals, Section 5.2

© Siemens AG, 2003. All rights reserved

Expansions SW 5.3

G643 is not used to generate a separate rounding block, but axis-specific block-internal rounding movements are inserted.

Expansions SW 6

The expansions described in the following make the behavior with G642 and G643 more precisely, and

rounding with contour tolerance is introduced.

When rounding with G642 and G643, the maximum permissible deviations of each axis are normally specified.

MD 20480: SMOOTHING_MODE
can be used to configure rounding with G642 and G643 such that instead of the axis-specific tolerances a contour tolerance and an orientation tolerance can be specified. In this case, the tolerance of the **contour** and of the **orientation** is set using two independent setting data, which can be programmed in the NC program, so for each block transition other setting data can be specified.

Contour tolerance

A contour tolerance is specified using the setting data:

SD 42465: SMOOTH_CONTUR_TOL
This setting data is used to define the **maximum** tolerance for the the contour when rounding.

Orientation tolerance

To specify an orientation tolerance, the setting data:

SD 42466: SMOOTH_ORI_TOL is used.
This setting data is used to define the **maximum** tolerance for the **tool orientation** when rounding. This data defines the maximum permissible angular displacement of the tool orientation.
This data is only effective if an **orientation transformation** is active.

Possible combinations

The effect of axis tolerances for G642 and G643 in MD 33100: COMPRESS_POS_TOL and the setting data introduced above are specified in the Table below.

© Siemens AG, 2003. All rights reserved

Table 2-3 MD 20480: SMOOTHING_MODE must be loaded with two decimal places as follows:

Value	G642 (Value setting in the ten's place of MD 20480)	G643 (Value setting in the units place of MD 20480)
0	G642 uses axis-specific tolerances. These are set using the axis-specific MD 33100: COMPRESS_POS_TOL are set. (default). Response as in SW 4.3	G643 uses axis-specific tolerances. These are set using the axis-specific MD MA_COMPRESS_POS_TOL are set. (default).
1	When rounding with G642, the contour tolerance SD 42465: SMOOTH_CONTUR_TOL is used for the geometry axes. The remaining axes are rounded using the axis-specific tolerances in MD 33100: COMPRESS_POS_TOL.	When rounding with G643, the contour tolerance SD 42465: SMOOTH_CONTUR_TOL is used for the geometry axes. The remaining axes are rounded using the axis-specific tolerances in MD 33100: COMPRESS_POS_TOL.
2	The orientation movement with G642 is rounded using the angle tolerance SD 42466: SMOOTH_ORI_TOL. All the other axes use the axis-specific tolerances MD 33100: COMPRESS_POS_TOL.	The orientation movement with G463 is rounded using the angle tolerance SD 42466: SMOOTH_ORI_TOL. All the other axes use the axis-specific tolerances MD 33100: COMPRESS_POS_TOL.
3	Combination of the two options 1x and 2x. i.e. the following tolerances for G642are used: SD 42465: SMOOTH_CONTUR_TOL and SD 42466: SMOOTH_ORI_TOL. Further axes are rounded using the axis-specific tolerance.	Combination of the two options 1x and 2x. i.e. the following tolerances for G643 are used: SD 42465: SMOOTH_CONTUR_TOL and SD 42466: SMOOTH_ORI_TOL. Further axes are rounded using the axis-specific tolerance.

Differences G642, G643

With regard to their rounding behavior, the commands G642 and G643 differ as follows:

	G642	G643
	With G642, the rounding travel is determined based on the **shortest** rounding travel of all axes. This value is taken into account when generating a **rounding block**. With G642, the rounding area results from the smallest tolerance setting.	In the case of G643, the rounding travel of each axis can be **different**. The rounding travels are taken into account axis-specifically and **block-internally**. Very different specifications for the contour tolerance and the tolerance of the tool orientation can only have effect with G643.

Supplementary conditions

The expansions "contour tolerance" and "orientation tolerance" exist only in systems with the **option** polynomial interpolation. The orientation transformation option is required additionally for rounding of orientations with angle tolerance input.

2.3.3 Jerk limitation on path through velocity reduction

Introduction

With jerk limitation on the path, a 3rd method of influencing the continuous-path mode is introduced. While the "Velocity reduction according to overload factor" function (see Subsection 2.3.1), limits the rate of velocity change, the function described here restricts step changes in the acceleration rate (jerks). See also Section "Path-related jerk limitation" in the following References.

References: /FB/, B2, "Acceleration".

© Siemens AG, 2003. All rights reserved

Occurence of jerking

When sections of the contour consisting of blocks containing different degrees of curvature (e.g. circle straight line transitions) are machined, step changes in the acceleration rate occur in continuous-path mode.

Reduction of jerking

The severity of such jerks can be reduced by decreasing the path velocity at transitions between blocks containing different degrees of curvature. A smoother transition is thus achieved between the contour sections.

Jerk limit

The user sets the maximum jerk that may occur on a geometry axis during a block transition in MD 32432: PATH_TRANS_JERK_LIM (maximum axial jerk of geo axis at block limit).

Activation

Jerk limitation at block transitions becomes implicitly active if continuous-path mode is programmed with G64 or G641 and SOFT acceleration characteristics. MD 32432: PATH_TRANS_JERK_LIM must be set to a positive value.

Example

A program excerpt with activation of jerk limitation is shown in Chapter 6.

© Siemens AG, 2003. All rights reserved

2.3.4 Machine axis-specific jerk limitation (SW 5 and higher)

Introduction

Jerk limitation for the path movement was previously defined by machine data (MD 20600: MAX_PATH_JERK. This machine data specifies a maximum value for the change in acceleration for all machine axes.

MD 32431

The newly introduced machine data MD 32431 MAX_AX_JERK[..] allows you to set the change in acceleration separately for each machine axis, as is already the case for acceleration limitation in machine data MD 32300: MAX_AX_ACCEL[..]. MD 32431: MAX_AX_JERK[..] acts on the axes interpolated by the path and when SOFT (smooth acceleration curve) is active.

Benefits

The introduction of axis-specific machine data has the following benefits:

- Immediate allowance is made in the interpolation for the dynamic response of the axes, which can then be fully utilized for each axis.
- Jerk limitation for separate axes is performed not just in linear blocks, but also in curved contours.

Machine data in the block

A basic distinction is made between the axis acceleration curve within a block and at the transition between two blocks.

The global limitation of the path acceleration change for all axes of a channel is still implemented with the machine data:

- MD 20600: MAX_PATH_JERK
- MD 42510: SD_MAX_PATH_JERK

If necessary, it is therefore possible to set a higher jerk limitation than the dynamic path response on curved contours. If you want to suppress the impact of channel-related data, you should assign a very high value to this machine data.

Note

MAX_PATH_JERK and SD_MAX_PATH_JERK are not scaled in G00 blocks. They are therefore unaffected by the relevant scale factors (G00_JERK_FACTOR and G00_ACCEL_FACTOR).

© Siemens AG, 2003. All rights reserved

MD 20602 and MD 20603

The following channel machine data can be set to avoid short-term overloading of the axis acceleration and axis jerk when the path velocity is changed on curved contours:

- MD 20602: CURV_EFFECT_ON_PATH_ACCEL:
  Reduces the path velocity to leave a reserve for the path acceleration. This is valid for BRISK (time-optimized operation with abrupt changes in the acceleration curve) and SOFT (jerkfree acceleration curve).
- MD 20603: CURV_EFFECT_ON_PATH_JERK:
  Reduces the path velocity to leave a reserve for the path jerk (only valid for SOFT).

You use the settings of these machine data to shift the focus between velocity and acceleration time.

The setting has no effect on machining with linear machine axis motion (linear blocks and no kinematic transformation active). It has only a minimal effect on slightly curved contours.

The default settings of these machine data are selected such that they have no effect on machining. It is therefore possible to use old machine data without modification. The timing is unchanged.
For JOG and POS movements (asynchronous axis motion), the previous jerk limitation via MD 32430: JOG_AND_POS_MAX_JERK continues to apply whenever MD 32420: JOG_AND_POS_JERK_ENABLE is enabled.

Scaling

MD SOFT_ACCEL_FACTOR scales the permissible acceleration **for SOFT** defined in machine data MD 32300 MAX_AX_ACCEL. This machine data has the same effect on POS and JOG axes.

G00_ACCEL_FACTOR scales the permissible **axial** acceleration. A harder acceleration can therefore be used for a programmed G00 preparatory function than for machining. This axial machine data is only effective during path movements. The rapid traverse overlay in JOG mode is not affected (acceleration limitation for G00 = G00_ACCEL_FACTOR * MA_AX_ACCEL).

Machine data at block transition

The machine data at the block transition are only valid when continuous-path mode (G64, G641) is active. The following machine data determine the response at the block transition:

- MD 32210: MAX_ACCEL_OVL_FACTOR[..]
- MD 32432: PATH_TRANS_JERK_LIM[..]
- MD: G00_JERK_FACTOR[..]

MD 32210: MAX_ACCEL_OVL_FACTOR[..] limits the acceleration overshoot for each axis as a consequence of a corner at the block transition.

© Siemens AG, 2003. All rights reserved

If the value of this machine data is greater than 1, a jump in axis velocity occurs at a corner. The associated peak in axis acceleration can be offset by the following effects:

- By the interpolation cycle, since the position controller only receives new setpoints at the interpolation cycle rate (the longer the cycle, the stronger the smoothing effect)
- By the axial setpoint filter ahead of the position controller (MD 32410: AX_JERK_TIME)
- By the dynamic response of the position and drive controller and, if applicable, an inactive feedforward control (FFWOF).

Note

The effect of this machine data depends on machine data SOFT_ACCEL_FACTOR (acceleration overload during SOFT), G00_ACCEL_FACTOR and G00_JERK_FACTOR.

MD: G00_JERK_FACTOR[..] scales the permissible axial jerk. This enables a higher axial jerk to be applied during positioning (programmed preparatory function G00) than during machining
(jerk limitation for G00 = G00_JERK_FACTOR * MA_AX_JERK).

Note

This axial machine data is only effective during path movements. The rapid traverse overlay in JOG mode is not affected.

MD 32432: PATH_TRANS_JERK_LIM[..] limits the jerk overshoot as a result of a jump in curvature at a tangential block transition. The axis jerk is offset by the same effects as the axis acceleration. The effect on the machine at a jump in curvature is much less pronounced than at a corner. In order to avoid a disturbing reduction in velocity, this machine data is also scaled by G00_JERK_FACTOR.

(limits the jerk overshoot for G00
= G00_JERK_FACTOR * PATH_TRANS_JERK_LIM).

© Siemens AG, 2003. All rights reserved

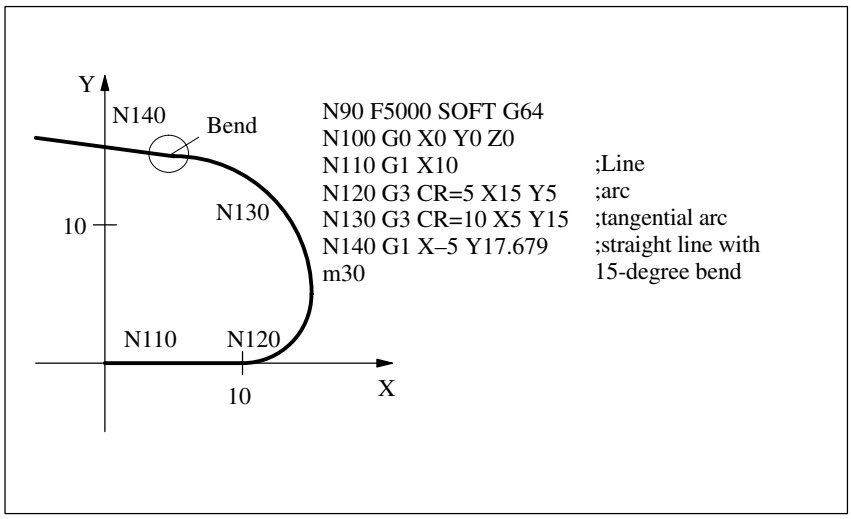


Fig. 2-6 Example: Contour consisting of straight lines and arcs

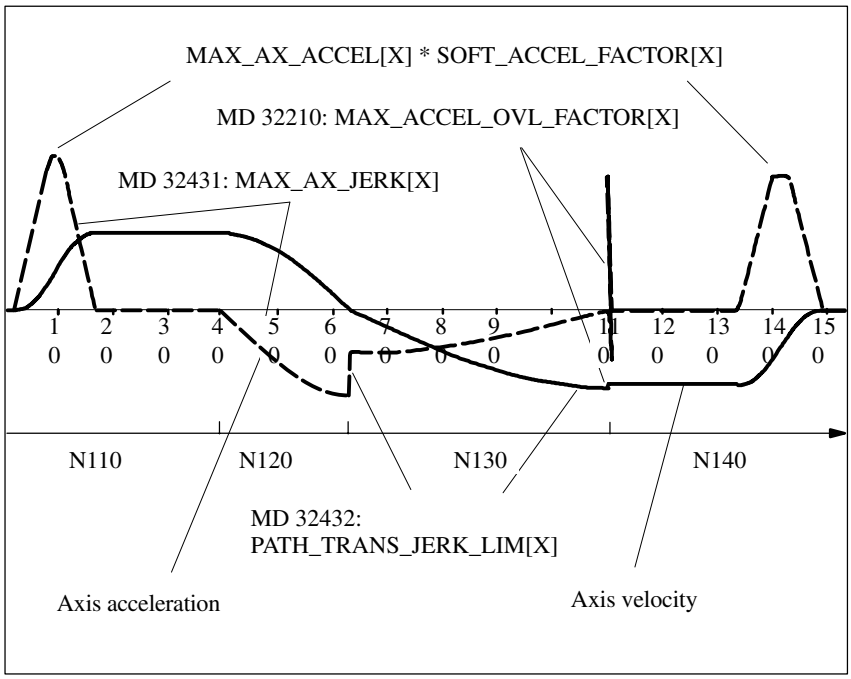


Fig. 2-7 Velocity and acceleration (X axis) for example contour

Short-term acceleration and jerk overshoots are non-critical in most situations. In such cases, the response at the block transition can be set separately from the response in the block.
In other cases, you must set MD 32210: MAX_ACCEL_OVL_FACTOR[..] approximate to 1 and MD 32432: PATH_TRANS_JERK_LIM[..] equal to MD 32431: MAX_AX_JERK[..].

© Siemens AG, 2003. All rights reserved

Note

MD 32432: PATH_TRANS_JERK_LIM[..] was previously only used for geometry axes. In SW 5 and higher, the data is used for **all** axes traversing synchronously on the path. If you have already set the machine data previously for rotary axes, the program execution may be significantly slower. You can achieve the original performance by increasing the appropriate values.

It is not possible to combine SOFT with a knee-shaped acceleration characteristic. The machine data MD 35240: ACCEL_TYPE_DRIVE[..] is referenced and the acceleration limit is correspondingly observed.

Activation

The path motion is defined by the G codes of the 1st G group (e.g. G1). The machine data MD 32431: MAX_AX_JERK[..] and
MD 20603: CURV_EFFECT_ON_PATH_JERK are only valid for a path movement with SOFT active.
MD 20602: CURV_EFFECT_ON_PATH_ACCEL is valid for a path movement with both SOFT and BRISK.

The jerk limitation for individual axes is set via machine data. The values can also be set from the parts program with the corresponding access rights. Please note that this modification is not canceled by a RESET.

Example

```
N100  G0 x0 y0 z20 BRISK
N200  G0 x100 SOFT G641 ADIS=.1 ADISPOS=3     ; Positioning
N300  z0                                      ; Infeed
N400  G1 F200 z–10                            ; Drilling
N500  G0 z20                                  ; Retraction
N600  x0                                      ; Repositioning
N700  M17
```

Choosing the appropriate machine data thus automatically achieves a harder acceleration on a highly dynamic Z axis than on a soft X axis.

- N100:
  MAX_AX_ACCEL is scaled with G00_ACCEL_FACTOR.
- N200, N300, N500, N600:
  MAX_AX_ACCEL is scaled with
  G00_ACCEL_FACTOR * SOFT_ACCEL_FAKTOR and
  MAX_AX_JERK is scaled with G00_JERK_FACTOR.
- N400:
  MAX_AX_ACCEL is scaled with SOFT_ACCEL_FACTOR.

© Siemens AG, 2003. All rights reserved

2.3.5 Smoothing of the path velocity (SW 5.3 and higher)

Application

In some applications in mold making, especially in the case of high speed cutting, it is desirable to achieve a constant path velocity.

Response without smoothing

The velocity control utilizes the given axis dynamic response. If the programmed feed rate cannot be achieved, the path velocity is brought to the parameterized axial limit values and the limit values of the path (velocity, acceleration, jerk). This can lead to repeated braking and acceleration on the path.

If a short acceleration takes place during a machining function with high path velocity, and is thus followed almost immediately by braking, the reduction in the machining time is only minimal. Such acceleration, however, can lead to undesirable effects, e.g. machine resonances.

Solution: Smoothing

In some cases, it can therefore be reasonable to sacrifice of transient acceleration processes in favor of a smoother tool path velocity. The following sections describe the conditions and possible settings to avoid less effective accelerations/decelerations.

Benefits

The following improvements are possible:

- Avoidance of excitations of possible machine resonance due to continuous, transient braking and acceleration processes (in the area of less IPO cycles).
- Avoiding of constantly varying cutting rates due to acceleration which brings no significant shortening of the program running time.

Smoother path movement

If the velocity is controlled smoother and not every acceleration process is carried out, both advantages can be achieved without any undesired extended processing time.

Decision criteria

The control system makes an appropriate decision based on

- Smoothing frequency for the tool path velocity (MD)
- Tolerable loss in productivity when suppressing accelerations/decelerations (MD).

© Siemens AG, 2003. All rights reserved

Setting parameters for path smoothing

The user can define the following parameters to set the path smoothing:

- Extension of the processing time via MD 20460.
  If any acceleration processes are not carried out, the processing time of this parts program will extend (see example).
- Specification of the resonance frequencies of the axes involved via MD 32440.
  Only such acceleration processes are to be removed which result in a substantial excitation of machine resonances.
- Taking into account the programmed feedrate for the smoothing MD 20462.

MD 20460

A percentage value representing the maximum tolerated extended duration must be entered here in MD 20460: LOOKAH_SMOOTH_FACTOR. The % value defines how much may a machining step without accelerations may be longer than the appropriate step when performing the accelerations/ decelerations. This would be a “worst case” value where all acceleration within a subprogram, with the exception of the initial approach motion, is removed in smoothing.
The actual extending of the program duration will always be less, and possibly even 0 if the criterion fails to become applicable for any of the programmed accelerations.It is thus feasible to enter values of 50–100%, without this resulting in a significant extending of the machining time.
It is thus feasible to enter values of 50–100%, without this resulting in a significant extending of the machining time.

MD 20462 (SW 6.1)

MD 20462: LOOKAH_SMOOTH_WITH_FEED defines that the programmed feedrate will also be taken into account in the smoothing. If the MD is set to 1 (default), the smoothing factor is observed especially exactly if the override is set to 100 %.

MD 32440

MD 32440: LOOKAH_FREQUENCY keeps a smoothing frequency for Look Ahead. Acceleration and braking processes running at a frequency higher than parameterized in this MD are smoothed or reduced in their dynamic response depending on their parameterization in
MD 20460: LOOKAH_SMOOTH_FACTOR and
MD 20465: ADAPT_PATH_DYNAMIC (see 2.3.6 Dynamic response adaptation).
A minimum of all axes involved in the path is always determined.

If vibration is excited in the mechanical system of these axes and if the corresponding frequency is known, then this MD should be set to a value smaller than this frequency.

The integrated measuring function, for example, can be used to determine the required resonance frequencies.
The minimum of these machine data 32440 is determined as a value *fpath* based on the axes involved in the path.
For the smoothing, only acceleration processes are taken into account where the starting or end velocity of this movement is reached again within a time $t = t2–t1 = 2/fpath$ (see example).

© Siemens AG, 2003. All rights reserved

Note

The smoothing of the path velocity does not lead to contour errors. Fluctuation of the axis velocity due to curves in the contour in the case of a constant path velocity can still occur and are not reduced here.

Fluctuations in the path velocity due to specification of a new feed rate are similarly unaffected. This remains the responsibility of the programmer of the subprogram.

Activation

The smoothing of the path velocity is activated with the NEW CONFIG machine data MD 20460: LOOKAH_SMOOTH_FACTOR. With the default value 0, the function is deactivated.

Example

The following parameters are assumed:

MD 20460: LOOKAH_SMOOTH_FACTOR = 10%

MD 32440: LOOKAH_FREQUENCY[AX1] = 20Hz
MD 32440: LOOKAH_FREQUENCY[AX2] = 20Hz
MD 32440: LOOKAH_FREQUENCY[AX3] = 10Hz

Drawings and explanations are to be found on the next page.

© Siemens AG, 2003. All rights reserved

The path involves the 3 axes X=AX1, Y=AX2, Z=AX3. The minimum from MD 32440: LOOKAH_FREQUENCY for these 3 axes is thus 10 Hz. This means that any acceleration which is completed within a period of t2–t1 = 2/10Hz = 200ms is examined. The time t2 is the time after which, following acceleration from velocity v1, the velocity returns to this velocity v1.

The extending of the execution time is also only considered within this range. The time characteristic without smoothing is shown in Fig. 2-8, the time characteristic resulting from smoothing in Fig. 2-9.

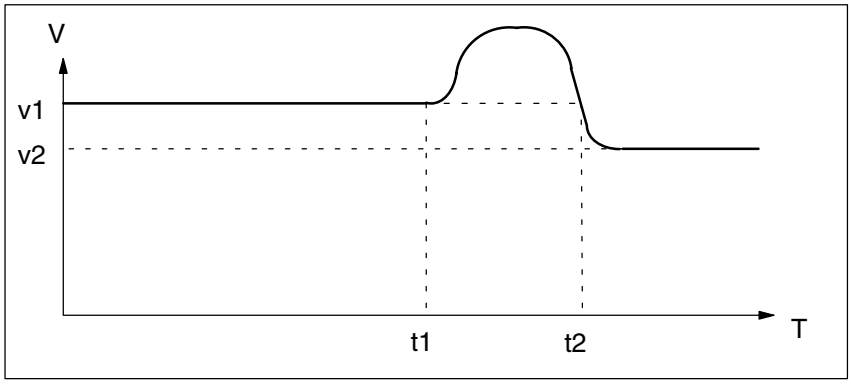


Fig. 2-8 Characteristic of the time-optimized path velocity

The course as shown in Fig. 2-8 results if the time t2–t1 is greater than 200ms or if the additional program execution time t3–t2 (illustration below) is greater than 10% (= MD 20460: LOOKAH_SMOOTH_FACTOR) of the value t2–t1.

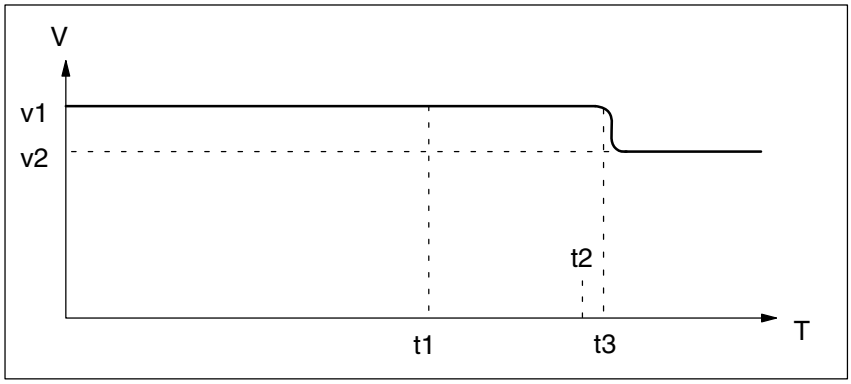


Fig. 2-9 Characteristic of the smoothed path velocity

The course as shown in Fig. 2-9 results if the time t2–t1 is less than 200ms or if the additional program execution time t3–t2 is 10% of t2–1 as the maximum.

© Siemens AG, 2003. All rights reserved

2.3.6 Dynamic response adaptation (SW 6.3)

Introduction

In order to protect the mechanical components and drives on a machine tool against overloading, the dynamic response of individual axes can be limited by setting parameters in axial machine data. The maximum values for velocity and acceleration are specified by physical limit values (e.g. maximum speed and torque of drive). The maximum jerk in MD 32431: MAX_AX_JERK does not serve, however, to maintain a real physical limit, but is used to generate a softer setpoint characteristic for SOFT. With BRISK the maximum acceleration value in MD 32300: MAX_AX_ACCEL must be taken into account for this purpose.

Operation without adaptation

With systems up to SW 5, the risk of overloading is reduced by setting the jerk and acceleration limits so low that highfrequency changes in the path velocity do not cause excessive oscillation.

The lower dynamic response settings often exclude higher path velocities and thus the possibility of shorter program processing times.

With dynamic response adaptation

In operation with dynamic response adaptation, **highfrequency** changes in the path velocity are **automatically** executed at **reduced jerk or acceleration values**.

Lowfrequency changes to the path velocity are applied at the full dynamic response limits, but highfrequency changes at reduced dynamic limits. The result is the shortest possible machining time without undesirable excitation of machine resonance if the process has been optimally parameterized.

Note

The procedure should be utilized in conjunction with the path velocity smoothing option described in Subsection 2.3.5. See example 2.

In the default setting, path smoothing is automatically activated with the adaptation option.

The parameter settings and mode of operation are described in detail in the following subsection.

Supplementary conditions

Dynamic adaptation for the path axes is active in

- continuous-path mode (see activation) and
- exact stop with SOFT and BRISK.

In continuous-path mode, the optimum effect of the adaptation of the path dynamic response is achieved at 100% override, since calculations for this purpose are performed at the preprocessing stage. Significant deviations from this override factor or even switching operations while machining is in progress will adversely affect the desired smoothing effect.

© Siemens AG, 2003. All rights reserved

Adaptation of the path velocity is active for all **contour motions**. It is activated internally under the following conditions:

- With G0 blocks
- Stop requests during axis motion, e.g. by NC STOP, RESET, override setting 0.
- Changes to the override in continuous-path mode.

Dependencies

Flexible adaptation of the dynamic response is based on

- a reduction factor
- an excitation frequency.

Reduction factor

Very brief changes in the path velocity are executed at

- low acceleration limits (with BRISK) or
- low jerk limits (with SOFT).

This factor for the dynamic path response can be specified in MD 20465: ADAPT_PATH_DYNAMIC.
ADAPT_PATH_DYNAMIC[0] is operative for BRISK and reduces the acceleration rate.
ADAPT_PATH_DYNAMIC[1] is operative for SOFT and reduces the permissible jerk.

Excitation frequency

The reduced limit values for jerk or acceleration must be applied only to braking or acceleration processes which will excite machine resonance.

This is excitation which occurs at frequencies above the frequency set for each individual axis in MD 32440: LOOKAH_FREQUENCY.

The minimum frequency is calculated from these machine data for all the axes involved in the path as the relevant frequency *f* at which the path velocity must be modified.

Activation

The function is activated if the reduction factor setting in MD 20465: ADAPT_PATH_DYNAMIC is > 1. A setting of 1 means no adaptation (default). MD 20465: ADAPT_PATH_DYNAMIC becomes operative after NEW CONFIG.

In continuous-path mode, MD 20465: ADAPT_PATH_DYNAMIC > 1 also activates smoothing of the path velocity with a smoothing factor of 100% (if this is not already activated) in addition to adaptation of dynamic path response. This means that a value of 0 in MD 20460: LOOKAH_SMOOTH_FACTOR corresponds to a setting of 100%.

Of the settings in MD 32440: LOOKAH_FREQUENCY for all axes involved in the path, the lowest frequency *f* is taken into account in defining the time window.

© Siemens AG, 2003. All rights reserved

Adaptations

In order to clarify the adaptation processes sketched below, please note the following basic principles:

The size of the time window is $t_{adapt} = 1 / f$

- The time needed to change the velocity is less than t_{adapt}
  - The acceleration rates are reduced by a factor of higher than 1 and less than/equal to the value in MD 20465: ADAPT_PATH_DYNAMIC. The reduction in acceleration rate increases the time taken to change the velocity. Please note the following 2 different cases:
    1.
    The acceleration is reduced at a value lower than MD 20465 such that the process takes t_{adapt} [s]. The maximum permissible reduction need not be utilized.
    2.
    The acceleration time is reduced at the value in MD 20465. In spite of the lower acceleration rate, the process takes less time than $t_{adapt.}$
    The maximum permissible reduction has been fully utilized.
- The time needed to change the velocity is greater than $t_{adapt.}$
  No dynamic response adaptation is necessary.

Example 1

The following parameters are assumed:

```
$MC_ADAPT_PATH_DYNAMIC[0] = 1.5
$MC_LOOKAH_SMOOTH_FACTOR = 1.0        ; Path smoothing deactivated
$MA_LOOKAH_FREQUENCY[AX1] = 20Hz
$MA_LOOKAH_FREQUENCY[AX2] = 10Hz
$MA_LOOKAH_FREQUENCY[AX3] = 20Hz
```

BRISK is active.

AX2
For path motions in which the 2nd axis (AX2) is involved, all braking and acceleration processes which take a maximum time of 1/10 Hz = 100 ms are adapted.

AX1, AX3
If only the axis AX1 or AX3 is involved, then all braking and acceleration processes which take a maximum time of 1/20 Hz = 50 ms are adapted. This time is designated tadapt*xy* in the following diagrams:

© Siemens AG, 2003. All rights reserved

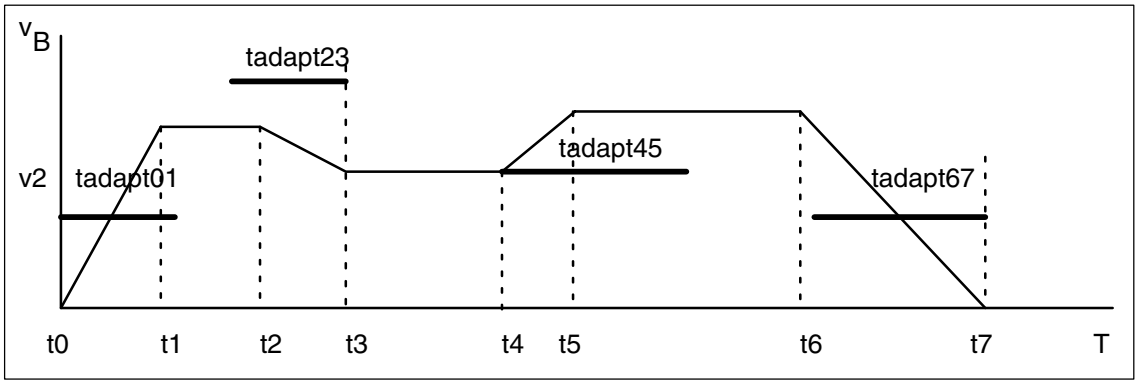


Fig. 2-10 Time-optimized path velocity characteristic for example 1

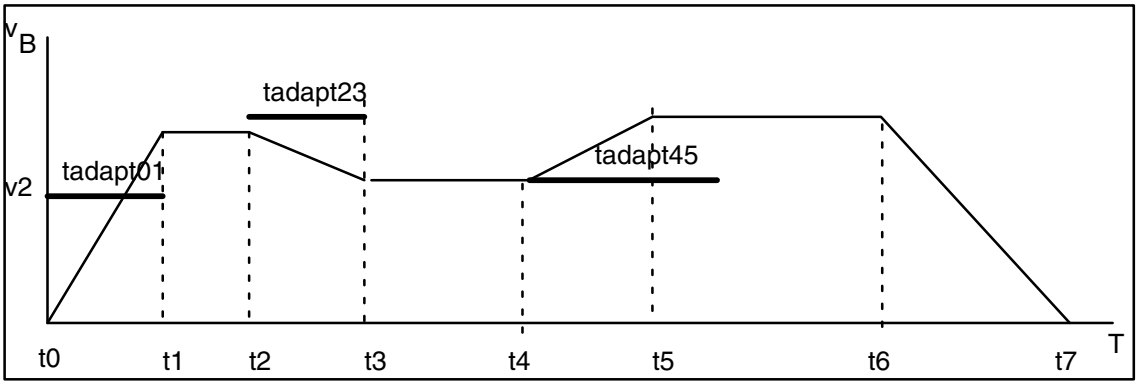


Fig. 2-11 Path velocity characteristic with dynamic response adaptation for example 1

The acceleration process between t0 and t1 and the braking process between t2 and t3 are “stretched” to time periods tadapt01 and tadapt23 respectively as a result of the acceleration adaptation.

The acceleration rate between t4 and t5 is reduced by a factor of 1.5, according to the parameterization in MD 20465: ADAPT_PATH_DYNAMIC[0]. Acceleration is still completed before the time period tadapt45 elapses.

The braking process between t6 and t7 remains unchanged because it takes longer than tadapt67.

© Siemens AG, 2003. All rights reserved

Example 2 Combination with path velocity smoothing

The following parameters are assumed:

$MC_LOOKAH_SMOOTH_FACTOR = 80%
$MC_ADAPT_PATH_DYNAMIC[0] = 3

$MA_LOOKAH_FREQUENCY[AX1] = 20Hz
$MA_LOOKAH_FREQUENCY[AX2] = 20Hz
$MA_LOOKAH_FREQUENCY[AX3] = 20Hz

BRISK is active.

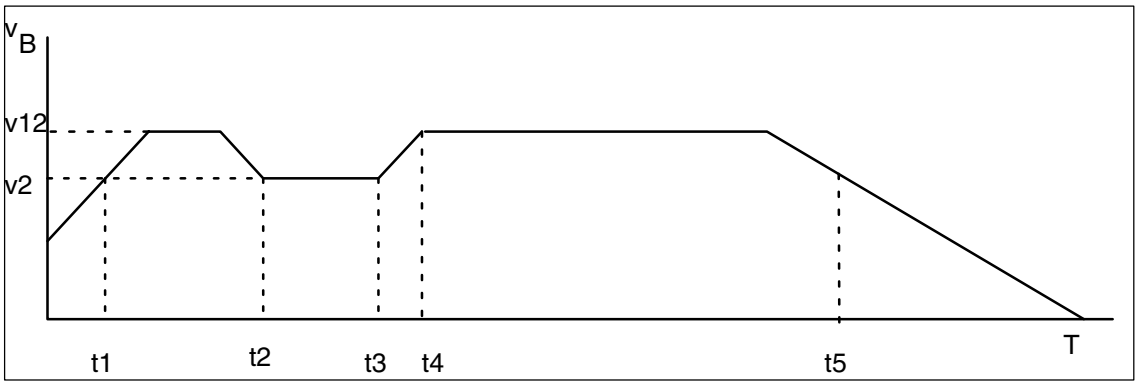


Fig. 2-12 Time-optimized path velocity characteristic for example 2

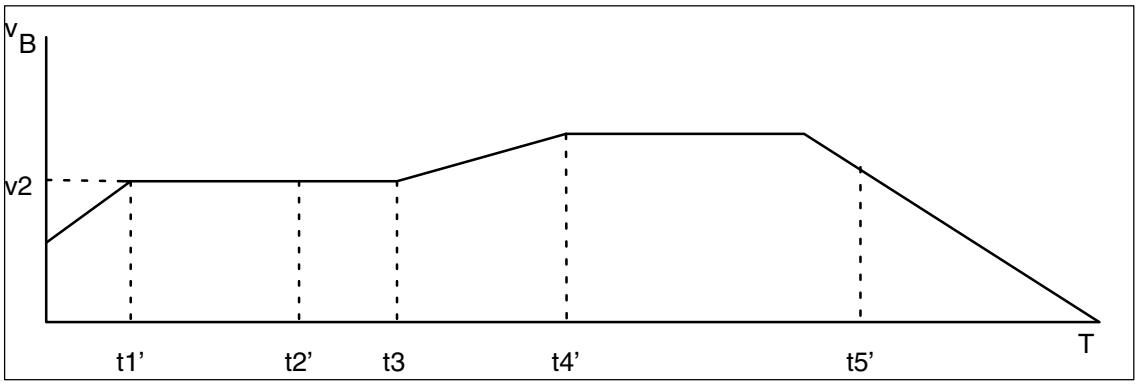


Fig. 2-13 Path smoothing + dynamic response adaptation for example 2

Figures 2-12 and 2-13 clearly illustrate the difference the difference in the parameterization in MD 20465: ADAPT_PATH_DYNAMIC and MD 20460: LOOKAH_SMOOTH_FACTOR.
The entire acceleration and braking characteristic between t1 and t2 is omitted because the increase in machining time without acceleration to v12 is less than the setting in MD 20460: LOOKAH_SMOOTH_FACTOR.

The entire acceleration and braking characteristic between t3 and t5 does not fulfill this condition or takes longer than parameterized in MD 32440: LOOKAH_FREQUENCY (in this instance: 2/20Hz = 100ms). The acceleration process from t3 to t4 is, however, shorter than 50ms (=1/20Hz) and is therefore executed at a rate reduced by a factor of 3 (= MD 20465: ADAPT_PATH_DYNAMIC[0]).

© Siemens AG, 2003. All rights reserved

The acceleration up to t1 left over after path smoothing is stretched to the time period up to t1' by the dynamic path adaptation.

This example shows why the dynamic path adaptation function should be used, whenever possible, in conjunction with path smoothing in continuous-path mode. Using a combination of the two functions means that only those acceleration or braking processes on the path that are not eliminated by path smoothing need to be optimized for the relevant machine.

Example 3

Effect of adaptation and smoothing of path velocity with SOFT

The following parameters are assumed:

$MC_LOOKAH_SMOOTH_FACTOR = 100%
$MC_ADAPT_PATH_DYNAMIC[1] = 4

$MA_LOOKAH_FREQUENCY[AX1] = 10Hz
$MA_LOOKAH_FREQUENCY[AX2] = 10Hz
$MA_LOOKAH_FREQUENCY[AX3] = 20Hz

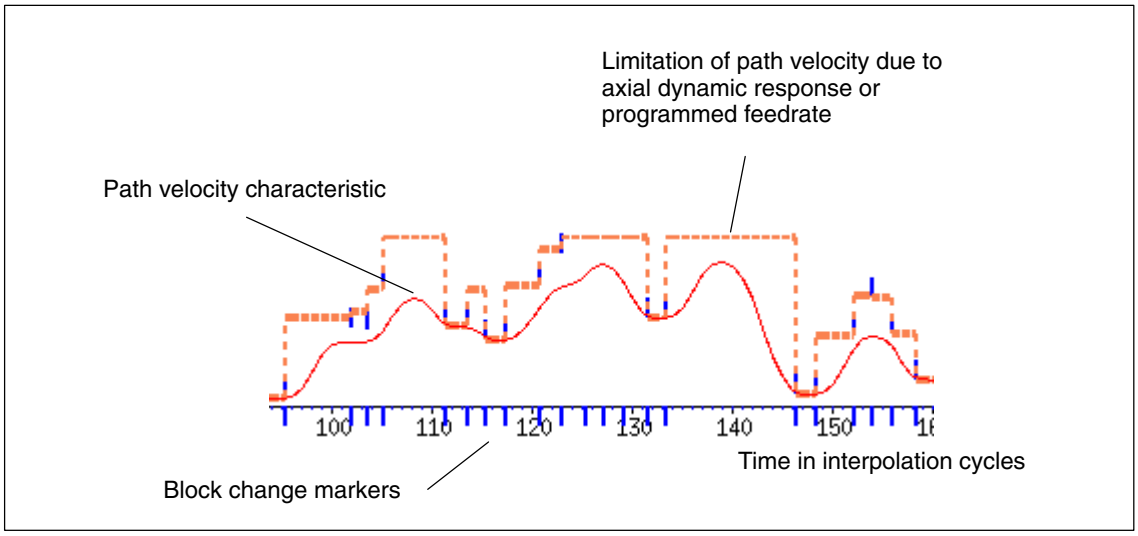


Fig. 2-14 Without dynamic path response adaptation or path smoothing

© Siemens AG, 2003. All rights reserved

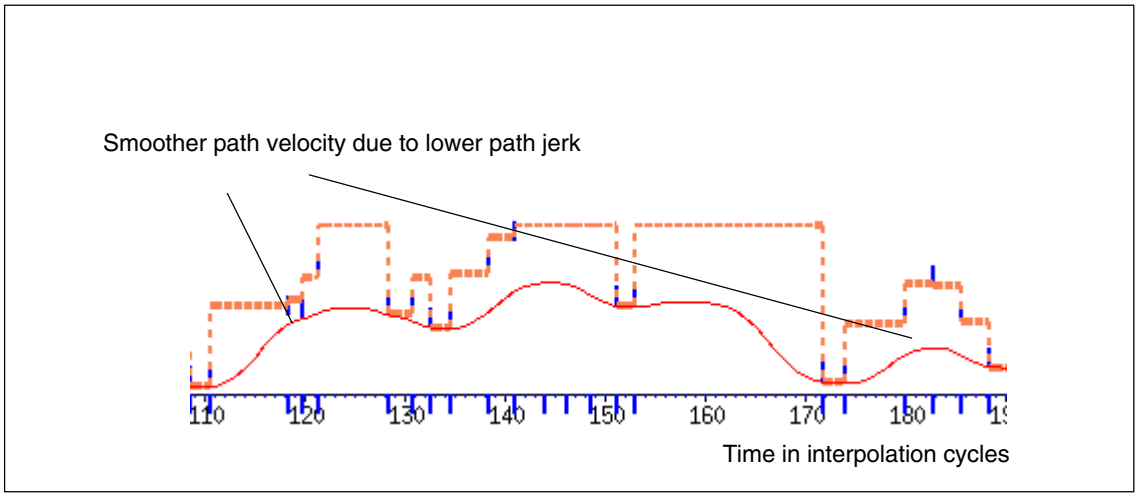


Fig. 2-15 With dynamic path response adaptation, without path smoothing

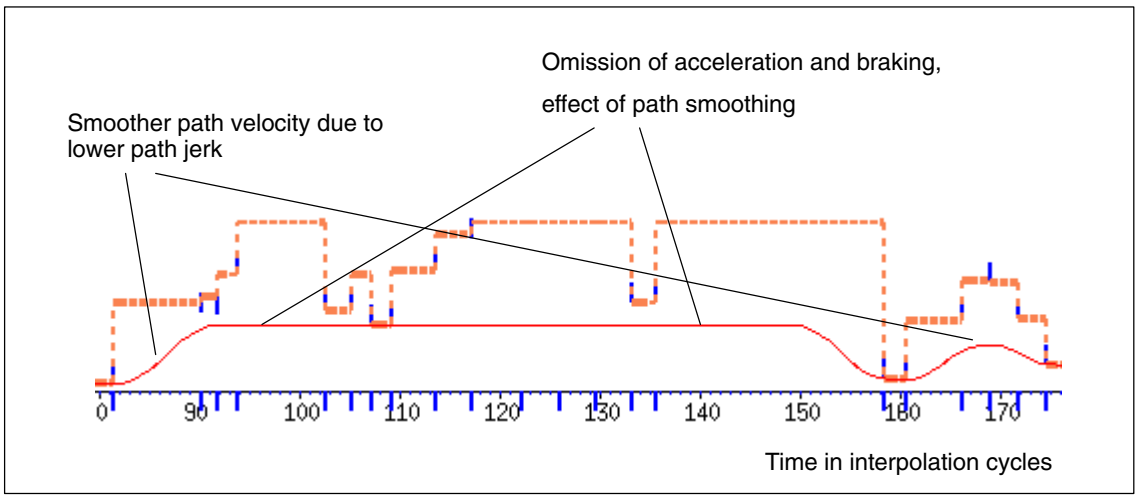


Fig. 2-16 With dynamic path response adaptation and path smoothing

The path velocity characteristic illustrated in Fig. 2-14 has been obtained through deselection of dynamic path response adaptation and path smoothing. This corresponds to the following parameter settings:
$MC_ADAPT_PATH_DYNAMIC[1] = 1
$MC_LOOKAH_SMOOTH_FACTOR = 0%

The path velocity characteristic illustrated in Fig. 2-15 has been obtained through selection of dynamic path response adaptation with minimum, and thus virtually inactive, path smoothing. The relevant parameters were set as follows:
$MC_ADAPT_PATH_DYNAMIC[1] = 4
$MC_LOOKAH_SMOOTH_FACTOR = 1%

© Siemens AG, 2003. All rights reserved

The path velocity characteristic illustrated in Fig. 2-16 has been obtained through selection of dynamic path response adaptation and path smoothing. For this purpose, path smoothing parameterization was selected that results per default when path smoothing is deselected and dynamic path response adaptation activated:
$MC_ADAPT_PATH_DYNAMIC[1] = 4
$MC_LOOKAH_SMOOTH_FACTOR = 0%
(equivalent to $MC_LOOKAH_SMOOTH_FACTOR = 100%)

The last two examples show why the dynamic path adaptation function should be used, whenever possible, in conjunction with path smoothing in continuous-path mode. Using a combination of the two functions means that only those acceleration or braking processes on the path that are not eliminated by path smoothing need to be optimized for the relevant machine.

Startup

The basic prerequisites for starting up the dynamic path response adaptation function are as follows:

- Measurement of axes involved with the startup functions, calculation of the lowest natural frequency of these axes
- Entry of this natural frequency in MD 32440: LOOKAH_FREQUENCY
- Calculation of the dynamic response values.

Calculation of the dynamic response values

The procedure for using the jerk limitation option (SOFT) is described below, since this is a frequent application for the dynamic path adaptation function. The procedure described can, however, be applied equally to BRISK.

1. MD 20465: ADAPT_PATH_DYNAMIC[1] to default 1.
2. Now examine the positioning behavior of the individual axes from different velocities. The jerk for each axis must be set such that the desired positioning tolerance is maintained.
   It will be easier to maintain the tolerance with a higher jerk on positioning processes from higher velocities than when the axes are positioned from low velocities.
3. Enter the maximum permissible jerk for the least critical positioning velocity in MD32431: MAX_AX_JERK.
4. Now calculate the factor highest jerk / lowest jerk (for most critical velocity) for all axes and enter the maximum of all axes in MD 20465: ADAPT_PATH_DYNAMIC[1].

© Siemens AG, 2003. All rights reserved

2.4 Look Ahead

Look Ahead is a procedure in continuous-path mode (G64, G641) that achieves velocity control with Look Ahead over several NC parts program blocks beyond the current block. If the program blocks only contain very small paths, a velocity per block is achieved that permits deceleration of the axes at the block end point without violating acceleration limits. This means that the programmed velocity was not actually reached although a sufficient number of prepared blocks with virtually tangential path transitions was available. With the Look Ahead function it is possible to plan the acceleration and deceleration phase with approximately tangential path transitions in order to achieve a higher feedrate with shorter distances. Deceleration to velocity limits is possible with Look Ahead such that violation of the acceleration and velocity limit is prevented.

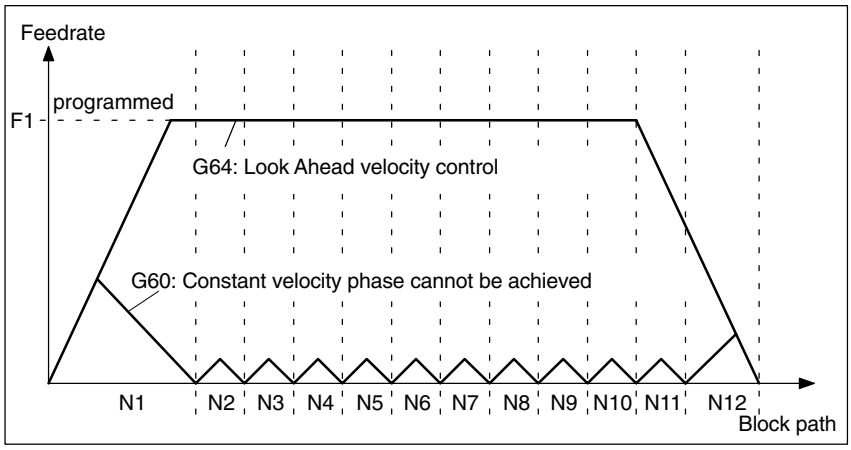


Fig. 2-17 Comparison of velocity response with exact stop G60 and continuous-path mode G64 with Look Ahead with short traversing paths.

Look Ahead takes plannable velocity limits into consideration such as:

- Exact stop at block end
- Velocity limit in the block
- Acceleration limit in the block
- Velocity limit on block transition
- Synchronization with block change at block transition.

© Siemens AG, 2003. All rights reserved

Scope

Look Ahead functionality is available only for path axes, but not for spindles or positioning axes.
Look Ahead performs block related analysis of the plannable velocity limits and determines the required braking ramp profiles on this basis. The LookAhead function is automatically adapted to block length, braking capacity and permissible path velocity.
For safety reasons, the velocity at the end of the last prepared block must initially be assumed to be zero because the next block might be very small or be an exact stop block and the axes must have been stopped by the end of the block. With a series of blocks with high set velocity and very short paths, the speed can be increased in each block depending on the velocity value currently calculated by the Look Ahead function in order to achieve the required set velocity. After this it can be reduced so that the velocity at the end of the last block considered by the Look Ahead function can be zero. This results in a serrated velocity profile (see the following Fig.) which can be avoided by reducing the set velocity or increasing the number of blocks considered by the Look Ahead function.

Number of blocks

To achieve reliable axis traversal in continuous-path mode, the feedrate must be adapted over several blocks. The number of blocks considered by the Look Ahead function is calculated automatically and can optionally be limited by a machine data. The standard setting is 1 which means that LookAhead only considers the following block for velocity control.

Because Look Ahead is especially important for short blocks (relative to the deceleration path), the number of blocks required is of interest for Look Ahead braking (see Fig. below). It is enough to consider the path length to be equal to the deceleration path that is required to brake from maximum velocity to rest. For a machine with a low axial acceleration of a = 1m/s^2 and a high feedrate of v_{path} = 10 m/min, the following number of blocks is obtained with an attainable block cycle time of the control of TB = 10 ms.

$n_{Look\ Ahead}$ = Deceleration path/Block length = ($v_{path}^2/(2a)$) / (v_{path} * TB) = 9.

Considering these aspects, it is advisable to adapt the feedrate over 10 blocks. The specified number of blocks for the Look Ahead function does not affect the Look Ahead algorithm and the memory requirement.
Because the machining speed is often less than the maximum speed in a program, more blocks would be considered than is necessary, which uses unnecessary additional computer power. For this reason, the required number of blocks is derived from the velocity which is calculated from the programmed velocity multiplied by the value in MD 12100: OVR_FACTOR_LIMIT_BIN (limitation with binary-coded override switch) or by the 31st override value in MD 12030: OVR_FACTOR_FEEDRATE (weighting of the path feedrate override switch). The 31st override value must have the value of the highest override factor used.
The number of blocks considered by the Look Ahead function is limited by the maximum number of NC blocks in the IPO buffer.

© Siemens AG, 2003. All rights reserved

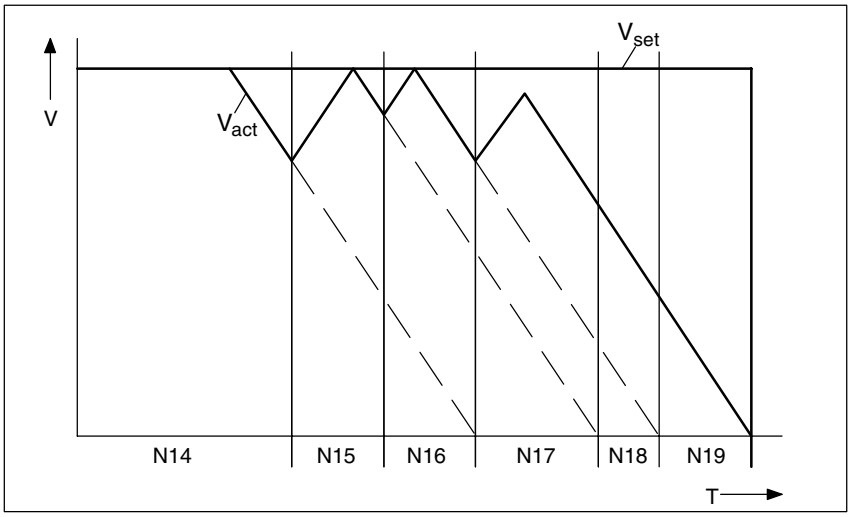


Fig. 2-18 Example of velocity control over several blocks. Number of Look Ahead blocks= 2

Velocity profiles

In addition to the fixed, plannable velocity limitations, Look Ahead can also take account of the programmed velocity. This makes it possible to achieve a lower velocity by applying Look Ahead beyond the current block.

Following block velocity

One possible velocity profile contains the determination of the following block velocity. Using information from the current and the following NC block, a velocity profile is calculated from which, in turn, the required velocity reduction for the current override is derived. The calculated maximum value of the velocity profile is limited by the maximum path velocity. With this function it is possible to initiate a speed reduction in the current block taking override into account such that the lower velocity of the following block can be achieved. If the reduction in velocity takes longer than the traverse time of the current block, the velocity is further reduced in the following block. Velocity control is only ever considered for the following block. MD 20400: LOOKAH_USE_VELO_NEXT_BLOCK (LookAhead for following block velocity).

© Siemens AG, 2003. All rights reserved

Override points

If the velocity profile of the following block velocity is not sufficient because, for example, very high override values, e.g. 200%, or constant cutting rate G96 is used with the result that the velocity must be further reduced in the following block, Look Ahead provides a way of reducing the programmed velocity over several NC blocks. By defining override points, Look Ahead then calculates a limiting velocity profile for each value. The required velocity reductions for the current override are derived from these profiles. The calculated maximum value of the velocity profile is limited by the maximum path velocity. The upper point should cover the velocity range that will be reached by the maximum value set in MD 12030: OVR_FACTOR_FEEDRATE (weighting of the path feed override switch) or by the value set in MD 12100: OVR_FACTOR_LIMIT_BIN (limitation with binary-coded override switch). In this way, a reduction of the velocity continuing into the block in which it is programmed can be avoided. If velocity reductions across block boundaries are required already at 100% override, a point must be set in the lower override range as well. The number of override points used per channel is specified in MD 20430:
LOOKAH_NUM_OVR_POINTS (number of override switch points for Look Ahead). The associated points are stored in MD 20440:
LOOKAH_OVR_POINTS (override switch points for Look Ahead).
A combination of both procedures for calculating the velocity profile is possible and generally advisable because even with the default machine data for this function, the greater part of the range of overridedependent velocity restrictions is covered.
Plannable velocity restrictions limit the overridedependent velocity restrictions. If neither of these procedures is activated, the set velocity is only ever approached in the current blocks.

© Siemens AG, 2003. All rights reserved

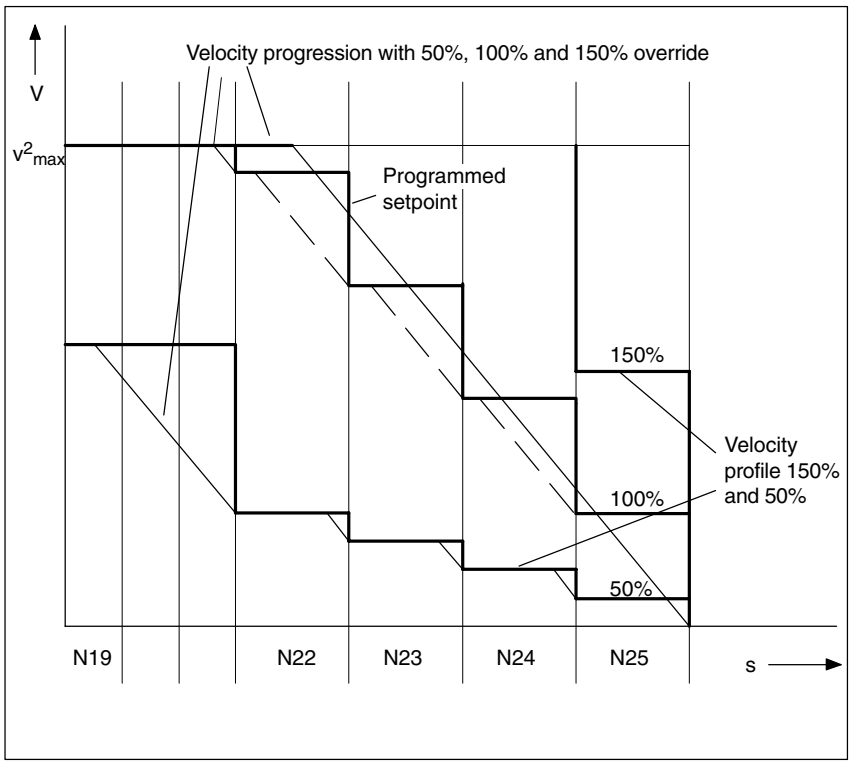


Fig. 2-19 Example of restricted velocity progressions with Look Ahead block selection = 4 and the following settings:
MD 20430: LOOKAH_NUM_OVR_POINTS = 2
MD 20440: LOOKAH_OVR_POINTS = 1.5, 0.5
MD 20400: LOOKAH_USE_VELO_NEXT_BLOCK = 1

Block cycle problem

Block cycle programs are encountered in cases where the traversing distances of the NC blocks to be processed are so short that the Look Ahead function has to reduce the machine velocity to provide enough time for block preparation. In this situation, the path motion can be subject to continual acceleration and deceleration.
The MD 20450: LOOKAH_RELIEVE_BLOCK_CYCLE (relief factor for the block cycle time) can be set to smooth such velocity fluctuations.

Selection and deselection of Look Ahead

The Look Ahead function is selected and deselected with continuous-path mode (G64 and G641 respectively).

© Siemens AG, 2003. All rights reserved

2.5 NC block compressor COMPON, COMPCURV, -CAD

COMPON, COMPCURV

The modal G code COMPON or COMPCURV can be used to activate an “NC block compressor”. This function collects a series of linear blocks during linear interpolation (the number is limited to 10) and approximates them within a tolerance specified in machine data via a 3rd-degree (COMPON) or 5th-degree (COMPCURV) polynomial. One traversing block is processed by the NC instead of a large number of small blocks.

COMPCAD

The COMPCAD G code can be used to select a further compression which optimizes the **surface quality and velocity**. The interpolation accuracy can again be specified in machine data. COMPCAD is processor and memory-intensive. It should only be used if surface quality enhancement measures cannot be incorporated in the CAD/CAM program.

The programming is described in
References:
Programming: /PGA/ Programming Guide, Advanced, Chapter 5.
Compressor for orientation transformation:
/FB3/ F2: “3 to 5-axis transformation”

The following three machine data are available for the compressor function:

- MD 20170 COMPRESS_BLOCK_PATH_LIMIT
  This MD specifies the maximum path length for block compression. Longer blocks are not compressed.
- MD 33100 COMPRESS_POS_TOL
  A tolerance can be specified for each axis. This value specifies the maximum deviation of the generated spline curve from the programmed end points. The higher the values, the more blocks can be compressed.
- MD 20172 COMPRESS_VELO_TOL
  The maximum deviation of the path feed while the compressor is active in conjunction with FLIN and FCUB can be specified here (does **not** apply to the COMPCAD command).

Recommendation for MD settings

The following machine data affect the compressor function and should contain the following values (specified in mm):

MD 18360 MM_EXT_PROG_BUFFER_SIZE = 100
MD 28520 MM_MAX_AXISPOLY_PER_BLOCK = 3
MD 28530 MM_PATH_VELO_SEGMENTS=5
MD 28540 MM_ARCLENGTH_SEGMENTS = 10
MD 28070 MM_NUM_BLOCKS_IN_PREP=60
MD 28060 MM_IPO_BUFFER_SIZE=100
SD 42470 CRIT_SPLINE_ANGLE = 36
MD 20170 COMPRESS_BLOCK_PATH_LIMIT = 20
MD 20172 COMPRESS_VELO_TOL = 100

© Siemens AG, 2003. All rights reserved

SW 5.2 and lower
MD 32310 MAX_ACCEL_OVL_FACTOR[AX1]=1.01
MD 32310 MAX_ACCEL_OVL_FACTOR[AX2]=1.01
MD 32310 MAX_ACCEL_OVL_FACTOR[AX3]=1.01

SW 5.3 and higher
MD 32310 MAX_ACCEL_OVL_FACTOR[AX1] = <value for G64 mode>
MD 32310 MAX_ACCEL_OVL_FACTOR[AX2] = <value for G64 mode>
MD 32310 MAX_ACCEL_OVL_FACTOR[AX3] = <value for G64 mode>
MD 20490 IGNORE_OVL_FACTOR_FOR_ADIS = 1

A tolerance can be specified for each axis: This value specifies the maximum deviation of the generated spline curve from the programmed end points. The higher the values, the more blocks can be compressed.
MD 33100 COMPRESS_POS_TOL[AX1] = 0.01
MD 33100 COMPRESS_POS_TOL[AX2] = 0.01
MD 33100 COMPRESS_POS_TOL[AX3] = 0.01

Experience has shown that a value of 0.01 is suitable for most applications. If necessary, the value can be increased, e.g. to 0.02:
MD 33100 COMPRESS_POS_TOL[AX1] = 0.02
MD 33100 COMPRESS_POS_TOL[AX2] = 0.02
MD 33100 COMPRESS_POS_TOL[AX3] = 0.02
NEWCONF

Activating the MD values

The new values are activated after the NEWCONF command.

The corner rounding function G642 and jerk limitation SOFT can be used to achieve further improvements in surface quality. These commands must be entered at the start of the program.

```
COMPCAD SOFT G642
```

COMPOF terminates the compressor function.

Any motion block with the following simple syntax is compressed:

```
N... G1 X... Y... Z... F...
```

where "..." is a number and X, Y, Z are axis names.
Blocks with extended addresses, e.g. C=100 or A=AC(100), and blocks with auxiliary functions are not compressed.

NC SW 6.3 and higher:
Motion blocks with extended syntax are not also compressed.

■

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

3.1 Smoothing of the path velocity (SW 5.3 and higher)

Several blocks with SOFT and BRISK

Smoothing of the path velocity is only effective in continuouspath mode with Look Ahead over several blocks with SOFT and BRISK, but not with G0.

The cycle times of the control must be parameterized such that the preprocessing is provided with enough blocks to be able to analyze an acceleration process.

■

4 Data Descriptions (MD, SD)

4.1 General machine data

10110 MD number	**PLC_CYCLE_TIME_AVERAGE** Average PLC acknowledgment time		
Default setting: 0.1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Time information for the CNC about OB1 cycle time in which it is guaranteed that the auxiliary functions can be acknowledged. In continuous-path mode with auxiliary function output, the time set in the MD is used to control the path feed during motion so that the minimum traversing time corresponds to the time specified. This enables a smooth velocity progression which is not affected by waiting for the PLC acknowledgment. The internal time basis is the IPO cycle. From P5.1, the MD is no longer evaluated. Using this MD, it is possible to delay status transitions: “Channel operating / Channel RESET/Channel failed —> Channel interrupted” on RESET for the PLC. When “Channel interrupted” is output, the NCK waits for at least the time set in the MD + one IPO cycle.		

© Siemens AG, 2003. All rights reserved

4.2 Channel-specific machine data

20400 MD number	**LOOKAH_USE_VELO_NEXT_BLOCK** Look Ahead for programmed following block velocity		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Look Ahead is active for the programmed following block velocity. A velocity profile is automatically generated for the required velocity reduction of the current override.		

20430 MD number	**LOOKAH_NUM_OVR_POINTS** Number of override switch points for Look Ahead		
Default setting: 1, 1, 1, 1, 1, ...	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	This data defines the number of calculated override points for the overridedependent velocity restrictions per channel to be considered by Look Ahead. LOOKAH_NUM_OVR_POINTS = 0: No velocity profile based on the override points stored in MD 20440: LOOAH_OVR_POINTS (override switch points for Look Ahead) is calculated. LOOKAH_NUM_OVR_POINTS = 1: A velocity profile base on the first override point in MD: LOOHAH_OVR_POINTS (override switch points for Look Ahead) is calculated. LOOKAH_NUM_OVR_POINTS = 2: Two velocity profiles based on the two override points in MD: LOOHAH_OVR_POINTS (override switch points for Look Ahead) is calculated.		
Related to	MD 20440: LOOKAH _OVR_POINTS (override switch points for Look Ahead)		

20440 MD number	**LOOKAH_OVR_POINTS** Override switch points for Look Ahead	
Default setting: {1.0, 0.2},{1.0, 0.2}, ... {1.0, 0.2}	Minimum input limit: 0.2	Maximum input limit: 2.0
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: DOUBLE	Applies from SW 1.1	
Significance:	For defining the override points for the override-dependent velocity restrictions per channel to be considered by Look Ahead. Look Ahead calculates a limiting velocity profile for each override point enabled by MD: LOOKAH_NUM_OVR_POINTS (number of override switch points for Look Ahead). The velocity reductions required for the current override are derived using these profiles. The points must be entered for each channel in descending order, e.g. 2.0, 1.0 (2.0) correspond to 200%, 1.0 corresponds to 100% override). The upper override point should be close to the upper value in MD: OVR_FACTOR_FEEDRATE (weighting of path feedrate override switch) or in MD: OVR_FACTOR_LIMIT_BIN (limitation with binary-coded override switch). The override points should match the override range valid for the machine and control.	
Related to	MD 20430: LOOKAH_NUM_OVR_POINTS (number of override switch points for Look Ahead)	

© Siemens AG, 2003. All rights reserved

20450 MD number	**LOOKAH_RELIEVE_BLOCK_CYCLE** Relief factor for the block cycle time		
Default setting: 0.0	Minimum input limit: ***	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DOUBLE	Applies from SW 2		
Significance:	Block cycle problems occur for the following reason: The traversing length of the NC blocks to be processed is so short that the Look Ahead function must reduce the machine velocity to provide enough time for block preparation. In this situation, the path motion can be subject to continual acceleration and deceleration. With this machine data you define how much such velocity fluctuations are to be smoothed.		
Special cases, errors, ...	Values up to approx. 1.0 are useful. The value 0.0 means that the function is deactivated.		

20460 MD number	**LOOKAH_SMOOTH_FACTOR** Smoothing factor for Look Ahead	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 500
Changes effective after NEW_CONF	Protection level: 2/7	Unit: %
Data type: REAL	Applies from SW 5.3	
Significance:	A smoothing factor can be specified in the interest of a smooth path velocity control. This factor determines the maximum permissible loss of productivity. Acceleration processes contributing to a shorter programming time less than this factor are not carried out. Only those acceleration processes whose frequencies lie above the frequency parameterized in MD 32440: MA_LOOKAH_FREQUENCY are considered. With the default value 0, this functionality of the smoothing is disabled.	
Related to	MD 32440: LOOKAH_FREQUENCY MD 32431: MAX_AX_JERK MD 32300: MAX_AX_ACCEL MD 20600: MAX_PATH_JERK	

20462 MD number	**LOOKAH_SMOOTH_WITH_FEED** Taking into account the programmed feedrate for smoothing the tool path velocity	
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: –
Data type: BOOLEAN	Applies from SW 6	
Significance:	The MD defines whether the smoothing of the tool path velocity will also take into account the programmed feedrate. In these cases, the given factor resulting from MD 20460: LOOKAH_SMOOTH_FACTOR can be observed better as if the override is set to 100%.	
Related to	MD 32440: LOOKAH_FREQUENCY MD 20460: LOOKAH_SMOOTH_FACTOR	

© Siemens AG, 2003. All rights reserved

20465 MD number	**ADAPT_PATH_DYNAMIC** Adaptation of dynamic path response		
Default setting: 1.0, 1.0	Minimum input limit: 1.0	Maximum input limit: 100.0	
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: –	
Data type: DOUBLE	Applies from SW 6.2		
Significance:	This adaptation factor can reduce the dynamic response of changes to the path velocity. ADAPT_PATH_DYNAMIC[0] is operative for Brisk and reduces the permissible acceleration rate. ADAPT_PATH_DYNAMIC[1] is operative for Soft and reduces the permissible jerk. Only those acceleration processes whose frequencies lie above the frequency parameterized in MD 32440: LOOKAH_FREQUENCY are considered. The function is deactivated by entering 1.0.		
Related to	MD 32440: LOOKAH_FREQUENCY, MD 20460: LOOKAH_SMOOTH_FACTOR. MD 32431: MAX_AX_JERK, MD 32300: MAX_AX_ACCELL; MD 20600: MAX_PATH_JERK		

20480 MD number	**SMOOTHING_MODE** Rounding behavior with G642, G643, G644	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3333
Changes effective after NEW_CONF	Protection level: 7 / 7	Unit: –
Data type: BYTE	Applies from SW 6	

© Siemens AG, 2003. All rights reserved

20480 MD number	**SMOOTHING_MODE** Rounding behavior with G642, G643, G644
Significance:	**G642** (setting in the **tens place** of MD 20480) **0:** G642 uses axis-specific tolerances. These are set using the axis-specific MD 33100: COMPRESS_POS_TOL are set. (default). Response as in SW 4.3 **1:** When rounding with G642, the contour tolerance SD 42465: SMOOTH_CONTUR_TOL is used for the geometry axes. The remaining axes are rounded using the axis-specific tolerances in MD 33100: COMPRESS_POS_TOL. **2:** The orientation movement with G642 is rounded using the angle tolerance SD 42466: SMOOTH_ORI_TOL. All the other axes use the axis-specific tolerances MD 33100: COMPRESS_POS_TOL. **3:** Combination of the two options 1x and 2x. i.e. the following tolerances for G642 are used: SD 42465: SMOOTH_CONTUR_TOL and SD 42466: SMOOTH_ORI_TOL. Further axes are rounded using the axis-specific tolerance. **Function G642:** With G642, the rounding travel is determined based on the **shortest** rounding travel of all axes. This value is taken into account when generating a **rounding block**. With G642, the rounding area results from the smallest tolerance setting. **G643** (setting in the **units place** of MD 20480) **0:** G643 uses axis-specific tolerances. These are set using the axis-specific MD 33100: COMPRESS_POS_TOL (default). Response as in SW 4.3. **1:** When rounding with G643, the contour tolerance SD 42465: SMOOTH_CONTUR_TOL is used for the geometry axes. The remaining axes are rounded using the axis-specific tolerances in MD 33100: COMPRESS_POS_TOL. **2:** The orientation movement with G643 is rounded using the angle tolerance SD 42466: SMOOTH_ORI_TOL. All the other axes use the axis-specific tolerances MD 33100: COMPRESS_POS_TOL. **3:** Combination of the two options 1x and 2x. i.e. the following tolerances for G643 are used: SD 42465: SMOOTH_CONTUR_TOL and SD 42466: SMOOTH_ORI_TOL. Further axes are rounded using the axis-specific tolerance. **Function G643:** In the case of G643, the rounding travel of each axis can be **different**. The rounding travels are taken into account axis-specifically and **block-internally**. Very different specifications for the contour tolerance and the tolerance of the tool orientation can only have effect with G643. The values of the units, tens, hundreds and thousands places are added up.
Special cases, errors, ...	
Related to	SD 42465: SMOOTH_CONTUR_TOL SD 42466: SMOOTH_ORI_TOL MD 33100: COMPRESS_POS_TOL

© Siemens AG, 2003. All rights reserved

20550 MD number	**EXACT_POS_MODE** Exact stop conditions with G0 and G1, ...(1st G code group)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 33	
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 6.1		
Significance:	Configuring the exact stop conditions with G00 and other G codes of the 1st G code group. The MD is coded decimally. The **unit's** places define the response with **G00** (infeed motion), and the **ten's** places define the response with the **remaining G codes** of the 1st group ("machining G codes"). Unit's place: 0: With G00, the programmed exact stop conditions become active in each case. 1: With G00, irrespective of the programmed exact stop G601(positioning window fine) becomes active. 2: With G00, irrespective of the programmed exact stop G602(positioning window coarse) becomes active. 3: With G00, irrespective of the programmed exact stop condition, G603 (setpoint reached) becomes active. Ten's place: 0: With the machining G codes, the programmed exact stop conditions become active in each case. 1: With the machining G codes, irrespective of the programmed exact stop condition, G601 (positioning window fine) becomes active. 2: With the machining G codes, irrespective of the programmed exact stop condition, G602 (positioning window coarse) becomes active. 3: With the machining G codes, irrespective of the programmed exact stop condition, G603 (setpoint reached) becomes active. The values of the unit's and ten's places are added true to the digits.		
Application example(s)	For example, the value of EXACT_POS_MODE = 2 means that with G00 the exact stop condition G602 always becomes automatically active, irrespective of which exact stop condition has been programmed. With the remaining G codes of the 1st group, however, the programmed exact stop condition comes into effect.		
References	Programming Guide, Fundamentals		

© Siemens AG, 2003. All rights reserved

20602 MD number	**CURV_EFFECT_ON_PATH_ACCEL** Influence of path curvature on dynamic path response		
Default setting: 0	Minimum input limit: 0	Maximum input limit: .95	
Changes effective after NEW_CONF	Protection level: USER	Unit: –	
Data type: DOUBLE	Applies from SW 5		
Significance:	This variable makes it possible to allow for the effect of the path curvature on the path acceleration and the path velocity. 0: No consideration > 0: If necessary, the path velocity and path acceleration are reduced, in order to provide sufficient reserve on the machine axes for centripetal acceleration. 0.75: Recommended setting. CURV_EFFECT_ON_PATH_ACCEL specifies the proportion of axis accelerations (see MAX_AX_ACCEL[..]) which can be used for the centripetal acceleration. The remainder is used to modify the path velocity.		
Special cases, errors, ...	For linear blocks, no centripetal acceleration is required so that the full axis acceleration of the path acceleration is available. CURV_EFFECT_ON_PATH_ACCEL has little or no effect on slightly curved contours or if the maximum path feed is sufficiently small. Consequently, the path acceleration is higher than that defined by (1 – CURV_EFFECT_ON_PATH_ACCEL) * MAX_AX_ACCEL[..].		
Related to	MD 32431: MAX_AX_ACCEL[..]		

20603 MD number	**CURV_EFFECT_ON_PATH_JERK** Influence of path curvature on path jerk		
Default setting: 0	Minimum input limit: 0	Maximum input limit: .95	
Changes effective after NEW_CONF	Protection level: USER	Unit: –	
Data type: DOUBLE	Applies from SW 5		
Significance:	This variable makes it possible to allow for the effect of the path curvature on the path acceleration and the path jerk. The value of CURV_EFFECT_ON_PATH_JERK is only relevant if CURV_EFFECT_ON_PATH_ACCEL > 0 and the jerklimited velocity control (SOFT, $MC_GCODE_RESET_VALUES[20] = 2) is active. 0: No consideration > 0: If necessary, the path acceleration and path jerk are reduced, in order to provide sufficient reserve on the machine axes for the change in path velocity on curved contours. 0.5: Recommended setting. This machine data specifies the proportion of axis jerk (see also MAX_AX_JERK[..]) used to apply the path acceleration for deceleration or acceleration on a curved contour. The remainder of (1 – CURV_EFFECT_ON_PATH_JERK) is available for the change in path acceleration, as well as the jerk proportion required in order to reach the velocity alone. The jerk proportion available for moving through changes in curvature is set in CURV_EFFECT_ON_PATH_ACCEL. If the value of this machine data is increased, it is possible to traverse through points with a higher rate of curvature change at a higher path velocity (determine the optimum!)		
Special cases, errors, ...	CURV_EFFECT_ON_PATH_JERK is not relevant for linear blocks, since the total axis jerk can be used to change the path acceleration. $MC_CURV_EFFECT_ON_PATH_JERK does not have the full effect on slightly curved contours or if the maximum path feed is sufficiently small and the changes in path velocity are small. Consequently, the deceleration ramps are shorter.		
Related to	MD 32431: MAX_AX_JERK[..]		

© Siemens AG, 2003. All rights reserved

4.3 Axis-specific machine data

32310 MD number	**MAX_ACCEL_OVL_FACTOR** Overload factor for axial velocity jumps		
Default setting: 1.2	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 3/3	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The overload factor restricts step changes in the machine axis velocity at block ends. The value entered refers to the value of MD: MAX_ AX_ ACCEL (axis acceleration) and states by how much the maximum acceleration for one IPO cycle can be exceeded.		
Related to	MD 32300: MAX_AX_ACCEL(axis acceleration) MD 10070: IPO_SYSCLOCK_TIME_RATIO (interpolator clock)		

32431 MD number	**MAX_AX_JERK** Max. axis jerk for path motion		
Default setting: 1000000.0	Minimum input limit: 0	Maximum input limit: –	
Changes effective after NEW_CONF	Protection level: USER	Unit: m/s^3	
Data type: DOUBLE	Applies from SW 5		
Significance:	Jerk limitation for path motion		

32432 MD number	**PATH_TRANS_JERK_LIM** Max. axial jerk of a geometry axis at block boundary		
Default setting: 1 000000.0	Minimum input limit: >0.0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 3/3	Unit: m/s^3, deg/s^3	
Data type: DOUBLE	Applies from SW 3.2		
Significance:	The control limits the jerk (acceleration jump) at a block transition between contour sections of different curvature to the value set.		
MD irrelevant for ...	Exact stop		
Application example(s)	See Chapter 6		
Related to	Continuous-path mode, SOFT type of acceleration		

32433 MD number	**SOFT_ACCEL_FACTOR** Scaling the acceleration limitation for SOFT		
Default setting: 1.0	Minimum input limit: plus	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 3 / 3	Unit: –	
Data type: DOUBLE	Applies from SW 5		
Significance:	Scaling the acceleration limitation for SOFT Acceleration limitation = SOFT_ACCEL_FACTOR * MA_AX_ACCEL		
Related to	MD 32300: MA_AX_ACCEL		

© Siemens AG, 2003. All rights reserved

32434 MD number	**G00_ACCEL_FACTOR** Scaling the acceleration limitation for G00		
Default setting: 1.0	Minimum input limit: plus	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 3 / 3	Unit: –	
Data type: DOUBLE	Applies from SW 5		
Significance:	Scaling the acceleration limitation for G00 Acceleration limitation for G00 = G00_ACCEL_FACTOR * MA_AX_ACCEL		
Related to	MD 32300: MA_AX_ACCEL		

32435 MD number	**G00_JERK_FACTOR** Scaling the axial jerk limitation for G00		
Default setting: 1.0	Minimum input limit: plus	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 3 / 3	Unit: –	
Data type: DOUBLE	Applies from SW 5		
Significance:	Scaling the axial jerk limitation for G00 Jerk limitation for G00 = G00_JERK_FACTOR * MA_AX_JERK limits the jerk overshoot for G00 = G00_JERK_FACTOR * PATH_TRANS_JERK_LIM		
Related to	MD 32300: MA_AX_JERK MD 32432: PATH_TRANS_JERK_LIM		

32440 MD number	**MA_LOOKAH_FREQUENCY** Smoothing limit frequency for Look Ahead		
Default setting: 10	Minimum input limit: 0	Maximum input limit: –	
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: Hz	
Data type: REAL	Applies from SW 5.3		
Significance:	Acceleration processes in continuous-path mode with Look Ahead, which run at a frequency higher than parameterized in this machine data are either smoothed or reduced in their dynamic response, depending on the parameterization in $MC_LOOKAH_SMOOTH_FACTOR and $MC_ADAPT_PATH_DYNAMIC. ADAPT_PATH_DYNAMIC (see Dynamic response adaptation) A minimum of all axes involved in the path is always determined. If vibration is excited in the mechanical system of these axes and if the corresponding frequency is known, then the frequency of this machine data should be set to a value smaller than this frequency.		
Related to	MD 20460: LOOKAH_SMOOTH_FACTOR		

© Siemens AG, 2003. All rights reserved

36000 MD number	**STOP_LIMIT_COARSE** Exact stop coarse		
Default setting: 0.04	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	An NC block has been completed when the distance of the actual position of the path axes from the setpoint entered for the exact stop limit. If the actual position of a path axis is not within these limits, the NC block is not considered to be completed and continued parts program processing is not possible. Transition to the next block can be influenced by the value entered. The larger the value, the earlier block change can be initiated. If the defined exact stop limit is not reached, – the block is not considered completed, – it is not possible to continue traversing the axis, – alarm 25080 positioning monitoring is output after the time in MD: POSITIONING_TIME (monitoring time exact stop fine) has elapsed, – the direction of movement +/– is displayed for the axis in the positioning display. The exact stop window is also evaluated for spindles in position control mode (SPCON instruction).		
Special cases, errors, ...	MD: STOP_LIMIT_COARSE must not be smaller than MD: STOP_LIMIT_FINE (exact stop fine). In order to achieve the same block change behavior as for exact stop, the exact stop coarse window can be the same as that for exact stop fine. The MD: STOP_LIMIT_COARSE must not be the same or greater than MD: STAND-STIL_POS_TOL (zero speed tolerance).		
Related to	MD 36020: POSITIONING_TIME (delay time exact stop fine)		

36010 MD number	**STOP_LIMIT_FINE** Exact stop fine		
Default setting: 0.01	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	See MD: STOP_LIMIT_COARSE (Genauhalt grob)		
Special cases, errors, ...	MD: STOP_LIMT_FINE must not be greater than that set in MD: STOP_LIMIT_COARSE (exact stop coarse). The MD: STOP_LIMIT_FINE must not be the same or greater than MD: STAND-STILL_POS_TOL (zero speed tolerance).		
Related to	MD 36020: POSITIONING_TIME (delay time exact stop fine)		

36012 MD number	**STOP_LIMIT_FACTOR[n]** Factor for exact stop coarse/fine and zero speed control		
Default setting: 1.0	Minimum input limit: 0.001	Maximum input limit: 1000.0	
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: –	
Data type: DOUBLE	Applies from SW 5.1		
Significance:	You can use this factor to reevaluate MD 36000: STOP_LIMIT_COARSE, MD 36010: STOP_LIMIT_FINE, MD 36030: STANDSTILL_POS_TOL depending on the parameter set. The ratio between the three values is always the same.		
Application example(s)	Modification of the positioning response when mass ratios change significantly as a result of a gear change or in order to save positioning time at the expense of precision in various operating states.		
Related to	MD 36000: STOP_LIMIT_COARSE, MD 36010: STOP_LIMIT_FINE, MD 36030: STANDSTILL_POS_TOL		

© Siemens AG, 2003. All rights reserved

36020 MD number	**POSITIONING_TIME** Delay time exact stop fine		
Default setting: 5.0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	If a block is ended with exact stop, the axis must have reached the exact stop window fine/ coarse within the positioning time. Otherwise the positioning action is aborted with alarm 25080 “Positioning monitoring” and switches to followup mode. The monitoring time is started with interpolator end for the axis. Position-controlled spindles are also subject to this timed positioning monitoring. If an error occurs, alarm 25080 “Positioning monitoring” is again triggered. IS “Mode group ready” (DB11, DBX6.3) is reset with alarm 25080, the path axis triggering the alarm is braked to reset with the braking ramp set in MD: AX_EMERGENCY_STOP_TIME (duration of the braking ramp for error states) and then switched to follow-up mode. The other path axes in the mode group are brought to reset with setpoint = 0 from the IPO. A traversing positioning axis belonging to the mode group is decelerated according to its acceleration ramp set in MD: MAX_AX_ACCEL (axis acceleration).		
Related to	MD 36000: STOP_LIMIT_COARSE (exact stop coarse) MD 36010: STOP_LIMIT_FINE (exact stop fine)		

© Siemens AG, 2003. All rights reserved

4.4 Channel-specific setting data

42465 SD number	**SMOOTH_CONTUR_TOL** Maximum contour deviation on rounding		
Default setting: 0.05	Minimum input limit: 0.000001	Maximum input limit: 999999.0	
Changes effective immediately	Protection level: 7 / 7	Unit: mm	
Data type: DOUBLE	Applies from SW 6		
Significance:	This setting data is used to define the maximum tolerance for the **contour** when rounding.		
Special cases, errors, ...			
Related to	MD 20480: SMOOTHING_MODE SD 42466: SMOOTH_ORI_TOL is used.		

42466 MD number	**SMOOTH_ORI_TOL is used.** Maximum deviation of the tool orientation on rounding		
Default setting: 0.05	Minimum input limit: 0.000001	Maximum input limit: 90.0	
Changes effective immediately	Protection level: 7 / 7	Unit: degrees	
Data type: DOUBLE	Applies from SW 6		
Significance:	This setting data is used to define the maximum tolerance for the contour when rounding. **tool orientation** when rounding. This data defines the maximum permissible angular displacement of the tool orientation. This data is only effective if an orientation transformation is active.		
Special cases, errors, ...			
Related to	MD 20480: SMOOTHING_MODE SD 42465: SMOOTH_CONTUR_TOL		

■

© Siemens AG, 2003. All rights reserved

Signal Descriptions 5

5.1 Channel-specific signals

DB21, ... **DBX36.3** Data block	**All axes stationary** Signal(s) from channel (NCK –> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	All axes assigned to the channel are stationary with interpolator end. No other traversing movements are active.		

5.2 Axis-specific signals

DB 31, ... **DBX60.6** Data block	**Position reached with exact stop coarse** Signal(s) from axis/spindle (NCK –> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The axis is in the appropriate exact stop and no interpolator is active for the axis and – the control is in Reset mode (Reset key or Program End), – the axis was last programmed as a positioning axis or positioning spindle (initial setting of special axis: Positioning axis). – the path movement was stopped with NC Stop. – the spindle is in position-controlled mode (SPCON/SPOS instruction) and is stationary, – the axis is switched from speed-controlled to position-controlled mode with IS “Position measuring system”.		
Signal state 0 or signal transition 1 ——> 0	The axis is not in the appropriate exact stop or the interpolator is active for the axis or – the path movement was stopped with NC Stop. – the spindle is in speed-controlled mode (SPCOF/SPOSA instruction), – “Follow-up mode” is active for the axis. – “Parking mode” is active for the axis. – the axis is switched from position-controlled to speed-controlled mode with IS “Position measuring system”.		
Signal irrelevant for	For rotary axes that are programmed as rounding axes.		
Related to	MD 36000: STOP_LIMIT_COARSE (exact stop coarse)		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX60.7** Data block	**Position reached with exact stop fine** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	See IS “Position reached with exact stop coarse”.	
Signal state 0 or signal transition 1 ——> 0	See IS “Position reached with exact stop coarse” (DB31, ... DBX60.6)	
Signal irrelevant for	For rotary axes that are programmed as rounding axes.	
Related to	MD 36010: STOP_LIMIT_FINE (exact stop fine)	

■

© Siemens AG, 2003. All rights reserved

6 Example

Example of jerk limitation on the path

```
...
N1000 G64 SOFT          ; Continuous-path mode with SOFT acceleration response
N1004 G0 X-20 Y10       ;
N1005 G1 X-20 Y0        ; Line
N1010 G3 X-10 Y-10 I10  ; Block transition with jump in path curvature (line – circle)
N1011 G3 X0 Y0 J10      ; Block transition with continuous path curvature
N1020 G2 X5 Y5 I5       ; Block transition with jump in path curvature (circle – circle)
N1021 G2 X10 Y0 J-5     ;
...
```

■

7 Data Fields, Lists

7.1 Interface signals

DB number	Bit, byte	Name	Refer-ence
Channel-specific			
21, ...	36.3	All axes stationary	
Axis/spindle-specific			
31, ...	60.6	Position reached with exact stop coarse	
31, ...	60.7	Position reached with exact stop fine	

© Siemens AG, 2003. All rights reserved

7.2 Machine data

Number	Identifier	Name	Reference
General ($MN_ ...)			
10110	PLC_CYCLE_TIME_AVERAGE	Maximum PLC acknowledgment time	
18360	MM_EXT_PROG_BUFFER_SIZE	FIFO buffer size for processing from external	A2
Channel-specific ($MC_ ...)			
20170	COMPRESS_BLOCK_PATH_LIMIT	Maximum traversing length of NC block for compression	K1
20400	LOOKAH_USE_VELO_NEXT_BLOCK	Look Ahead for constant accel. velocity control	
20430	LOOKAH_NUM_OVR_POINTS	No. of override switch points for Look Ahead	
20440	LOOKAH_OVR_POINTS	Override switch points for Look Ahead	
20450	LOOKAH_RELIEVE_BLOCK_CYCLE	Relief factor for the block cycle time	
20460	LOOKAH_SMOOTH_FACTOR	Smoothing factor for Look Ahead (≧ SW 5.3)	
20462	LOOKAH_SMOOTH_WITH_FEED	Smoothing takes feedrate into account (SW 6 and higher)	
20465	ADAPT_PATH_DYNAMIC	Adapt. of dynamic path response (≧ SW 6.2)	
20480	SMOOTHING_MODE	Response of rounding with G642 and G643 (SW 6 and higher)	
20490	IGNORE_OVL_FACTOR_FOR_ADIS	G641/642 independent of overload factor	
20550	EXACT_POS_MODE	Exact stop conditions with G0/G1 (≧ SW 6)	
20602	CURV_EFFECT_ON_PATH_ACCEL	Influence of path curvature on dynamic response of the path (SW 5 and higher)	
20603	CURV_EFFECT_ON_PATH_JERK	Influence of path curvature on jerk of path (SW 5 and higher)	
28060	MM_IPO_BUFFER_SIZE	Number of NC blocks in IPO buffer (DRAM)	S7
28070	MM_NUM_BLOCKS_IN_PREP	Number of NC blocks for block preparation (DRAM)	S7
28520	MM_MAX_AXISPOLY_PER_BLOCK	Max. number of axis polynomials per block	S7
28530	MM_PATH_VELO_SEGMENTS	Number of storage elements for limiting path velocity in block	K1
28540	MM_ARCLENGTH_SEGMENTS	Number of storage elements for arc length function representation per block	K1
Axis/channel-specific ($MA_ ...)			
32310	MAX_ACCEL_OVL_FACTOR	Overload factor for velocity jump	
32431	MAX_AX_JERK	Max. axis jerk for path motion (SW 5 or higher)	
32432	PATH_TRANS_JERK_LIM	Max. axial jerk of a GEO axis at block limit (SW 3.2 and higher)	
32433	SOFT_ACCEL_FACTOR	Scaling of the SOFT acceleration limitation (SW 5 and higher)	
32434	G00_ACCEL_FACTOR	Scaling of the acceleration limitation for G00 (SW 5 and higher)	
32435	G00_JERK_FACTOR	Scaling of the axial jerk limitation for G00 (SW 5 and higher)	
32440	LOOKAH_FREQUENCY	Smoothing limit frequency for Look Ahead (SW 5.3 and higher)	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
33100	COMPRESS_POS_TOL	Maximum deviation with compensation	K1
36000	STOP_LIMIT_COARSE	Exact stop coarse	
36010	STOP_LIMIT_FINE	Exact stop fine	
36012	STOP_LIMIT_FACTOR	Factor for exact stop coarse/fine and standstill monitoring (SW 5.2 and higher)	
36020	POSITIONING_TIME	Delay time exact stop fine	

© Siemens AG, 2003. All rights reserved

7.3 Setting data

Channel-specific setting data ($SC_ ...)			
42465	SMOOTH_CONTUR_TOL	Max. contour deviation on rounding	
42466	SMOOTH_ORI_TOL is used.	Max. deviation of the tool orientation on rounding	
42470	CRIT_SPLINE_ANGLE	Limit angle for spline and polynomial interpolation and compressor	W1

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Acceleration (B2)

1	**Brief Description**		**1/B2/1-3**
2	**Detailed Description**		**1/B2/2-5**
	2.1	Acceleration profiles	1/B2/2-5
	2.2	Path-related jerk limitation	1/B2/2-7
	2.3	Acceleration limitation	1/B2/2-11
	2.4	Axis-related jerk limitation	1/B2/2-12
	2.5	Knee-shaped acceleration characteristic	1/B2/2-15
	2.6	Programmable acceleration	1/B2/2-17
	2.7	Current path acceleration specification	1/B2/2-18
	2.7.1	Example for $AC_PATHACC, BRISK	1/B2/2-20
	2.7.2	Example for $AC_PATHJERK, SOFT	1/B2/2-21
3	**Supplementary Conditions**		**1/B2/4-23**
4	**Data Descriptions (MD, SD)**		**1/B2/4-23**
	4.1	Axis-specific machine data	1/B2/4-23
	4.2	Channel-specific machine data	1/B2/4-28
	4.3	Channel-specific setting data	1/B2/4-29
	4.4	System variables	1/B2/4-30
5	**Signal Descriptions**		**1/B2/7-31**
6	**Example**		**1/B2/7-31**
7	**Data Fields, Lists**		**1/B2/7-31**
	7.1	Machine data	1/B2/7-31
	7.2	Setting data	1/B2/7-33
	7.3	System variables	1/B2/7-34
	7.4	Alarms	1/B2/7-34

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

1 Brief Description

The NC axes are accelerated and decelerated with preset acceleration values. This makes it possible to control the axes at the required velocities. The acceleration values which are possible depend on the torques of the drive motors, on the current supplying capacity of the drive controller and on the load bearing capabilities of the machine mechanics as well as on the acceleration profile of the CNC. The CNC checks whether the acceleration value stored in a machine data for each machine axis has been violated, taking path velocity and curvature into account. If necessary, the velocity is lowered in order to keep to the axis-specific maximum acceleration.

The characteristics and possibilities offered by the existing acceleration modes and acceleration limitations (see Chapter 3) are described in this document.

The following description contains information describing acceleration modes used in conjunction with the FM 353 positioning module.

References: /S7S/, SIMATIC S7-300
FM 353 Positioning Module for Stepper Drives

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

2 Detailed Description

2.1 Acceleration profiles

Abrupt acceleration

With the v/t-linear control of the axis velocity that is normally applied, the motion is controlled such that the acceleration rate changes abruptly over time. The aim of this type of velocity control is to maintain the programmed velocity over the greatest area covered by the block. For this reason, acceleration to the programmed feedrate is performed at the beginning of the block at a maximum value and deceleration to rest performed at the end of the block, also at the maximum value. With the discontinuous, stepped acceleration, jerkfree starting and braking of the axes is not possible, but a time-optimized velocity/time profile can be implemented.

Jerklimited acceleration

In contrast to the method involving abrupt changes in acceleration rate, the jerk-limited acceleration profile controls the motion such that the axis setpoint characteristic is smooth (without jerks). This is achieved by limitation of changes in acceleration. Because of the softer acceleration progression, the traverse time is longer than with stepped acceleration for the same distance, velocity and acceleration. This time loss could be compensated for by setting a higher acceleration for the axes.

As well as utilizing the acceleration capabilities of the machine to the full, acceleration with jerk limitation has the following advantages:

- Reduced wear to mechanical parts of the machine
- Reduction of the excitation of high-frequency, difficult to control vibrations of the machine.

© Siemens AG, 2003. All rights reserved

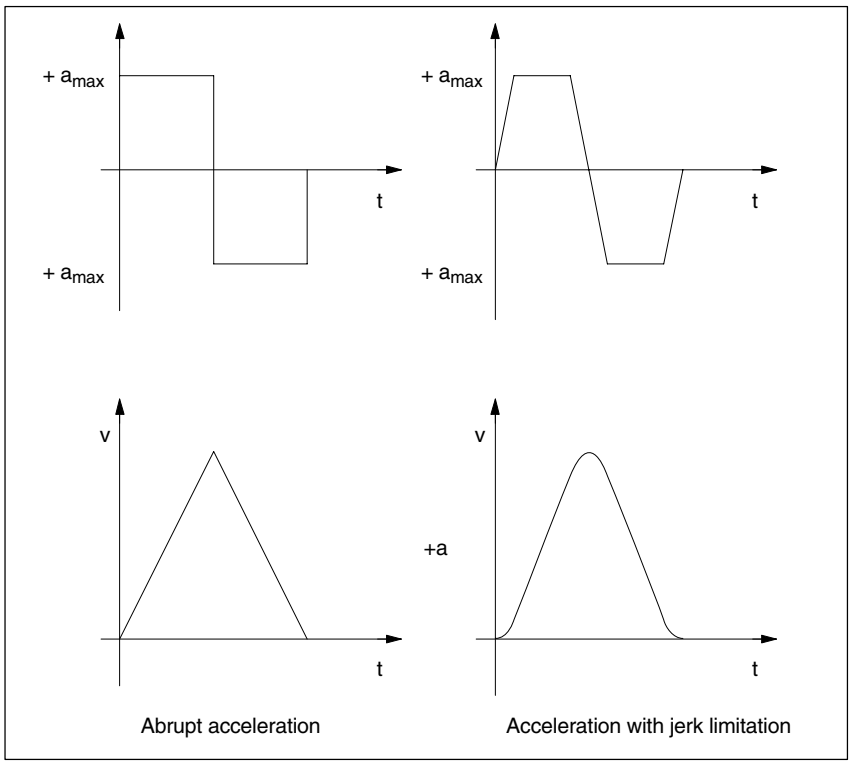


Fig. 2-1 Schematic illustration of velocity and acceleration profiles showing abrupt acceleration and acceleration with jerk limitation

a: Acceleration
v: Velocity
t: Time

© Siemens AG, 2003. All rights reserved

2.2 Path-related jerk limitation

In order to meet the special requirements of path interpolation, it is possible to choose between a stepped acceleration profile and an acceleration profile with jerk limitation. Jerk limitation is implemented entirely on the interpolator level.

Jerklimited acceleration

Acceleration with jerk limitation can be selected for the axes in the path grouping. The path-related maximum permissible jerk is stored in the channel-specific MD 20600: MAX_PATH_JERK (path-related maximum jerk). It limits the change of path acceleration. Using the permissible jerk value and the path acceleration, a time is calculated by which the change in acceleration must have been completed. The possible changes in acceleration have values between zero and the path acceleration. The acceleration limit is the maximum possible path acceleration. Because the path acceleration depends on the number of interpolating axes and on the shape of the path, the time calculated from the jerk limitation does not have a fixed length. The duration is proportional to path acceleration for constant jerk limitation.

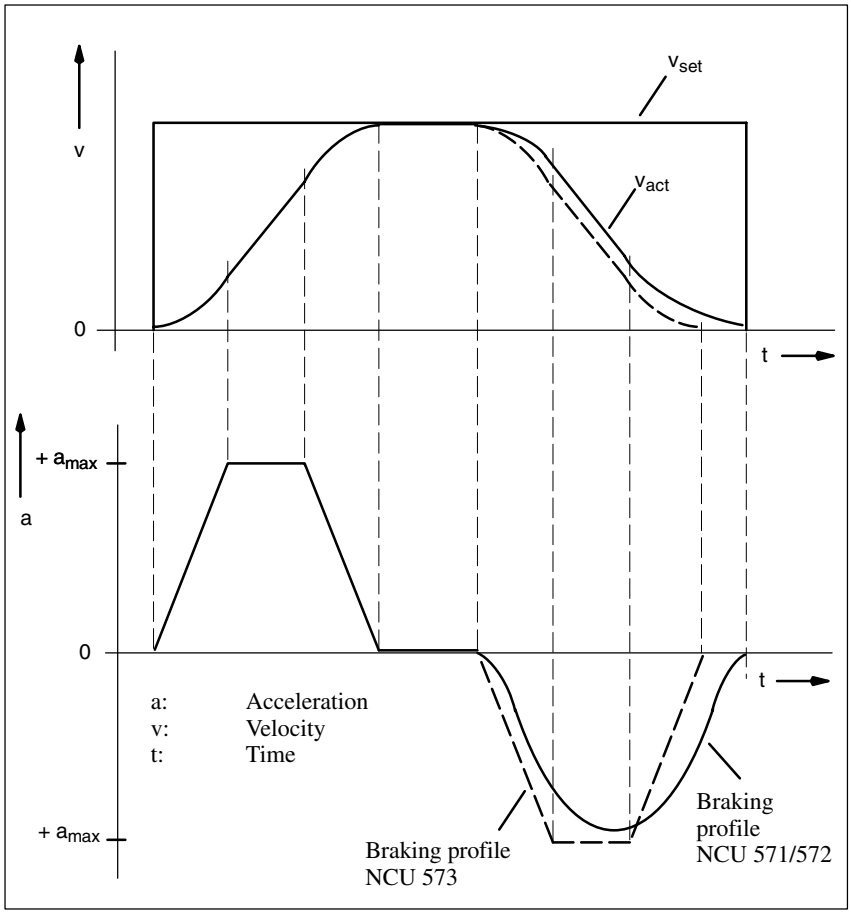


Fig. 2-2 Representation of velocity and acceleration progression when acceleration with jerk limitation has been implemented

© Siemens AG, 2003. All rights reserved

Note

With SW 5.3, the NCU 572 has the same braking profile as the NCU 573. The same applies to CCU1, CCU2, CCU3.

It can be seen from the previous Fig. that the acceleration profile is trapezoidal. The height of the trapeze is limited by the path acceleration. The gradient of the sides of the trapeze depends on the time which is calculated by dividing the acceleration by the path related maximum jerk. When an NCU 571 or 572 is used, the braking profile is bellshaped. With an NCU 573 the acceleration profile is trapezoidal. The height is limited by the path acceleration. The jerk limitation value set in MD 20600: MAX_PATH_JERK (path-related maximum jerk) limits the gradient of the curve.

Activation/ deactivation of acceleration with jerk limitation

The acceleration profile with jerk limitation based on the value set in MD 20600: MAX_PATH_JERK (path-related maximum jerk) is activated by:

- Program code SOFT in NC parts program

  Programming SOFT in the parts program is modal and deselects stepped acceleration profiles. If SOFT is programmed in a block with path axes, the previous block is ended with exact stop.

- SOFT as a default setting

  If SOFT is preset for the initial setting in MD 20150: GCODE_RESET_ VALUES (initial setting of G groups) (GCODE_RESET_ VALUES[20] = 2), the jerk limitation acts modally when processing of the parts program commences.
  When SOFT is selected, the acceleration profile shown in the previous Fig. is activated irrespective of the setting in MAX_PATH_JERK (path-related maximum jerk). A change in the active jerk value cannot be achieved in the NC parts program by overwriting MD:20600 MAX_PATH_JERK because this value is only activated on POWER ON.
  The active jerk value can be changed in the NC parts program by overwriting SD 42510: SD_MAX_PATH_JERK.

Acceleration with jerk limitation is deactivated by:

- Program code BRISK in NC parts program

  The program code BRISK is modal. If path axes are programmed in a block with BRISK, the previous block is ended with exact stop. BRISK activates the profile with abrupt acceleration changes associated with v/t-linear velocity control.

- BRISK as a default value

  BRISK can be specified as the initial setting of machine data MD 20150: GCODE_RESET_VALUES (initial setting of G groups) (GCODE_RESET_VALUES[20] = 1). The stepped acceleration profile is active modally from the time when processing of the parts program begins. This is the setting made with the standard installation.

© Siemens AG, 2003. All rights reserved

The acceleration profiles SOFT and BRISK are generated in different ways. For this reason, the SOFT profile will always require a longer traversing time for the same distance and acceleration rate than BRISK, regardless of the setting in MD 20600: MAX_PATH_JERK.

Applicability

Path-related jerk limitation is available for interpolating path axes in operation modes AUTO and MDA. The SOFT and BRISK acceleration profiles can be used in traverse modes exact stop G09, G60, continuous-path modes G64, G641 and with Look Ahead. The profiles are also active with the dry run feedrate function. With alarms that trigger a rapid stop, both acceleration profiles are inactive.

Abrupt acceleration profile with constant velocity phase

With the BRISK acceleration profile, a direct switchover from acceleration to braking takes place in short blocks in which the setpoint velocity cannot be reached. This transition entails a very large jump in acceleration. In order to attenuate this jump, the axes can be traversed with a constant velocity for a set minimum duration between the acceleration and the deceleration phases. One consequence of this is that the acceleration jump is split into two halves. The duration of the constant velocity phase can be set in MD 20500: CONST_VELO_MIN_TIME (minimum time with constant velocity).
The constant velocity phase according to the setting in MD 20500: CONST_VELO_MIN_TIME is not implemented:

- With Look Ahead
- In blocks in which the traveling time is less than or equal to one IPO cycle.

© Siemens AG, 2003. All rights reserved

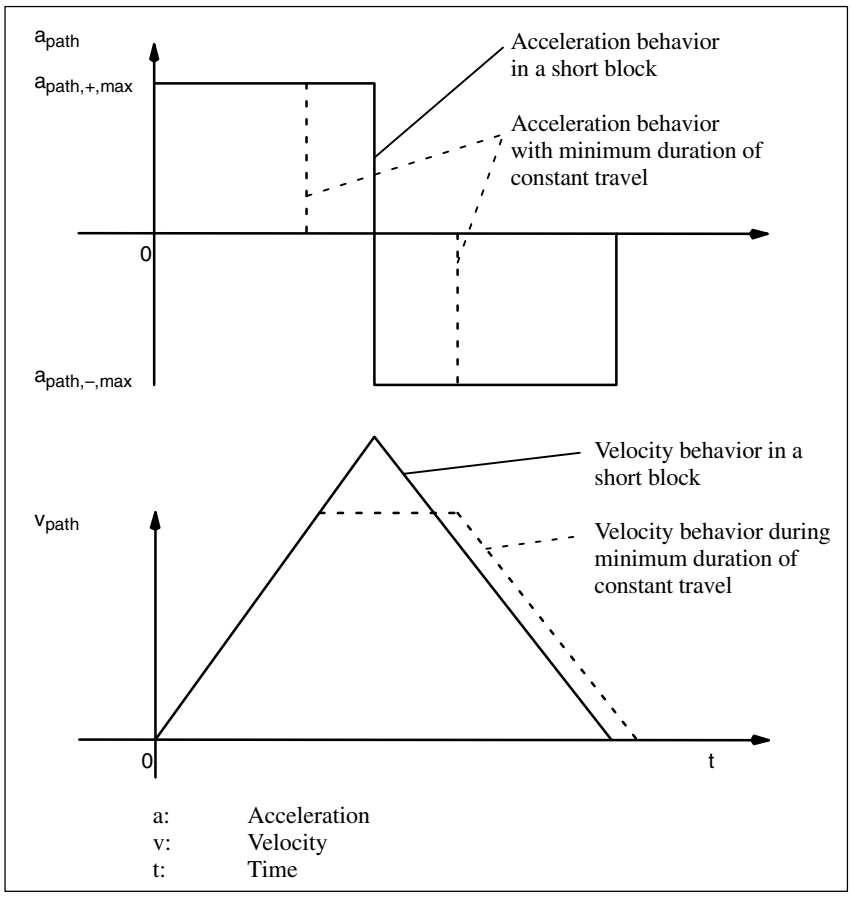


Fig. 2-3 Acceleration and velocity behavior in short blocks with and without a minimum constant travel duration

© Siemens AG, 2003. All rights reserved

2.3 Acceleration limitation

Request

The requirements on motion control can be fulfilled by means of the permanently set machine data. Special requirements, e.g. varying moment of inertia, different operating modes (positioning, roughing, finish cut) require an adaptation of the dynamic response of the acceleration and braking ramps.

One possible method of influence is the programmable axial acceleration limitation. Another additional limitation is the acceleration limitation via setting data. Whether this limitation value is to be included in the calculation must be specified via a separate setting data.

Furthermore, the dynamic response can be adapted via the keywords BRISK and SOFT. From time to time, however, the switchover is too coarse; the permissible jerk value can therefore also be adapted via an additional setting data.

Functionality

The call path can be set flexibly in SD 42500: SD_MAX_PATH_ACCEL can be set to specify a path acceleration limit value which acts in addition to the limit derived from the axial limitation values. This setting data is included in the jerk limitation calculation if SD 42502: IS_SD_MAX_PATH_ACCEL is set.

The call path can be set flexibly in SD 42510: SD_MAX_PATH_JERK can be set to specify an additional jerk limitation to the value set in MD 20600: MAX_PATH_JERK. This setting data is included in the jerk limitation calculation if SD 42512: IS_SD_MAX_PATH_JERK is set.

Note

SD 42500: SD_MAX_PATH_ACCEL is taken into account only if its value is lower than the limit value calculated from the axis motion and machine data MA_MAX_AX_ACCEL[AxNo], MA_MAX_AX_VELO[AxNo], allowing for the programmable reduction in acceleration.

SD 42510: SD_MAX_PATH_JERK is taken into account only if its value is lower than the setting in MD 20600: MAX_PATH_JERK.

Activation

The setting data can be set and activated via download, MMC or parts program. The programmed values remain valid until the SRAM is erased.

Examples for setting SDs in parts program

$SC_SD_MAX_PATH_ACCEL=2

$SC_SD_MAX_PATH_JERK=(0.8*$MC_MAX_PATH_JERK)=0.8·100=80

$SC_IS_SD_MAX_PATH_ACCEL=TRUE

$SC_IS_SD_MAX_PATH_JERK=TRUE

© Siemens AG, 2003. All rights reserved

Control system response

Control system response to POWER ON, operating mode switchover, Reset, block search and REPOS.

The last value to be set is valid. The default settings apply after a cold start.

§SC_IS_SD_MAX_PATH_ACCEL=FALSE

§SC_SD_MAX_PATH_ACCEL=10

$SC_IS_SD_MAX_PATH_JERK=FALSE

$SC_SD_MAX_PATH_JERK=1000

2.4 Axis-related jerk limitation

In addition to path-related jerk limitation, a jerk limitation can be set for each individual axis independently of path interpolation.

Jerk limitation on interpolator level

An axis-specific jerk limitation can be programmed for axes that are traversed in the conventional operating modes as well as for those that operate in positioning axis mode. The acceleration behavior is the same as for the SOFT acceleration profile of path-related jerk limitation and is set permanently and axis-specifically in machine data. This limitation cannot be deselected for the axes in the relevant modes.

The axes for which jerk limitation is to be programmed can be selected with MD32420: JOG_AND_POS_JERK_ENABLE (initial setting for axial jerk limitation). The maximum permissible axis jerk is stored in MD 32430: JOG_AND_POS_MAX_JERK (axial jerk). It limits the permissible change in axis acceleration. Taking the jerk value and acceleration rate programmed in MD32300: MAX_AX_ACCEL (axis acceleration) into account, a time is calculated in which the acceleration change is performed. The possible change in acceleration rate is in the range between 0 and the axis acceleration value stored in MD 32300: MAX_AX_ACCEL.

A change in the active jerk value cannot be achieved in the NC parts program by overwriting MD:32430 JOG_AND_POS_MAX_JERK because it only becomes active on RESET. A new value set becomes effective on the next program run.

The axial jerk limitation is active in AUTO and MDA modes for both positioning axis types.

The jerk limitation is active for axes in JOG mode during

- jogging
- handwheel jogging
- repositioning
- Teach In.

The jerk limitation is not active during

- reference point approach
- alarms that initiate a rapid stop.

© Siemens AG, 2003. All rights reserved

Jerk limitation on position controller

It is possible to set an axis-specific jerk limitation in the position controller in addition to and independently of the jerk limitations that can be activated on interpolator level. The jerk limitation causes smoothing of the axis setpoint progression with any axis movement irrespective of the mode the axis is traversed in. The acceleration limitations on the interpolator level are subjected to further smoothing with this limitation. It cannot be influenced by a program code or interface signal.

The limitation is enabled axis-specifically in
MD 32400: AX_JERK_ENABLE (axial jerk limitation) and set in associated with a timer for the smoothing filter by means of
MD 32410: AX_JERK_TIME (time constant for axial jerk filter).

With **SW 5.1** and higher, it is also possible to control the jerk limitation in the position controller with a new filter based on a smoothing method that incurs few contour errors:

MD 32402: AX_JERK_MODE = 1 ; 2nd degree filter (default)
corresponds to SW 1 to SW 4.4
MD 32402: AX_JERK_MODE = 2 ; floating averaging
(new jerk filter, available with SW 5.1 and higher).
MD 32402: AX_JERK_MODE = 3 ; bandstop filter in SW 6.3 and higher

Modus 2 requires a bit more computation time, but with the same smoothing effect, it results in lower contour errors or, with the same accuracy, in a smoother contour with smoother movements. Mode 2 is recommended. Mode 1 is the default mode in SW 1 to SW 4.4 for compatibility reasons.

For further information on the method of operation of the jerk filter available with SW 5.1 and higher (balancing filter for improving the position setpoints of the position controller), please refer to:

References: /FB/, G2, “Velocities, Setpoint/Actual Value Systems, Optimization of Control”

Mode 3
Bandstop filter can be parameterized in 2 variants:

- “Real bandstop filter”:
  When numerator and denominator frequencies are set identically (=blocking frequency). If you select (numerator) damping setting zero, the blocking frequency is equivalent to complete attenuation. The 3dB bandwidth is determined in this case by $f_{Bandwidth} = 2 * f_{Blocking}$
  If you do not want complete attenuation, but only a reduction by a factor k, then select the numerator damping according to k.

- “Bandstop filter with additional amplitude response increase/decrease at high frequencies”:
  In this case, the numerator and denominator natural frequencies are set to different values. The numerator natural frequency determines the blocking frequency. By selecting a lower (higher) denominator natural frequency than the numerator natural frequency, you can increase (decrease) the amplitude response at high frequencies. An amplitude response increase at high frequencies can be justified in most cases, as the controlled system generally possesses a lowpass characteristic itself, i.e. the amplitude response drops at high frequencies anyway.

© Siemens AG, 2003. All rights reserved

Note

Axial jerk limitation is useful in the following situations:

- When the smoothing times of the IPO level are tool long, i.e. when limitations are required within one IPO cycle,
- When the effect of the speed feedforward control needs to be diminished because traversal at 100% feedforward control appears to be too hard. By increasing the setting in MD 32410: AX_JERK_TIME, the motional response of the axes can be “softened” steplessly (but with a corresponding reduction in contour accuracy).
- When it is necessary to reduce oscillations at certain block transitions (e.g. straight line —> circle).

Except for linear blocks, this smoothing causes a contour falsification. This also applies if the same filter time is set for interpolating machine axes. Under this prerequisite, with a circle, the contour error results in a circle radius that is too small. If the same velocity and the same circle radius are programmed, the contour error will increase with increasing filter time. The jerk limitation is only disabled with alarms switching the axes into the follow-up mode.



Caution

The effective servo gain factor is reduced when an axial jerk limitation is programmed. This must be noted with respect to axes/spindles that must have an identical servo gain factor (e.g. for Rigid tapping).
There is generally no point in entering values higher than approximately 20–30ms in MD: AX_JERK_TIME (because the servo gain factor, and thus the contour accuracy, would then be reduced) (pay attention to the service display!).
Path-related jerk limitation (MD: MAX_PATH_JERK, MD: JOG_POS_MAX_JERK) should always be used first. If this option is not sufficient, then MD: AX_JERK_TIME should be set to a very low value (< 5ms) and used. If an even smoother axis motion is required and the degree of contour accuracy is sufficient (check with circularity test), then the value in AX_JERK_TIME can be increased.

© Siemens AG, 2003. All rights reserved

2.5 Knee-shaped acceleration characteristic

Note

This function is only available for positioning axes.

The speed-controlled motors used for axes and spindles in machine tools have a speed-dependent torque characteristic and therefore also a speed-dependent acceleration capability. In the lower speed range, the speed range under armature control, the torque is constant, but from a specific motor speed upwards, the torque decreases in the speed range under field control. The speed-dependent torque development is limited by the following factors:

- Field weakening

  Field weakening can cause an increase in the motor speed with DC motors. The torque output is correspondingly reduced by the speed in the field setting range.

- Commutation limit

  With DC motors the armature current must be reduced at high motor speeds because of brush sparking. This causes a reduction of the torque as the speed increases.

- Control

  Three-phase motors can also be controlled such that they increase their speed. However, the available torque is reduced as speed increases.

- DC link voltage

  The DC link voltage of the power section limits the maximum possible dynamic torque of three-phase drives because the current that can be fed in is reduced because of the mutual induction.

Until now, the setpoint was calculated and output by the CNC on the basis of the acceleration of the machine axes, making use of the entire speed range of the setpoint. From the moment at which motor torque started to develop as a function of speed, the drive continued to accelerate along its current limit, but the difference between setpoint and actual value changed continually as the drive was accelerating in the torque-reducing speed range. The function “Knee-shaped acceleration characteristic” means that the dependency of the acceleration on the speed of the motors and velocity can be taken into account for spindles and positioning axes. This makes it possible to minimize the following error in the field setting range. The knee-shaped acceleration characteristic is produced using motor-specific characteristic data and machine-specific requirements. With respect to the maximum velocity/speed of the axis/spindle, a factor f_n stored in machine data MD 35220: ACCEL_REDUCTION_SPEED_POINT (speed for reduced acceleration) is used to calculate the velocity/speed at which torque reduction begins. The permitted acceleration rate to maximum velocity/speed corresponds to the permissible acceleration rate minus the amount representing the product of factor f_a in MD 35230: ACCEL_REDUCTION_FACTOR (reduced acceleration) multiplied by the permissible acceleration.

© Siemens AG, 2003. All rights reserved

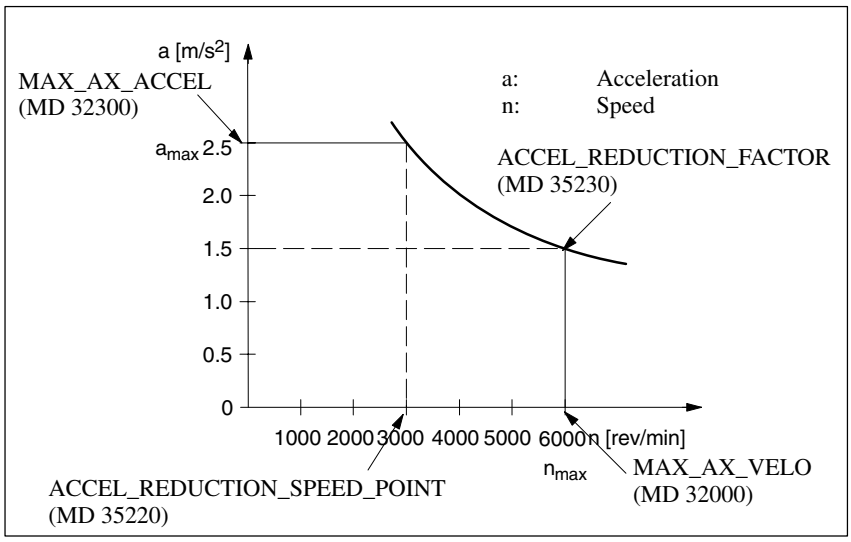


Fig. 2-4 Knee-shaped acceleration characteristic with $f_a = 0.4$ and $f_n = 0.5$

The progression of the acceleration characteristic should, as far as possible, correspond to the progression of the torque of the motor. The torque that is required because of the acceleration should be less than or equal to the available motor torque.

For the knee-shaped acceleration characteristic of the spindles, the maximum speeds of the gear stages stored in MD: GEAR_STEP_MAX_VELO_LIMIT (maximum speed of the gear step) and the acceleration values permissible for gear steps stored in MD: GEAR_STEP_SPEEDCRTL_ACCEL (acceleration with speed control) or in MD: GEAR_STEP_POSCRTL_ACCEL (acceleration with position control) are relevant. Because greater accuracy is required for the spindle in position-controlled mode, for example thread cutting, than is required in speed-controlled mode, it is advisable to set the acceleration values in MD: GEAR_STEP_POSCRTL_ACCEL to be smaller than that in MD: GEAR_STEP_SPEEDCRTL_ACCEL in order to make use of the advantages of the smaller following error.

The maximum reference quantities for calculating the knee-shaped acceleration characteristic for positioning axes are the values stored in MD: MAX_AX_VELO (maximum axis velocity) and in MD: MAX_AX_ACCEL (axis velocity). Factors f_a and f_n must be derived from the data sheets of the motors used.

The knee-shaped acceleration characteristic can be combined with the function axis-related jerk limitation on the interpolator level. The knee-shaped acceleration characteristic takes the changes velocity settings into account using override.

Selection and deselection of knee-shaped acceleration characteristic

When the default settings in MD: ACCEL_REDUCTION_FACTOR and MD: ACCEL_REDUCTION_SPEED_POINT are changed, the knee-shaped acceleration characteristic function is activated. Selection and deselection using the NC parts program or the PLC interface is not possible.

© Siemens AG, 2003. All rights reserved

2.6 Programmable acceleration

Request

It may be required in critical program sections to limit the maximum axis acceleration rates which are programmed in MD 32300: MAX_AX_ACCEL (axis acceleration).

The “Programmable acceleration function” (see Chapter 3) can therefore be used, for example, to prevent mechanical vibration.

Functionality

The command

ACC[machine axis name] = percentage

can be programmed to specify a percentage value of > 0% and ≤ 200% for each machine axis. Interpolation is performed with an acceleration that does not exceed this percentage of the maximum axis acceleration stored in MD: MAX_AX_ACCEL.

Activation

The limitation is effective in all interpolation types of the AUTOMATIC and MDA modes.

Please note the following:

- The limitation is not active in JOG mode and during reference point approach.
- The command ACC is active immediately and modally.
- With the command ACC[machine axis name] = 100 the limitation is deactivated.
- On RESET and end of program, 100% of the maximum axis acceleration is activated.
- The limitation is also active with dry run feedrate.

Error states

The acceleration limitation is not active in error states that cause a rapid stop with an open position-controlled loop (because the axis is stopped according to a braking ramp of the set speed).

© Siemens AG, 2003. All rights reserved

2.7 Current path acceleration specification

Introduction

The methods described in Sections 2.1 to 2.6 for acceleration and jerk control apply to:

a) Normal machining in continuous-path mode and

b) reactions to path-related realtime events:
- Stop
- Feedrate change
  - by override switch
  - by synchronized actions via system variable $AC_OVR
- Safe speed SG for Safety Integrated
  (s. **References:** /FBSI/, Description of Functions SINUMERIK Safety Integrated)

For example, if reduced acceleration and jerk values are set in order to reduce the vibrations on a laser cutting head, the operations in b) are also performed more sluggishly than desired.

Solution

Two system variables available with SW 6.3 can be used to accelerate the reaction in the operations in b).

System variables $AC_PATHACC and $AC_PATHJERK can be modified from the parts program and from static synchronized actions. Each read and write operation that accesses these system variables from the parts program entails an implicit preprocessing stop.

Table 2-1 Effect of $AC_PATHACC with BRISK

Value of the system variable	Effect
$AC_PATHACC < acceleration specified during preprocessing	None
$AC_PATHACC > acceleration specified during preprocessing	Increases the path acceleration to allow a faster reaction to realtime path velocity demands. $AC_PATHACC maximum = 2 x MAX_AX_ACCEL[ ]: Larger values are limited to double the value of MD 32300: MAX_AX_ACCEL[ ] for the relevant axis.
The conditions apply to all axes involved in the path.	

Effect

System variable $AC_PATHACC allows the path acceleration [m/s**2 or inch/s**2] to be specified in the interpolation cycle for the application of override changes, stop/start events and alteration of the safe velocity SG.

© Siemens AG, 2003. All rights reserved

Exception

The variable does NOT have any effect in the case of changes to the tool path velocity resulting from path plans involving e.g. contour curvature, corners, kinematic limitations of the transformation, ...).

SOFT

System variable $AC_PATHJERK is also referenced for SOFT.

Table 2-2 Effect of $AC_PATHJERK for jerk

Value of the system variable	Effect
$AC_PATHJERK not equal to 0	Specification of the path jerk [m/s**3 or inch/s**3] in the interpolation cycle
$AC_PATHJERK can only overincrease the path jerk in the same way as $AC_PATHACC. The control does not limit the jerk allowance relative to MD 32431: MAX_AX_JERK[axis].	

This function is practical in cases where the tool path velocity is to be controlled more quickly than is appropriate during normal program operation.

Supplementary conditions

- RESET deactivates $AC_PATHACC, $AC_PATHJERK (i.e. sets it to zero).
- Mit $AC_PATHACC = 0, $AC_PATHJERK = 0 deactivates the function and saves processor time during the main run. – The assignment of values less than 0 sets the value implicitly to 0.
- Programming $AC_PATHACC or $AC_PATHJERK in the parts program triggers an implicit STOPRE.
- $AC_PATHACC, $AC_PATHJERK can be used in static synchronized actions.
- A change in feedrate override with increased acceleration also applies during a slow brake ramp specified during preprocessing.
- $AC_PATHACC, $AC_PATHJERK is only effective while the tool path velocity is being changed in order to implement the override changes or the stop/start event. Once the new tool path velocity is reached, the acceleration and jerk limitation values specified during preprocessing are active.
- The $AC_PATHACC excess is allowed without taking centripetal acceleration into account.
- In order to match the jerk to $AC_PATHACC via $AC_PATHJERK, you can set

  $AC_PATHJERK = $AC_PATHACC / smoothing time

  (with smoothing time e.g. .02s).

© Siemens AG, 2003. All rights reserved

2.7.1 Example for $AC_PATHACC, BRISK

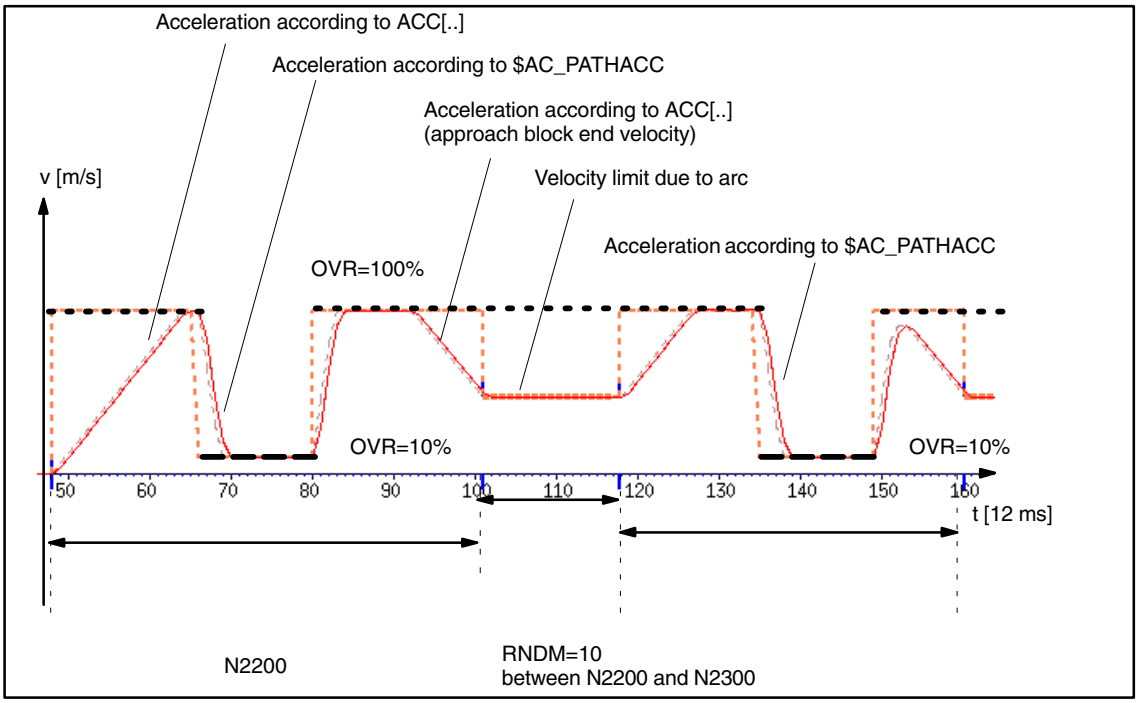


Fig. 2-5 Switching between path acceleration specified during preprocessing and $AC_PATHACC

Parts program (excerpt, schematic):

```
; Switch acceleration depending on high-speed input 1:
;------------------------------------------------------------------------------
N53 ID=1 WHEN $A_IN[1] == 1 DO $AC_PATHACC = 2.*$MA_MAX_AX_ACCEL[X]

; Test override profile (simulates external intervention):
;------------------------------------------------------------------------------
N54 ID=2 WHENEVER ($AC_TIMEC > 16) DO $AC_OVR=10
N55 ID=3 WHENEVER ($AC_TIMEC > 30) DO $AC_OVR=100
; Approach
;------------------------------------------------------------------------------
N1000 G0 X0 Y0 BRISK
N1100 TRANS Y=–50
N1200 AROT Z=30 G642


; Contour
;------------------------------------------------------------------------------
N2100 X0 Y0
N2200 X = 70 G1 F10000 RNDM=10 ACC[X]=30 ACC[Y]=30
N2300 Y = 70
N2400 X0
N2500 Y0
M30
```

© Siemens AG, 2003. All rights reserved

2.7.2 Example for $AC_PATHJERK, SOFT

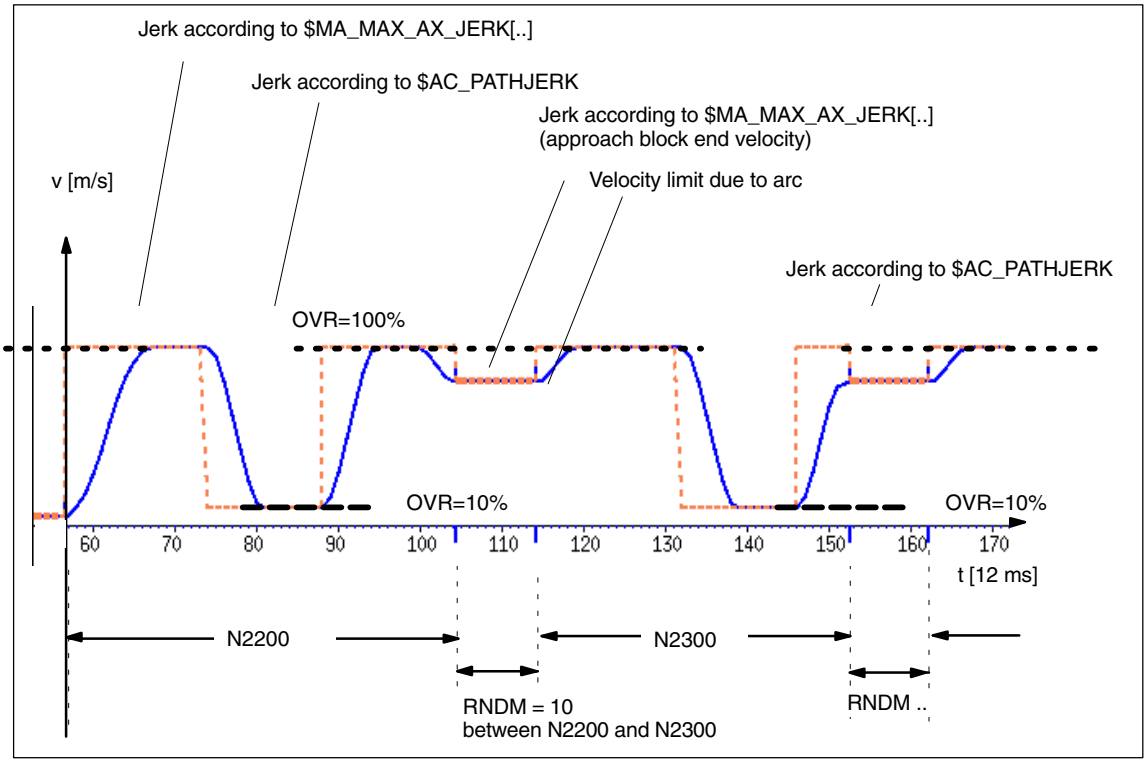


Fig. 2-6 Switching between path jerk specified during preprocessing and $AC_PATHJERK

Parts program (excerpt, schematic):

```
; Set path acceleration and path jerk on external intervention:
;------------------------------------------------------------------------------
 N0100 $AC_PATHACC = 0.
N0200 $AC_PATHJERK = 4. * ($MA_MAX_AX_JERK[X] + $MA_MAX_AX_JERK[Y]) / 2.

; Synchronized action for varying the override (simulates external intervention):
;------------------------------------------------------------------------------
N53 ID=1 WHENEVER ($AC_TIMEC > 16) DO $AC_OVR=10
N54 ID=2 WHENEVER ($AC_TIMEC > 30) DO $AC_OVR=100

; Approach
;------------------------------------------------------------------------------
N1000 G0 X0 Y0 SOFT
N1100 TRANS Y=–50
N1200 AROT Z=30 G642

; Contour
;------------------------------------------------------------------------------
N2100 X0 Y0
N2200 X = 70 G1 F10000 RNDM=10
N2300 Y = 70
N2400 X0
N2500 Y0
M30
```

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

Availability of the function

"Programmable acceleration"

The function is an option and is available for

- SINUMERIK 840D with NCU 572/573, SW 2 and higher
- SINUMERIK 810D with CCU2, SW 2 and higher
- SINUMERIK 810D with CCU1, SW 3.2 and higher

Acceleration limitation

The function is an option and is available for

- SINUMERIK 840D with NCU 572/573, SW 3 and higher

Knee-shaped acceleration characteristic

This function is only available for positioning axes.

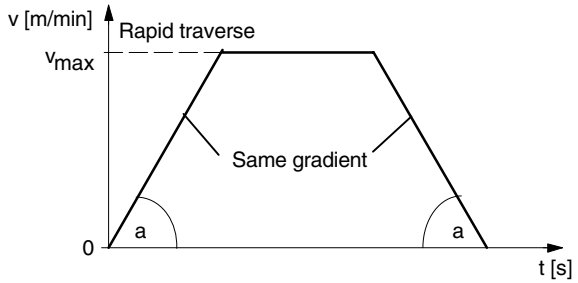
■

4 Data Descriptions (MD, SD)

4.1 Axis-specific machine data

32300 MD number	**MAX_AX_ACCEL** Axis acceleration		
Default setting: 1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: m/s^2, rev/s^2	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The acceleration defines a change in velocity of the axis against time. Different axes do not have to have the same acceleration. The lowest acceleration value of all the axes involved in interpolation is taken into account. In the case of rotary axes, the entered value corresponds to the angular acceleration. The machine manufacturer should determine for which continuous deceleration and acceleration the machine is suitable. This value is entered in the machine data. The acceleration value is active for every type of acceleration and delay operation.		
MD irrelevant for ...	Error states that lead to rapid stop.		

© Siemens AG, 2003. All rights reserved

32300 MD number	**MAX_AX_ACCEL** Axis acceleration
Fig. 4-1	
Related to	MD 32210: MAX_ACCEL_OVL_FACTOR (overload factor for axial velocity step changes) MD 20610: ADD_MOVE_ACCEL_RESERVE (acceleration margin for overlaid motions)

32400 MD number	**AX_JERK_ENABLE** Axial jerk limitation		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after NEW_CONF	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Enables the axial jerk limitation function in every mode. The limitation is set via a time constant and is always active. The limitation functions independently of the limitations “path-related maximum jerk”, “knee-shaped acceleration characteristic” and the axial jerk limitation of the axes that are operated in JOG or positioning mode.		
Related to	MD 32410: AX_JERK_TIME Time constant for axis jerk limitation)		

32402 MD number	**AX_JERK_MODE** Filter type for axial jerk limitation	
Default setting: 1	Minimum input limit: 1	Maximum input limit: 2
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –
Data type: BYTE	Applies from SW 5.2, max. of 32 position control cycles	

© Siemens AG, 2003. All rights reserved

32402 MD number	**AX_JERK_MODE** Filter type for axial jerk limitation
Significance:	Filter type for axial jerk limitation: 1: 2nd degree filter (same as SW 1 to 4) 2: Floating average generation (as of SW 5.1, max. of 16 position control cycles up to 0.032 s) 3: Bandstop (SW 6 and higher) Type 1 set as default for compatibility reasons. Type 2 is recommended. Type 2 requires a bit more computation time, but with the same smoothing effect, it results in lower contour errors or, with the same accuracy, in a smoother contour with smoother movements. In addition, type 2 is suited to dampen critical frequencies in the position setpoint, thus less exciting vibrations of the machine. The damping acts especially strong at the frequency f = 1 / (filter time of MD 32410) and multiples of it. The maximum jerk is set with the time constant MD 32410: AX_JERK_TIME are set. The greater the time, the smaller the maximum jerk. With the types 1 and 2, any setting times are accepted. Recommended values for type 1: min. 0.03s to max. 0.06s. Recommended values for type 2: min. 1 position control cycle to max. 32 position control cycles (31 position control cycles available). For example, at a position control cycle of 2ms, this corresponds to 0.002 s to 0.062 s for MD AX_JERK_TIME. MD 32402: AX_JERK_MODE is only effective when MD 32400: AX_JERK_ENABLE is set to 1. Type 3 requires the setting of machine data AX_JERK_TIME, AX_JERK_FREQ and AX_JERK_DAMP. To parameterize a pure bandstop filter, it is advisable to parameterize AX_JERK_TIME=0 which automatically sets “Denominator frequency = numerator frequency = blocking frequency = AX_JERK_FREQ”. When you set AX_JERK_TIME>0, you set a different denominator frequency. This enables you to implement a bandstop filter with amplitude boost at frequencies higher than the blocking frequency.
Special cases, errors, ...	The machine data must be the same for all axes of an axis container. /FB/, B3, Distributed Systems, Several Operator Panels and NCUs, NCU Link
Related to	MD 32400: AX_JERK_ENABLE MD 32410: AX_JERK_TIME MD 32412: AX_JERK_FREQ MD 32414: AX_JERK_DAMP
References	/FB/, G2, Velocities, Setpoint/Actual Value Systems, Closed-Loop Control

32410 MD number	**AX_JERK_TIME** Time constant for axial jerk filter		
Default setting: 0.001	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Time constant of the axial jerk filter which causes a softer axis setpoint progression. The jerk filter is only active when the time constant is greater than the position control cycle. When you use jerk filter variant “Bandstop” (MD 32402: AX_JERK_MODE = 3), you must enter the denominator equivalent time constant in this machine data: $T_N = 1 / (2 * PI * f_N)$ where f_N = denominator natural frequency. When you set a value of “0”, denominator natural frequency = numerator natural frequency = blocking frequency is set automatically in the software, i.e. with $MA_AX_JERK_TIME=0, $T_N = 1 / (2 * PI * f_Z)$ applies in the software.		
MD irrelevant for ...	Error states that cause a change to follow-up mode (e.g. EMERGENCY STOP)		
Special cases, errors, ...	An identical time constant must be programmed for machine axes which interpolate with one another (e.g. with Rigid tapping).		
Related to	MD 32400: AX_JERK_ENABLE (axial jerk limitation) MD 32402: AX_JERK_MODE		

© Siemens AG, 2003. All rights reserved

32412 MD number	**AX_JERK_FREQ** Blocking frequency axial jerk filters		
Default setting: 10.0	Minimum input limit: 0.0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: Hz	
Data type:	Applies from SW		
Significance:	Blocking frequency axial jerk filter bandstop [1/s]		
Related to	MD 32400: AX_JERK_ENABLE (axial jerk limitation) MD 32402: AX_JERK_MODE		

32414 MD number	**AX_JERK_DAMP** Damping axial jerk filters		
Default setting: 0.0	Minimum input limit: 0.0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: –	
Related to	MD 32400: AX_JERK_ENABLE (axial jerk limitation) MD 32402: AX_JERK_MODE MD 32412: AX_JERK_FREQ		

32420 MD number	**JOG_AND_POS_JERK_ENABLE** Initial setting for axial jerk limitation		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Enables the axis-specific jerk limitation function for JOG and REF modes and positioning axis operation.		
Related to	MD 32430: JOG_AND_POS_MAX_JERK (axial jerk)		

32430 MD number	**JOG_AND_POS_MAX_JERK** Axial jerk	
Default setting: 1000	Minimum input limit: 0.0	Maximum input limit: plus
Changes effective after POWER ON	Protection level: 2	Unit: m/s^3, rev/s^3
Data type: DOUBLE	Applies from SW 1.1	
Significance:	The jerk limit value limits the rate of change of axis acceleration in the JOG and REF modes and in positioning axis operation. Settings and time calculation are as for MD 20600: MAX_PATH_JERK (pathrelated maximum jerk).	
MD irrelevant for ...	Path interpolation and error states that lead to rapid stop.	
Related to	MD 32420: JOG_AND_POS_JERK_ENABLE (initial setting of axial jerk limitation)	

© Siemens AG, 2003. All rights reserved

35220 MD number	**ACCEL_REDUCTION_SPEED_POINT** Speed for reduced acceleration		
Default setting: 1.0	Minimum input limit: 0.0	Maximum input limit: 1.0	
Changes effective after POWER ON	Protection level: 2	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	This machine data defines the speed/velocity for spindles/positioning axes from which the acceleration reduction is to start. The reference is the defined maximum speed/velocity. The starting point is a percentage of the maximum values. Example: MD: ACCEL_REDUCTION_SPEED_POINT = 0.7, the maximum speed is 3000rpm. Acceleration reduction begins with v_{on}= 2100rpm, i.e. the maximum acceleration capacity is utilized in the speed range 0 ...2099.99rpm. From 2100rpm to the maximum speed, it is operated with reduced acceleration.		
Related to	MD 32000: MAX_AX_VELO (maximum axis velocity) MD 35130: GEAR_STEP_MAX_VELO_LIMIT (maximum gear stage speed) MD 35230: ACCEL_REDUCTION_FACTOR (reduced acceleration)		

35230 MD number	**ACCEL_REDUCTION_FACTOR** Reduced acceleration	
Default setting: 0.0	Minimum input limit: 0.0	Maximum input limit: 1.0
Changes effective after POWER ON	Protection level: 2	Unit: factor
Data type: DOUBLE	Applies from SW 1.1	
Significance:	The machine data contains the factor by which the acceleration of the spindle/positioning axes is reduced with reference to the maximum speed/velocity. The acceleration is reduced by the factor between the starting speed/velocity defined in MD: ACCEL_REDUCTION_SPEED_POINT and the maximum speed/velocity. Example: : a= 10 rev/s^2, v_{on}= 2100rpm, MD: ACCEL_REDUCTION_FACTOR = 0.3. Acceleration and deceleration takes place within the speed range 0 ...2099.99rpm with an acceleration of 10 rev/s^2. From a speed of 2100rpm up to the maximum speed, the acceleration is reduced from 10 rev/s^2 down to 7 rev/s^2.	
MD irrelevant for ...	Errors that lead to rapid stop.	
Related to	MD 32300: MAX_AX_ACCEL (axis acceleration) MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL (acceleration in open-loop speed mode) MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in closed-loop position mode) MD 35242: ACCEL_REDUCTION_SPEED_POINT (speed for reduced acceleration)	

© Siemens AG, 2003. All rights reserved

4.2 Channel-specific machine data

20500 MD number	**CONST_VELO_MIN_TIME** Minimum time with constant velocity		
Default setting: 0.0	Minimum input limit: 0.0	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Defines the minimum time for constant velocity during transition from acceleration to deceleration in short blocks in which the set velocity cannot be reached. Entering a time of at least several IPO cycles prevents a direct transition from the acceleration to the deceleration phase and thus limits the acceleration jump to 50%. This acceleration limitation is only active with the acceleration profile BRISK.		
MD irrelevant for ...	Look Ahead does not take account of this function.		

20600 MD number	**MAX_PATH_JERK** Path-related maximum jerk		
Default setting: 100	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: m/s^3	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The jerk limitation restricts path acceleration in SOFT mode. The path acceleration divided by the jerk limitation value produces a time in which the acceleration change takes place. Examples: **Jerk limitation value** / **Path acceleration** / **Time** 100 m/s^3 / 1 m/s^2 / 0.01s 100 m/s^3 / 2 m/s^2 / 0.02s 100 m/s^3 / 3 m/s^2 / 0.03s 200 m/s^3 / 2 m/s^2 / 0.01s 300 m/s^3 / 3 m/s^2 / 0.01s		
MD irrelevant for ...	Error states that lead to a rapid stop. The limitation continues to be ineffective for positioning axes.		

20610 MD number	**ADD_MOVE_ACCEL_RESERVE** Acceleration reserve for overlaid movements		
Default setting: 0.2	Minimum input limit: 0.0	Maximum input limit: 0.9	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	This machine data contains that factor that defines the acceleration margin retained by a path movement in order to provide sufficient acceleration reserves for an overlaid movement for the velocity control. A factor of 0.2 means that the path axes utilize 80% of the path acceleration in normal operation. Only when a request for overlaid movement is made, can 100% of the path acceleration be utilized.		
MD irrelevant for ...	Error states that lead to a rapid stop. The limitation continues to be ineffective for positioning axes.		
Special cases, errors, ...	At the moment the machine data is only taken into account if the function “Fast retraction” is first activated.		
Related to	MD 32300: MAX_AX_ACCEL (axis acceleration)		

© Siemens AG, 2003. All rights reserved

4.3 Channel-specific setting data

42500 MD number	**SD_MAX_PATH_ACCEL** SD Maximum path acceleration		
Default setting: 10	Minimum input limit: 0	Maximum input limit:	
Changes effective after POWER ON	Protection level: 7	Unit: m/s^2	
Data type: DOUBLE	Applies from SW 3.1		
Significance:	Setting data for additional limitation of (tangential) path acceleration		
Related to	MD 32300: MAX_AX_ACCEL SD 42502: IS_SD_MAX_PATH_ACCEL		

42502 MD number	**IS_SD_MAX_PATH_ACCEL** SD for activation of additional limitation of path acceleration	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1
Changes effective after POWER ON	Protection level: 7	Unit:
Data type: BOOLEAN	Applies from SW 3.1	
Significance:	Setting data SD_MAX_PATH_ACCEL is included in the limit calculations if SD: IS_SD_MAX_PATH_ACCEL=TRUE	
Related to	SD 42500: SD_MAX_PATH_ACCEL	

42510 MD number	**SD_MAX_PATH_JERK** SD for additional limitation of (tangential) path jerk	
Default setting: 100	Minimum input limit:	Maximum input limit:
Changes effective after POWER ON	Protection level: 7	Unit: m/s^3
Data type: DOUBLE	Applies from SW 3.1	
Significance:	The maximum path-related jerk can be applied in addition to MD: MAX_PATH_JERK.	
Related to	MD 20600: MAX_PATH_JERK SD 42510: IS_SD_MAX_PATH_JERK	

42512 MD number	**IS_SD_MAX_PATH_JERK** SD for activating the additional limitation of the (tangential) path jerk via SDs	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1
Changes effective after POWER ON	Protection level: 7	Unit:
Data type: BOOLEAN	Applies from SW 3.1	
Significance:	Setting data SD_MAX_PATH_JERK is included in the limit calculations if SD: IS_SD_MAX_PATH_JERK=TRUE	
Related to	SD 42510: SD_MAX_PATH_JERK (SD for additional limitation of (tangential) path jerk)	

© Siemens AG, 2003. All rights reserved

4.4 System variables

Name	$AC_PATHACC			
Meaning	Specifies an increased path acceleration for override changes and Stop/Start events. $AC_PATHACC is only taken into account if the value is greater than the prepared acceleration limit. The value 0 disables the function. Values generating a machine axis acceleration which would exceed twice the value configured in $MA_MAX_AX_ACCEL[..] are limited internally to twice the value in $MA_MAX_AX_ACCEL[..].			
Data type	REAL			
Access	Read in parts program	Write in parts program	Read in Synchronized action	Write in Synchronized action
	x	x	x	x
Implicit preprocessing stop	x	x		

Name	$AC_PATHJERK			
Meaning	Specifies an increased path jerk for override changes and Stop/Start events. $AC_PATHJERK is only taken into account if the value is greater than the jerk limit specified during preprocessing. The value 0 disables the function.			
Data type	REAL			
Access	Read in parts program	Write in parts program	Read in Synchronized action	Write in Synchronized action
	x	x	x	x
Implicit preprocessing stop	x	x		

■

© Siemens AG, 2003. All rights reserved

5 Signal Descriptions

None

6 Example

None

7 Data Fields, Lists

7.1 Machine data

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
20500	CONST_VELO_MIN_TIME	Minimum time with constant velocity	
20600	MAX_PATH_JERK	Path-related maximum jerk	
20610	ADD_MOVE_ACCEL_RESERVE	Acceleration reserve for overlaid movements	
Axis-specific ($MA_ ...)			
32300	MAX_AX_ACCEL	Axis acceleration	
32400	AX_JERK_ENABLE	Axial jerk limitation	
32402	AX_JERK_MODE	Filter type for axial jerk limitation	
32410	AX_JERK_TIME	Time constant for axial jerk filter	
32412	AX_JERK_FREQ	Blocking frequency axial jerk filters	
32414	AX_JERK_DAMP	Damping axial jerk filters	
32420	JOG_AND_POS_JERK_ENABLE	Initial setting for axial jerk limitation	
32430	JOG_AND_POS_MAX_JERK	Axial jerk	
32432	PATH_TRANS_JERK_LIM	Max. axial jerk of a geometry axis at block boundary	B1

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
35220	ACCEL_REDUCTION_SPEED_POINT	Speed for reduced acceleration	
35230	ACCEL_REDUCTION_FACTOR	Reduced acceleration	

© Siemens AG, 2003. All rights reserved

7.2 Setting data

Number	Identifier	Name	Refer-ence
Channel-specific ($SC_ ...)			
42500	SD_MAX_PATH_ACCEL	Maximum path acceleration	
42502	IS_SD_MAX_PATH_ACCEL	SD for activation of additional limitation of path acceleration	
42510	SD_MAX_PATH_JERK	SD for additional limitation of (tangential) path jerk	
42512	IS_SD_MAX_PATH_JERK	SD for activating the additional limitation of the (tangential) path jerk via SD	

© Siemens AG, 2003. All rights reserved

7.3 System variables

Number	Identifier	Name	Refer-ence
Channel-specific ($SC_ ...)			
	$AC_PATHACC	Excess acceleration for Stop, feedrate change, safe velocity (Safety Integrated)	
	$AC_PATHJERK	Excess path jerk for Stop, feedrate change, safe velocity (Safety Integrated)	

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Diagnostic Tools (D1)

1 Brief Description 1/D1/1-3

2 Detailed Description 1/D1/2-5

2.1 Alarms, messages and alarm log 1/D1/2-5
2.1.1 General 1/D1/2-5
2.1.2 NCK alarm handler 1/D1/2-5

2.2 Service displays 1/D1/2-7
2.2.1 Axis/spindle service display 1/D1/2-8
2.2.2 Drive service display (for digital drives only) 1/D1/2-15
2.2.3 Service display PROFIBUS DP 840Di 1/D1/2-24
2.2.4 Communication log 1/D1/2-28
2.2.5 Log book 1/D1/2-29
2.2.6 Version 1/D1/2-29

2.3 PLC status 1/D1/2-30

2.4 Other diagnostics tools 1/D1/2-31

2.5 Identification defective drive modules (SW 6.3 and higher) 1/D1/2-32

3 Supplementary Conditions 1/D1/4-35

4 Data Descriptions (MD, SD) 1/D1/4-35

4.1 General machine data 1/D1/4-35

5 Signal Descriptions 1/D1/6-39

6 Example 1/D1/6-39

7 Data Fields, Lists 1/D1/7-41

7.1 Interface signals 1/D1/7-41

7.2 Machine data 1/D1/7-42

7.3 Alarms 1/D1/7-43

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

1 Brief Description

Integrated diagnostic tools

Test aids are integrated in the control system as diagnostic tools for the end customer or the service staff on site. The following information can be displayed on the operator panel in the “Diagnostics” operating area:

- Display of alarms and messages from the control system or drives in plaintext
- Status displays for:
  - Interface signals between the NCK, MMC and PLC as well as PLC and PLC I/O modules
  - Data blocks
  - PLC flags, timers and counters
  - Inputs and outputs of the PLC.
- Service displays
  - Information about the actual values and states of axes/spindles (e.g. position setpoint/actual value, drive status, actual current value, following error, contour deviation, ...)
  - Communications error log NC/PLC/MMC
  - Logbook with the relevant system modifications (e.g. altered password)
  - Version display showing the currently available system SW versions.

  Signal states and signal combinations can be changed for test purposes.

External diagnostic tools

The 611D startup tool also provides support in optimizing drives and analyzing mechanical components and properties.

The PCIN program can be used to archive machine data, setting data, parts programs, etc.

Identification of defective drive modules (SW 6.3 and higher)

Using MD 13030 DRIVE_MODULE_TYPE it is possible to remove individual modules from the drive bus configuration at the NC end (the affected axes are switched to Simulation mode). In this way, for example, you can extract a 611D module specified in an alarm text from the bus.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 Alarms, messages and alarm log

2.1.1 General

The currently active or not yet acknowledged alarms and messages are displayed in the Diagnostics operating area.

The alarms which have been generated and the time that they occurred are listed in the alarm log.
Detailed explanations of the alarms which may occur are given in
References: /DA/ Diagnostics Guide or
in the **online help** in systems with MMC 101/102/103.

For alarms and messages that have been programmed by the machine tool manufacture (value range ... ...) please refer to the relevant explanations in the documentation provided by the manufacturer.

2.1.2 NCK alarm handler

Application

The alarm handler provides an infrastructure for the activation and management of alarms on the NCK.

Functions

- Battery backup of a maximum of 16 alarms that have been activated since system power-up and are not yet reset.
- Alarm reactions can be programmed as channel-specific, modegroup-specific or NCK-specific reactions.
- The “NoReady” alarm reaction can be programmed as a channel-specific reaction.

Activation

The alarm handler is activated when an error status is detected in the NCK, causing an alarm to be output.

An alarm can also be triggered by a parts program containing the language command **SETAL**.
References: /PGA/ Programming Guide, Advanced

© Siemens AG, 2003. All rights reserved

Note

The currently active alarms in the NCK are read via the operator panel interface.

It is not possible to set alarms externally in the NCK.

Alarms with an alarm ID in the 60000 to 60999 range can be activated in a parts program.

Data backup

On POWER ON, the alarm handler data are reinitialized completely since they are not stored in the buffered SRAM.

Compatibility

SW 4 and higher

With SW 4.1 and higher, it is possible to set the channel-specific signal CHANNEL_NOREADY in the VDI interface in response to alarms.

Up to SW 3.x

The MD 11412: ALARM_REACTION_CHAN_NOREADY is set to determine whether the channel-specific signal CHANNEL_NOREADY is used for the function, thus ensuring that earlier PLC versions remain compatible.

Default setting: CHANNEL_NOREADY signal is not used.
Alarms that have specified a channel-specific NOREADY signal are reconfigured to the modegroup-specific NOREADY signal.

Delete criterion

You must specify for each alarm whether the alarm can be cleared again. The following criteria are possible options:

- POWERONCLEAR
  The alarm is cleared by switching the control off and then on again.
- RESETCLEAR
  The alarm is cleared when the Reset key is pressed in the channel in which it has occurred.
- CANCELCLEAR
  The alarm is cleared when the Cancel key is pressed in any channel. The alarm can also be canceled by means of an NC start or Reset.
- NCSTARTCLEAR
  The alarm is cleared when a program is started in the channel in which the alarm has occurred. The alarm can also be canceled by means of a Reset.
- CLEARHIMSELF
  The alarm is not cleared by an operator input or action, but explicitly by a “clearAlarm” programmed in the NCK source code.
- BAGRESETCLEAR
  The alarm is canceled by a “BAGRESETCLEAR” command or by the execution of a Reset in every channel of this mode group.
- NCKRESETCLEAR
  The alarm is canceled by an “NCKRESETCLEAR” command or by the execution of a RESET in every channel.

© Siemens AG, 2003. All rights reserved

Alarm display control

The scope of alarm outputs can be modified via machine data.

- MD 11410: SUPPRESS_ALARM_MASK
  Screen form for suppressing special alarm outputs
- MD 11411:ENABLE_ALARM_MASK
  Screen form for activating special alarm outputs

For details of these machine data, please refer to Chapter 4.

2.2 Service displays

Operator actions

For how to operate the service displays see:
References: /BA/, “Operator’s Guide”

Note

It is possible to switch over between the displays using the vertical soft key **PARTVIEW** / **OVERALLVIEW** on the MMC 102/103. The data in the partial view are updated at significantly shorter intervals.

General

For diagnostic purposes, the SINUMERIK 840Di provides various service displays via HMI Advanced (formerly: MMC 103) and 840Di StartUp.

Diagnostics

The following service displays are provided for SINUMERIK 840Di:

- HMI Advanced
  - Service display “Axis/spindle”
- 840Di StartUp
  - Service displays “PROFIBUS DP”.

Supplementary conditions

Compared with SINUMERIK 840D, the following service displays and operating areas are not implemented in the SINUMERIK 840Di:

- HMI Advanced
  - Service display “Drive”
  - Operating level “Safety Integrated”

© Siemens AG, 2003. All rights reserved

2.2.1 Axis/spindle service display

For the purposes of installing and diagnosing

- axes and
- spindles,

the information shown in the following figure can be called up for each axis/spindle in the “Diagnostics” operating area via the operator panel.

Selection and operation of the “Diagnostics” area is described in
References: /BA/, “Operator’s Guide”

Application

This information is used to

- check the setpoint branch
  (e.g. **position setpoint, speed setpoint, spindle speed setpoint prog.**)
- check the actual value branch
  (e.g. **actual position value measuring system 1/2, actual speed value**)
- optimize the position control loop
  (e.g. **following error, control deviation, servo gain factor**)
- check the whole closed-loop system of the axis
  (e.g. by a comparison between the position setpoint and actual position value, speed setpoint and actual speed value)
- check for hardware errors
  (e.g. by checking the encoder, if the axis is moved mechanically, a change in the actual position value must result)
- set and check axis monitoring functions.

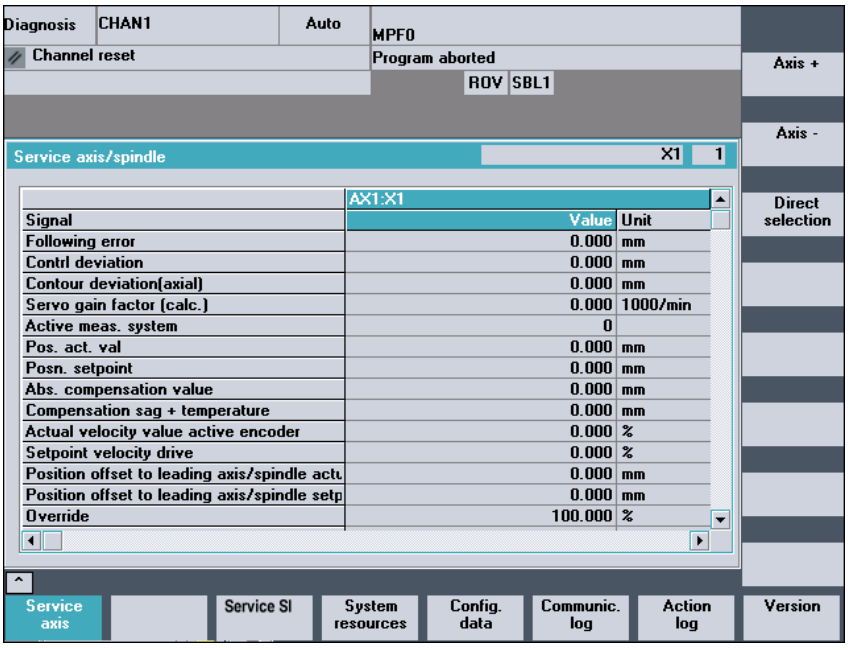


Fig. 2-1 Axis/spindle service display

© Siemens AG, 2003. All rights reserved

Diagram description

Following error

The difference between the position setpoint and actual position value of the active measuring system 1 or 2.
Unit: mm, inch or degrees

Control deviation

The difference between the position setpoint at the position controller input and the actual position value of the active measuring system 1 or 2.
Unit: mm, inch or degrees

Contour deviation

This value indicates the current contour deviation (fluctuations in following error caused by compensatory processes on the speed controller due to changes in load).
The contour deviation is based on the difference between a precalculated actual position value derived from the position setpoint and the actual position value of the active measuring system 1 or 2.
Unit: mm, inch or degrees

Servo gain factor (calculated)

The servo gain factor in the display is calculated by the NC according to the following equation:

$$\text{Servo gain factor} = \frac{\text{Velocity setpoint}}{\text{Following error}};$$

$$\text{unit (of the default setting):} \quad \frac{[\text{m/min}]}{[\text{mm}]};$$

Velocity setpoint = setpoint currently being output to the axis/spindle.

References: /FB1/, G2, “Velocities, Setpoint/Actual Value Systems, Closed-Loop Control”

Active measuring system

This line indicates whether measuring system 1 or 2 is active.

Actual position value measuring system 1/2

Actual position of the axis as measured by means of measuring system 1/2.
The position is displayed in the machine coordinate system (no zero point offsets or tool offsets taken included).
Unit: mm, inch or degrees

Position setpoint

Setpoint position transferred from the interpolator to the position control.
Unit: mm, inch or degrees

Absolute compensation value meas. system 1 or 2

Displays the absolute compensation value for measuring system 1 or 2.
The compensation value is the sum of the backlash compensation and leadscrew error compensation of the current axis position.
Unit: mm, inch or degrees

Sag and temperature compensation

Display of compensation value calculated for the current axis position based on the total of the sag and temperature compensations.
Unit: mm, inch or degrees

© Siemens AG, 2003. All rights reserved

Velocity actual value of active encoder (only 840Di)

Display of velocity actual value of the encoder currently active.

Velocity setpoint of drive (only 840Di)

Display of velocity setpoint of drive.

Speed actual value

The pulses supplied by the encoder are evaluated and displayed by the NC.
Unit: %
100% signifies maximum speed (corresponds to 10V on analog interface; maximum speed with 611D, programmed in MD 1401: MOTOR_MAX_SPEED (speed for maximum useful motor speed)).

Speed setpoint

Speed setpoint transferred to the drive (= speed setpoint from position controller and feedforward control).
Unit: %
100% corresponds to the maximum speed setpoint (10V for an analog interface, maximum speed for 611D).

Spindle speed setpoint programmed

Speed setpoint programmed by the user.
Unit: rpm
e.g.: Input: S1000; display: 1000rpm

Display applies to spindles only.

Spindle speed setpoint current

Current active speed setpoint with correct sign, incl. calculated compensation value and any operative speed limitation (programmed by means of setting or machine data).
Unit: U/min

Display applies to spindles only.

Override

The active override factor of the feedrate and spindle override switch is displayed.
Unit: %

Position offset for master axis / spindle actual value

The currently applicable position offset value is displayed on this line (referred to the actual value) if such an offset (angular offset between master and slave axes) has been programmed for the “Synchronous spindle” function.
Unit: mm, inches, degrees
References: /FB/, S3, Synchronous Spindles

Position offset for master axis / spindle actual value

The currently applicable position offset value is displayed on this line (referred to the setpoint) if such an offset (angular offset between master and slave axes) has been programmed for the “Synchronous spindle” function.
Unit: mm, inches, degrees
References: /FB/, S3, Synchronous Spindles

© Siemens AG, 2003. All rights reserved

Current gear stage

The current actual gear stage is displayed here.

In the case of axes, the value is only displayed if a spindle is assigned to the axis.
The display corresponds to IS “Actual gear stage” (DB31, ... DBX16.0 to 16.2).
References: /FB1/, S1, “Spindles”

Parameter set (axis)

Display indicating which of the 6 parameter sets of the position controller is active.
For example, a gear change causes a change of parameter set.

References: /FB1/, G2, “Velocities, Setpoint/Actual Value Systems, Closed-Loop Control”

Controller mode

Display of current controller states.

0:	Position control
1:	Speed control
2:	Hold
3:	Parking
4:	Follow-up
5:	Braking

For a more detailed description of the controller states see:
References: /FB1/, A2, “Various Interface Signals“

“Referenced” status display

Status display for reference point approach (axis).

Bit0=status 0:	The machine axis with position measuring system 1 or 2 is not referenced.
Bit0=status 1:	The machine axis reached the reference point (incremental measuring system) or the target point (linear measurement system with distancecoded reference marks) with reference point approach.
Bit1=status 0:	Referencing not mandatory (this axis need not be referenced)
Bit1=status 1:	Referencing not mandatory for NC Start.

Dependent on machine data
MD 34110: REFP_CYCLE_NR and
MD 20700: REFP_NC_START_LOCK

Display corresponds
IS “Referenced/synchronized” 1 or 2 (DB31, ... DBX60.4 and 60.5).

References: /FB1/, R1, “Reference Point Approach”

QEC learning active

Indicates whether or not the learning process for quadrant error compensation is active.

Fixed stop reached

Indicates whether or not the axis has fulfilled the conditions for “Fixed stop reached” when the “Travel to fixed stop” function is active (IS DB31 ..., DBX62.5)
References: /FB/, F1, Travel to Fixed Stop

© Siemens AG, 2003. All rights reserved

Torque limitation value

Displays the value programmed via FXST[x] or SD 43510: FIXED_STOP_TORQUE or the clamping torque defined via MD 37010: FIXED_STOP_TORQUE_DEF for the Travel to fixed stop function.
Unit: % of the maximum torque
References: /FB/, F1, Travel to Fixed Stop

Safe actual position axis

Displays the current axis actual position that has been measured via the NC. This actual position should have the same value as “Safe actual position” drive.
References: /FBSI/ Description of Functions, Safety Integrated

Safe actual position drive

Displays the current axis actual position that has been measured via the drive. This actual position should have the same value as “Safe actual position” axis.
References: /FBSI/ Description of Functions, Safety Integrated

Safe input signals axis

Displays the safe input signals of the PLC defined for the “Safety Integrated” function. The state of these input signals should correspond to that of “Safe input signals drive”.
References: /FBSI/ Description of Functions, Safety Integrated

Safe input signals drive

Displays the safe input signals of the drive (DMP on drive bus) defined for the “Safety Integrated” function. The state of these input signals should correspond to that of the “Safe input signals axis”.
References: /FBSI/ Description of Functions, Safety Integrated

Safe output signals axis

Displays the safe output signals of the PLC defined for the “Safety Integrated” function. The state of these output signals should correspond to that of “Safe output signals” drive.
References: /FBSI/ Description of Functions, Safety Integrated

Safe output signals drive

Displays the safe output signals of the drive (DMP on drive bus) defined for the “Safety Integrated” function. The state of these output signals should correspond to that of “Safe output signals axis”.
References: /FBSI/ Description of Functions, Safety Integrated

The following description contains more information about problems which may be encountered and clarifies further queries.

© Siemens AG, 2003. All rights reserved

Data readoff

The figure below shows at which points in the closedloop system the axis and spindle information is read off.

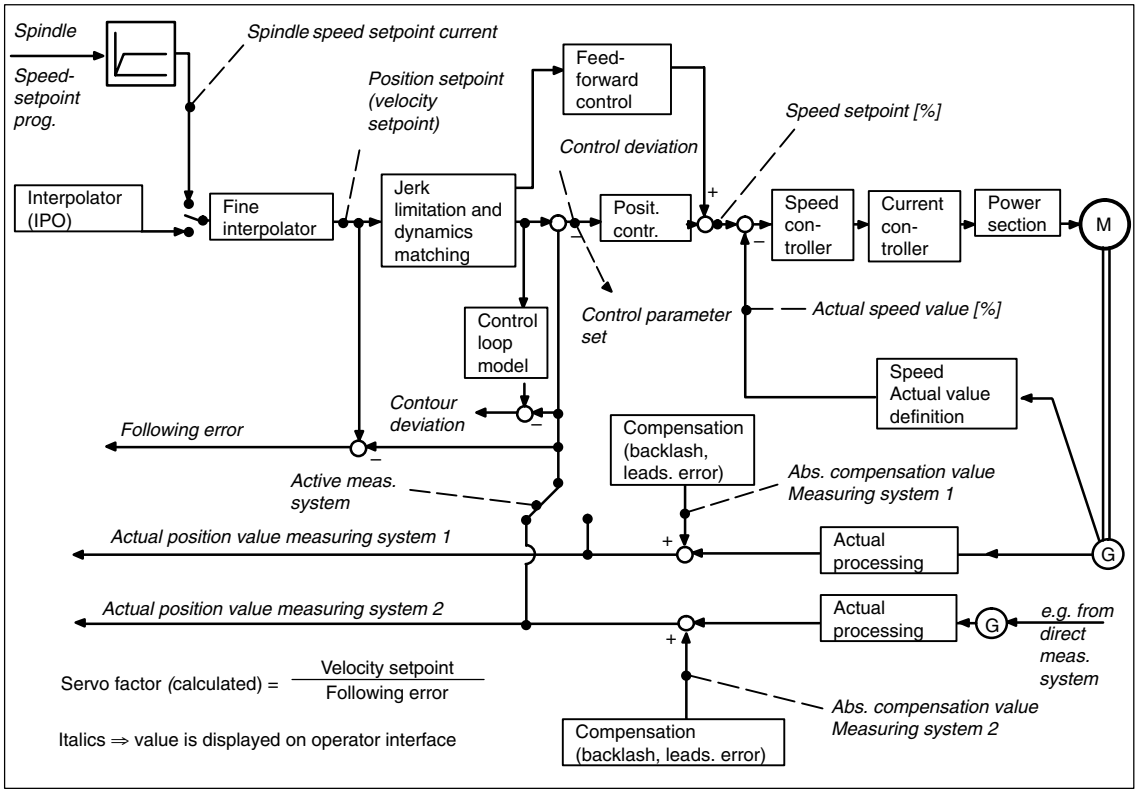


Fig. 2-2 Overview diagram of axis and spindle information

Checking the position controller setting

A simple check of the position controller settings can be made via the service axis display by entering the number 1 (corresponding to KV = 1) in MD 32200: POSCTRL_GAIN[n] (servo gain factor). The change takes immediate effect.

Because the servo gain factor is defined as

$$KV = \frac{\text{Velocity}}{\text{Following error}}; \quad \frac{[\text{m/min}]}{[\text{mm}]}; \text{ (default setting)}$$

a **following error** of 1mm must be measured (with KV = 1 and constant velocity) at a feedrate of 1 m/min.

© Siemens AG, 2003. All rights reserved

If the desired servo gain factor does not correspond to the actual factor, the possible causes and remedial optimization options are as follows:

- MD 32260: RATED_VELO (rated motor speed), MD 32250: RATED_OUTVAL (rated output voltage).
  (Applies to FMNC only)
- Speed and torque feedforward control is switched on.
  In this case, a higher servo gain factor is displayed than set in MD 32200: POSCTRL_GAIN[n] (servo gain factor).
- Filter for jerk limitation and dynamics matching is activated.
  In this case, a lower servo gain factor is displayed than set in MD 32200: POSCTRL_GAIN[n] (servo gain factor).

Diagnostics for alarms

This information is also provided as a diagnostic tool for diagnosing the causes of alarms such as:

- 25040 “Zero speed monitoring”
  ⇒ **following error** > MD 36030: STANDSTILL_POS_TOL (zero speed tolerance)
- 25050 “Contour monitoring”
  ⇒ **contour deviation** > MD 36400: CONTOUR_TOL
  (tolerance band for contour monitoring)
- 25060 “Speed setpoint limitation”
  ⇒ **speed setpoint** > MD 36210: CTRLOUT_LIMIT
  (maximum speed setpoint)
- 25080 “Positioning monitor”
  ⇒ **following error** > MD 36010: STOP_LIMIT_FINE
  (exact stop fine)
- 25100 “Measuring systems cannot be switched
  ⇒ The difference between **the actual position value of the measuring system 1 or 2** > MD 36500: ENC_CHANGE_TOL
  (max. tolerance for actual position value acquisition)
- 26000 “Clamping monitor”
  ⇒ **following error** > MD 36050: CLAMP_POS_TOL
  (clamping tolerance with interface signal “Clamping active”)

Please refer to
References: /DA/, “Diagnostics Guide”

© Siemens AG, 2003. All rights reserved

Diagnostics of defective state errors

The following information is provided to assist in the analysis of operational state errors such as:

- The axis does not traverse even though a travel command has been given.
  ⇒ Check whether the servo enable signal is applied.
  Position control or speed control (for spindle control) must be activated in **controller mode**.
- Feedrate fluctuations.
  ⇒ Detected via **following error** or **actual speed value.**
- Incorrect positioning.
  ⇒ Comparison **position setpoint** with **actual position of measuring system 1/2** and **abs. compensation value measuring system 1/2**.
- The cam is not detected by the PLC during referencing.
  ⇒ Check **Status display "referenced"**
- The reference point value displayed is incorrect.
  ⇒ The wrong measuring system may have been used for referencing.
- The **actual speed value** of the main spindle drive shows heavy fluctuations.
  ⇒ Speed range for encoder selected too high or MD 36300: ENC_FREQ_LIMIT[n] (encoder limit frequency) set higher than specified on the encoder data sheet.
- Spindle positioning is incorrect.
  ⇒ Wrong measuring system may be selected or synchronization with wrong zero marker.

2.2.2 Drive service display (for digital drives only)

For the purposes of installing and diagnosing

- feed drives (FDD) and
- main spindle drives (MSD)

the information shown in the following figure can be called up for each axis/spindle in the "Diagnostics" operating area via the operator panel.

Note

The parameters in the service display "Drive" are not necessary for connecting drives via PROFIBUS DP. With SINUMERIK 840Di, the drives are defined as PROFIBUS nodes. The appropriate service data are displayed in 840DiStartup in the menu Diagnostics –> PROFIBUS.

© Siemens AG, 2003. All rights reserved

Application

This information is used to

- check the status of enabling and control signals
  (e.g. **pulse enable**, **drive enable**, **motor selection**, **setpoint parameter set**)
- check the status of FDD/MSD operating modes
  (e.g. **setup mode**, **parking axis**)
- display temperature warnings
- check the current setpoint/actual value display
  (e.g. **actual position value measuring system 1/2**, **speed setpoint**, **actual speed value**)
- check the drive status **(drive ready)**
- display the current **ramp-up phase**
- display the group error message (**message status class 1**)
- display the drive status messages
  (e.g. **Threshold torque not reached, Minimum speed not reached**, **Actual speed = set speed**)

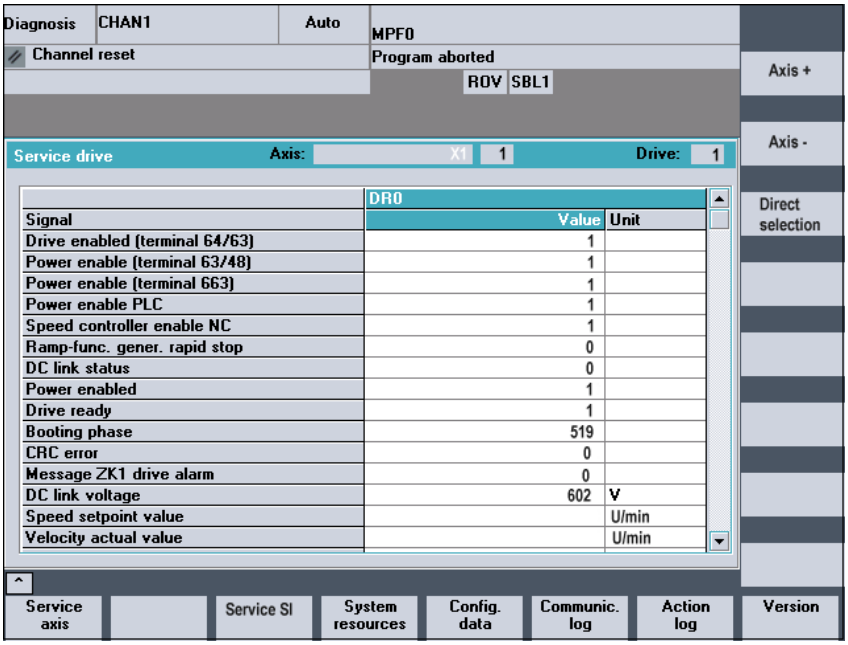


Fig. 2-3 Drive service display

Diagram description

The individual status displays, warnings, messages etc. are explained below.
For more detailed information please consult:
References: /IAD/, “Installation and StartUp Guide”

© Siemens AG, 2003. All rights reserved

Drive enable (terminal 64/63)

The display corresponds to the status of terminal 64/63 on the 611D infeed/regenerative feedback unit:

State 1: Central drive enable
State 0: Central drive disable

The display corresponds to MD 1700: TERMINAL_STATE (state of binary inputs).

Pulse enable (terminal 63/48)

The display corresponds to the status of terminal 63/48 on the 611D infeed/regenerative feedback unit:

State 1: Central pulse enable
State 0: Central pulse suppression

The display corresponds to MD 1700: TERMINAL_STATE (state of binary inputs).

Pulse enable (terminal 663)

The display shows the status of terminal 663 (relay: safe operational stop) on the drive module.

State 1: module-specific pulse enable
State 0: module-specific pulse disable

The display corresponds to MD 1700: TERMINAL_STATE (state of binary inputs).

PLC pulse enable

Indicates whether the pulse enable from the PLC is available for the drive.

State 1: The pulses for the drive module have been disabled by the PLC.
State 0: Pulse enable for this drive is active.

The display corresponds to IS “Pulse enable” (DB31, ... DBX21.7).
References: /FB1/, A2, “Various Interface Signals”

Speed controller enable NC

This display indicates whether the speed controller for the drive has been enabled by the NC.

State 1: Speed controller enable = OFF
State 0: Speed controller enable = ON

Ramp function generator rapid stop

Status display for ramp function generator rapid stop.

State 1: Ramp function generator rapid stop is not active for the drive.
State 0: Ramp function generator rapid stop is active. The drive is stopped without a ramp function with speed setpoint = 0 and without pulse suppression.

The display corresponds to IS “Ramp function generator rapid stop” (DB31, ... DBX92.1).
References: /FB1/, A2, “Various Interface Signals”

© Siemens AG, 2003. All rights reserved

DC link status (ON/OFF)

The display contains the following drive warning:

State 0: DC link voltage = ON
State 1: DC link voltage is below warning threshold.

The warning threshold corresponds to the value set in MD 1604: LINK_VOLTAGE_WARN_LIMIT (DC link undervoltage warning threshold).

Impulse enabled

Message indicating whether the drive pulses have been enabled.

State 0: The pulses have been disabled for the drive module. The axis/spindle cannot be traversed.
State 1: The pulses have been enabled for the drive module. The axis/spindle can be traversed.

The display corresponds to IS “Pulses enabled” (DB31, ... DBX93.7).
References: /FB1/**,** A2, “Various Interface Signals”

Drive ready

Display of the current status of the selected drive.

State 0: The drive is not ready.
State 1: The drive is ready.

The display corresponds to IS “Drive Ready” (DB31, ... DBX93.5).
References: /FB1/**,** A2, “Various Interface Signals”

Rampup phase

Display indicating the current rampup phase of the selected drive.

Significance:

XYZ
Internal coding

0	=	Software loaded in drive module
1	=	Basic initialization of drive module completed.
2, 3	=	Drive machine data are initialized.
4	=	Synchronization
5	=	Cyclic operation

CRC error

Display of communications errors between NC and drive detected in hardware.

Note

If the display shows a value other than “0”, please contact your SIEMENS Regional Office!

ZK1 messages

Display indicates whether messages of status class 1 are active.

State 0: No status class 1 message is active.
State 1: One or several status class 1 messages are active.

© Siemens AG, 2003. All rights reserved

Status class 1 messages are alarms with the following characteristics:

- They cause internal responses
  (e.g. regenerative braking, immediate pulse suppression)
- They are modal.

DC link voltage

Indicates the current DC link voltage level within the drive grouping.
Unit: Volts

Speed setpoint

The displayed speed setpoint represents the unfiltered total setpoint value. It is made up of the position controller output component and the speed feedforward branch.
Unit: U/min

The display corresponds to MD 1706: DESIRED_SPEED (speed setpoint).

Speed actual value

The actual value displayed represents the unfiltered actual speed value.
Unit: U/min

The display corresponds to MD 1707: ACTUAL_SPEED (actual speed value).

Smoothed actual current value

Display of the smoothed actual current value. The torque-generating current actual value is smoothed by a PT1 element with parameterizable time constant.
Unit: %
100 % corresponds to the maximum current of the power section.

The display corresponds to MD 1708: ACTUAL_CURRENT (smoothed actual current value).

Motor temperature

Display of motor temperature measured via temperature detectors.
Unit: Degrees Celsius

The display corresponds to MD 1702: MOTOR_TEMPERATURE (motor temperature).

Speed setpoint filter 1

Status display of speed setpoint smoothing function.

State 0: No speed setpoint smoothing is active.
State 1: Speed setpoint smoothing requested by the PLC with (DB31, ... DBX20.3) of the IS “Speed setpoint smoothing” is active because speed setpoint filter 1 is configured as a low pass filter.

The display corresponds to IS “Speed setpoint smoothing active” (DB31, ... DBX92.3).
References: /FB1/, A2, “Various Interface Signals”

2nd torque limit

Display of active torque limit

State 0: Torque limit 1 is active.
State 1: Torque limit 2 is active.

The display corresponds to IS “Torque limit 2 active” (DB31, ... DBX92.2).
References: /FB1/, A2, “Various Interface Signals”

Integrator disable

This display indicates whether the speed controller integrator is active.

© Siemens AG, 2003. All rights reserved

State 0: The integrator of the speed controller is enabled.
The speed controller functions as a PI controller.
State 1: Deactivation of the speed controller integrator requested by the PLC with IS “Speed controller integrator disabled” (DB 31, ... DBX 21.6) is active for the drive module. The speed controller has therefore switched from a PI to a P controller.

The display corresponds to IS “Speed controller integrator disabled” (DB31, ... DBX93.6).
References: /FB1/, A2, “Various Interface Signals”

Setup mode

Operating mode display of 611D:

State 0: Normal operation is active for the drive:
State 1: Setup mode is active for the drive.

The display corresponds to IS “Setup mode active” (DB31, ... DBX92.0).
References: /FB1/, A2, “Various Interface Signals”

Parking axis

Operating mode display of 611D:

State 0: Axis/spindle in normal mode
State 1: Axis/spindle in parking position, i.e. all encoder-specific monitoring functions and evaluations are disabled.
The encoder can be removed without triggering an alarm.

Setpoint parameter set (drive)

Indicates which of the 8 drive parameter sets of the 611D are to be activated by the PLC.

The display corresponds to IS “Parameter set selection A, B, C” (DB31, ... DBX21.0 to 21.2).
References: /FB1/, A2, “Various Interface Signals”

Actual parameter set (drive)

Display indicating which of the 8 drive parameter sets of the 611D is currently active.

The display corresponds to IS “Active parameter set A, B, C” (DB31, ... DBX93.0 to 93.2).
References: /FB1**/**, A2, “Various Interface Signals”

Operating mode

Display indicating whether the motor is operating as a feed drive or main spindle drive.

Motor selection (start/delta)

Display indicating which motor data set is to be activated by the PLC. At the moment the motor data record is used for the star/delta switchover on main spindle drives.

The following assignment applies:

Motor selection:	Application:	Coding:
Motor 1	MSD Star operation	0 0
Motor 2	MSD Delta operation	0 1
Motor 3	Reserved	1 0
Motor 4	Reserved	1 1

The display applies to MSD drives only. The display corresponds to IS “Motor selection A, B” (DB31, ... DBX21.3 to 21.4).
References: /FB1/, A2, “Various Interface Signals”

© Siemens AG, 2003. All rights reserved

Actual motor (star/delta)

Display indicating which of the motor data sets is currently active. At the moment the motor data record is used for the star/delta switchover on main spindle drives.

The following assignment applies:

Motor selection:	Application:	Coding:
Motor 1	MSD Star operation	0 0
Motor 2	MSD Delta operation	0 1
Motor 3	Reserved	1 0
Motor 4	Reserved	1 1

The display applies to MSD drives only.
The display corresponds to IS “Active motor A, B” (DB31, ... DBX93.3 to 93.4).
References: /FB1**/**, A2, “Various Interface Signals”

Position actual value measuring system 1/2

Actual position of the axis as measured by means of measuring system 1/2. The position is displayed in the machine coordinate system (no zero point offsets or tool offsets taken included).
Unit: mm, inch or degrees

Heatsink temperature warning

Warning signal output by drive:

State 0: The heatsink temperature monitoring has not responded.
State 1: The heatsink temperature monitoring has responded.

The display corresponds to IS “Heatsink temperature prewarning” (DB31, ... DBX94.1).
References: /FB1/, A2, “Various Interface Signals”

Motor temperature warning

Warning signal output by drive:

State 0: The motor temperature is below the warning threshold.
State 1: The motor temperature has exceeded the defined warning threshold.

The warning threshold corresponds to the value set in MD 1602: MOTOR_TEMP_WARN_LIMIT (maximum motor temperature).

The display corresponds to IS “Motor temperature warning” (DB31, ... DBX94.0).
References: /FB1/, A2, “Various Interface Signals”

Ramp-up function complete

Status display of drive.

State 0: The ramp-up function has not yet been completed after a new speed setpoint was defined.
State 1: The actual speed value has reached the speed tolerance band after a new speed setpoint was defined.

The speed tolerance band corresponds to the setting in MD 1426: SPEED_DES_EQ_ACT_TOL (tolerance band for ’$n_{set}n_{act}$’ signal).

The display corresponds to IS “Ramp-up function complete” (DB31, ... DBX94.2)
References: /FB1/, A2, “Various Interface Signals”

Torque lower than threshold setting

Status display of drive.

© Siemens AG, 2003. All rights reserved

State 0: In the stationary condition (i.e. ramp-up procedure completed), the torque setpoint is greater than the threshold torque.
State 1: In the stationary condition, the torque setpoint has not reached the threshold torque.

The threshold torque corresponds to the setting in MD 1428: TORQUE_THRESHOLD_X (threshold torque).

The display corresponds to IS "$M_d < M_{dx}$" (DB31, ... DBX94.3).
References: /FB1/, A2, "Various Interface Signals"

© Siemens AG, 2003. All rights reserved

Speed lower than minimum setting

Status display of drive.

State 0: The speed actual value is higher than the minimum speed.
State 1: The actual speed value is smaller than the minimum speed.

The minimum speed corresponds to the setting in MD 1418:
SPEED_THRESHOLD_MIN (n_{min} for '$n_{act} < n_{min}$' signal).

Display corresponds to IS "$|n_{act}| < n_{min}$" (DB31, ... DBX94.4).
References: /FB1/, A2, "Various Interface Signals"

Speed lower than threshold setting

Status display of drive.

State 0: The speed actual value is higher than the threshold speed.
State 1: The actual speed value is smaller than the threshold speed.

The threshold speed corresponds to the setting in MD 1417:
SPEED_THRESHOLD_X (n_x for '$n_{act} < n_x$' signal).

Display corresponds to IS "$|n_{act}| < n_x$" (DB31, ... DBX94.5).
References: /FB1/, A2, "Various Interface Signals"

Actual speed = set speed

Status display of drive.

State 0: The actual speed value is outside the speed tolerance band after a new speed setpoint was defined.
State 1: The actual speed value has reached the speed tolerance band after a new speed setpoint was defined.

The speed tolerance band corresponds to the setting in MD 1426:
SPEED_DES_EQ_ACT_TOL (tolerance band for '$n_{set}–n_{act}$' signal).

Display corresponds to IS "$|n_{act}| = n_{set}$" (DB31, ... DBX94.6).
References: /FB1/, A2, "Various Interface Signals"

Variable signal 1

Status display of variable signaling function of 611D.
The variable signal function monitors a memory location for violation of the defined threshold. In addition to the threshold, a tolerance band can be defined which is also taken into account when scanning for violation of the threshold value. The signal can be combined with an ON delay or OFF delay time.

State 0: Threshold value not reached
State 1: Threshold value exceeded
The variable signal function is parameterized in the following 611D machine data:

PROG_SIGNAL_FLAGS	(bits variable signal function)
PROG_SIGNAL_NR	(signal number variable signal function)
PROG_SIGNAL_ADDRESS	(address variable signal function)
PROG_SIGNAL_THRESHOLD	(threshold variable signal function)
PROG_SIGNAL_HYSTERESIS	(hysteresis variable signal function)
PROG_SIGNAL_ON_DELAY	(ON delay variable signal function)
PROG_SIGNAL_OFF_DELAY	(OFF delay variable signal function)

© Siemens AG, 2003. All rights reserved

Diagnostics for alarms

This information is also provided as a diagnostic tool for diagnosing the causes of alarms such as:

- 300300 Acceleration error
  ⇒ Check the **acceleration phase** to determine what stage of acceleration the drive has reached.
- 25201 “Drive fault”
  ⇒ **Message ZK1** is set.
  ⇒ Check **setpoint parameter set**, **motor selection, DC link status**.
  Check alarms 300500 to 300515.
- 25040 “Zero speed monitoring”
  25050 “Contour monitoring”,
  25060 “Speed setpoint limitation”
  25080 “Positioning monitor”
  ⇒ The drive enable signal(s) may be missing (e.g. **pulse enable**, **drive enable**, **pulse enable PLC** not active); this state will generate display **Pulses enabled** = OFF.
- 300614 “Motor temperature exceeded”
  ⇒ Check the current motor temperature.

Please refer to
References: /DA/, Diagnostics Guide

2.2.3 Service display PROFIBUS DP 840Di

General

The user interface 840Di StartUp provides diagnostic screen forms for PROFIBUS DP and its nodes. These diagnostic screens are only intended for information. You cannot modify them.

The following detailed information is displayed:

- PROFIBUS configuration
- Information on the slaves regarding their assignment to PLC/NC
- Detailed information on the slaves and the corresponding slots
- Information on the axes.

General notes for handling

To obtain a quick overview, the current states of certain functions are represented using colored lamps. The following general conventions are used for the meaning of the individual colors:

- Green: Function is OK.
- Red: Failure or no communication at the moment
- Gray: Function is not available for the present communication.

© Siemens AG, 2003. All rights reserved

Diagnostic screen PROFIBUS DP Configuration

Display of PROFIBUS Configuration

The diagnostic screen PROFIBUS Configuration provides general information on PROFIBUS DP.

The following parameters are displayed:

Table 2-1 Diagnostic screen PROFIBUS Configuration

Function/subfunction	Explanation/meaning
Bus configuration	
Baud rate in MBd	Data transfer rate
Cycle time in msec	Configured bus cycle time; also defines the position controller cycle
Synchronous portion (TDX) in msec	Configured time for cyclic data exchange within a PROFIBUS DP cycle
Status	
Configuration OK.	Status of configuration • Green lamp: DP master has powered up. • Red lamp: Failure or no communication.
Bus status	Current bus status is displayed in this field. Each bus status is explained in the screen form in brief. Possible states are: • POWER_ON • OFFLINE • CLEAR • OPERATE • Error

Diagnostic screen of the DP slaves

Information on the slaves

These diagnostic screen forms provide an overview of the DP slaves configured and detected on the bus.

The following information is provided:

© Siemens AG, 2003. All rights reserved

Table 2-2 Diagnostic screen Information on the slaves

Function/subfunction	Explanation/meaning
Slave No. [DP address]	Configured DP address of DP slave
Assignment	It is displayed whether the DP slave is assigned to the NC or to the PLC. • NC: NC: e.g. one or several drives controlled by the NC. • PLC: e.g. I/O modules or an axis controlled by the PLC.
Active on the bus	Displays whether the DP slave has been detected on the bus • Green lamp: DP slave has been detected on PROFIBUS DP and the data exchange with the assigned component (NC or PLC) operates. • Red lamp: Failure or no communication.
Sync. with NC	Displays whether DP slave operates on the bus synchronously to the NC. • Green lamp: DP slave operates on PROFIBUS DP synchronously to the NC, i.e. the equidistant data exchange takes place. • Gray lamp: DP slave is not assigned to the NC, but to the PLC. • Red lamp: Failure or no communication.
Number of slots	Number of configured slots within DP slave
Details	Pressing this button will open another diagnostic screen that provides detailed information on the corresponding DP slave.

Detailed information of the slots within a slave

Detailed information on the slots

The button **Details** opens the dialog box Detailed information on the slave. This screen form provides detailed information on the slots assigned to the DP slave.

In addition, the Slave dialog box display important information on the DP slave currently selected.

The following information is displayed for the slots:

Table 2-3 Diagnostic screen Detailed information on the slave

Function/subfunction	Explanation/meaning
Slave	
Slave No. [DP address]	Configured DP address of DP slave

© Siemens AG, 2003. All rights reserved

Table 2-3 Diagnostic screen Detailed information on the slave

Function/subfunction	Explanation/meaning
Assignment	Displays whether the DP slave is assigned to the NC or to the PLC. • NC: e.g. one or several drives controlled by the NC. • PLC: e.g. I/O modules or an axis controlled by the PLC.
Active on the bus	Displays whether the DP slave has been detected on the bus • Green lamp: DP slave has been detected on PROFIBUS DP and the data exchange with the assigned component (NC or PLC) operates. • Red lamp: Failure or no communication.
Synchr.	Displays whether DP slave operates on the bus synchronously to the NC. • Green lamp: DP slave operates on PROFIBUS DP synchronously to the NC, i.e. the equidistant data exchange takes place. • Gray lamp: DP slave is not assigned to the NC, but to the PLC. • Red lamp: Failure or no communication.
Slots	
No.	Slot number within DP slave
I/O address	I/O address in the I/O address space of the PLC assigned to this slot. For NC axes, setpoint and actual value must always be configured using the same I/O address.
Logical drive no.	Drive number assigned for the appropriate axis in the NC machine data.
Length [bytes]	Length of the I/O area in the STEP7 I/O address space, which is reserved for the slot
Type	Specification whether the slot is an input, output or diagnostic slot. If the slot is assigned to an NC axis, an output is always designated as a setpoint and an input always as an actual value.
Machine axes	Display of the name defined for this slot in the machine data. If the slot is not assigned to any NC axis, <No NC axis> is displayed.
Message frame type	Message frame type configured in the NC machine data. If the slot is not assigned to any NC axis, the message frame type will not be occupied (–).
Status	Current slot status is only displayed for NC axes • Green: Slot is used by the NC; communication active • Gray: No NC axis • Red: Slot is used by the NC; communication currently not active

© Siemens AG, 2003. All rights reserved

Diagnostic screen for the axes

Diagnostic screen AxisInfo

The diagnostic screen AxisInfo displays axisspecific detailed information. The diagnostic screen provides an NCoriented view of the axis information.

The following information is displayed for the axes:

Table 2-4 Diagnostic screen AxisInfo

Function/subfunction	Explanation/meaning
Machine axes	Name of the axis defined in the NC machine data
Output	
Slave/slot	Configured assignment
Status	Current slot status • Green lamp: Cyclic communication • Red lamp: (Not yet) cyclic communication
Message frame failures	The display shows the number of message frame failures since the NC was booted. This value indicates the quality (fault susceptibility) of the PROFIBUS DP line.
Encoders 1	
Slave/slot	Configurable assignment
Status	Current slot status • Green lamp: Cyclic communication • Red lamp: (Not yet) cyclic communication
Message frame failures	The display shows the number of message frame failures since the NC was booted. This value indicates the quality (fault susceptibility) of the PROFIBUS DP line.
Type	Display of encoder type configured in the NC machine data • ABS: Absolute value encoder • INC: Incrementalvalue encoder
Encoders 2	(If configured the same display as with encoder 1)

2.2.4 Communication log

The communications which have occurred between the MMC and NC are displayed in chronological order under soft key **Comm. log** in the “Diagnostics” operating area. This error list is provided to assist developers of OEM applications in localizing sporadic errors. The list has no relevance for normal operation.

© Siemens AG, 2003. All rights reserved

2.2.5 Log book

The Log book display selected by means of soft key **Log book** in the “Diagnostics” operating area automatically lists details of all alterations to the control that are relevant for the system (e.g. changes in access level).

With SINUMERIK 840Di, the log book is displayed in 840Di StartUp.

2.2.6 Version

The version of the MMC or NC software installed can be read from this display (“Diagnostics” operating area under soft key **Version**) by service personnel. The software versions of all software modules installed are also listed.

© Siemens AG, 2003. All rights reserved

2.3 PLC status

PLC status signals can be checked and altered via the operator panel in operating area “Diagnostics”.

Application

The end customer or service personnel can use this function on site without a programming device to do the following:

- Check the input and output signals of the PLC I/Os.
- Carry out limited troubleshooting
- Check the interface signals for diagnostic purposes.

Operator actions

For how to operate the service displays and alter them see:
References: /BA/, “Operator’s Guide”

Status display

The status of the following data can be displayed on the operator panel.

- Interface signals from the machine control panel
- Interface signals to the machine control panel
- Interface signals between the NCK and PLC
- Interface signals between the MMC and PLC
- Data blocks (DB 0–127)
- Flags (FB 0–255)
- Timers (T 0–127)
- Counters (C 0–63)
- Inputs (IB 0–127)
- Outputs (QB 0–127)

Distribution of interface signals (DBx, DBBy) see:
References: /IAD/, “Installation and StartUp Guide”
/IAF/, “Installation and StartUp Guide”

Change in status

The status of the above signals can be changed for test purposes. Signal combinations are also possible. A maximum of ten operands can be altered at any one time.

© Siemens AG, 2003. All rights reserved

2.4 Other diagnostics tools

611D startup tool

One of the functions of this program is to provide a tool

- for assessing the major values in position, speed and current control
- for archiving drive and control data, and
- for analyzing the given mechanical properties.

For handling and complete range of functions see:
References: /IAD/, “Installation and StartUp Guide”

Archiving of data

The PCIN software package can be used to archive machine data, setting data, parts programs, etc.
Handling is described in the relevant documentation.
References: /PI/ PCIN 4.3

840Di StartUp

For diagnosing the SINUMERIK 840Di, the WINDOWS program 840Di StartUp can be used. This provides information, e.g. on the current operating mode and the nodes of PROFIBUS DP.

© Siemens AG, 2003. All rights reserved

2.5 Identification defective drive modules (SW 6.3 and higher)

Function

Troubleshooting may involve a situation where a drive module (SIMODRIVE 611 digital) displayed in an alarm text needs to be removed from the bus to prove that it is precisely this module that has caused the displayed error.

Using MD 13030 DRIVE_MODULE_TYPE it is possible to remove individual modules from the drive bus configuration at the NC end (the affected axes are switched to Simulation mode).

Note

You must **remove** the defective module from the drive bus configuration (SIMODRIVE 611 digital) **before** you activate the function. To do this, connect up the drive bus excluding the relevant module.

Since this internal modification to the machine configuration can result in damage to the machine if implemented incorrectly, the axes are prevented from moving.

If Safety Integrated has been activated for the modules concerned, you must disable it manually (safety, logged, EMERGENCY STOP scheme).

Remove drive module on NC end

A drive module (SIMODRIVE 611 digital) specified in an alarm text must be removed from the bus:

1. Remove the module from the drive bus network
2. Set entries for drive module to zero in MD 13030 DRIVE_MODULE_TYPE (zeroaxis module).
3. Perform an NC RESET.

The axes which were controlled by the removed drive modules are now replaced by simulation axes. The 611D bus with its drive modules is now in a state in which it could normally move axes, but axis traversal has been disabled internally.
Alarm 300020 “Drive %1 removed for diagnostics” is displayed to indicate this status.

Restore initial configuration

Following diagnostics, the initial configuration on the drive bus must be restored:

1. Replace or re-install the removed drive module.
2. Reenter the original values for the settings of the drive module in MD 13030 DRIVE_MODULE_TYPE.
3. Perform an NC RESET.

© Siemens AG, 2003. All rights reserved

Example

The 2-axis module with drive numbers “1” and “2” must be removed from a drive grouping.

Note

You must **remove** the defective module from the drive bus configuration (SIMODRIVE 611 digital) **before** you activate the function. To do this, connect up the drive bus excluding the relevant module.

If Safety Integrated has been activated for the modules concerned, you must disable it manually (safety, logged, EMERGENCY STOP scheme).

Table 2-5 Bus configuration example

Module	Drive no.	Active	Type	Module type	Power section code
1	10	1	ARM/MSD	1-axis	6
2 left	1	1	SRM/FDD	2-axis	14
2 right	2	1	SRM/FDD	2-axis	14
3 left	4	1	HLA	2-axis	
3 right	5	1	ANA	2-axis	
4	12	1	SLM	1-axis	11
5	11	1	I/O	DMPC	

Module “2” must now be removed:

- Select MD 13030 DRIVE_MODULE_TYPE in MD display “General MD”.
- DRIVE_MODULE_TYPE[0] = 1
  DRIVE_MODULE_TYPE[1] = 2 <– set this to zero
  DRIVE_MODULE_TYPE[2] = 2 <– set this to zero
  DRIVE_MODULE_TYPE[3] = 2
  DRIVE_MODULE_TYPE[4] = 2
  DRIVE_MODULE_TYPE[5] = 1
  DRIVE_MODULE_TYPE[6] = 9
- After modification, the table looks like this:
  DRIVE_MODULE_TYPE[0] = 1
  DRIVE_MODULE_TYPE[1] = 0
  DRIVE_MODULE_TYPE[2] = 0
  DRIVE_MODULE_TYPE[3] = 2
  DRIVE_MODULE_TYPE[4] = 2
  DRIVE_MODULE_TYPE[5] = 1
  DRIVE_MODULE_TYPE[6] = 9
- Alarms 300020 “Drive 1 removed for diagnostics” and 300020 “Drive 2 removed for diagnostics” are displayed.

Internally simulated drives are used for all axes which contain settings to the removed drive numbers. If the closed-loop control is activated for the drives that are still installed, these drives operate in the normal way. Interpolative traversal of all axes is disabled.

© Siemens AG, 2003. All rights reserved

Note

If alarm 300003 “Axis xx drive yy incorrect module type zz” appears, then you have removed only one part of a 2-axis module. In this case, you should check the module type in the drive configuration display. “NO” axis type is shown for removed modules.

■

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

Identification of defective drive modules

The “Identification of defective drive modules” function is available with SW 6.3 and higher for SINUMERIK 840D/810D.

■

4 Data Descriptions (MD, SD)

4.1 General machine data

11410 MD number	**SUPPRESS_ALARM_MASK** Screen form for suppressing special alarm outputs		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 2		

© Siemens AG, 2003. All rights reserved

11410 MD number	**SUPPRESS_ALARM_MASK** Screen form for suppressing special alarm outputs
Significance:	Screen form for suppressing special alarm outputs Bit set: The corresponding alarm (warning) is NOT generated. Bit 0: Alarm 15110 “Channel %1 block %2 REORG not possible” Bit 1: Alarm 10763 “Channel %1 block %2 The path component of the block in the contour plane becomes zero” Bit 2: Alarm 16924 “Channel %1 Caution: Program testing can modify tool/magazine data”. The alarm is only a message. Bit 3: Alarm 22010 “Channel %1 spindle %2 block %3 actual gear stage does not match the set gear stage”. The alarm is only a message. Bit 4: Alarm 17188 “Channel %1 D number %2 for tool T No. %3 and %4 defined” 17189 Channel %1 D number %2 of the tools in magazine/position %3 and %4 defined Both alarms are of equal significance and are messages only. Bit 5: Alarm 22071 “TO unit %1 tool %2 duplo no. %3 is active, but not in the active wear group.” The alarm is only a message. Bit6: Alarm 4027 “Attention: MD %1 was also changed for the other axes of the axis container %2” Alarm 4028 “Attention: At the next ramp-up the axial MD in the axis container will be amended” Bit7: Alarm 22070 “TO unit %1 Please change tool T= %2 into the magazine. Repeat data backup”. The alarm is only a message. Bit8: Alarm 6411 “Channel %1 tool %2 with duplo no. %3 has reached the tool warning limit” 6413 “Channel %1 tool %2 with duplo no. %3 has reached the tool monitoring limit”. Both alarms are only messages. They arise from the execution of the program. Bit9: Alarm 6410 “TO unit %1 tool %2 with duplo no. %3 has reached the tool warning limit”. 6412 “Channel %1 tool %2 with duplo no. %3 has reached the tool monitoring limit”. Both alarms are only messages. They occur following an operator action. Bit10: Alarm 10604 “Channel %1 Block %2 Thread lead increase too high. 10605 “Channel %1 Block %2 Thread lead decrease too high. Bit11: Alarm 14088 “Channel 1% Block %2 Axis %3 in unexpected position”. Bit12: Alarm 10607 “Channel %1 Block %2 Thread with frame cannot be executed. Bit14: Alarm 21701 “Channel %1 Block %3 Axis %2 Measuring not possible. Bit16: Alarm 21600 Monitoring for ESR active
Related to	
References	

© Siemens AG, 2003. All rights reserved

11411 MD number	**ENABLE_ALARM_MASK** Display special alarms		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	Mask for generating alarms that are normally suppressed. Bit set: Alarms of this alarm group are output. Bit not set: Alarms of this alarm group are **not** output. Bit 0: Alarms which have the response SHOWALARMAUTO are output. (Alarm 16905) The alarm response is to be set when an alarm is to be output only in Automatic mode without a manual operator action. Bit 1: Alarms which have the response SHOWWARNING are output. The response is used for warnings that are normally suppressed. SW 5.1 and higher: Bit 2: Technological alarm 22280 “Thread ramp-up distance too short” is output. SW 5.2 and higher: Bit 3: Alarms triggered by the NCU-LINK MODULE are bit 2 = 1.		
Related to			
References			

11412 MD number	**ALARM_REACTION_CHAN_NOREADY** Alarm reaction CHAN_NOREADY permitted		
Default setting:	Minimum input limit:	Maximum input limit:	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 4.1		
Significance:	If this MD is not set, then BAG_NOREADY is executed instead of CHAN_NOREADY→ With SW 4.1 and higher, it is possible to set CHANNEL_NOREADY on the PLC in response to alarms. If this MD is not set, then the alarm handler internally reconfigures CHAN_NOREADY into BAG_NOREADY. The purpose of this MD is solely to provide compatibility with PLC systems with versions lower than SW 4.1.		
Related to			

11413 MD number	**ALARM_PAR_DISPLAY_TEXT** Texts as alarm parameters		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 0 / 0	Unit: –	
Data type: BOOLEAN	Applies from SW 3		
Significance:	If the MD is set, texts can be output as alarm parameters instead of numbers.		

11420 MD number	**LEN_PROTOCOL_FILEX** File size for protocol files (KB)		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 1 000 000	
Change effective after RESET	Protection level: 7 / 2	Unit: KB	
Data type: DWORD	Applies from SW 4.3		
Significance:	The WRITE command can be used to store blocks from the parts program in a file. The length of the protocol file is limited. An alarm is output.		

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Signal Descriptions 5

None

■

Example 6

– None –

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

7 Data Fields, Lists

7.1 Interface signals

DB number	Bit, byte	Name	Refer-ence
Axis/spindle-specific		**Signals from PLC to axis/spindle**	
31, ...	16.0 to 16.2	Actual gear step	S1
31, ...	21.0 to 21.2	Parameter set selection A, B, C	A2
31, ...	21.3 to 21.4	Motor selection A, B	A2
31, ...	21.7	Pulse enable	A2
		Signals from axis/spindle to PLC	
31, ...	60.4	Referenced/synchronized 1	R1
31, ...	60.5	Referenced/synchronized 2	R1
31, ...	62.5	Fixed stop reached	F1
31, ...	92.0	Setup mode active	A2
31, ...	92.1	Ramp function generator rapid stop	A2
31, ...	92.2	Torque limit 2 active	A2
31, ...	92.3	Speed setpoint smoothing active	A2
31, ...	93.0 to 93.2	Active parameter set A, B, C	A2
31, ...	93.3 to 93.4	Active motor A, B	A2
31, ...	93.5	Drive Ready	A2
31, ...	93.6	Integrator n-controller disabled	A2
31, ...	93.7	Pulses enabled	A2
31, ...	94.0	Motor temperature warning	A2
31, ...	94.1	Heatsink temperature prewarning	A2
31, ...	94.2	Ramp-up function complete	A2
31, ...	94.3	$\|M_d\| < M_{dx}$	A2
31, ...	94.4	$\|n_{act}\| < n_{min}$	A2
31, ...	94.5	$\|n_{act}\| < n_x$	A2
31, ...	94.6	$\|n_{act}\| = n_{set}$	A2
31, ...	16.0 to 16.2	Actual gear step	S1
31, ...	21.0 to 21.2	Parameter set selection A, B, C	A2
31, ...	21.3 to 21.4	Motor selection A, B	A2
31, ...	21.7	Pulse enable	A2

© Siemens AG, 2003. All rights reserved

7.2 Machine data

Number	Identifier	Name	Refer-ence
General ($MN_ ...)			
11410	SUPPRESS_ALARM_MASK	Mask for suppressing special alarms	
11411	ENABLE_ALARM_MASK	Activation of special alarms	
11412	ALARM_REACTION_CHAN_NOREADY	Alarm reaction CHAN_NOREADY permitted	
11413	ALARM_PAR_DISPLAY_TEXT	Texts as alarm parameters (Siemens Rights)	
11420	LEN_PROTOCOL_FILEX	File size for protocol files (KB)	
13030	DRIVE_MODULE_TYPE	Module identifier (SIMODRIVE 611 digital)	G2
Axis-specific ($MA_ ...)			
32200	POSCTRL_GAIN [n]	Servo gain factor	G2
32250	RATED_OUTVAL	Rated output voltage	G2
32260	RATED_VELO	Rated motor speed	G2
36010	STOP_LIMIT_FINE	Exact stop fine	B1
36030	STANDSTILL_POS_TOL	Zero speed tolerance	A3
36050	CLAMP_POS_TOL	Clamping tolerance with interface signal “Clamping active”	A3
36210	CTRLOUT_LIMIT	Maximum speed setpoint	G2
36300	ENC_FREQ_LIMIT[n]	Encoder limit frequency	A3
36400	CONTOUR_TOL	Tolerance band contour monitoring	A3
36500	ENC_CHANGE_TOL	Max. tolerance for position actual value acquisition	G2
37010	FIXED_STOP_TORQUE_DEF	Clamping torque	F1
43510	FIXED_STOP_TORQUE	Fixed stop clamping torque	F1
Drive machine data ($MD_ ...)			
1401	MOTOR_MAX_SPEED	Speed for max. useful motor speed	/IAD/
1417	SPEED_THRESHOLD_X	n_x for ‘$n_{act} < n_x$’ signal	/IAD/
1418	SPEED_THRESHOLD_MIN	n_{min} for ‘$n_{act} < n_{min}$’ signal	/IAD/
1426	SPEED_DES_EQ_ACT_TOL	Tolerance band for ’n_{set}–n_{act}’ signal	/IAD/
1428	TORQUE_THRESHOLD_X	Threshold torque	/IAD/
1602	MOTOR_TEMP_WARN_LIMIT	Maximum motor temperature	/IAD/
1604	LINK_VOLTAGE_WARN_LIMIT	DC link under voltage warning threshold	/IAD/
1620	PROG_SIGNAL_FLAGS	Bits variable signal function	/IAD/
1621	PROG_SIGNAL_NR	Signal number variable signal function	/IAD/
1622	PROG_SIGNAL_ADDRESS	Address variable signal function	/IAD/
1623	PROG_SIGNAL_THRESHOLD	Threshold variable signal function	/IAD/
1624	PROG_SIGNAL_HYSTERESIS	Hysteresis variable signal function	/IAD/
1625	PROG_SIGNAL_ON_DELAY	ON Delay variable signal function	/IAD/
1626	PROG_SIGNAL_OFF_DELAY	OFF Delay variable signal function	/IAD/
1700	TERMINAL_STATE	Status of binary inputs	/IAD/
1702	MOTOR_TEMPERATURE	Motor temperature	/IAD/
1706	DESIRED_SPEED	Speed setpoint	/IAD/

© Siemens AG, 2003. All rights reserved

32200	POSCTRL_GAIN [n]	Servo gain factor	G2
1707	ACTUAL_SPEED	Speed actual value	/IAD/
1708	ACTUAL_CURRENT	Smoothed current actual value	/IAD/

7.3 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Interactive Programming (D2)

1 Brief Description 1/D2/1-3

2 Detailed Description 1/D2/2-5

2.1 General, applications 1/D2/2-5
2.2 Configuring the user interface 1/D2/2-8
2.3 Adapting the technology memory 1/D2/2-10
2.4 Creation of a tool catalog 1/D2/2-11
2.5 Setting up interactive programming 1/D2/2-12
2.5.1 Selecting the milling or turning machining technology 1/D2/2-12
2.5.2 Adaptation of axis identifiers 1/D2/2-13
2.5.3 Adaptation of simulation data up to SW 3.1 ..... 1/D2/2-14
2.5.4 Adaptation of simulation data with SW 3.2 and higher 1/D2/2-18
2.5.5 Special conditions for adapting simulation data 1/D2/2-22
2.5.6 Matching the tool offset data 1/D2/2-27
2.5.7 Modifications with SW 3.6 and higher 1/D2/2-29
2.6 Archiving of configuration data 1/D2/2-31

3 Supplementary Conditions 1/D2/3-33

4 Data Descriptions (MD, SD) 1/D2/5-35

5 Signal Descriptions 1/D2/5-35

6 Example 1/D2/6-37

7 Data Fields, Lists 1/D2/7-39

7.1 Machine data 1/D2/7-39
7.1.1 General machine data 1/D2/7-39
7.1.2 Channel-specific machine data 1/D2/7-39
7.1.3 Axis-specific machine data 1/D2/7-40

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

1 Brief Description

The “Interactive Programming” function can be used to create and process

1. parts programs and
2. program data.

Integrated help features and testing tools support the programmer.

- The system supports the direct input of CNC code in conformity with DIN 66025 and the processing of DIN 66025 programs which have been created externally.
- An ASCII editor is included for the editing of programs and data created with “interactive programming”.
- The user interface, the technological parameters and the integrated tool catalog can be customized.

Basic programming

Programs are created in a series of steps which are generated by parameter inputs in interactive screen forms after selection of one of the eight dialog levels.

Note

A program step can contain up to 20KB of ASCII characters.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 General, applications

Applications

The Interactive programming function allows machine operators who have no in-depth knowledge of the NC programming code to

- create,
- edit and
- test (in a simulated machining environment under realistic conditions) parts programs.

In the supplied state, the dialogs, and particularly the graphical machining simulation, are intended for:

- 2-axis turning machines (axes X, Z)
- 3-axis milling machines (axes X, Y, Z)
  with working planes

  X–Y, infeed in Z (G17)
  X–Z, infeed in Y (G18)
  Y–Z, infeed in X (G19)

The “free input” dialog level enables all the language elements of CNC code described in the Programming Guide to be used at any time in order to program the current workpiece.

If the programmer frequently uses language elements which are not contained in the standard dialogs (e.g. Transmit, 5-axis machining), these can also be included in the dialog by configuring “user displays” (see Section 2.2).

Note

Assignments to turning or milling machines must be made via initial settings “DPTURN” or “DPMILL” before the MMC 102/103 runs up and the “Interactive programming” function is called (see Subsection 2.5.1).

References: /PG/, “Programming Guide: Fundamentals”

© Siemens AG, 2003. All rights reserved

Geometry model

The geometry model of the graphical machining simulation is generally oriented to the representation of prismatic parts without inclined orientation for milling, and rotationally symmetrical parts without eccentric drilling and milling operations for turning.

Program interpreter

The program interpreter of the machining simulation function in the MMC 102/103 is based on the porting of **one** NC channel with an appropriate data environment.

Note

Components of the parts program which operate outside this framework (such as program coordination commands INIT, START, WAIT, or machine kinematics which are not configured in the ported data environment) can cause error states in the simulation, even though no errors are encountered during program execution in the NC area.

Such program parts can be skipped during simulation in SW 3.2 and higher (by evaluating the system variable $P_SIM in the program), without restricting the machining sequence on the machine tool:

```
PROGRAM_START
:
:
IF $P_SIM GOTOF dest_label
... cannot be interpreted for simulation
... program section
dest_label:
:
:
PROGRAM_END
```

Program code

In addition to the DIN 66025 compliant code with high-level language extensions, the program code generated by “Interactive Programming” also contains internal structure identifiers which are required for the management of the program steps.

ASCII editor

Programs which have been created with “interactive programming” **should** be edited within the framework of “interactive programming” or, provided the internal program structure is not modified and only values are altered, with the ASCII editor.
The ASCII editor and “interactive programming” are designed for interoperation insofar as they use the same data management system.

Other programming tools

Programs which have been modified with other (external) programming tools can be further edited with “interactive programming”, provided the internal program structure is still intact.

© Siemens AG, 2003. All rights reserved

General conditions

The “Interactive programming” function is implemented in the MMC 102/103 and can be used in conjunction with this component in the

- SINUMERIK 840D
- SINUMERIK 810D
- SINUMERIK FMNC

Restrictions

1. In the standard version of “interactive programming”, the dialogs are oriented to “80 % probable use” for milling, drilling and turning operations.
2. Special technological functions can be configured additionally at any time if required.
3. Language instructions used in conjunction with function G33 (thread cutting) are executed correctly during machining simulation with SW 3.2 and higher.

Compatibility

Parts programs which already exist in system 800 controls (from SINUMERIK 805 to 840C) cannot be processed further in the “Interactive programming” environment until they have been converted with a program converter (@, cycle interfaces ...).
It is not recommended to load system 800 programs directly and make subsequent modifications in the MMC area with the ASCII editor or “interactive programming”.

A program template for the conversion of system 800 programs is available from Siemens AG, AUT 2.

© Siemens AG, 2003. All rights reserved

2.2 Configuring the user interface

Adapting the help screens

After selecting a help screen with the Information key and double-clicking any point in the graphics area with a mouse, the existing or initially empty graphic can be edited with the PAINTBRUSH graphics editor from the Windows 3.x platform.

After the changes have been saved, the edited graphic is not activated until the help screen is called up again.

The cursor for help texts is always ready for input in the text field.

The default help text contains the name of the associated window, and is saved together with the user extension (user notepad).

The default settings are called up again when all characters, including spaces, are deleted from the text field.

Adapting the basic info

The graphics and text of the basic information for the dialog levels can edited separately, in the same way, by double-clicking the mouse in the graphics area or in the header.

Creation of user displays

The user can create his own list displays for the following 5 dialog levels (horizontal soft keys). He can then call them using vertical soft key “User” on the corresponding dialog level.

1. Program section
2. Path/contour
3. Continuous-path control
4. Tools and material
5. Machining

The list screens are created in the menu path

“Display Mode / User Displays”.

References: /BA/, “Operator’s Guide”

This enables the necessary parameters to be entered and called up for user-specific programs (e.g. cycles), in the same way as the standard cycle dialogs.

CNC code can also be generated directly from the parameters for the user display (e.g. user display for the generation of the necessary language elements for a machine transformation).

An initially empty help screen is assigned automatically to each user display.

© Siemens AG, 2003. All rights reserved

Note

- A **mouse** is always required for the configuration of graphics.
- The graphics data for the help screens must be stored in bitmap format (*.bmp) in the system path **C:\dh\dp.dir\hlp.dir**.
- Special care should be taken after execution of the PAINTBRUSH operation “Edit/Insert From...” that the target directory matches the required system path before the modified graphics file is changed (“Save File As...”).
- The configuration of the user interface is subject to the general multilevel **access protection**. The password should be set before the configuration process.

© Siemens AG, 2003. All rights reserved

2.3 Adapting the technology memory

Structure of the technology memory

For each of the access keys

"Material\Cutting Material\Machining Quality\Machining Type"

it is possible to store one technology data block (empirical values), each with a maximum of ten parameters.

Material memory

The materials used in "interactive programming" must be defined before being used in the technology memory. In the menu path

"Tools and Material\Enter Material"

the appropriate data can be entered and saved in the menu path.

Cutter material memory

Before they are used in "Interactive Programming", the cutter materials of the available tools must also be stored in the menu path

"Tools and Material\Enter Cutting Material".

Technological suggestion values

In the menu path

"Tools and Material\Enter Technology"

for each of the access keys technological suggestion values can be initialized by the appropriate inputs.

The suggested values are not calculation results from "interactive programming", but are empirical values of the machine manufacturer or end user.

Depending on the current machining type and the tool used, the appropriate suggestion values can be called up again in order to initialize the technology-dependent dialog screens (path screens and cycle displays).

Note

- Sets of technological data can not be edited. They must first be deleted and then entered again.
- Only the "Sequence No." is significant for the management of workpiece and cutting materials.
- "Classes" and "Keywords" are verbal differentiation criteria and cannot be used for sorting.
- As with standard dialogs, up to ten parameters can be linked to the technology memory during the configuration of user displays.
- In SW 3.2 and higher, the technological memory is subject to the general access protection.

© Siemens AG, 2003. All rights reserved

2.4 Creation of a tool catalog

Structure of the tool catalog

Two sets of data are basically available for each tool:

1. “Tool master data”,
2. “Tool operating data”.

These can be customized.

Tool master data

The “Tool master data” include:

- Data for tool geometry and kinematics
- A subset of data which can be mapped onto TOA data.

Tool operating data

The “Tool operating data” relevant for a tool change:

- Active cutting edge on tool call
- Cutting material of the tool for the technology access
- Tool location number in the magazine
- Preselection of radius compensation
- Preselection of spindle motion

can be initialized in the tool change screen, in the same way as the suggested values in the technology memory, so that they are subsequently available for the use of the tool catalog.

Creating the tool catalog

For the creation of the tool catalog, reference tools are available to the user in the menu path

“Tools and material\Tool entry”

as a starting point for the customization of the master and operating data of the tools to be included in the tool catalog.

Using the tool catalog

In the menu path

“Tools and material\Tool change”

the selection of a tool from the tool catalog causes the tool change screen to be initialized automatically with the previously presented tool operating data.

Note

- In the supplied state, the tool catalog only contains the master data of the reference tools.
- When using a tool management system (option), the master data of the tool catalog are matched to the data of the tool management system by using the same database.
- In SW 3.2 and higher, machining simulation access to the tool master data is initialized when a tool is called from the tool catalog.

© Siemens AG, 2003. All rights reserved

2.5 Setting up interactive programming

The following steps must be taken when setting the “interactive programming” on the MMC 102/103:

1. Select milling or turning machining technology
2. Adaptation of axis identifiers
3. Adapt simulation data up to SW 3.1 or with SW 3.2 and higher
4. Special conditions for adapting simulation data

2.5.1 Selecting the milling or turning machining technology

Assuming that at least the manufacturer password is active, the machine manufacturer and user can perform the following actions, in order to assign “interactive programming” to a turning or milling machine:

Operator actions



Press the user area key,



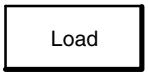
switch to startup mode,



press the “MMC” soft key, followed by



the “Editor” key and then



the “Load” key, to display the file overview.



Press the “END” key twice, to call up the file selection window.





Use the direction keys to look in “C:\MMC2” for the “REGIE.INI” file



and open the file with “OK”.



Under the “TaskConfiguration” section you will find:

Task2=name:=dpmill, ... (for milling machine)
;Task2=name:=dpturn, ... (for turning machine)

© Siemens AG, 2003. All rights reserved

Now place the semicolon (;) in front of the task which is **not to be active**, and delete the semicolon from in front of the task which is to be activated (in this case: Milling version active).

Save

Save the modified file



and press “Recall” to return to startup mode.

When the MMC 102/103 is powered up again, “interactive programming” is assigned to the desired machining technology.

Note

In the supplied state, the “milling” machining technology is active.
In SW 3.2 and higher, the task name for the milling version is “DP” (instead of “DPMILL”).

2.5.2 Adaptation of axis identifiers

“Interactive programming” allows the axis identifiers to be adapted flexibly to the geometry axes and the associated machine and interpolation axes.

The defined identifiers are included in both the CNC code generation and the CNC code analysis (step modifications).

Note

Automatic matching does not take place.

To ensure that the generated CNC code is executable on the target machine (NCK or simulation interpreter), the axis identifiers of “interactive programming” should be matched as follows to the real axis identifiers (NCK machine data):

Operator actions

After opening a program,

Display Mode

select the “Display Mode” dialog level,

Default Settings

activate the “Default Settings” dialog screen,



and use the “Page Down” key to select the second page of the screen.

© Siemens AG, 2003. All rights reserved

A ••• Z

0 ; ••• 9 :

You can now adapt the axis identifiers to your needs.

OK

The settings are saved with OK,

and included in the CNC code generation for the current workpiece from this point on.

Note

In SW 3.2 and higher, this setting can be used for all new workpieces (prompt window).

2.5.3 Adaptation of simulation data up to SW 3.1

As already described in Section 2.1, the machining simulation on the MMC-102/103 can be run with a separate set of machine data by means of a pseudochannel.

This set of machine data for “interactive programming” should be matched to the real set of machine data such that a realistic, errorfree simulation can be performed on standard parts programs.

Program sequences (such as channel coordination commands) which cannot be interpreted cannot be skipped in SW 3.1, and cause alarm messages to be output by the simulation interpreter.

Overview of the sets of data up to and including SW 3.1:

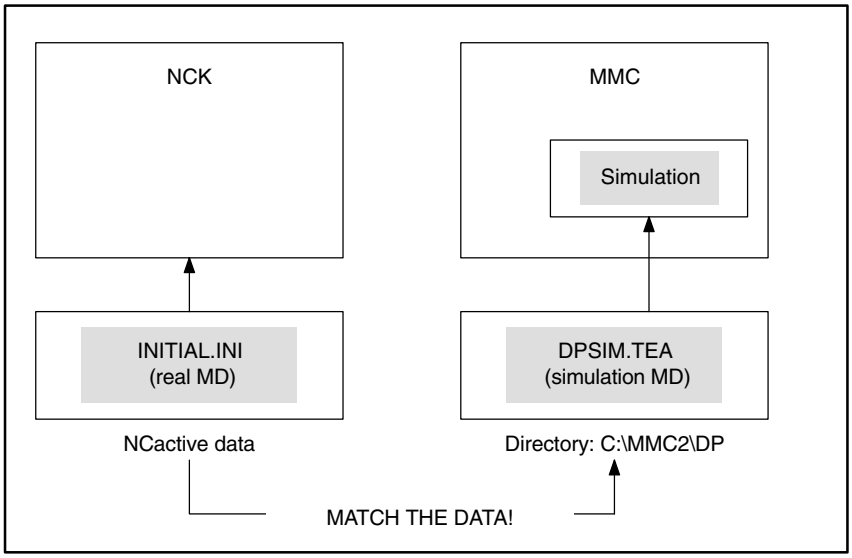


Fig. 2-1 Overview of the sets of data up to and including SW 3.1

© Siemens AG, 2003. All rights reserved

In the supplied state, the set of simulation data in DPSIM.TEA is oriented to the milling version.

For the “turning” technology, you should open this file and **make turning-specific changes** (XZ kinematics, working plane G18, diameter programming, etc.).

For the standard configuration adaptation, you only need to delete the semicolon in the following turning-specific machine data:

```
$MC_GCODE_RESET_VALUES[5]=2 ;BHf from turn.md ... G18
:
:
$MC_GCODE_RESET_VALUES[28]=2 ;BH from turn.md diamprog (DIAMON)
```

Note

The same conditions that apply to the adaptation of NCK machine data should be observed in all modifications of the simulation machine data in the DPSIM.TEA file. See

References: /IAD/, “Installation and StartUp Guide” SW 2

The flow chart in Fig. 2-2 represents the procedure for matching the machine data up to and including SW 3.1:

© Siemens AG, 2003. All rights reserved

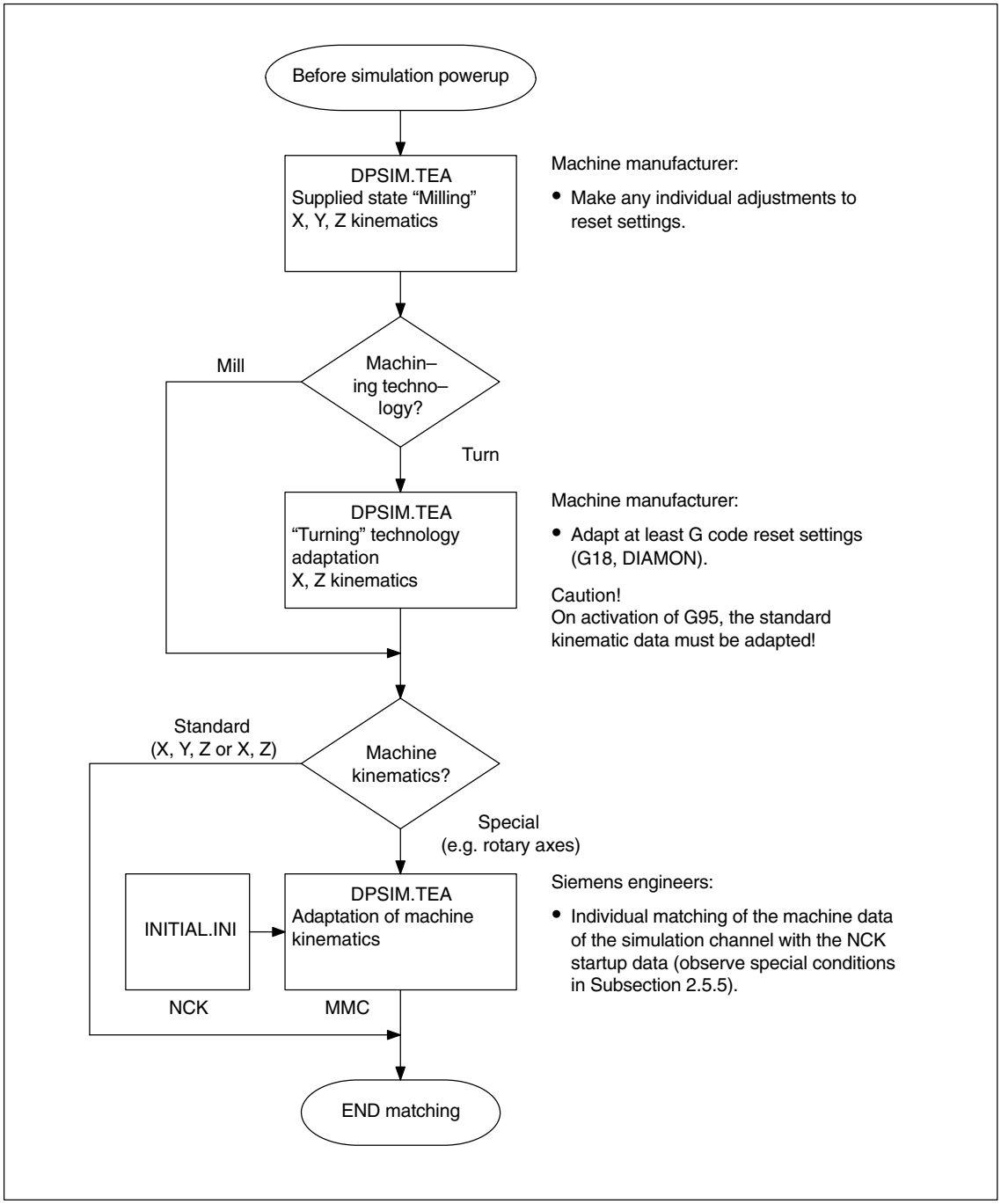


Fig. 2-2 Procedure for matching the machine data up to SW 3.1

© Siemens AG, 2003. All rights reserved

Operator actions

Operator actions for adapting the file DPSIM.TEA:

Note

Before adapting the file DPSIM.TEA, a full PC keyboard must be connected to the keyboard interface (round connector) of the MMC 102/103, and the manufacturer's password must be activated.



Press the user area key,

Startup mode

switch to "Startup mode",

MMC

press the "MMC" soft key

DOS Shell

followed by the "DOS Shell" key.

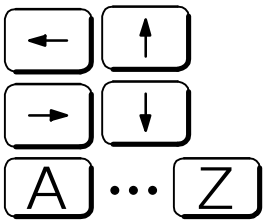
You are now on the DOS level in the directory C:\MMC2>.

CD DP

Enter the DOS command shown opposite to change to the DP subdirectory C:\MMC2\DP>.

EDIT DPSIM.TEA

Now, open the file DPSIM.TEA for editing with the DOS editor.



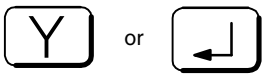
Use the direction keys to move the text cursor to the desired modification point, and make the changes.

ALT

When you have made the changes, press the "Alt" key on the full PC keyboard to activate the top menu line "File" (in the American Windows Version!).



Use the direction key to open the "File" menu, and press the "X" key to exit the editor.



If you have made any changes, answer "Yes" to the prompt window.



Return to startup mode by entering the DOS command "EXIT".

© Siemens AG, 2003. All rights reserved

The modified data in the DPSIM.TEA file are activated the next time the simulation is powered up.

Note

- In SW 3.1 of the MMC 102/103, the file DPSIM.TEA cannot be opened with the editor in the menu path Startup... (the file is too large).
- The individual matching of the DPSIM.TEA file (by Siemens engineers) should preferably be carried out on an external PC/PG:
  - File DPSIM.TEA must first be overwritten with the complete contents of initialization program INITIAL.INI. The data affected by the special conditions (see Subsection 2.5.5) must then be found and modified to the appropriate state.
  - The modified DPSIM.TEA can subsequently be copied back to the path C:\MMC2\DP.
- In addition to the startup machine data, the initialization program INITIAL.INI also contains basic settings for various user parameters (R parameters, settable frames, tool offsets, ...). These basic settings are **not** included in the simulation!

2.5.4 Adaptation of simulation data with SW 3.2 and higher

The machining simulation also uses a pseudochannel to run a separate, albeit restructured set of machine data in SW 3.2. and higher.

The technological basic sets of data have also been completed for “milling” and “turning”.

© Siemens AG, 2003. All rights reserved

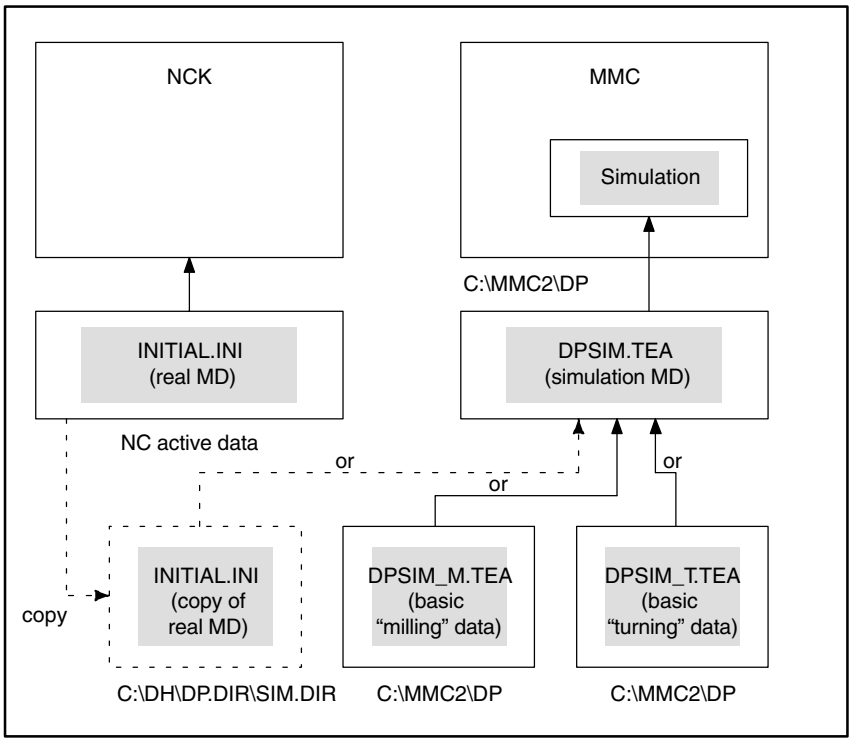


Fig. 2-3 Overview of data sets in SW 3.2 and higher

For more convenient adaptation in software SW 3.2 and higher, the set of simulation machine data is already contained in two basic sets of data in the C:\MMC2\DP directory:

- DPSIM_T.TEA for “turning”
- DPSIM_M.TEA for “milling”.

The valid set of data is automatically assigned to the machining simulation (if a copy of the real startup data is not located in the .... DIR directory in the form of an INITIAL.INI file), according to the selection of the machining technology (see Subsection 2.5.1).
Otherwise, the INITIAL.INI file is assigned automatically.

Finally, the file DPSIM.TEA with the simulationspecific machine data is used, in order to ensure that the special conditions in Subsection 2.5.5 are met.

© Siemens AG, 2003. All rights reserved

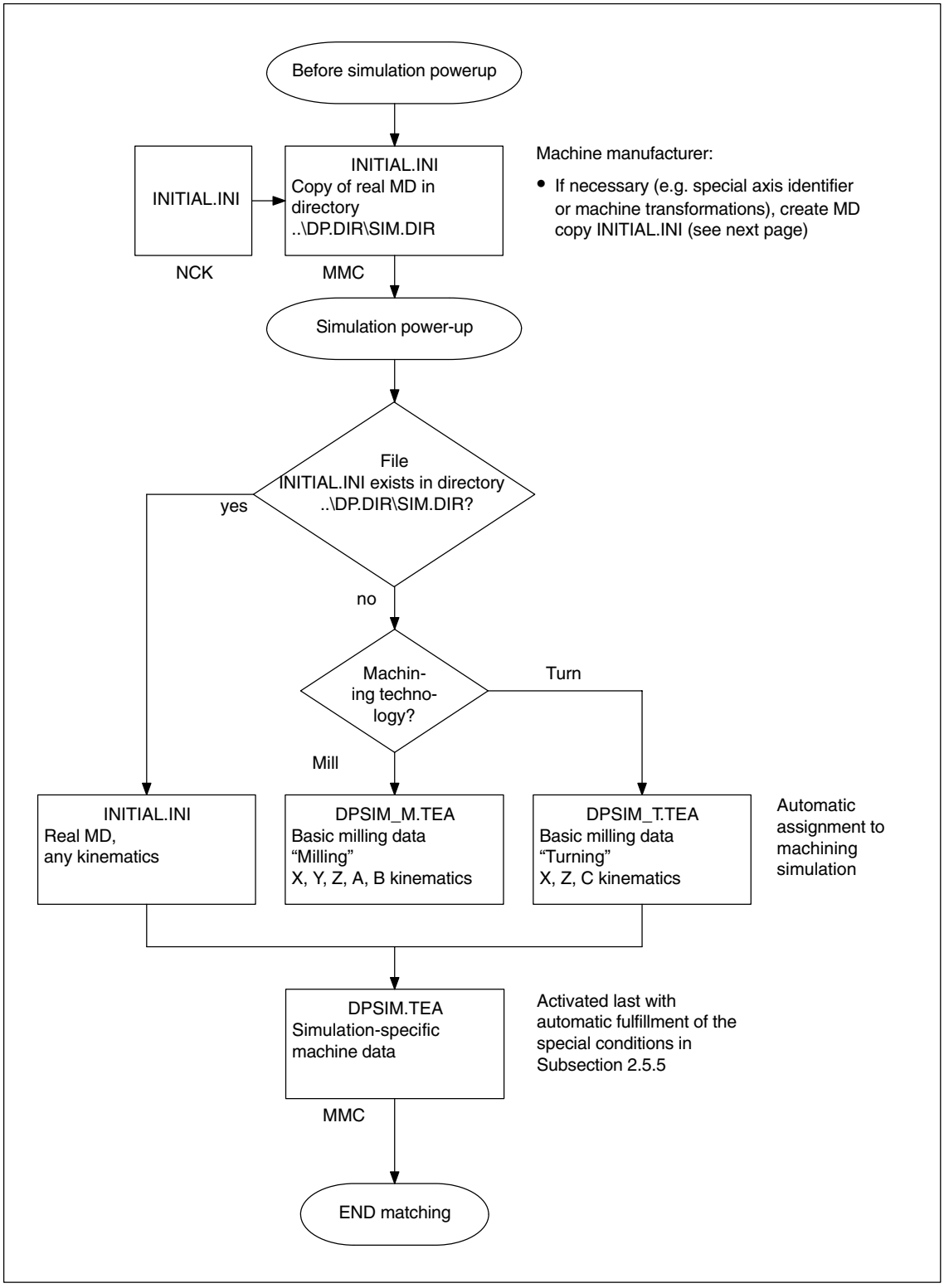


Fig. 2-4 Semiautomatic procedure for matching the machine data in SW 3.2 and higher

© Siemens AG, 2003. All rights reserved

Operator actions

Operator actions for creating the file INITIAL.INI in the simulation directory:

In order to adapt the set of simulation data to the real set of machine data, the active initialization data, including the machine data, are copied directly into the simulation directory of the interactive programming system (the manufacturer's password must be entered correctly).



Press the user area key,



switch to the "Services" mode,



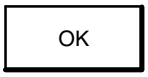
and activate the "Data Selection" function area.



Use the direction keys to position the cursor on the directory "NC-active data"



and select the data with the Selection key.



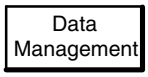
Press OK to confirm the operation, and return to the "Programs/Data" overview.



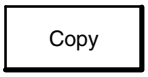
Pressing the Input key opens the "NC-active data" directory,



after which you can use the direction keys to select the file "INITIALIZATION PROGRAM (INI)".



Now, change to the "Data Management" function area,

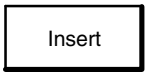


and select the function "Copy".



Use the direction keys and the Input key to move to the target directory "Interactive Programming/Simulation Data",





and press the "Insert" soft key.

The copy operation is subsequently started, and the file INITIAL.INI is created in the simulation directory.

The real machine data are included the next time the simulation is powered up.

© Siemens AG, 2003. All rights reserved

2.5.5 Special conditions for adapting simulation data

Secondary conditions up to SW 3.1

Contrary to the NCK Start up Guide, the flexible assignment of geometry axes to channel axes cannot be used for individual matching of the simulation channel machine data in the file DPSIM.TEA. The simulation interpreter always assumes a permanent assignment:

1st geometry axis => 1st channel axis or none
2nd geometry axis => 2nd channel axis or none
3rd geometry axis => 3rd channel axis or none

Consequently, the corresponding machine data in the DPSIM.TEA file can only have the following values:

$MC_AXCONF_GEOAX_ASSIGN_TAB[0]=1 or 0
$MC_AXCONF_GEOAX_ASSIGN_TAB[1]=2 or 0
$MC_AXCONF_GEOAX_ASSIGN_TAB[2]=3 or 0

The following configuration example is intended to illustrate this.

Other assignments to channel axes (e.g. during the configuration of machine transformations) should be adapted accordingly.

Note

The restriction described above does not apply to SW 3.2 and higher!

Configuration example

```
;_____________________________________________________________

; SW 3.1 – Machine data excerpt for adaptation

; of the axis configuration for machining simulation

; on a turning machine (intended only as an example !)

; 2 geometry axes ( X, Z ) ==> 1st and 2nd machine axes

; 1 spindle (C, rotary axis) ==> 4th machine axis

; 1 auxiliary axis (Q, linear axis) ==> 3rd machine axis

;_____________________________________________________________

CHANDATA(1)
$MN_AXCONF_MACHAX_NAME_TAB[0]="X1"
$MN_AXCONF_MACHAX_NAME_TAB[1]="Z1"
$MN_AXCONF_MACHAX_NAME_TAB[2]="Q1"
$MN_AXCONF_MACHAX_NAME_TAB[3]="C1"
$MN_AXCONF_MACHAX_NAME_TAB[4]="A1"
$MN_AXCONF_MACHAX_NAME_TAB[5]="Y1"
$MN_AXCONF_MACHAX_NAME_TAB[6]="U1"
$MN_AXCONF_MACHAX_NAME_TAB[7]="V1"

CHANDATA(1)
$MC_CHAN_NAME="CHAN1" '385e
$MC_AXCONF_GEOAX_ASSIGN_TAB[0]=1
```

© Siemens AG, 2003. All rights reserved

```
$MC_AXCONF_GEOAX_ASSIGN_TAB[1]=0
$MC_AXCONF_GEOAX_ASSIGN_TAB[2]=3
$MC_AXCONF_GEOAX_NAME_TAB[0]="X"
$MC_AXCONF_GEOAX_NAME_TAB[1]="Y"
$MC_AXCONF_GEOAX_NAME_TAB[2]="Z"
$MC_AXCONF_MACHAX_USED[0]=1
$MC_AXCONF_MACHAX_USED[1]=4
$MC_AXCONF_MACHAX_USED[2]=2
$MC_AXCONF_MACHAX_USED[3]=3
$MC_AXCONF_MACHAX_USED[4]=0
$MC_AXCONF_MACHAX_USED[5]=0
$MC_AXCONF_MACHAX_USED[6]=0
$MC_AXCONF_MACHAX_USED[7]=0
$MC_AXCONF_CHANAX_NAME_TAB[0]=" "
$MC_AXCONF_CHANAX_NAME_TAB[1]="C"
$MC_AXCONF_CHANAX_NAME_TAB[2]=" "
$MC_AXCONF_CHANAX_NAME_TAB[3]="Q"
$MC_AXCONF_CHANAX_NAME_TAB[4]="A"
$MC_AXCONF_CHANAX_NAME_TAB[5]="Y"
$MC_AXCONF_CHANAX_NAME_TAB[6]="U"
$MC_AXCONF_CHANAX_NAME_TAB[7]="V"

:

$MA_SPIND_ASSIGN_TO_MACHAX[AX4]=1

:

M30
```

Other conditions for SW 3.1 and SW 3.2

Certain machine data of the simulation channel must have a defined state (simulation state) for the error-free operation of the machining simulation. This defined state differs from the real state of the corresponding NCK machine data.

- In **SW 3.1**, the machine data must be set explicitly to this state in the file DPSIM.TEA (see Subsection 2.5.3).
- In **SW 3.2** and higher, the machine data are included in the DPSIM.TEA file (which was last active) from the start, with the result that the data are matched automatically.

The file DPSIM.TEA is printed on the following three pages, as supplied with SW 3.2 and higher:

© Siemens AG, 2003. All rights reserved

DPSIM.TEA file

```
; This dpsim.tea serves as a supplement to machine data input,

; in order to adapt any inappropriate settings for simulation

; made by the user – INITIAL.INI.

;

;

CHANDATA (1)

; No referencing for required for NC start
$MC_REFP_NC_START_LOCK = 0

; Deactivate PLC time out monitoring:
$MN_PLC_RUNNINGUP_TIMEOUT = 10.
$MN_PLC_CYCLIC_TIMEOUT = 10.

; Memory configuration
$MN_MM_USER_MEM_DYNAMIC = 5000

; Override to respond digitally:
$MN_OVR_FEED_IS_GRAY_CODE = 0
$MN_OVR_RAPID_IS_GRAY_CODE = 0
$MN_OVR_AX_IS_GRAY_CODE = 0
$MN_OVR_SPIND_IS_GRAY_CODE = 0

; No 611D drive required:
$MN_DRIVE_IS_ACTIVE[0]=0
$MN_DRIVE_IS_ACTIVE[1]=0
$MN_DRIVE_IS_ACTIVE[2]=0
$MN_DRIVE_IS_ACTIVE[3]=0
$MN_DRIVE_IS_ACTIVE[4]=0
$MN_DRIVE_IS_ACTIVE[5]=0
$MN_DRIVE_IS_ACTIVE[6]=0
$MN_DRIVE_IS_ACTIVE[7]=0
$MN_DRIVE_IS_ACTIVE[8]=0

; Position measurement system simulated
$MA_ENC_TYPE[0,AX1]=0
$MA_ENC_TYPE[1,AX1]=0
$MA_ENC_TYPE[0,AX2]=0
$MA_ENC_TYPE[1,AX2]=0
$MA_ENC_TYPE[0,AX3]=0
$MA_ENC_TYPE[1,AX3]=0
$MA_ENC_TYPE[0,AX4]=0
$MA_ENC_TYPE[1,AX4]=0
$MA_ENC_TYPE[0,AX5]=0
$MA_ENC_TYPE[1,AX5]=0
$MA_ENC_TYPE[0,AX6]=0
$MA_ENC_TYPE[1,AX6]=0
$MA_ENC_TYPE[0,AX7]=0
$MA_ENC_TYPE[1,AX7]=0
$MA_ENC_TYPE[0,AX8]=0
$MA_ENC_TYPE[1,AX8]=0
```

© Siemens AG, 2003. All rights reserved

```
; Setpoint output simulated:
$MA_CTRLOUT_TYPE[0,AX1]=0
$MA_CTRLOUT_TYPE[0,AX2]=0
$MA_CTRLOUT_TYPE[0,AX3]=0
$MA_CTRLOUT_TYPE[0,AX4]=0
$MA_CTRLOUT_TYPE[0,AX5]=0
$MA_CTRLOUT_TYPE[0,AX6]=0
$MA_CTRLOUT_TYPE[0,AX7]=0
$MA_CTRLOUT_TYPE[0,AX8]=0

; Axis output signals to PLC even for simulated axes
$MA_SIMU_AX_VDI_OUTPUT[AX1]=1
$MA_SIMU_AX_VDI_OUTPUT[AX2]=1
$MA_SIMU_AX_VDI_OUTPUT[AX3]=1
$MA_SIMU_AX_VDI_OUTPUT[AX4]=1
$MA_SIMU_AX_VDI_OUTPUT[AX5]=1
$MA_SIMU_AX_VDI_OUTPUT[AX6]=1
$MA_SIMU_AX_VDI_OUTPUT[AX7]=1
$MA_SIMU_AX_VDI_OUTPUT[AX8]=1

; The IPO cycle is adjusted to match the servo cycle

; therefore $MA_FIPO_TYPE[AXn] = 1 not required for SIM
$MN_IPO_SYSCLOCK_TIME_RATIO=1

; The only one allowed, and even the default value is not set separately

; $MN_POSCTRL_SYSCLOCK_TIME_RATIO=1

; Deactivate spindle states for IPO start
$MA_SPIND_ON_SPEED_AT_IPO_START [AX1] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX1] = FALSE
$MA_SPIND_ON_SPEED_AT_IPO_START [AX2] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX2] = FALSE
$MA_SPIND_ON_SPEED_AT_IPO_START [AX3] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX3] = FALSE
$MA_SPIND_ON_SPEED_AT_IPO_START [AX4] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX4] = FALSE
$MA_SPIND_ON_SPEED_AT_IPO_START [AX5] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX5] = FALSE
$MA_SPIND_ON_SPEED_AT_IPO_START [AX6] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX6] = FALSE
$MA_SPIND_ON_SPEED_AT_IPO_START [AX7] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX7] = FALSE
$MA_SPIND_ON_SPEED_AT_IPO_START [AX8] = FALSE
$MA_SPIND_STOPPED_AT_IPO_START [AX8] = FALSE

;Simulation without gearbox switchover (requires PLC acknowledgment!)
$MA_GEAR_STEP_CHANGE_ENABLE[ AX1 ]=0
$MA_GEAR_STEP_CHANGE_ENABLE[ AX2 ]=0
$MA_GEAR_STEP_CHANGE_ENABLE[ AX3 ]=0
$MA_GEAR_STEP_CHANGE_ENABLE[ AX4 ]=0
$MA_GEAR_STEP_CHANGE_ENABLE[ AX5 ]=0
$MA_GEAR_STEP_CHANGE_ENABLE[ AX6 ]=0
$MA_GEAR_STEP_CHANGE_ENABLE[ AX7 ]=0
$MA_GEAR_STEP_CHANGE_ENABLE[ AX8 ]=0
```

© Siemens AG, 2003. All rights reserved

```
;Enable all options for simulation:
$ON_NC_CODE_CONF_NAME_TAB[ 0 ]= " "          ; ASPLINE
$ON_NC_CODE_CONF_NAME_TAB[ 2 ]= " "          ; BSPLINE
$ON_NC_CODE_CONF_NAME_TAB[ 4 ]= " "          ; CSPLINE
$ON_NC_CODE_CONF_NAME_TAB[ 6 ]= " "          ; COMPON
$ON_NC_CODE_CONF_NAME_TAB[ 8 ]= " "          ; POLY
$ON_NC_CODE_CONF_NAME_TAB[ 10 ]= " "         ; REPOSL
$ON_NC_CODE_CONF_NAME_TAB[ 12 ]= " "         ; REPOSQ
$ON_NC_CODE_CONF_NAME_TAB[ 14 ]= " "         ; REPOSH
$ON_NC_CODE_CONF_NAME_TAB[ 16 ]= " "         ; COUPON
$ON_NC_CODE_CONF_NAME_TAB[ 18 ]= " "         ; TANGON
$ON_NC_CODE_CONF_NAME_TAB[ 20 ]= " "         ; FTOCON
$ON_NC_CODE_CONF_NAME_TAB[ 22 ]= " "         ; CUT3DC
$ON_NC_CODE_CONF_NAME_TAB[ 24 ]= " "         ; CUT3DF
$ON_NC_CODE_CONF_NAME_TAB[ 26 ]= " "         ;
$ON_NC_CODE_CONF_NAME_TAB[ 28 ]= " "         ;

;=============================================================
; The following screens are initially enabled directly as values !!!
;
;=============================================================

; Delete active tool on reset

; since only bit 1 is enabled, TOOL_RESET_VALUE is evaluated (default 0)
$MC_RESET_MODE_MASK=1

; Simulation of PLC acknowledgments on tool motion and tool change active.

; This bit is only used for testing purposes. It is used to test

; the data transport on NCK and MMC.
$MC_TOOL_MANAGEMENT_MASK='H0200'

;*************************************************************

; CAUTION: No blank lines after M30!

M30
```

© Siemens AG, 2003. All rights reserved

2.5.6 Matching the tool offset data

The following basic options are available for representing tools in the machining simulation:

1. Create a TOA file in the workpiece directory of the simulated program
2. Call a subprogram with the TOA data from the simulated program
3. Insert TOA data directly at the beginning of the simulated program
4. With SW 3.2 and higher, tool data from the tool catalog can also be used as representative tools.

To enable the tools to be represented with the real offset values in the machining simulation, the active tool data in the simulation must first be matched to the tool data from the NCactive data.

You can do this by copying the desired tool data from the NCactive data, and inserting the data in one of the previously addressed TOA data areas included in the simulation.

Note

- No information about the exact geometry of the tool shank or the cutting edge can be derived from the TOA parameters.
- The simulation uses standard default values for the above. These values can deviate from the real dimensions of the tool used, and incorrectly indicate collisions during the machining simulation.
- The exact dimensions can only be referenced in SW 3.2 and higher, by including the tool catalog.

Operator actions

The operating sequence for the indirect copying of the NC-active tool data is mainly identical in SW 3.1 and SW 3.2:



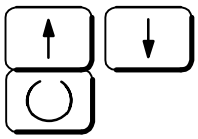
Press the user area key,



switch to the “Services” mode,



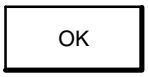
and activate the “Data Selection” function area.



Use the direction keys to position the cursor on the directory “NC-active data”,

select the data with the Selection key,

© Siemens AG, 2003. All rights reserved



and confirm with OK.



In the “Programs/Data” overview, press the Input key to open the directory “NC-active data”,



and use the direction keys to move to the directory “Tool/magazine data”.



Open this directory, too.



Now, select the directory “Tool offsets”



and open this directory as well.



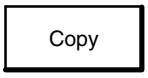
Use the direction keys to move to “Tool offsets – complete.INI”.



In order to copy the contents of the file via the Clipboard, open the file with the Input key,



select the desired contents of the file



and copy them to the Clipboard.

You can now insert use the ASCII editor to insert the contents of the Clipboard in one of the TOA areas included in the simulation:



For example, create an empty file of the type “.TOA” in “Interactive programming”, and insert the contents of the Clipboard in this TOA file.



“Interactive Programming” SW 3.2 and higher

- In order to create a workpiece-specific TOA file or a TOA subprogram, you do not have to use the Clipboard, but you can copy straight from the “Tool offsets – complete” source to the “workpieces” target. From there, you can derive a global subprogram, for example.
- Please ensure that the copied file in your target area has the correct type (TOA or SPF) and rename the file if necessary.
- The real turret or magazine loading is then always available in the machining simulation.

© Siemens AG, 2003. All rights reserved

Note

Please refer to Chapter 6 “Services User Area” of the Operator’s Guide for a detailed description of file handling.

References: /BA/, “Operator’s Guide”

2.5.7 Modifications with SW 3.6 and higher

a)
Layout changes

Basic information + Help

- The basic information was simplified:
  When displaying the basic information the current protection level is evaluated: The upper section of text is always displayed, the lower section (information concerning configuration options) is displayed only when the selected protection level is “User password” and higher.
- If the Help scratchpad text for a screen form does not have the same heading as the form, then the Help display is output together with the screen form (If not: Press i key first). This mechanism uses the interactive programming function originally for cycle displays, user displays and for tool master data. However, the user can apply it to all screen forms.

Free input

- The screen form for ‘Free inputs’ is now a fullframe display.
- If a new step of the Free input type is created and block numbering is selected, then a block number is entered in the screen form.

Contour

- The contour display can no longer be selected explicitly, but is now used only implicitly by the contour computer.
- The contour display covers the full screen width.
- The soft key ‘Setting geometry’ has been removed from the basic menu for ‘Display mode’. It can now be reached only by changing to the ’Display mode’ level when the contour computer is active.

© Siemens AG, 2003. All rights reserved

Simulation

- The simulation display is now a full-frame display
- The soft key ‘Setting simulation’ has been shifted from the basic menu of Display mode into the basic menu for Simulation and is now called ‘Blank + options’.

Program display

- Additional display areas are no longer set in the ‘Default Settings’ screen form (previously: Program + geometry, program + simulation).
- The soft key ‘Setting program’ in the basic menu of ‘Display mode’ has been replaced by soft key Change represent., which can be selected to switch between Step and Code representation.

Program identifier screen form

The program type display has been removed.

Tool change screen form

- The last selected change mode is present on the next call.
- Two different input fields are now provided for entering the name of the tool change program and the name of the tool. The option of calling a tool change program in Step can now also be utilized with the tool management function.
- The programming level can also be set in the screen form.

b)
New functions

1. The technology memory can be deactivated in the INI file:
   New entry in C:\MMC2\dpmill.ini / dpturn.ini (variables are read only during power-up):
   [ACCESSLEVEL]
   USE_TECHNO=7
   USE_TECHNO=0 can be entered to switch of the technology support function: Cutting edge/material selections and cutting edge/material editors are then no longer available.
2. The tool catalog can be deactivated in the INI file:
   New entry in C:\MMC2\dpmill.ini / dpturn.ini (variables are read only during power-up):
   [ACCESSLEVEL]
   USE_CATALOG=7
   When USE_CATALOG is set to 0, no tool catalog or tool editor is available.
3. If both the technology memory and tool catalog have been deactivated and no user displays are configured on dialog level 6, ‘Tool change’ is displayed as dialog level 6 rather than ‘Tools and material’. When the level is selected, the ‘Tool change’ screen form is displayed immediately.
4. The screen form ‘Blank + options’ contains the new soft key ‘Match tool data’ which can be used to copy a tool data image TO_INI.INI from the NC into directory ‘DP.DIR\SIM.DIR’.

© Siemens AG, 2003. All rights reserved

5. The screen form ‘Blank + options’ contains the new soft key ‘Match machine data’ which can be used to copy a machine data image INITIAL_INI from the NC into directory ‘DP.DIR\SIM.DIR’.
6. If neither of the above two files exists when the simulation routine runs up for the first time, the system asks whether data must be matched.
7. If no tool data are stored for a particular tool, the user must select one of the following procedure options. The following soft keys are displayed:
   - *Ignore*: All offset data are set to zero internally. A polymarker is displayed on the programmed path.
   - *Always ignore*: As for a). The same procedure is applied automatically for all other missing tool data in this simulation run.
   - *Abort*: The simulation run is aborted and alarm 17190 (“Illegal T number”) output.
   - *Default* (not provided for turning applications or in conjunction with tool management): The data of a default tool (type 120, length1 100, diameter from “Blank+options” screen form) are used in the simulation run.
   - *Always default* (not provided for turning applications or in conjunction with tool management): As for d). The same procedure is applied automatically for all other missing tool data in this simulation run.

2.6 Archiving of configuration data

Both the standard configuration data of “interactive programming” and the user-specific extensions of this configuration data can be read in and out or included in a batch start-up at any time in the “Services/Interactive Programming” data area.

Note

Please refer to the Operator’s Guide “Services User Area” for a detailed description of file handling.

References: /BA/ Operator’s Guide

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

The “Interactive programming function” is fully executable on all operator panel configurations with the MMC 102/103.

Note

Recommended operator panel platform:

- MMC 102/103 with 16MB RAM
- Color monitor (OP 031/032)
- Operation via a QWERTY keyboard

The response of the simulation channel to defined language statements is described in the documentation for SW 2 of the SINUMERIK 840D (Programming Guide, Descriptions of Functions).

The information in Section 2.1 (evaluation of system variable $P_SIM) should also be noted.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

The data descriptions in the following documentation apply to the simulation data:

References: /IAD/, “Installation and StartUp Guide”, SW 2

5 Signal Descriptions

None.

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

6 Example

The “interactive programming” function is to be used for the creation and simulation of parts programs on a turning machine with

- facing axis X,
- longitudinal axis Z,
- rotary axis C (master spindle),
- an auxiliary axis Q (loader) and
- a turret with permanent location.

Since the real machine environment differs from the standard state of the “interactive programming” system when supplied, at least the following modifications must be made:

1. Set up “Interactive programming” for “turning” applications according to instructions in Subsection 2.5.1
2. Adapt axis identifiers according to instructions in Subsection 2.5.2
3. Adapt simulation data according to instructions in Subsection 2.5.3 or 2.5.4
4. Adapt tool offset data according to instructions in Subsection 2.5.6

Only when these modifications have been made is the “interactive programming” system suitable in practice for use with the above turning machine, and for machining simulation in the real environment.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

7

7.1 Machine data

7.1.1 General machine data

Number	Identifier	Name	Refer-ence
General ($MN_ ...)			
10000	AXCONF_MACHAX_NAME_TAB[n]	Machine axis name	/K2/
10060	POSCTRL_SYSCLOCK_TIME_RATIO	Factor for position control cycle	/G2/
10070	IPO_SYSCLOCK_TIME_RATIO	Factor for interpolator cycle	/G2/
10100	PLC_CYCLE_TIMEOUT	Maximum PLC cycle time	/P3/
10120	PLC_RUNNINGUP_TIMEOUT	Monitoring time for PLC powerup	/H2/
12000	OVR_AX_IS_GRAY_CODE	Axis feed override switch, Graycoded	/V1/
12020	OVR_FEED_IS_GRAY_CODE	Path feed override switch, Graycoded	/V1/
12040	OVR_RAPID_IS_GRAY_CODE	Rapid traverse override switch, Graycoded	/V1/
12060	OVR_SPIND_IS_GRAY_CODE	Spindle override switch, Graycoded	/V1/
13000	DRIVE_IS_ACTIVE[n]	Drive activation (611D)	/G2/
18210	MM_USER_MEM_DYNAMIC	Dynamic user memory in DRAM	/S7/

7.1.2 Channel-specific machine data

Channel-specific ($MC_ ...)			
20000	CHAN_NAME	Channel name	/K1/
20050	AXCONF_GEOAX_ASSIGN_TAB[n]	Assignment of geometry axis to channel axis	/K2/
20060	AXCONF_GEOAX_NAME_TAB[n]	Geometry axis in channel	/K2/
20070	AXCONF_MACHAX_USED[n]	Machine axis name valid in channel	/K2/
20080	AXCONF_CHANAX_NAME_TAB[n]	Channel axis name in channel	/K2/
20110	RESET_MODE_MASK	Definition of control initial setting after parts pro-gram start	/K2/
20120	TOOL_RESET_VALUE	Tool whose tool length compensation is selected during power-up (Reset/parts program end)	/K2/
20310	TOOL_MANAGEMENT_MASK	Activation of tool management in various forms	/FBW/
20700	REFP_NC_START_LOCK	NC start disable without reference point	/R1/

© Siemens AG, 2003. All rights reserved

7.1.3 Axis-specific machine data

Axis-specific ($MA_ ...)			
30130	CTRLOUT_TYPE[n]	Output type of setpoint	/G2/
30240	ENC_TYPE[n]	Type of actual value acquisition (position actual value)	/G2/
30350	SIMU_AX_VDI_OUTPUT[n]	Output of axis signals for simulation axes	/G2/
35000	SPIND_ASSIGN_TO_MACHAX[n]	Assignment of spindle to machine axis	/S1/
35010	GEAR_STEP_CHANGE_ENABLE[n]	Gear stage change possible. Spindle has several gear stages.	/S1/
35500	SPIND_ON_SPEED_AT_IPO_START[n]	Feed enable with spindle in setpoint range	/S1/
35510	SPIND_STOPPED_AT_IPO_START[n]	Feed enable with stationary spindle	/S1/

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Travel to Fixed Stop (F1)

1	**Brief Description**		**1/F1/1-3**
2	**Detailed Description**		**1/F1/2-5**
	2.1	General, applications	1/F1/2-5
	2.2	General functionality	1/F1/2-6
	2.2.1	Functional sequence, programming, parameterization	1/F1/2-6
	2.2.2	Response to RESET and function abort	1/F1/2-13
	2.2.3	Block search response	1/F1/2-14
	2.2.4	Miscellaneous	1/F1/2-18
	2.2.5	Supplementary conditions for SW 5 extensions	1/F1/2-21
	2.2.6	Traversing with limited torque/force FOC (SW 5 and higher)	1/F1/2-23
	2.3	Travel to fixed stop with digital drives SIMODRIVE 611 digital (FDD/MSD)	1/F1/2-25
	2.3.1	Travel to fixed stop with hydraulic drives SIMODRIVE 611 digital (HLA module)	1/F1/2-27
	2.4	Travel to fixed stop with analog drives	1/F1/2-28
	2.4.1	SIMODRIVE 611 analog (FDD)	1/F1/2-28
	2.4.2	SIMODRIVE 611 analog (MSD)	1/F1/2-30
	2.4.3	Diagrams for travel to fixed stop for analog drives	1/F1/2-34
3	**Supplementary Conditions**		**1/F1/4-37**
4	**Data Descriptions (MD, SD)**		**1/F1/4-39**
	4.1	Axis-specific machine data	1/F1/4-39
	4.2	Axis-specific setting data	1/F1/4-44
5	**Signal Descriptions**		**1/F1/5-47**
	5.1	Axis/spindle-specific signals	1/F1/5-48
	5.1.1	Signals to axis/spindle	1/F1/5-48
	5.1.2	Signals from axis/spindle	1/F1/5-49
6	**Example**		**1/F1/6-51**
	6.1	Traveling to fixed stop, examples (SW 5.3 and higher)	1/F1/6-51
7	**Data Fields, Lists**		**1/F1/7-53**
	7.1	Interface signals	1/F1/7-53

© Siemens AG, 2003. All rights reserved

7.2 Machine data 1/F1/7-54
7.3 Setting data 1/F1/7-54
7.4 Alarms 1/F1/7-54

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

The “Travel to fixed stop” function can be used for operations such as traversing tailstocks or sleeves to an end limit position in order to clamp workpieces. The clamping torque and a fixed stop monitoring window can be programmed in the parts program and can also be altered via setting data once the fixed stop has been reached.

The “Travel to fixed stop” function can be implemented for axes as well as for spindles with axis-traversing capability.

The function can be implemented for several axes simultaneously and in parallel to the motion of other axes.

SW 5 and higher

In this software version, the functionality is extended such that

- torques or the force can be adapted block-specifically
- traversing is possible with limited torque/force (Force Control, FOC)
- the “Travel to fixed stop” functions can be enabled from synchronized actions.

Additional expanded machine data and system variables are made available for this purpose.

SW 6 and higher

In this SW, the “Travel to Fixed Stop” functionality is also available for the SINUMERIK 810D with CCU3.

SINUMERIK 840D with NCU 573.3. The following expansions have been implemented:

- Block search with calculation, multi-channel (SERUPRO). Move axes with FXS and FOC in simulation mode.
- Suspended axes can be moved to fixed stop even when FXS alarms are active.
- With SW 6.4 and higher, VDI signals can be used to set a REPOS offset for each axis and display the FXS status currently active on the machine after the search target has been located.

Additional, expanded machine data, system variables and interface signals are provided for this purpose.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 General, applications

Area of application

The “Travel to fixed stop” (FXS = Fixed Stop) function can be used to build defined forces for clamping workpieces, such as those required for tailstocks, quills and grippers. Mechanical reference points can also be approached with the function. With a sufficient reduction in torque, it also possible to implement simple measuring operations without connecting a probe.

The “Travel to fixed stop” function can be implemented for axes as well as for spindles with axis-traversing capability.

It can also be applied to several axes simultaneously and in parallel to the motion of other axes.

The fixed stop can be approached on any selected path (straight line, circle, spline, ...).

Availability

The “Travel to fixed stop” function is available if
MD 37000: FIXED_STOP_MODE
(Travel to fixed stop mode) is set to 1. The function can then be started from the NC program with command “FXS[x]=1”.
With SW 5 and higher, it is additionally possible to use synchronized actions.

Supplementary conditions

Travel to fixed stop has been implemented for

- SINUMERIK 840D with digital drives (SIMODRIVE 611 digital)
- SINUMERIK FMNC with analog drives
- SINUMERIK 840Di with SIMODRIVE 611 universal

The function can be used in analog drives only if the drive actuator used has a switchable torque limitation. In order to obtain programmable pressure forces or torques in analog drives, a drive actuator which can be switched over between speed and torque control without a sign change is required.

The series SIMODRIVE 611 analog (FDD/MSD) meets these requirements.

With the actuator SIMODRIVE 611 analog MSD, make sure that the actuator alarm F11 (speed controller at limit) is switched off.

© Siemens AG, 2003. All rights reserved

Restrictions

"Travel to fixed stop" **cannot** be used for

- suspended axes (on **840D** with 611D**, possible with SW 2.2 and higher**)
- gantry axes
- concurrent positioning axes which are controlled exclusively by the PLC (FXS must be selected from the NC program.).

Note

It is not permissible to program a new position for any axis/spindle for the "Travel to fixed stop" function has already been activated (Exception: Synchronized actions, examples).

Before the function is selected for a spindle, the spindle must be switched to position-controlled mode.

Note

When programming the clamping torque via FXST[x] or SD
SD 43510: FIXED_STOP_TORQUE, with SINUMERIK 840Di, this value can only be programmed within a range between 0% and 100%. The admissible step value is 1%.
Programmable values greater than 100% (e.g. 180%) will be taken into account only with a maximum of 100%.

2.2 General functionality

2.2.1 Functional sequence, programming, parameterization

Programming

Travel to fixed stop is selected with command

FXS[machine axis identifier] = 1 selected.

FXS[machine axis identifier] = 0 deselected.

The commands are effective modally. The clamping torque is set with command

FXST[machine axis identifier] = <torque>.

It is entered as a % of the static torque (MD 1118: MOTOR_STANDSTILL_CURRENT) of the drive/
as a % of the motor torque (MD 1103: MOTOR_NOMINAL_CURRENT) on main spindle drives.

The command

FXSW[machine axis identifier] = <monitoring window>

is used for setting the width of the fixed stop monitoring window.

The unit is dependent on the default setting: mm, inch or degrees.

© Siemens AG, 2003. All rights reserved

Channel axis identifiers SW 4 and higher

Instead of the machine axis identifiers, it is also possible to use channel axis identifiers if the channel axis identifiers are assigned exactly to one machine axis.
Restrictions: Channel identifiers may not be used (option disabled) for machine axes which have an active transformation or frame. In cases where the machine axis is acting as a coupled axis (e.g. slave axis), the system disables programming of channel axis identifiers and alarm 14092 ”Wrong axis type” is output.

The movement towards the target point can be defined as a path or positioning axis motion. In the case of positioning axes, the “FXS” function can also be applied beyond block limits. The function may also be selected for several machine axes simultaneously. The FXST and FXSW commands are optional. The travel path and the command which activates the function **must be programmed in the one block** (Exception: Synchronized actions).

Examples

With machine axis identifiers:

```
X250 Y100 F100 FXS[X1]=1
X250 Y100 F100 FXS[X1]=1 FXST[X1]=12.3
X250 Y100 F100 FXS[X1]=1 FXST[X1]=12.3 FXSW[X1]=2          ; mm
X250 Y100 F100 FXS[X1]=1 FXSW[X1]=2                        ; mm
```

References: /PG/, “Programming Guide: Fundamentals”

Channel axis identifier with unambiguous machine axis assignment:

For the purpose of illustrating the differences in programming, channel axis X is programmed as the image of machine axis AX1 (or X1 (name in MD 10000 $MN_AXCONF_MACHAX_NAME_TAB)) in the example below.

Programming with machine axis identifiers	Programming with channel axis identifiers
FXS[X1] = 1; Selection for X1	FXS[X] = 1; Selection for X –> X1
FXST[X1] = 10; New torque for X1	FXST[X] = 10; New torque for X –> X1
FXSW[X1] = 5; New window for X1	FXSW[X] = 5; New window for X –> X1

All of the following four programming lines have the same effect if channel axis X is programmed as the image of machine axis AX1, X1:

```
Z250 F100 FXS[AX1]=1 FXST[AX1]=12.3 FXSW[AX1]=2000
Z250 F100 FXS[X1]=1 FXST[X1]=12.3 FXSW[X1]=2000
Z250 F100 FXS[X]=1 FXST[X]=12.3 FXSW[X]=2000
Z250 F100 FXS[X]=1 FXST[X1]=12.3 FXSW[AX1]=2000
```

© Siemens AG, 2003. All rights reserved

Functional sequence

The function is explained by the example below (sleeve is pressed onto workpiece).

The special points to be noted with regard to “Travel to fixed stop” in digital or analog drives are described in Sections 2.3 and 2.4.

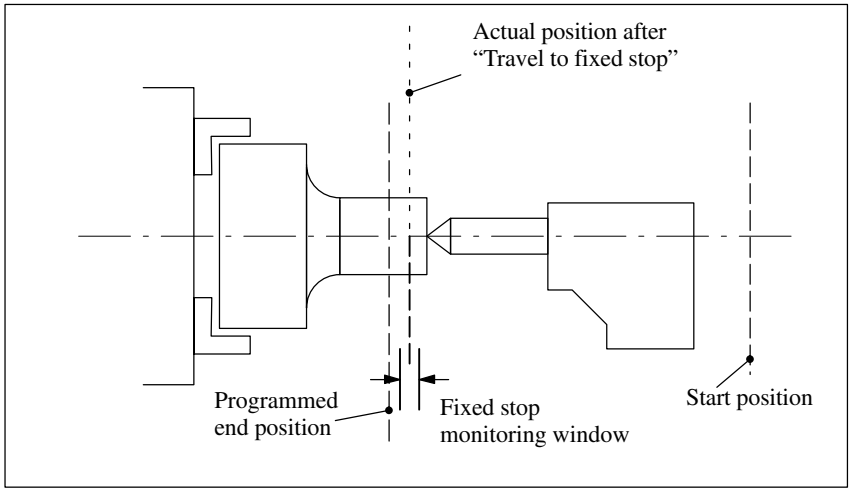


Fig. 2-1 Example of travel to fixed stop

Selection

The NC detects that the function “Travel to fixed stop” is selected via the command FXS[x]=1 and signals the PLC via the IS “Activate travel to fixed stop” (DB31, ... DBX62.4) that the function has been selected.

If MD 37060: FIXED_STOP_ACKN_MASK (register PLC acknowledgments for travel to fixed stop) is set accordingly, the NC waits for acknowledgment by the PLC through IS “Activate travel to fixed stop” (DB31, ... DBX3.1).

The programmed target position is then approached from the start position at the programmed velocity. The fixed stop must be located between the start and target positions of the axis/spindle. Any programmed torque limitation is effective from the block start, i.e. the fixed stop is also approached at a reduced torque. Allowance for this limitation is made in the NC through an automatic reduction in the acceleration rate.

If no torque has been programmed in the block or since the start of the program, then the value set in axis-specific

MD 37010: FIXED_STOP_TORQUE_DEF

(default setting for clamping torque) is applicable.

Fixed stop reached

As soon as the axis comes into contact with the mechanical fixed stop (workpiece), the closed-loop control in the drive raises the torque so that the axis can move on. The torque rises up to the programmed limit value and then remains constant.

© Siemens AG, 2003. All rights reserved

The “Fixed stop reached” status can be determined in the following ways depending on the setting in MD 37040: FIXED_STOP_BY_SENSOR (detection of fixed stop via sensor):

- FIXED_STOP_BY_SENSOR = 0 The “Fixed stop reached” status is achieved when the axial contour deviation (= difference between actual and expected following error) has exceeded the value set in MD 37030: FIXED_STOP_THRESHOLD (threshold for fixed stop detection).
- FIXED_STOP_BY_SENSOR = 1 External sensor transmits “Fixed stop reached” status to NC via the PLC by means of IS “Sensor fixed stop” (DB31, ... DBX1.2).
- FIXED_STOP_BY_SENSOR = 2 The “Fixed stop reached” status is reached either when it is detected by the contour monitoring function or signalled by the external sensor in response to signal edge change 0 → 1 to (DB31,...DBX 1.2).

The axis contour deviation is explained in:

References: /FB/, D1, “Diagnostics Tools”

Internal processes

Once the NC has detected the “Fixed stop reached” status, it deletes the distance-to-go and the position setpoint is made to follow. The controller enabling command remains active.

The PLC is then informed of the state by means of IS “Fixed stop reached” (DB31, ... DBX62.5).

If MD 37060: FIXED_STOP_ACKN_MASK is set accordingly, then the NC waits for acknowledgment by the PLC in the form of IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1).

The NC then executes a block change or considers the positioning motion to be completed, but still leaves a setpoint applied to the drive actuator to allow the clamping torque to take effect.

The fixed stop monitoring function is activated as soon as the stop position is reached.

Monitoring window

If no fixed stop monitoring window is programmed in the block or since the beginning of the program, then the value set in MD 37020: FIXED_STOP_WINDOW_DEF (default setting for fixed stop monitoring window) is applicable.

If the axis leaves the position it was in when the fixed stop was detected, then alarm 20093 “Fixed stop monitoring has responded” is output and the “Travel to fixed stop” function deselected.

The window must be selected by the user such that the alarm is activated only when the axis leaves the fixed stop position.

© Siemens AG, 2003. All rights reserved

Enabling the fixed stop alarms

The machine data MD 37050 FIXED_STOP_ALARM_MASK can be used to set the enabling of the fixed stop alarms as follows:

- MD 37050 = 0 fixed stop not reached (suppress alarm 20091)
- MD 37050 = 2 fixed stop not reached (suppress alarm 20091) **and** fixed stop aborted (SW 4 and higher: Suppress alarm 20094)
- MD 37050 = 3
  Fixed stop aborted (SW 4 and higher: Suppress alarm 20094)
- MD 37050: No other values ≤ 7 suppress alarms
- MD 37050: **In SW 5 and higher**, it is possible to activate a new setting using the parts program command NEWCONF.

Fixed stop is not reached

If the programmed end position is reached without the “Fixed stop reached” status being detected, then alarm 20091 “Fixed stop not reached” is output depending on the status of MD 37050: FIXED_STOP_ALARM_MASK (enable fixed stop alarms).

Function abort

If the “Travel to fixed stop” function is aborted owing to a pulse disable command, cancellation of PLC acknowledgments or a Reset in the approach block, the display or suppression of alarm 20094 can be controlled via MD 37050: FIXED_STOP_ALARM_MASK. See Chapter 4.

Abort without alarm

The PLC can initiate travel to fixed stop without an alarm in the approach block (for example, when the operator actuates a key) if alarm 20094 is suppressed in machine data MD 37050: FIXED_STOP_ALARM_MASK.

The Travel to fixed stop function is deselected in response to both “Fixed stop not reached” and “Fixed stop aborted”.

Alarms

- If the fixed stop position is not reached when the function is active, alarm 20091 “Fixed stop not reached” is output and a block change executed.
- If a traversing request (e.g. from the parts program, the PLC, from compile cycles or from the operator panel) is provided for an axis after the fixed stop has been reached, the alarm 20092 “Travel to fixed stop still active” is output and the axis is not moved.
- If an axis has reached the fixed stop and is then moved out of this position by more than the value specified in SD: FIXED_STOP_WINDOW (fixed stop monitoring window), alarm 20093 “Fixed stop monitoring has responded” is output; the “Travel to fixed stop” function is then deselected for this axis and system variable $AA_FXS[x] is set to 2.

No function abort after alarm in SW 6.2

The travel to fixed stop function remains active after an alarm if the bit values have been set in machine data MD 37052: FIXED_STOP_ALARM_REACTION. IS “Mode group ready” (DB11, ... DBX6.3). For further information about this functionality, see Subsection 2.2.4 Miscellaneous.

© Siemens AG, 2003. All rights reserved

Alarm suppression after new programming (SW 5 and higher)

Travel to fixed stop can be used for simple measuring processes.

For example, it is possible to carry out a check for tool breakage by measuring the tool length by traversing onto a defined obstacle. To do so, the fixed stop alarm must be suppressed. When the function for clamping workpieces is then used “normally”, the alarm can be activated using parts program commands.

Sequence in case of a fault or abnormal termination

IS “Activate travel to fixed stop” (DB31, ... DBX62.4) is reset.

Depending on the setting in MD 37060: FIXED_STOP_ACKN_MASK, the NC waits for the PLC to acknowledge by resetting IS “Activate travel to fixed stop” (DB31, ... DBX3.1).

The torque limitation is then canceled and a block change executed.

Deselection

The NC detects deselection of the function on the basis of command FXS[x]=0. Then, a preprocessing stop (STOPRE) is initiated internally, since it can not be foreseen where the axis will be after deselection.

The torque limitation and monitoring of the fixed stop monitoring window are canceled. IS “Activate travel to fixed stop” (DB31, ... DBX62.4) and “Fixed stop reached” (DB31, ... DBX62.4) reset.

Depending on the setting in MD 37060: FIXED_STOP_ACKN_MASK, the NC waits for the PLC to acknowledge by resetting IS “Activate travel to fixed stop” (DB31, ... DBX3.1) and/or IS “Fixed stop reached” (DB31, ... DBX1.1).

The axis will then change to position control. The follow-up mode of the position setpoints is ended and a synchronization to the new actual position is carried out.

A programmed traverse motion can then be executed. This motion must lead away from the fixed stop or else the stop or the even machine may sustain damage.

A block change is executed after the target position has been reached.

Multiple selection

A selection may only be carried out once. If the function is called once more due to faulty programming (FXS[axis] = 1), the alarm 20092 “Travel to fixed stop still active” is initiated. An example is to be found in Chapter 6.

Block-related synchronized actions

By programming a block-related synchronized action, travel to fixed stop can be connected during an approach motion.

Programming example:

```
N10 G0 G90 X0 Y0
N20 WHEN $AA_IW[X]>17 DO FXS[X]=1    ; If X reaches a position greater
N30 G1 F200 X100 Y110                ; than 17mm, FXS is activated.
```

© Siemens AG, 2003. All rights reserved

Changing clamping torque and fixed stop monitoring window

The commands FXST[x] and FXSW[x] can be used to change the clamping torque and the fixed stop monitoring window in the parts program. The changes become effective prior to traversing motions programmed in the same block. If a new torque is programmed, then a step change transition to this torque is executed at the beginning of the appropriate block (up to and including SW 5.2).

If a new fixed stop monitoring window is programmed, then it is not only the window width which changes, but also the reference point for the window center if the axis moved beforehand. The actual position of the machine axis at the instant the window is changed constitutes the new window center.

SW 5.3 and higher

New machine data and synchronized actions FXS, FXST, FXSW

Terminal 663 controllable with MD 37002

Individual basic frames can be deleted with MD 37002: FIXED_STOP_CONTROL can be used to control the response in the case of pulse blocking at limit. Deleting the pulses by terminal 663 or the IS “Pulse enable” DBX31, ...DBX21.7 will not abort the function. As a result, the drive will press against the fixed stop again without any further operating action when the machine is restarted.

Further details about the response to pulse disabling are given in Subsection 2.2.5.

The rise time of the torque corresponds to the time needed by the current controller of the drive to reach the limitation again.

If the pulses are deleted when a deselection is active (waiting for PLC acknowledgments), the torque limit will be reduced to zero. If the pulses are reactivated during this phase, no torque is no longer built up. Once the response has been completed, you can continue traversing as normally.

FXS commands programmable in synchronized actions

The parts program commands FXS, FXST and FXSW can be programmed in synchronized actions/technology cycles.

The function FXS[x]=1 can also be activated without movement; the torque is limited immediately. Once the axis is moved using the setpoint, a monitoring for stop is carried out.

In static and block-related synchronized actions, the same commands FXS, FXST and FXSW can be used as in the normal parts program run. The values assigned can be resulted from a calculation.

Examples are to be found in Chapter 6.

Ramp for the torque limit with MD 37012

Up to SW 4, the new torque limit was directly transmitted to the drive. Since the drive sets the torque limit step-like, the limitation came into effect immediately. For example, for pushing in a quill, the setting of the torque limit would be too jerky.

In SW 5 and higher, a ramp has been implemented. A time is set in machine data

MD 37012: FIXED_STOP_TORQUE_RAMP_TIME

defining the length of the ramp until the new torque limit is reached.

For further information on the machine data, please refer to Section 4.1.

© Siemens AG, 2003. All rights reserved

2.2.2 Response to RESET and function abort

Response to RESET

During selection (fixed stop not yet reached) the function FXS can be aborted with RESET. The abortion is carried out such that an “almost achieved” fixed stop (setpoint already beyond the fixed stop, but still within the threshold for the fixed stop detection) will not result in a damage.

This is achieved by synchronizing the position setpoint to the new actual position. As soon as the fixed stop is reached, the function remains operative even after RESET.

Function abort

A function abort can be triggered by the following events:

EMERGENCY STOP:

- The “Travel to fixed stop” function is canceled in the drive with an FM-NC control.
- With an 840D control, the NC and drive are disconnected from the supply after EMERGENCY STOP, i.e. the PLC must react.
- With an 840Di control, the NC and drive are disconnected from the supply after EMERGENCY STOP, i.e. the PLC must react.



Caution

Make sure that after the function “Travel to fixed stop” due to EMERGENCY STOP has been canceled, no dangerous machine situation (MD 37002: FIXED_STOP_CONTROL e.g. Cancel pulse blocking) can result.

The fixed stop monitoring function responds in the case of:

- Exit from fixed stop position by axis
- Tool breakage
- Pulse disable

© Siemens AG, 2003. All rights reserved

2.2.3 Block search response

Block search with calculation

The response is as follows:

- If the target block is located in a program section in which the axis must stop at a fixed limit, then the fixed stop is approached if it has not yet been reached.
- If the target block is located in the program section in which the axis must not stop at a fixed limit, then the axis leaves the fixed stop if it is still positioned there.
- If the desired fixed stop status is reached, alarm message 10208 “Press NC Start to continue the program” is output. The program can be continued after pressing NC Start to acknowledge.
- At the start of the target block, FXST[x] and FXSW[x] are set to the same value as they would have during normal program processing.

Block search without calculation

The commands FXS, FXST and FXSW are ignored.

FOC SW 5 and higher

FOCON/FOCOF is activated modally. It is active already in the approach block.

SERUPRO (SW 6 and higher)

Block search with calculation, multi-channel

The block search in the Program test mode is called SERUPRO and has been derived from **“Se**arch**Ru**n by **Pro**gram Test”. This search mode allows the user a multi-channel block search with calculation of **all** required status data from the previous history.

- During the block search, the PLC interface is updated **and**
- machining processes covering the interaction of several channels are executed within the framework of this block search correctly.

Search process with FXS and FOC

The user select the function FXS or FOC in a program area of the searched target block in order to acquire all states and functions of this machining, which have been valid last. The NC will start the selected program in Program test mode automatically. After the target block has been found, the NC stops at the beginning of the target block, deselects Program test internally again and displays the Stop condition “Search target found” in its block display.

SW 6.1 and higher
If FXS “Travel to fixed stop” is programmed between the beginning of the program and the search target, the instruction is not really executed by the NC. The motion is only simulated up to the programmed end point.



Caution

SERUPRO approach does not really take the statement FXS into account. The approach to the programmed end position of the FXS block is only simulated **without torque limitation**.

© Siemens AG, 2003. All rights reserved

The user can log the turning on and turning off of FXS in the parts program. If necessary, the user can start an ASUB in order to activate or deactivate FXS in this SERUPRO ASUB.

$AA_FXS and $VA_FXS

SW 6.2 and higher
The meaning of system variable $AA_FXS is redefined for SERUPRO only and completely replaced by variable $VA_FXS. Variables $AA_FXS and $VA_FXS have the same values continuously **outside** the SERUPRO function.

Description	NCK variables
Axis not at fixed stop	$AA_FXS = 0 and $VA_FXS = 0
Fixed stop successfully approached	$AA_FXS = 1 and $VA_FXS = 1
Approach to fixed stop failed	$AA_FXS = 2 and $VA_FXS = 2
Travel to fixed stop selection active	$AA_FXS = 3 and $VA_FXS = 3
Fixed stop detected	$AA_FXS = 4 and $VA_FXS = 4
Travel to fixed stop deselection active	$AA_FXS = 5 and $VA_FXS = 5

Course of values

Course of values of system variables $VA_FXS[ ] with values 1 to 5

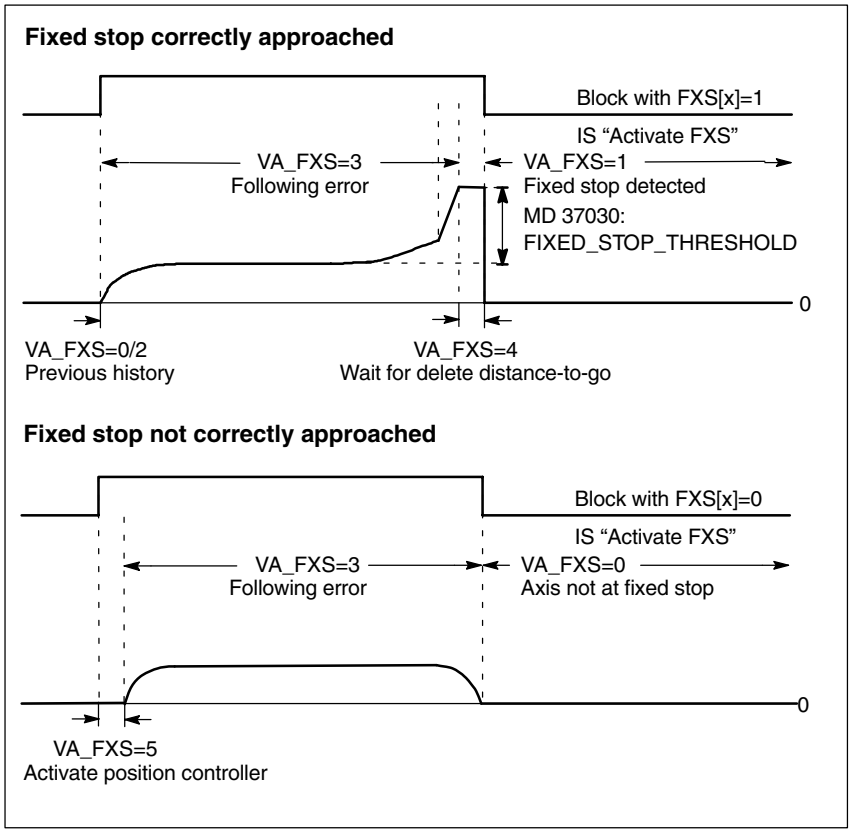


Fig. 2-2 Diagram for FXS with a digital drive (611 digital)

© Siemens AG, 2003. All rights reserved

$AA_FXS Simulate axis traversal

System variable $AA_FXS displays the current status of program simulation "program-sensitive system variable".

Example:
If in the SERUPRO operation

- axis Y traversal is simulated with FXS[Y]=1, then $AA_FXS has a value of 3.
- axis Y traversal is simulated with FXS[Y]=0, then $AA_FXS has a value of 0.

During simulation with SERUPRO,
$AA_FXS cannot have the values 1, 2, 4, 5, since the actual status of $VA_FXS Travel to Fixed Stop can never be detected.

Note

The state $AA_FXS = 1 is never reached during the SERUPRO operation. This means that other program branches can be processed which will produce different results due to the simulation.

If after the SERUPRO operation

- axis Y is traversed again, then variables $AA_FXS and $VA_FXS are assigned the same values again.

$VA_FXS Real machine status

Variable $VA_FXS **always** describes the real machine status.

The real machine status of the relevant axis is thus displayed

- with $VA_FXS during the SERUPRO operation.

Setpoint/actual status comparison

The two system variables $AA_FXS and $VA_FXS can be used to compare the setpoint and actual states in the parts program. The SERUPRO ASUB routine is then as follows:

SERUPRO ASUB
Asub fxsSeruproAsup.mpf

;The setpoint and actual states are compared in order to
;activate or deactivate FXS for the REPOSA block

```
N1000 WHEN ($AA_FXS[X]==3) AND ($VA_FXS[X]==0) DO FXS[X]=1
N2000 WHEN ($AA_FXS[X]==0) AND ($VA_FXS[X]==1) DO FXS[X]=0
N1020 REPOSA
```

SW 6.4 and higher

Display **REPOS offset**
Once the search target has been found, the FXS state active on the machine is displayed for each axis via axial VDI signals

IS "Activate travel to fixed stop" (DB31, ... DBX62.4) and
IS "Fixed stop reached" (DB31, ... DBX62.5) displayed.

Example:
If the machine is at the fixed stop and the block search has reached a block after deselection of FXS, the new target position is displayed as the REPOS offset via IS "Fixed stop reached" (DB31, ... DBX62.5).

© Siemens AG, 2003. All rights reserved

FXS fully automatic in REPOS

The functionality of FXS is repeated automatically with REPOS and designated FXS REPOS below. The sequence is comparable to the fxsSeruproAsup.mpf routine. Every axis is taken into account and the torque last programmed before the search target is applied.

The user can treat FXS separately in an SERUPRO ASUB. The following then applies:

Every FXS action executed in an SERUPRO ASUB

- automatically ensures $AA_FXS[X] == $VA_FXS[X].

This deactivates FXS REPOS for axis X.

FXS REPOS deactivation

FXS REPOS is deactivated

- by an FXS synchronous action which refers to REPOSA or
- $AA_FXS[X] == $VA_FXS[X] in SERUPRO_ASUB

Note

An SERUPRO ASUB **without** FXS handling, or no SERUPRO ASUB, results in fully automatic FXS REPOS.



Caution

FXS REPOS moves all path axes in a path grouping to the target position. Axes with and without FXS treatment thus traverse together with the G code and feedrate valid at the time the target block is processed. As a result, the fixed stop may be approached in rapid traverse (G0) or a higher velocity.

FOC fully automatically in REPOS

The FOC REPOS function behaves analogously to the FXS REPOS function.



Caution

A continuously changing torque characteristic can **not** be be implemented with FOCREPOS.

Example:
A program moves axis X from 0 to 100 and activates FOC every 20 millimeters for 10 millimeters at a time. This torque characteristic is generated with non-modal FOC and **cannot** therefore be traced by FOC REPOS. FOC REPOS will traverse axis X from 0 to 100, with or without FOC, according to the last programming.

For FXS “Travel to fixed stop” programming examples, please refer to:
References: /FB1/, K1, “Mode Group, Channel, Programming Mode”, Program Test

© Siemens AG, 2003. All rights reserved

2.2.4 Miscellaneous

Setting data

The following axis-specific setting data are provided for the “Travel to fixed stop” function:

- SD 43500: FIXED_STOP_SWITCH (selection of travel to fixed stop)
- SD 43510: FIXED_STOP_TORQUE (clamping torque for travel to fixed stop)
- SD 43520: FIXED_STOP_WINDOW (fixed stop monitoring window)

The setting data are effective only when the axis has reached the fixed stop.

The status of the setting data is displayed via the operator panel in the “Parameters” area.

Commands FXS[x], FXST[x] and FXSW[x] effect a block-synchronous change in these setting data. If FXST[x] and FXSW[x] are not programmed and “Travel to fixed stop” is activated, the defaults from

MD 37010: FIXED_STOP_TORQUE_DEF and
MD 37020: FIXED_STOP_WINDOW_DEF

are accepted into the appropriate setting data.

The setting data for clamping torque and fixed stop monitoring window can be changed by the operator and via the PLC. It is thus possible to specify a higher or lower clamping torque or a modified fixed stop monitoring window after the fixed stop has been reached.

SW 5.3 and higher

Changing the clamping torque using the ramp and values greater than 100%

A clamping torque change is transferred to the drive step-like. It is possible to specify a ramp always such that a modified torque limit is reached via MD 37012: FIXED_STOP_TORQUE_RAMP_TIME.

Clamping torque greater than 100%

Values greater than 100% are reasonable for SD 43510: FIXED_STOP_TORQUE only for a short time. Irrespectively, the maximum torque is limited by the drive.

For example, the drive machine data

MD1103 – rated motor current,
MD1104 – maximum motor current,
MD1105 – reduction of maximum motor current,
MD1230/1231 – torque limit value1/2 have a limiting effect.

For further information, please refer to the Planning Guide SIMODRIVE Three-Phase Motors for Feed and Main Spindle Drives and to the appropriate document regarding the hydraulic module /FBHLA/.

© Siemens AG, 2003. All rights reserved

Status interrogation in the parts program

System variable $AA_FXS[x] indicates the status of the “Travel to Fixed Stop” function. It has the following coding:

$AA_FXS= 0 Axis is not at fixed stop

1 Fixed stop has been approached successfully (axis is within fixed stop monitoring window)

2 Approach to fixed stop has failed (axis is not at fixed stop)

3 Travel to fixed stop activated (in SW 5.3 and higher extended)

4 Fixed stop has been detected (in SW 5.3 and higher extended)

5 Travel to fixed stop is deselected. The deselection is not yet completed (extended in SW 5.3 and higher)

Interrogation of the system variable in the parts program initiates a block search stop.

As a result of the status interrogation in the parts program, it is possible, for example, to react to an erroneous operational sequence of the “Travel to fixed stop” function.

The following applies to the example below:
MD 37050: FIXED_STOP_ALARM_MASK = 0 ⇒ No alarm is generated in response to an error. A block change therefore takes place and the error scenario can be evaluated via the system variable.

Example

```
X300 Y500 F200 FXS[X1]=1 FXST[X1]=25 FXSW[X1]=5
IF $AA_FXS[X1]=2 GOTOF FXS_ERROR
G01 X400 Y200
```

Inoperative interface signals

The following interface signals (PLC → NCK) are inoperative for axes at the fixed stop until the function is deselected (incl. traversing motion):

- IS “Axis/spindle lock” (DB31, ... DBX1.3)
- IS “Servo enable” (DB31, ... DBX2.1)

Actual position at fixed stop

System variable $AA_IM[x] can determine the actual position of the machine axis, e.g. for test purposes after successful travel to fixed stop.

Applicability with other functions

“Measure with deletion of distance-to-go” (“MEAS” command) and “Travel to fixed stop” cannot be programmed at the same time in one block.

Exception:

One function is acting on a path axis while the other is acting on a positioning axis or both functions are acting on positioning axes.

Contour monitoring

The axis contour monitoring function is inoperative while “Travel to fixed stop” is active.

Positioning axes

When “Travel to fixed stop” is applied to POSA axes, block changes are made independently of the fixed stop motion.

© Siemens AG, 2003. All rights reserved

Vertical axes

In SW 6.2 and higher, the “Travel to fixed stop” function can be used for vertical axes even when alarms are active. Should the traversal of vertical axes be aborted as a result of an FXS alarm when “Travel to fixed stop” is active, the relevant drives are not disconnected from the supply via the VDI interface. This functionality has the same effect on vertical axes as an electronic weight compensation and can be configured via machine data

MD 37052: FIXED_STOP_ALARM_REACTION.

Note

For further details about adaptations for SIMODRIVE 611 digital or digital (HLA module), please see:

References: /FB2/, K3, “Compensation”, Section 2.9 Electronic counterbalancing.

MD 37052

Machine data MD 37052: FIXED_STOP_ALARM_REACTION does not result in disconnection of the drive from the power supply when an alarm is generated, as interface signal “Mode group ready” DB11, ... DBX6.3 remains active.

Bit value=0:
Alarms have an impact on FXS (drives are deenergized as in earlier versions)
IS “Mode group ready” DB11, ... DBX6.3 is deleted.

Bit value=1:
Alarms have no impact on FXS (SW 6.2 and higher)
IS “Mode group ready” DB11, ... DBX6.3 remains active.

- Bit0: Alarm 20090 Travel to fixed stop impossible
- Bit1: Alarm 20091 Fixed stop not reached
- Bit2: Alarm 20092 Travel to fixed stop still active
- Bit3: Alarm 20093 Zero speed monitoring at fixed stop has responded
- Bit4: Alarm 20094 Travel to fixed stop aborted

© Siemens AG, 2003. All rights reserved

2.2.5 Supplementary conditions for SW 5 extensions

Response to pulse blocking

The cancellation of pulse enabling either by terminal 663 or by IS “Pulse enable” (DB31, ... DBX21.7) is in the following referred to as pulse blocking.

The machine data MD 37002: FIXED_STOP_CONTROL can be used to influence the interaction of travel to fixed stop and pulse blocking.

Bit 0: Response in case of pulse blocking at limit as follows:

Bit 0 = 0: Travel to fixed stop is aborted
Bit 0 = 1: Travel to fixed stop is interrupted, i.e. the drive is made dead.

Once the pulse blocking is canceled again, the drive will press at the limited torque again. The torque is connected steplike. At the fixed stop, the drive can be controlled either via

- IS “Pulse enable” (DB31, ... DBX21.7) or via
- drive terminal 663 pulse enable.

The NC evaluates IS “Pulse enable” (DB31, ... DBX21.7) itself.

Depending on drive machine data MD 1012: FUNC_SWITCH, the NC reacts with FXS to a change in the status of terminal 663 as follows:

Bit 2 = 0 The NC does **not** receive the status of terminal 663.

Bit 2 = 1 The NC does receive the status of terminal 663.

Terminal 663

When pulse enabling is canceled by terminal 663, the drive is deenergized and coasts to a standstill immediately. This status is not passed to the NC when

MD 1012: FUNC_SWITCH, Bit 2 = 0.

The status can be checked in line “Pulse enable” (terminal 663) in service display **Service drive**.

Note

Travel to fixed stop can be aborted through blocking the pulses
IS “Pulse enable” (DB31, ... DBX21.7) or through terminal 663 only if

MD 37002: FIXED_STOP_CONTROL Bit 0 = 0 and
MD 1012: FUNC_SWITCH Bit 2 = 1 are set.

If FXS must be interrupted and travel to fixed stop aborted, the following settings are required:

MD 37002: FIXED_STOP_CONTROL Bit 0 = 0 and
MD 1012: FUNC_SWITCH Bit 2 = 0

© Siemens AG, 2003. All rights reserved

Programming FXS in synchronized actions

The function is not available for analog axes (PLC acknowledgment cannot be awaited). If nevertheless a programming is done, the alarm 20090 “Travel to fixed stop not possible. Check programming and axis data” is output.

Select FXS[ ]=1:

The following interface signal is set:
Message to PLC:
IS “Activate travel to fixed stop” (DB31, ... DBX62.4)

The FXS selection command can only be used in systems with digital drives (FDD, MSD, HLA).

The following condition must be observed:

- In MD 37060: FIXED_STOP_ACKN_MASK, Bit 0 = 0 must be set.

Bit 0 = 1 (waiting for PLC acknowledgment) must not be set; otherwise, an interpolator stop would be required to acknowledge the signal, interrupting the movement.

Deselect FXS[ ]=0:

The following interface signal is reset:
Message to PLC:
IS “Activate travel to fixed stop” (DB31, ... DBX62.4)

- MD 37060: FIXED_STOP_ACKN_MASK must have the value zero for signal deselection without motion stop.

Without ramp

The torque limit is changed **without taking into account** the ramp if:

- FXS is activated with (FXS[]=1) to make sure that the reduction is activated immediately (especially for synchronized actions).
- The drive is made dead in case of error as fast as possible.

Selection of FXS with G64

The MD 37060: FIXED_STOP_ACKN_MASK, bit 0 must be 0 (do not wait for PLC input signal “Travel to fixed stop not activated”), since the selection of FXS may not initiate a motion stop. If nevertheless a programming is done, the alarm 20090 “Travel to fixed stop not possible. Check programming and axis data” is output.

Modify the torque FXST

- The clamping torque can already be modified when approaching the stop.
- The torque limit FXST acts in addition to the acceleration limitation with ACC.
- The new torque taking into account the ramp (MD 37012: FIXED_STOP_TORQUE_RAMP_TIME) comes into effect one interpolation cycle after the change in the drive. A change of the effective torque can be checked by reading the synchronized action variable $VA_TORQUE[axis].

© Siemens AG, 2003. All rights reserved

2.2.6 Traversing with limited torque/force FOC (SW 5 and higher)

For applications for which torque or force are to be changed dynamically depending on the travel or on the time or on other parameters (e.g. pressing), the following FOC functionality (Force Control) is provided.

Force/travel or force/time profiles are thus possible using the “Interpolation cycle” resolution.

The function allows torque/force to be modified at any time using synchronized actions. The function can be activated modally or blockrelated.

Modal activation (FOCON/FOCOF)

Activation of the function after POWER_ON and RESET is defined by MD 37080: FOC_ACTIVATION_MODE.

Table 2-1 Controlling the initial setting of the modal limitation of torque/force

After	Bit 0	Effect
POWER ON	0	FOCOF
	1	FOCON
RESET	0	FOCOF
	1	FOCON

FOCON Activation of the modally effective torque/force limitation

FOCOF Disable torque/force limitation

The modal activation acts beyond the program end.
If already programmed, the torque/force set with FXST is active. FXST can be programmed irrespectively of FOCON; it comes into effect, however, only after the function has been activated.

Programming

The programming of the axis is carried out in square brackets. The following is permissible:

- Geometry axis identifiers
- Channel axis identifiers
- Machine axis identifiers

Example:

```
N10 FOCON[X]                    ; modal activation of torque limit

N20 X100 Y200 FXST[X]=15        ; X traverses at reduced torque (15%)
N30 FXST[X]=75 X20              ; change of torque to 75%, X traverses at
                                  this limited torque
N40 FOCOF[X]                    ; disable torque limit
```

Block-related limit (FOC)

The parts program command FOC activates the torque limit for a block.
An activation from a synchronized action acts up to the end of the current parts program block.

© Siemens AG, 2003. All rights reserved

Priority FXS/FOC

The activation of FXS with FOC active has priority, i.e. FXS is executed. A deselection of FXS will cancel the clamping. A modal torque/force limitation remains active. After POWER ON, in the case of activation, MD 37010: FIXED_STOP_TORQUE_DEF is active.
This torque can be modified at any time by programming FXST.

Synchronized actions

The language commands FOC, FOCON, FOCOF can also be programmed in synchronized actions, as the commands for travel to fixed stop.

Determine FOC status

The activation status can be read at any time via the state variable $AA_FOC. If FXS is additionally activated, the status is not changed.

0: FOC not active
1: FOC modally active
2: FOC active blockrelated

Determine torque limit status

The system variables $VA_TORQUE_AT_LIMIT can be used at any time to read in systems with digital drives (FDD, MSD, HLA) whether the currently active torque corresponds to the given torque limit.

0: Effective torque less than torque limit value
1: Effective torque has reached the torque limit value

Restrictions

The function FOC is subject to the following restrictions:

- The change of the torque/force limitation representing itself as an acceleration limitation is only taken into account in the traversing movement at block limits (see command ACC).
- FOC only:
  Monitoring for reaching the active torque limit is not possible from the VDI interface.
- If the acceleration limitation is not adapted accordingly, the following error increases.
- If the acceleration limitation is not adapted accordingly, the end-of-block position is possibly reached later than specified in
  MD 36040: STANDSTILL_DELAY_TIME. Instead of this,
  MD 36042: FOC_STANDSTILL_DELAY_TIME is introduced, which is monitored in this status. If this virtual time is exceeded, the
  alarm 25042 “Zero speed control at torque/force limitation” is triggered and the drive is stopped as fast as possible.

Functional range Link and container axes

All axes that can be traversed in a channel, i.e. also link axes and container axes, can be traversed to fixed stop.

References: /FB2/, B3, “Several Operator Panels and NCUs, Distributed Systems”

The status of the machine axis is kept in the case of a container switch, i.e. a clamped machine axis remains at the stop.

If a modal torque limitation has been activated with FOCON, this is kept for the machine axis even after a container switch.

© Siemens AG, 2003. All rights reserved

2.3 Travel to fixed stop with digital drives SIMODRIVE 611 digital (FDD/MSD)

Selection

The NC detects that the function “Travel to fixed stop” is selected (via the command FXS[x]=1) and signals the PLC that the function has been selected via the IS “Activate travel to fixed stop” (DB31, ... DBX62.4) that the function has been selected.

If MD 37050: FIXED_STOP_ACKN_MASK is set accordingly, then the NC waits for acknowledgment by the PLC in the form of IS “Enable travel to fixed stop” (DB31, ... DBX3.1); and then the function is started. This acknowledgment is not required by the NC for digital drives.

The axis now traverses to the target position at the programmed velocity. At the same time, the clamping torque (specified via FXST[x]) is transferred to the drive via the digital interface, and this limits its effective torque. In addition, the acceleration is reduced accordingly in the NC automatically acc. to FXST[x]).

Fixed stop reached

As soon as the axis reaches the fixed stop, the axial contour deviation increases. If the speed entered in MD 37030: FIXED_STOP_THRESHOLD is exceeded or IS “Sensor fixed stop” (DB31, ... DBX1.2) set, the control system detects that the fixed stop has been reached.

The NC will then delete the distance-to-go and will synchronize the position setpoint to the current actual position value. The controller enabling command remains active.

Once the NC has transferred IS “Fixed stop reached” (DB31, ... DBX62.5) are output to the PLC.

If MD 37060: FIXED_STOP_ACKN_MASK is set accordingly, then the NC waits for acknowledgment by the PLC in the form of IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1). This acknowledgment is not required by the NC for digital drives.

A block change is executed. The clamping torque continues to be applied.

Fixed stop is not reached

If the programmed end position is reached without the “Fixed stop reached” status being recognized, then the torque limitation in the drive is canceled via the digital interface and IS “Activate travel to fixed stop” (DB31, ... DBX62.4) reset.

Depending on the setting in MD 37060: FIXED_STOP_ACKN_MASK, the NC waits for the PLC to acknowledge by resetting IS “Activate travel to fixed stop” (DB31, ... DBX3.1); the program then advances to the next block.

Deselection

Deselection is performed as described in Subsection 2.2.1. In addition, this Section provides further explanations on how to enable the fixed stop alarms and how to enable pulses from the PLC via terminal 663 or the IS “Pulse enable” (DBX31, ... DB21.7).

© Siemens AG, 2003. All rights reserved

Diagram

The following diagram shows the sequence of the motor current, following error and interface signals for “Activate travel to fixed stop” (DB31, ... DBX62.4) and “Fixed stop reached” (DB31, ... DBX62.5) with digital drive (SIMODRIVE 611 digital).

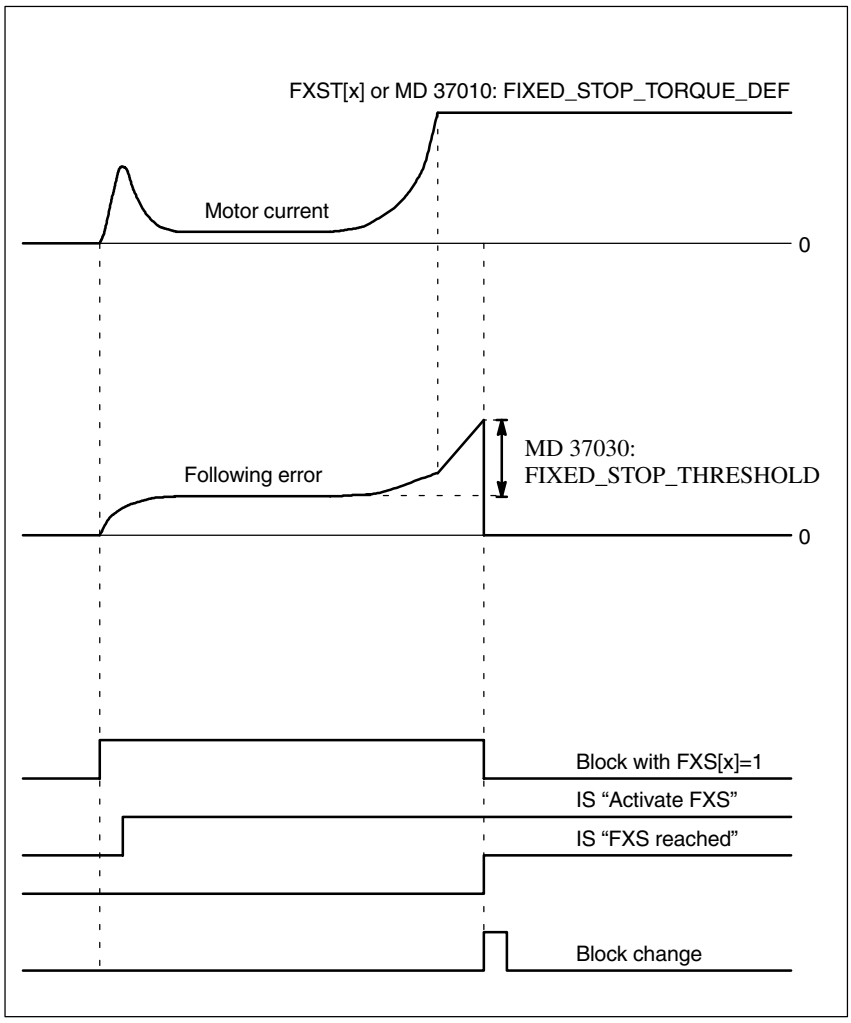


Fig. 2-3 Diagram for FXS with a digital drive (611 digital)

© Siemens AG, 2003. All rights reserved

2.3.1 Travel to fixed stop with hydraulic drives SIMODRIVE 611 digital (HLA module)

Velocity/force control

If the function FXS (FXS[x]=1) is activated for the hydraulic module 611 digital (HLA module), only a change from velocity control to force control takes place. Positioning from the NC is no longer possible in this case.

NC SW6
611 digital SW 5.1

When traveling to fixed stop, the NC evaluates a torque/force limit acting in addition to the limits set on the drive

- Current,
- Force/torque,
- Power, pull-out power,
- Setup mode

Note

For appropriate explanations of velocity and force control as well as all special points to be noted with respect to adaptations for SIMODRIVE 611 digital or digital (HLA module), please see:

References: /FBHLA/, “Drive Functions Firmware”, Chapter 4
/FB2/, K3, “Compensation”, Section 2.9
Electronic counterbalancing.

© Siemens AG, 2003. All rights reserved

2.4 Travel to fixed stop with analog drives

Current/torque control

With the 611 analog (FDD), the torque control and torque limitation has been realized as a current control/current limitation.

2.4.1 SIMODRIVE 611 analog (FDD)

Fixed clamping torque

A fixed current limitation is preset in the drive actuator by means of a resistor circuit (or via R12). This current limitation is activated by the control via a PLC output (which acts on terminal 96 of the actuator), thus ensuring that a fixed clamping torque is available on the axis.

Setpoints can be injected via terminals 56/14 or 24/20.

Programmable clamping torque

In this case, the PLC switches the drive actuator from speed-controlled into current-controlled operation as soon as the fixed stop is reached. Activation of terminal 22 causes the voltage level applied to terminals 20/24 to take effect as a current setpoint rather than a speed setpoint.

The NC is thus able to specify a variable clamping torque.

Setpoints must be injected via terminals 24/20.

References: /IAA/, Installation and StartUp Guide for SIMODRIVE 611
SIMODRIVE 611 Analog System

Hardware connection

Fig. 2-4 shows the hardware connections between FMNC, PLC and SIMODRIVE 611 analog (FDD).

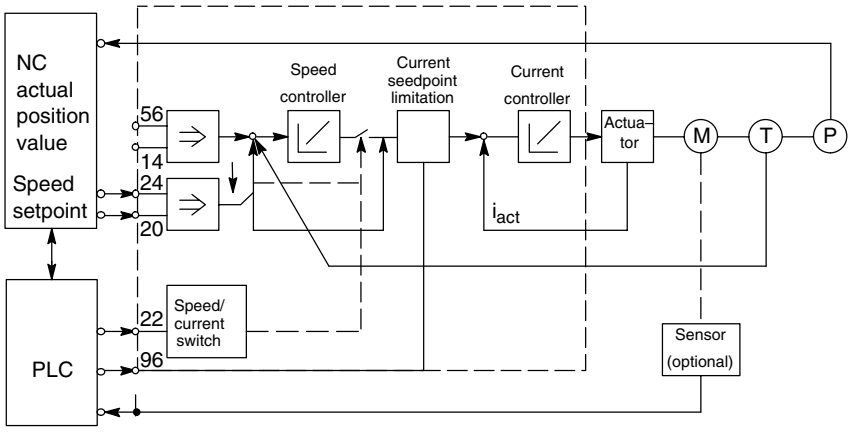


Fig. 2-4 Hardware connection for FM-NC, PLC and SIMODRIVE 611A (FDD)

© Siemens AG, 2003. All rights reserved

Selection

The NC detects that the function “Travel to fixed stop” is selected (via the command FXS[x]=1) and signals the PLC that the function has been selected via the IS “Activate travel to fixed stop” (DB31, ... DBX62.4) that the function has been selected.

The PLC must then activate the current limitation on the actuator (terminal 96).

If MD 37060: FIXED_STOP_ACKN_MASK is set accordingly, then the NC waits for acknowledgment by the PLC in the form of IS “Enable travel to fixed stop” (DB31, ... DBX3.1); and then the function is started.
(This acknowledgment should always be programmed for analog drives for safety reasons, i.e. so that the motion is not started before the current, and thus also the torque, has been limited).

The control then sets the torque limit internally to the value specified in MD 37070: FIXED_STOP_ANA_TORQUE (torque limit on approach to fixed stop for analog drives). This value must correspond to the torque limit value activated via terminal 96. The NC also reduces the acceleration rate automatically in accordance with MD 37070: FIXED_STOP_ANA_TORQUE.

The axis now traverses to the target position at the programmed velocity.

Fixed stop reached

As soon as the axis reaches the fixed stop, the axial contour deviation increases. If the speed entered in MD 37030: FIXED_STOP_THRESHOLD is exceeded or IS “Sensor fixed stop” (DB31, ... DBX1.2) set, the control system detects that the fixed stop has been reached.

The position controller then responds by outputting a speed setpoint corresponding to the value set in MD 37070: FIXED_STOP_ANA_TORQUE. The speed controller, of which the output is limited by terminal 96, forces the drive to the current limit by means of this continuously applied setpoint.

The NC then deletes the remaining distance-to-go and forces the position setpoint to follow. The controller enabling command remains active.

Once the NC has transferred IS “Fixed stop reached” (DB31 , ... DBX62.5) are output to the PLC.

If a **programmable clamping torque** input (via FXST[x] or setting data) from the NC is required, then the PLC must switch the drive over from speed-controlled to current-controlled operation. To do so, it activates terminal 22 and switches off the current limitation (terminal 96) after a period of > 10 ms. The torque set in MD 37070: FIXED_STOP_ANA_TORQUE is now applied to the drive.

If MD 37060: FIXED_STOP_ACKN_MASK is set accordingly, then the NC waits for acknowledgment by the PLC in the form of IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1). Then the requested clamping torque is transferred from the select block to the drive steplike.

A block change is executed. The clamping torque continues to be applied.

Fixed stop is not reached

If the programmed end position is reached without the “Fixed stop reached” status being detected, then the internal torque limitation set in MD 37070: FIXED_STOP_ANA_TORQUE is canceled and the IS “Activate travel to fixed stop” (DB31, ... DBX62.4) reset.

© Siemens AG, 2003. All rights reserved

The PLC must then deactivate the current limitation (terminal 96).

Depending on the setting in MD 37060: FIXED_STOP_ACKN_MASK, the NC waits for the PLC to acknowledge by resetting IS “Activate travel to fixed stop” (DB31, ... DBX3.1); the program then advances to the next block.

Deselection

The NC detects that the function has been deselected on the basis of command FXS[x]=0 and specifies a speed or current setpoint of “0”, i.e. zero clamping torque.

The NC then resets IS “Activate travel to fixed stop” (DB31, ... DBX62.4) and (DB31, ... DBX62.4) and “Fixed stop reached” (DB31, ... DBX62.5).

If current-controlled operation is activated, the PLC must first switch on the current limitation (terminal 96) and switch the drive over to speed-controlled operation (terminal 22) (a speed setpoint of “0” is applied by NC).

Then current limitation must then be deactivated (terminal 96).

Depending on the setting in MD 37060: FIXED_STOP_ACKN_MASK, the NC waits for the PLC to acknowledge by resetting
IS “Enable travel to fixed stop” (DB31, ... DBX3.1) and/or
IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1).

The axis then switches over to position control mode (follow-up mode is terminated) and synchronization with the new actual position takes place.

The programmed travel motion is then executed.

A block change is executed after the target position has been reached.

2.4.2 SIMODRIVE 611 analog (MSD)

Fixed clamping torque

A fixed clamping torque is implemented by entering a fixed torque limitation in a free gear stage in the drive actuator (setting parameter 039). When the “Travel to fixed stop function” is selected, the PLC switches over to the unassigned gear stage of the drive actuator, thus activating the torque limitation.

Setpoints must be injected via terminals 56/14.

Programmable clamping torque

In this case, the PLC switches the drive actuator from speed-controlled into torque-controlled operation after the fixed stop is reached.

The NC can therefore input a variable clamping torque.

Setpoints must be injected via terminals 56/14.

References: /IAA/, Installation and StartUp Guide for SIMODRIVE 611
SIMODRIVE 611 Analog System

© Siemens AG, 2003. All rights reserved

Hardware connection

Fig. 2-5 shows the hardware connections between FM-NC, PLC and SIMODRIVE 611 analog (MSD).

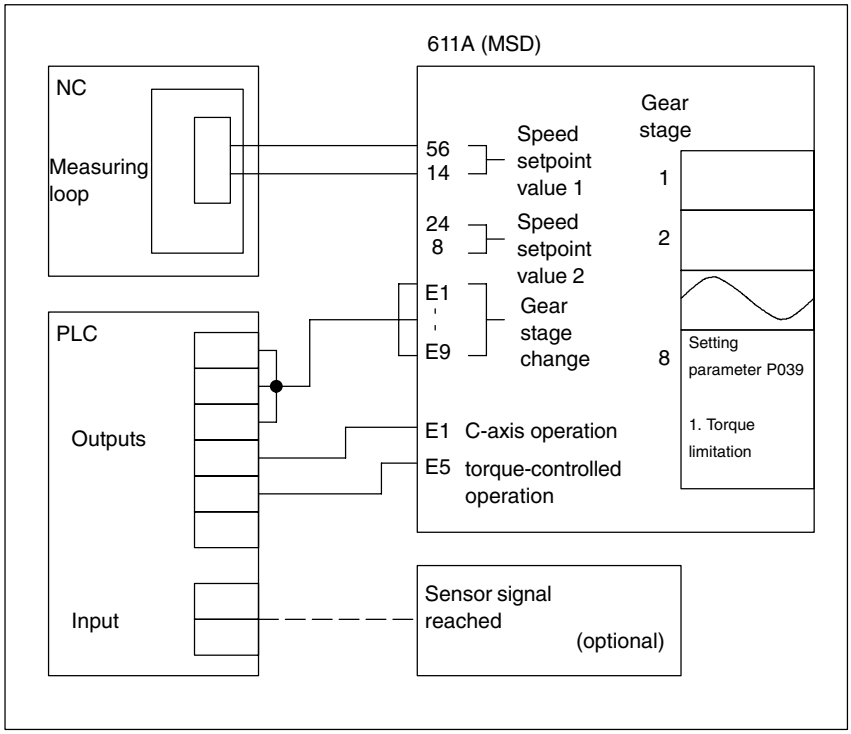


Fig. 2-5 Hardware connection for FM-NC, PLC and SIMODRIVE 611A (MSD)

Caxis operation

The control system has to switch the spindle into C axis mode before the “Travel to fixed stop” function is selected. The PLC does this by activating one of the programmable terminals E1-D9 (e.g. terminal E1) of the drive actuator.

Selection

The NC detects that the function “Travel to fixed stop” is selected (via the command FXS[x]=1) and signals the PLC that the function has been selected via the IS “Activate travel to fixed stop” (DB31, ... DBX62.4) that the function has been selected.

As a result, the PLC activates the unassigned gear stage, in which the torque limitation is effective, by means of programmable terminals E1 - E9 of the drive actuator. It then switches the speed controller monitor off (one terminal (E1-E9) to deactivate error F11 of drive actuator).

If MD 37060: FIXED_STOP_ACKN_MASK is set accordingly, then the NC waits for acknowledgment by the PLC in the form of IS “Enable travel to fixed stop” (DB31, ... DBX3.1); and then the function is started. **(This acknowledgment should always be programmed for analog drives for safety reasons, i.e. so that the motion is not started before the torque has been limited).**

© Siemens AG, 2003. All rights reserved

The control then sets the torque limit internally to the value specified in MD 37070: FIXED_STOP_ANA_TORQUE (torque limit on approach to fixed stop for analog drives). This value must correspond to the torque limit value set in the actuator. The NC also reduces the acceleration rate automatically in accordance with MD 37070: FIXED_STOP_ANA_TORQUE.

The rotary axis now traverses to the target position at the programmed velocity.

Fixed stop reached

As soon as the C-axis reaches the fixed stop, the axial contour deviation increases. If the speed entered in MD 37030: FIXED_STOP_TESHOLD is exceeded or IS Sensor fixed stop (DB31, ... DBX1.2) set, the control system detects that the fixed stop has been reached.

The position controller then responds by outputting a speed setpoint corresponding to the value set in MD 37070: FIXED_STOP_ANA_TORQUE. The speed controller forces the drive to the torque limit by means of this continuously applied setpoint.

The NC then deletes the remaining distance-to-go and forces the position setpoint to follow. The controller enabling command remains active.

Once the NC has transferred IS “Fixed stop reached” (DB31 , ... DBX62.5) are output to the PLC.

If a **programmable clamping torque** input (via FXST[x] or setting data) from the NC is required, then the PLC must switch the drive over from speed-controlled to torque-controlled operation. To do so, it activates one of the programmable terminals E1 – E9 (e.g. terminal E5) and switches off the torque limitation after a period of > 80 ms by selecting the preceding gear stage. The torque set in MD 37070: FIXED_STOP_ANA_TORQUE is now applied to the drive.

If MD 37060: FIXED_STOP_ACKN_ASK is set accordingly, then the NC waits for acknowledgment by the PLC in the form of IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1). Then the requested clamping torque is transferred from the select block to the drive steplike.

A block change is executed. The clamping torque continues to be applied.

Fixed stop is not reached

If the programmed end position is reached without the “Fixed stop reached” status being detected, then the internal torque limitation set in MD 37070: FIXED_STOP_ANA_TORQUE is canceled and the IS “Activate travel to fixed stop” (DB31, ... DBX62.4) reset.

The PLC then activates the preceding gear stage, thus deactivating the torque limitation. It also switches on the speed control monitoring function again.

Depending on the setting in MD 37060: FIXED_STOP_ACKN_MASK, the NC waits for the PLC to acknowledge by resetting IS “Activate travel to fixed stop” (DB31, ... DBX3.1); the program then advances to the next block.

© Siemens AG, 2003. All rights reserved

Deselection

The NC detects that the function has been deselected on the basis of command FXS[x]=0 and specifies a speed or torque setpoint of “0”, i.e. zero clamping torque.

The NC then resets IS “Activate travel to fixed stop” (DB31, ... DBX62.4) and (DB31, ... DBX62.4) and “Fixed stop reached” DB31, ... DBX62.5).

If torque-controlled operation is activated, the PLC must first select the unassigned gear stage in which the torque limitation is effective and switch the drive over to speed-controlled operation (a speed setpoint of “0” is applied by NC). The PLC must also deactivate the speed controller monitoring function.

The PLC then activates the preceding gear stage, thus deactivating the torque limitation. It also switches on the speed control monitoring function again.

Depending on the setting in MD 37060: FIXED_STOP_ACKN_MASK, the NC waits for the PLC to acknowledge by resetting IS “Enable travel to fixed stop” (DB31, ... DBX3.1) and/or IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1).

The C-axis then switches over to position control mode (follow-up mode is terminated) and synchronization with the new actual position takes place.

The programmed travel motion is then executed.

A block change is executed after the target position has been reached.

© Siemens AG, 2003. All rights reserved

2.4.3 Diagrams for travel to fixed stop for analog drives

FXS selection (fixed stop is reached)

The following diagram shows the sequence of the following error and interface signals for “FXS selection” (fixed stop is reached) on analog drives.

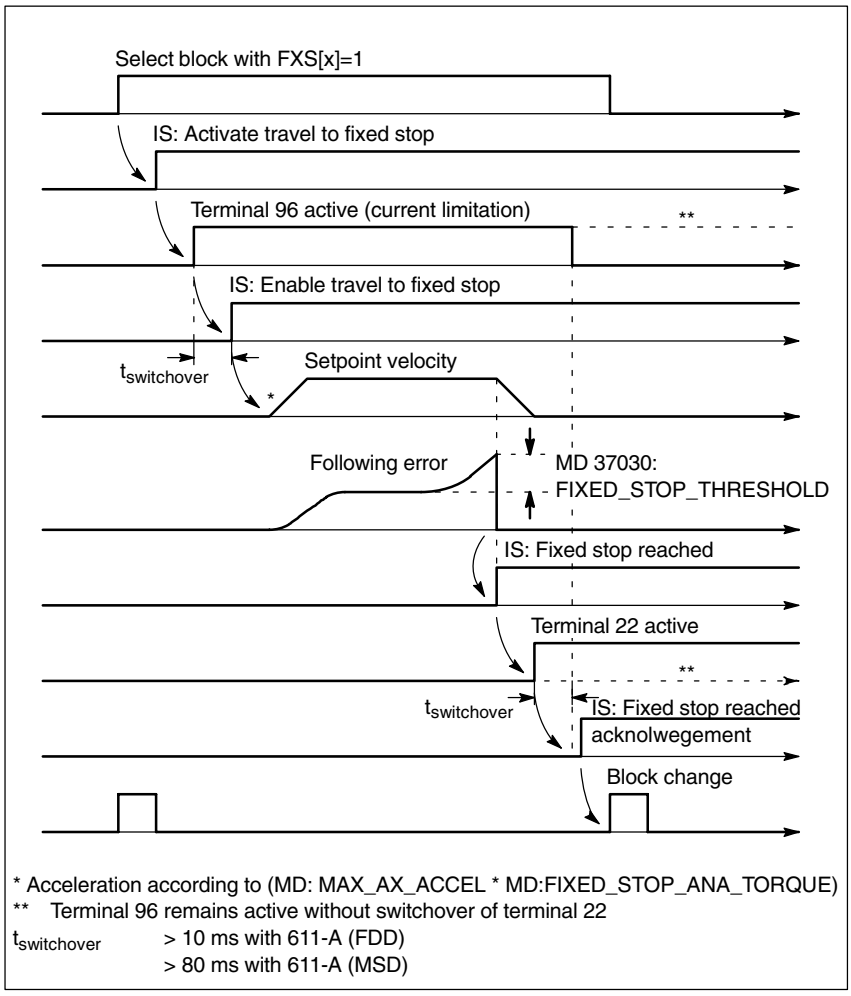


Fig. 2-6 Diagram for FXS selection (fixed stop is reached) with analog drive

© Siemens AG, 2003. All rights reserved

FXS selection (fixed stop is not reached)

The following diagram shows the sequence of the following error and interface signals for “FXS selection” (fixed stop is **not** reached) on analog drives.

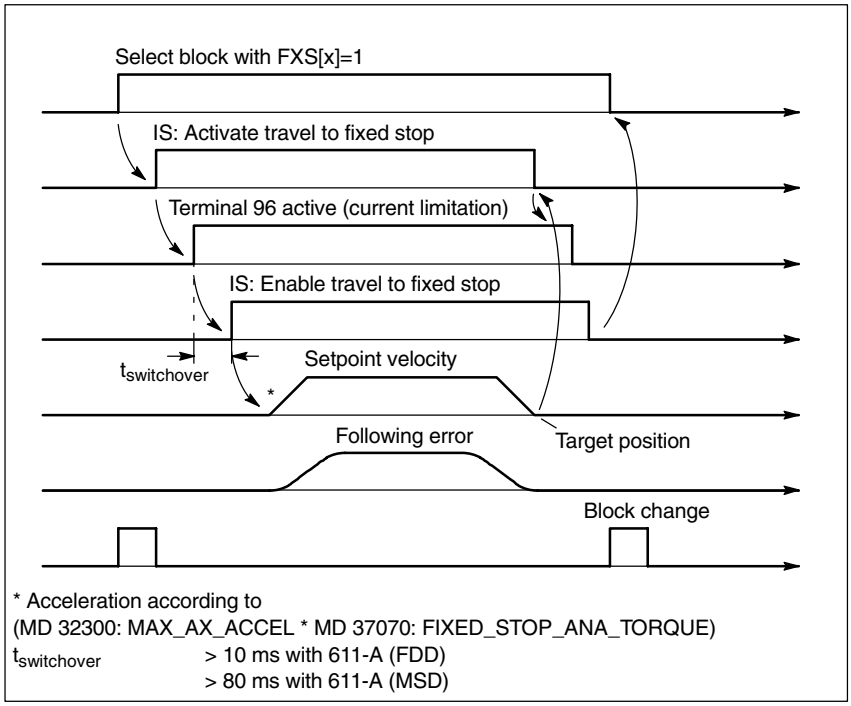


Fig. 2-7 Diagram for FXS selection (fixed stop is not reached) with analog drive

© Siemens AG, 2003. All rights reserved

FXS deselection

The following diagram shows the sequence of the following error and interface signals for “FXS Deselection” on analog drives.

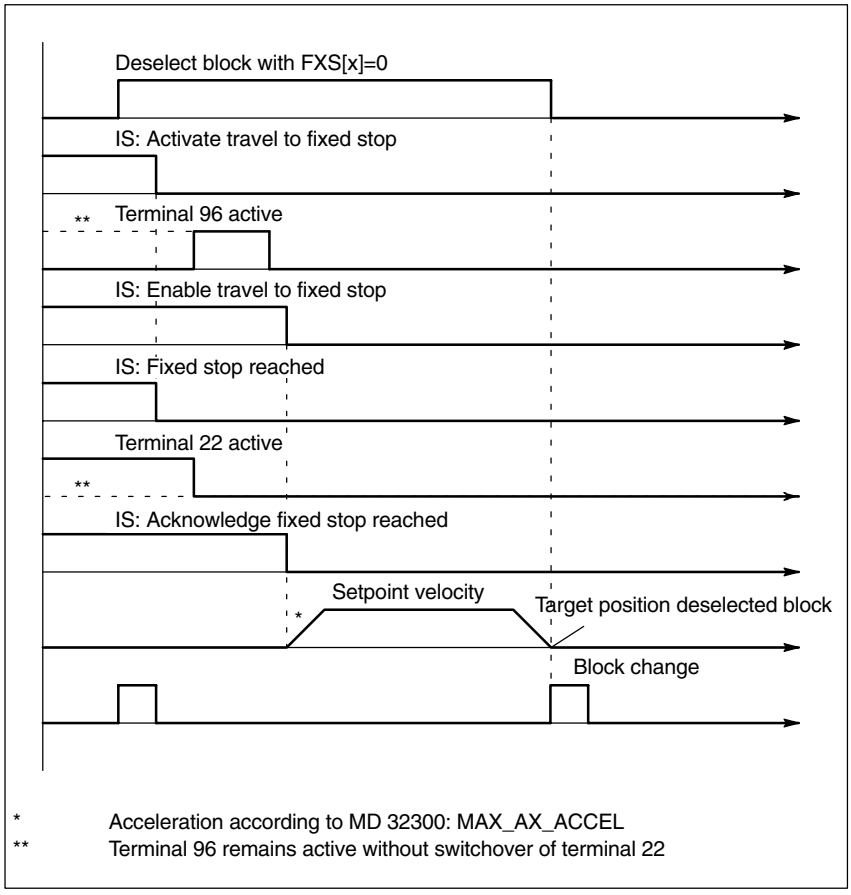


Fig. 2-8 Diagram for FXS deselection with analog drive

© Siemens AG, 2003. All rights reserved

Supplementary Conditions

3

Availability of the "Travel to fixed stop" function

The function is an option and is available for

- SINUMERIK FMNC with NCU 570 with SW 2 and higher
- SINUMERIK 840D with NCU 571/572/573 with SW 2 and higher
- SINUMERIK 840D with NCU 572.3/573, expansions with SW 5.3 and higher
- SINUMERIK 810D with CCU3, SW 6.3 and higher
- SINUMERIK 840Di SW 1.1

Availability of the Force Control (FOC) function

The FOC function is available is available only with

- SINUMERIK 840D with NCU 573, expansions with SW 5.2 and higher.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

4.1 Axis-specific machine data

36042 MD number	**FOC_STANDSTILL_DELAY_TIME** Delay time of zero speed control with FOC and FXS		
Default setting: 0.4	Minimum input limit: 0.0	Maximum input limit: plus	
Change effective after NEWCONF	Protection level: 2 / 7	Unit: s	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	Wait time between the end of a movement (IPO setpoint = 0) and activation of the zero speed control with active torque limit. If the configurable end-of-block criterion occurs during this time, the zero speed control is activated. If the configurable end-of-block criterion does not occur during this delay time, alarm 25042 is triggered.		

37000 MD number	**FIXED_STOP_MODE** Travel to fixed stop mode		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/4	Unit: –	
Data type: BYTE	Applies from SW 2.1		
Significance:	This machine data defines how the “Travel to fixed stop” function can be started. MD=0 Travel to fixed stop not available (option missing). MD=1 Travel to fixed stop can be started from the NC program with command FXS[x]=1.		

37002 MD number	**FIXED_STOP_CONTROL** Special function when traveling to fixed stop		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 5.3		
Significance:	Switch for special functions when traveling to fixed stop. Bit 0: Response in case of pulse blocking at limit = 0: Travel to fixed stop is aborted = 1: Travel to fixed stop is interrupted, i.e. the drive is made dead. As soon as the pulse blocking is deactivated the drive operates with, limited torque again. The torque is connected step-like.		

© Siemens AG, 2003. All rights reserved

37010 MD number	**FIXED_STOP_TORQUE_DEF** Default for clamping torque		
Default setting: 5.0	Minimum input limit: 0	Maximum input limit: 100.0	
Changes effective after POWER ON	Protection level: 2/4	Unit: %	
Data type: DOUBLE	Applies from SW 2.1		
Significance:	This machine data indicates the clamping torque as a % of the static torque MD 1118: MOTOR_STANDSTILL_CURRENT of the drive/as a % of the motor torque MD 1103: MOTOR_NOMINAL_CURRENT on main spindle drives. The clamping torque becomes operative as soon as the fixed stop is reached or “IS Acknowledge fixed stop reached” (DB31, ... DBX1.1) has been set. The entered value is a default and is effective only on the condition that • no clamping torque has been programmed with command FXST[x]. • the clamping torque set in SD 43510: FIXED_STOP_TORQUE has not been changed (after fixed stop was reached). When “Traveling to fixed stop” with an analog drive (611–A) and **fixed clamping torque**, the torque limit set on the drive should be equal to the torque limit set in MD 37070: FIXED_STOP_ANA_TORQUE.		
Related to	MD 37070: FIXED_STOP_ANA_TORQUE (torque limit on approach to fixed stop for analog drives) SD 43510: FIXED_STOP_TORQUE (clamping torque for travel to fixed stop)		

37012 MD number	**FIXED_STOP_TORQUE_RAMP_TIME** Virtual time until reaching the new clamping torque when traveling to fixed stop		
Default setting: 0.0.	Minimum input limit: 0.0	Maximum input limit: plus	
Change valid after NEWCONF	Protection level: 2 / 7	Unit: s	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	Virtual time until reaching the modified torque limit. The subdivision takes place in the position controller cycle and takes place there step-like. The value 0.0 disables the ramp function.		

37020 MD number	**FIXED_STOP_WINDOW_DEF** Default for fixed stop monitoring window		
Default setting: 1.0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/4	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 2.1		
Significance:	The default setting for the fixed stop monitoring window is set in this machine data. The fixed stop monitoring window becomes operative as soon as the fixed stop is reached, i.e. IS “Fixed stop reached” (DB31, ... DBX62.5) is set. If the axis/spindle leaves the position at which the fixed stop has been detected by more than the tolerance specified in MD: FIXED_STOP_WINDOW_DEF, then alarm 20093 “Fixed stop monitoring has responded” is output and the “FXS” function deselected. The entered value is a default and is effective only on the condition that • no fixed stop monitoring window has been programmed with command FXSW[x]. • the fixed stop monitoring window set via SD 43520: FIXED_STOP_WINDOW has not been changed (after fixed stop was reached).		
Related to	SD 43520: FIXED_STOP_WINDOW (fixed stop monitoring window)		

© Siemens AG, 2003. All rights reserved

37030 MD number	**FIXED_STOP_THRESHOLD** Threshold for fixed stop detection		
Default setting: 2.0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/4	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 2.1		
Significance:	The contour monitoring threshold for fixed stop detection is entered in this machine data. This machine data is effective only if MD: FIXED_STOP_BY_SENSOR is set to 0. IS “Fixed stop reached” (DB31, ... DBX62.5) is set if the axial contour deviation exceeds the threshold set in MD: FIXED_STOP_THRESHOLD.		
MD irrelevant for	MD 37040: FIXED_STOP_BY_SENSOR = 1		
Related to	IS “Fixed stop reached” (DB31, ... DBX62.5)		

37040 MD number	**FIXED_STOP_BY_SENSOR** Fixed stop detection via sensor		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 2.1		
Significance:	This machine data defines how the criterion “Fixed stop reached” is determined. MD=0 The criterion “Fixed stop reached” is determined internally on the basis of the axial contour deviation (threshold specified by MD: FIXED_STOP_THRESHOLD). MD=1 The criterion “Fixed stop reached” is detected via an external sensor and transmitted to the NC by means of IS “Sensor for fixed stop” (DB31, ... DBX1.2). MD=2 The criterion “Fixed stop reached” is assumed if either the contour monitoring (acc. to MD = 0) or the signal of the external sensor (acc. to MD = 1) has responded accordingly.		
Related to	MD 37030: FIXED_STOP_THRESHOLD (threshold for fixed stop detection) IS “Sensor for fixed stop” (DB31, ... DBX1.2)		

37050 MD number	**FIXED_STOP_ALARM_MASK** Enabling the fixed stop alarms		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 7	
Changes effective after POWER ON SW 5 and higher: NEWCONF	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 2.1		
Significance:	This machine data is set to define whether the alarms 20091 “Fixed stop not reached” and 20094 “Fixed stop aborted” must be output. MD=0 Suppression of alarm 20091 “Fixed stop not reached” MD=2 Suppression of alarms 20091 “Fixed stop not reached” and 20094 “Fixed stop aborted” (SW 4 and higher) MD=3 Suppression of alarm 20094 “Fixed stop aborted” (SW 4 and higher) All other permissible values of ≤ 7 do not suppress any alarms.		

© Siemens AG, 2003. All rights reserved

37052 MD number	**FIXED_STOP_ALARM_REACTION** Reaction to fixed stop alarms		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/4	Unit: –	
Data type: BYTE	Applies from SW 6.2		
Significance:	Reaction of IS “Mode group ready” (DB11, ... DBX6.3) to fixed stop alarms: Bit value = 0: IS “Mode group ready” (DB11, ... DBX6.3) is deleted (drive de-energized) Bit value = 1: IS “Mode group ready” (DB11, ... DBX6.3) remains active. Bit 0: Alarm 20090 Travel to fixed stop impossible. Bit 1: Alarm 20091 Fixed stop not reached Bit 2: Alarm 20092 Travel to fixed stop still active Bit 3: Alarm 20093 Zero speed monitor at fixed stop has responded. Bit 4: Alarm 20094 Travel to fixed stop aborted. None of the other bits are significant. Standard: Bit values = 0: All alarms result in deenergization of the drives.		
Related to	IS “Mode group ready” (DB11, ... DBX6.3)		

37060 MD number	**FIXED_STOP_ACKN_MASK** Monitoring of PLC acknowledgments for travel to fixed stop		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/4	Unit: –	
Data type: BYTE	Applies from SW 2.1		
Significance:	This machine data determines whether or not the NC waits for acknowledgment messages from the PLC when the “Travel to fixed stop“ function is active. Bit 0 = 0 Once the NC has transmitted IS “Activate travel to fixed stop” (DB31, ... DBX62.4) to the PLC, it starts the programmed travel motion. Bit 0 = 1 After the NC has transferred IS “Activate travel to fixed stop” (DB31, ... DBX62.4) to the PLC, it waits for the PLC to acknowledge with IS “Enable travel to fixed stop“ (DB31, ... DBX3.1) and then starts the programmed travel motion. Bit 0 should be set to “1” for analog drives so that the motion is not started until the PLC has limited the drive torque. Bit 1 = 0 Once the NC has transferred IS “Fixed stop reached” (DB31 , ...DBX62.5) to the PLC, the program advances to the next block. Bit 1 = 1 After the NC has transferred IS “Fixed stop reached” (DB31, ... DBX62.5) to the PLC, it waits for the PLC to acknowledge with IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1), outputs the programmed torque and then executes a block change. Bit 1 should be set to 1 for analog drives so that the PLC can switch the drive over to torque-controlled operation if a programmable clamping torque must be specified. The “Travel to fixed stop” function can be executed without acknowledgments in digital drives (611D) allowing program run times to be reduced.		
Related to	IS “Activate travel to fixed stop” (DB31, ... DBX62.4) IS “Enable travel to fixed stop” (DB31, ... DBX3.1) IS “Fixed stop reached” (DB31, ... DBX62.5) IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1)		

© Siemens AG, 2003. All rights reserved

37070 MD number	**FIXED_STOP_ANA_TORQUE** Torque limit on fixed stop approach for analog drives		
Default setting: 5.0	Minimum input limit: 0	Maximum input limit: 100.0	
Changes effective after POWER ON	Protection level: 2/4	Unit: %	
Data type: DOUBLE	Applies from SW 2.1		
Significance:	This machine data defines an internal NC torque limit for analog drives. It is specified as a % of the maximum drive torque (corresponds to % of max. current setpoint with FDD). This torque limit is effective in the NC from the start of the motion (acceleration torque) until the instant the fixed stop is reached. The torque limit must have the same effect as the torque limit set in the drive (611A). This limit is required to ensure that • there are no step changes in torque during switchover from speed-controlled to current-controlled or torque-controlled operation. • the acceleration is reduced to the correct value in the NC.		
MD irrelevant for	SINUMERIK 840D with SIMODRIVE 611D		

37080 MD number	**FOC_ACTIVATION_MODE** Status of activation of the modal function FOC		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level:	Unit: –	
Data type: BYTE	Applies from SW 5.3		
Significance:	The status of the FOC function after POWER ON and RESET can be set by setting or deleting individual bits of this MD: Bit 0: Response after POWER ON = 0: FOCOF = 1: FOCON Bit 1: Response after RESET = 0: FOCOF = 1: FOCON		

© Siemens AG, 2003. All rights reserved

4.2 Axis-specific setting data

43500 SD number	**FIXED_STOP_SWITCH** Selection of travel to fixed stop		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 7/7	Unit: –	
Data type: BYTE	Applies from SW 2.1		
Significance:	This setting data allows the “Travel to fixed stop” function to be selected and deselected. SD=0 Deselect “Travel to fixed stop” function SD=1 Select “Travel to fixed stop” function When SW 2.x is installed, this data can be overwritten only through the parts program by means of command FXS[x]=1/0. The status of the setting data is displayed via the operator panel in the “Parameters” area.		

43510 SD number	**FIXED_STOP_TORQUE** Clamping torque for travel to fixed stop		
Default setting: 5.0	Minimum input limit: 0.0	Maximum input limit: 800.0	
Changes effective immediately	Protection level: 7/7	Unit: %	
Data type: DOUBLE	Applies from SW 2.1, max. 800.0 (SW 5.3 and higher)		
Significance:	This setting data indicates the clamping torque as a % of the static torque MD 1118: MOTOR_STANDSTILL_CURRENT of the drive/as a % of the motor torque MD 1103: MOTOR_NOMINAL_CURRENT on main spindle drives. Please observe that a clamping torque greater than 100% may only be present for a short time; otherwise, the motor will be damaged. With hydraulic drives, it is limited to 100%, since the reference parameter is the maximum force of the drive. When selecting the function “Travel to fixed stop” through programming, the default of MD 37010 FIXED_STOP_TORQUE_DEF is active up to the programming with FXST[.]. In the case of FOC, the setting data will also act as a torque limit. Time of effect: Digital drives: SD acts already when approaching the stop. Analog drives: The SD comes into effect when the stop is detected. The **Fixed stop is reached**, if – with MD 37060: FIXED_STOP_ACKN_MASK, Bit 1 = 0 (no acknowledgment required) IS “Fixed stop reached” (DB31, ... DBX62.5) is set by the NC. – with MD 37060: FIXED_STOP_ACKN_MASK, Bit 1 = 1 (acknowledgment required) IS “Fixed stop reached” (DB31, ... DBX62.5) is set by the NC and acknowedged with IS “Acknowledge fixed stop reached (DB31, ... DBX1.1). The status of the setting data is displayed via the operator panel in the “Parameters” area. The FXST[x] command effects a block-synchronous change to this data. The setting data can be modified by the operator, by programming or via the PLC.		
Related to	MD 37010: FIXED_STOP_TORQUE_DEF (default setting for clamping torque) MD 10710: MN_PROG_SD_RESET_SAVE_TAB		

© Siemens AG, 2003. All rights reserved

43520 SD number	**FIXED_STOP_WINDOW** Fixed stop monitoring window		
Default setting: 1.0	Minimum input limit: 0.0	Maximum input limit: plus	
Changes effective immediately	Protection level: 7/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 2.1		
Significance:	The fixed stop monitoring window is entered in this setting data. The setting data is effective only if the fixed stop has been reached. The **Fixed stop is reached**, if – with MD: FIXED_STOP_ACKN_MASK, Bit 1 = 0 (no acknowledgment required) IS “Fixed stop reached” (DB31, ... DBX62.5) is set by the NC. – with MD: FIXED_STOP_ACKN_MASK, Bit 1 = 1 (acknowledgment required) IS “Fixed stop reached” (DB31, ... DBX62.5) is set by the NC and acknowledged with IS “Acknowledge fixed stop reached” (DB31, ... DBX1.1). If the position at which the fixed stop was detected leaves the tolerance band by more than the amount specified in SD 43520: FIXED_STOP_WINDOW_DEF, then alarm 20093 “Fixed stop monitoring has responded” is output and the function “FXS” is deselected. The status of the setting data is displayed via the operator panel in the “Parameters” area. The FXSW[x] command effects a block-synchronous change to this setting data. The setting data can also be changed by the user or via the PLC. The value is otherwise transferred from MD: FIXED_STOP_WINDOW_DEF to the setting data when “Travel to fixed stop“ is active.		
Related to	MD 37020: FIXED_STOP_WINDOW_DEF (default setting for fixed stop monitoring window) MD 10710: MN_PROG_SD_RESET_SAVE_TAB		

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

5 Signal Descriptions

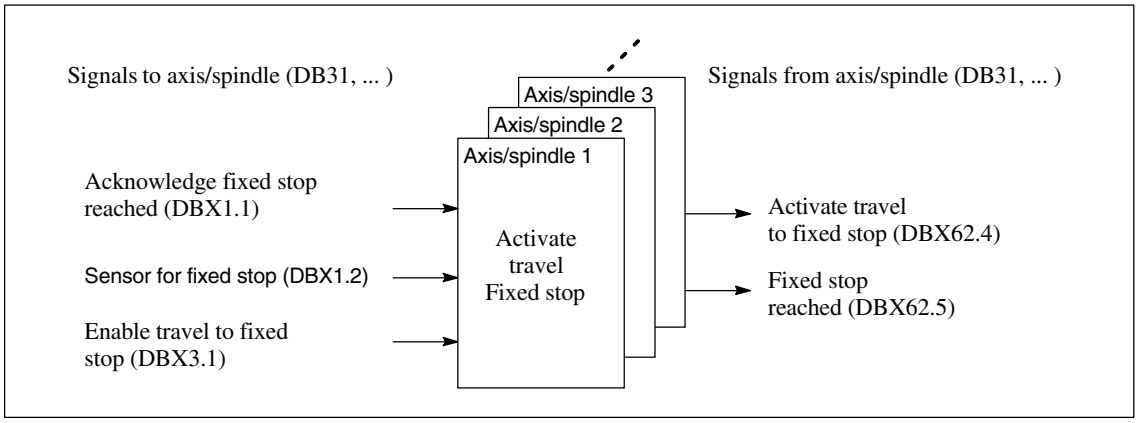


Fig. 5-1 PLC interface signals for “Travel to fixed stop”

© Siemens AG, 2003. All rights reserved

5.1 Axis/spindle-specific signals

5.1.1 Signals to axis/spindle

DB 31, ... **DBX1.1** Data block	**Acknowledge fixed stop reached** Signal(s) to axis/spindle (PLC → NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	After fixed stop has been reached: IS “Fixed stop reached” DB31, ... DBX62.5 = 1 → The axis pushes against the fixed stop with clamping torque → The fixed stop monitoring window is activated → The program advances to the next block		
Signal state 0 Signal transition 1 ——> 0	After fixed stop has been reached: IS “Fixed stop reached” DB31, ... DBX62.5 = 1 → The axis pushes against the fixed stop with clamping torque → The fixed stop monitoring window is activated → The program does not advance to the next block and channel message “Wait: Auxiliary function acknowledgment missing” is displayed After fixed stop has been reached: IS “Fixed stop reached” DB31, ... DBX62.5 = 1 The function is aborted, the alarm “20094 axis %1 Function aborted” is displayed. Meaning when function “FXS = 0” is deselected in parts program: The torque limitation and monitoring of the fixed stop window are canceled.		
IS irrelevant for	MD 37060: FIXED_STOP_ACKN_MASK (monitoring of PLC acknowledgments for travel to fixed stop) = 0 or 1		
Related to	MD 37060: FIXED_STOP_ACKN_MASK (monitoring of PLC acknowledgments for travel to fixed stop) IS “Fixed stop reached” (DB31, ... DBX62.5)		

DB 31, ... **DBX1.2** Data block	**Sensor for fixed stop** Signal(s) to axis/spindle (PLC → NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	Fixed stop is reached.		
Signal state 0 or signal transition 1 ——> 0	Fixed stop is not reached.		
Related to	The signal is effective only if MD 37040: FIXED_STOP_BY_SENSOR is set to 1.		

© Siemens AG, 2003. All rights reserved

DB 31, ... DBX3.1 Data block	**Enable travel to fixed stop** Signal(s) to axis/spindle (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	Meaning when “FXS” function is selected via parts program, (IS “Activate travel to fixed stop” = 1): Travel to fixed stop is enabled and the axis traverses from the start position at the programmed velocity to the programmed target position.	
Signal state 0 Signal transition 1 ——> 0	Meaning when “FXS” function is selected via parts program, (IS “Activate travel to fixed stop” = 1): → Travel to fixed stop is disabled. → The axis is stationary with reduced torque at the start position. → The channel message “Wait: Auxiliary function acknowledgment missing” is displayed. Meaning before the fixed stop is reached IS “Fixed stop reached” DB31, ... DBX62.5 = 0. → Travel to fixed stop is aborted → Alarm “20094: axis%1 Function aborted” is displayed Meaning after the fixed stop has been reached IS “Fixed stop reached” DB31, ... DBX62.5 = 1. The torque limitation and monitoring of the fixed stop monitoring window are canceled. Selection “Acknowledge fixed stop reached” DB31, ...DBX1.1.	
IS irrelevant for	MD 37060: FIXED_STOP_ACKN_MASK (monitoring of PLC acknowledgments for travel to fixed stop) = 0 or 2	
Related to	MD 37060: FIXED_STOP_ACKN_MASK (monitoring of PLC acknowledgments for travel to fixed stop) IS “Activate travel to fixed stop” (DB31, ... DBX62.4)	

5.1.2 Signals from axis/spindle

DB 31, ... DBX62.4 Data block	**Activate travel to fixed stop** Signal(s) from axis/spindle (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	The “Travel to fixed stop function” is active. This signal is used for analog drives in order, for example, to activate the current or torque limitation parameterized in the actuator.	
Signal state 0 or signal transition 1 ——> 0	The “Travel to fixed stop function” is not active.	

DB 31, ... DBX62.5 Data block	**Fixed stop reached** Signal(s) from axis/spindle (NCK → PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 2.1
Signal state 1 or signal transition 0 ——> 1	The fixed stop was reached after selection of the “FXS” function. This signal is used by analog drives, e.g. to switch the actuator from speed-controlled to current or torque-controlled mode so that a programmable clamping torque can be set.	
Signal state 0 or signal transition 1 ——> 0	The fixed stop has still not been reached after selection of the “FXS” function.	

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Example

6

6.1 Traveling to fixed stop, examples (SW 5.3 and higher)

Static synchronized actions

Travel to fixed stop (FXS), initiated by a synchronized action

```
N10 IDS=1 WHENEVER                ; Activate static synchronized action:
(($R1==1) AND                     ; Setting $R1=1
($AA_FXS[Y]==0)) DO               ; activates
$R1=0 FXS[Y]=1                    ; FXS for the Y axis
FXST[Y]=10                        ; the effective torque is reduced to 10%
FA[Y]=200                         ; and a traversing movement is started
POS[Y]=150                        ; in the direction of the stop

N11 IDS=2 WHENEVER                ; Once the stop has bee detected,
($AA_FXS[Y]==4) DO                ; ($AA_FXS[Y]==4), the torque is
FXST[Y]=30                        ; increased to 30%

N12 IDS=3 WHENEVER                ; After the stop has been reached,
($AA_FXS[Y]==1) DO                ; the torque is controlled
FXST[Y]=$R0                       ; depending on R0

N13 IDS=4 WHENEVER                ; Selection depending on
(($R3==1) AND                     ; R3 and
($AA_FXS[Y]==1)) DO               ; travel back
FXS[Y]=0                          ;
FA[Y]=1000 POS[Y]=0               ;

N20 FXS[Y]= 0                     ; Normal program run
G0 G90 X0 Y0                      ;

N30 RELEASE(Y)                    ; Enable Y axis for the movement in
                                  ; the synchronized action

N40 G1 F1000 X100                 ; Movement of another axis

N50 .....                         ;

N60 GET(Y)                        ; Include Y axis again in
                                  ; path group
```

© Siemens AG, 2003. All rights reserved

Multiple selection

A selection may only be carried out once. If the function is called once more due to faulty programming (FXS[axis]=1), the alarm 20092 “Travel to fixed stop still active” is initiated.
A programming that polls either $AA_FXS[] or its own flag (here: R1) in the condition, avoids that the function is activated several times.

Programming sample (parts program fragment):

```
N10 R1=0
N20 IDS=1 WHENEVER ($R1 == 0 AND $AA_IW[AX3]>7) DO R1=1
FXS[AX1]=1 FXST[AX1]=12
```

Block-related synchronized actions

By programming a block-related synchronized action, travel to fixed stop can be connected during an approach motion.

Programming example:

```
N10 G0 G90 X0 Y0
N20 WHEN $AA_IW[X]>17 DO FXS[X]=1 ; If X reaches a position greater
N30 G1 F200 X100 Y110             ; than 17mm, FXS is activated
```

■

© Siemens AG, 2003. All rights reserved

7 Data Fields, Lists

7.1 Interface signals

DB number	Bit, byte	Name	Refer-ence
Axis/spindle-specific			
31–48	1.1	Acknowledge fixed stop reached	
31–48	1.2	Sensor for fixed stop	
31–48	1.3	Axis/spindle disable	A2
31–48	2.1	Servo enable	A2
31–48	3.1	Enable travel to fixed stop	
31–48	62.4	Activate travel to fixed stop	
31–48	62.5	Fixed stop reached	

© Siemens AG, 2003. All rights reserved

7.2 Machine data

Number	Identifier	Name	Reference
Axis-specific ($MA_ ...)			
36042	FOC_STANDSTILL_DELAY_TIME	Delay time of zero speed control with FOC and FXS (SW 5.3 and higher)	
37000	FIXED_STOP_MODE	Travel to fixed stop mode	
37002	FIXED_STOP_CONTROL	Special functions when traveling to fixed stop (SW 5.3 and higher)	
37010	FIXED_STOP_TORQUE_DEF	Default for clamping torque	
37012	FIXED_STOP_TORQUE_RAMP_TIME	Virtual time until reaching the new clamping torque when traveling to fixed stop (SW 5.3 and higher)	
37020	FIXED_STOP_WINDOW_DEF	Default for fixed stop monitoring window	
37030	FIXED_STOP_THRESHOLD	Threshold for fixed stop detection	
37040	FIXED_STOP_BY_SENSOR	Fixed stop detection via sensor	
37050	FIXED_STOP_ALARM_MASK	Enabling the fixed stop alarms	
37052	FIXED_STOP_ALARM_REACTION	Reaction to fixed stop alarms with SW 6.2 and higher	
37060	FIXED_STOP_ACKN_MASK	Monitoring of PLC acknowledgments for travel to fixed stop	
37070	FIXED_STOP_ANA_TORQUE	Torque limit on fixed stop approach for analog drives	
37080	FOC_ACTIVATION_MODE	Status of activation of the modal function FOC (SW 5.3 and higher)	

7.3 Setting data

Number	Identifier	Name	Reference
Axis-specific ($SA_ ...)			
43500	FIXED_STOP_SWITCH	Selection of travel to fixed stop	
43510	FIXED_STOP_WINDOW	Clamping torque when traveling to fixed stop extended to a torque greater than 100% (SW 5.3 and higher)	
43520	FIXED_STOP_TORQUE	Fixed stop monitoring window	

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, Diagnostics Guide
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Velocities, Setpoint/Actual Value Systems, Closed-Loop Control (G2)

1 Brief Description 1/G2/1-3

2 Detailed Description 1/G2/2-5

2.1 Velocities, traversing ranges, accuracies 1/G2/2-5
2.1.1 Velocities 1/G2/2-5
2.1.2 Traversing ranges 1/G2/2-6
2.1.3 Positioning accuracy of the control system 1/G2/2-7
2.1.4 Block diagram of resolutions and normalization values 1/G2/2-7
2.1.5 Input/display resolution, calculation resolution 1/G2/2-9
2.1.6 Standardizing physical quantities of machine and setting data 1/G2/2-10

2.2 Metric/inch measuring system 1/G2/2-13
2.2.1 Conversion of basic system by parts program 1/G2/2-13
2.2.2 Manual switchover of the basic system (SW 5 and higher) 1/G2/2-17
2.2.3 FGROUP and FGREF (SW 5 and higher) 1/G2/2-21

2.3 Setpoint/Actual value system 1/G2/2-23
2.3.1 General 1/G2/2-23
2.3.2 Speed setpoint and actual value routing 1/G2/2-25
2.3.3 Configuration of drives 1/G2/2-30
2.3.4 Adapting the motor/load ratios 1/G2/2-31
2.3.5 Speed setpoint output and actual value processing 1/G2/2-34
2.3.6 Adjustments to actual-value resolution 1/G2/2-38

2.4 Closed-loop control 1/G2/2-43
2.4.1 Parameter sets of the position controller 1/G2/2-46
2.4.2 Extension of the parameter set (SW 5.1 and higher) 1/G2/2-47

2.5 Optimization of the control (SW 5.0 and higher) 1/G2/2-50
2.5.1 Position controller: Positional deviation injection (SW 5.1 and higher) 1/G2/2-50
2.5.2 Position controller position setpoint filter: New balancing filter (SW 5.0 and higher) 1/G2/2-51
2.5.3 Position controller position setpoint filter: New jerk filter (SW 5.1 and higher) 1/G2/2-56
2.5.4 Position feedback loop with PI controller (SW 6.4 and higher) 1/G2/2-58

© Siemens AG, 2003. All rights reserved

2.5.5 System variable for the status of pulse enable (SW 5.1 and higher) 1/G2/2-60
2.5.6 Extensions for “deceleration axes” (SW 5.3 and higher) 1/G2/2-61

3 Supplementary Conditions 1/G2/4-63

4 Data Descriptions (MD, SD) 1/G2/4-63

4.1 General machine data 1/G2/4-63

4.2 Axis-specific machine data 1/G2/4-75

5 Signal Descriptions 1/G2/7-91

6 Example 1/G2/7-91

7 Data Fields, Lists 1/G2/7-91

7.1 Machine data 1/G2/7-91

7.2 Alarms 1/G2/7-94

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

This Section describes the parameter settings for

- the actual value or measuring systems
- the setpoint value system
- the operating accuracy
- the traversing ranges
- the axis velocities and
- the control parameters

on a machine axis.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 Velocities, traversing ranges, accuracies

2.1.1 Velocities

The maximum path/axis velocities and spindle speed are influenced by the machine design, the dynamic response of the drive and the limit frequency of the actual value sensing system (encoder limit frequency).

The maximum axis velocity is defined in MD 32000: MAX_AX_VELO (maximum axis velocity). The maximum permissible spindle speed is preset in MD 35100: SPIND_VELO_LIMIT (maximum spindle speed). For an explanation see:
References: /FB/, S1, “Spindles”

In addition to the limitation through MAX_AX_VELO, the control system also limits the maximum path velocity according to the prevailing situation according to the following formula:

$$V_{max} \leq \frac{\textit{progr. path length in a part program block [mm or deg]}}{\textit{IPO clock cycle [s]}} * 0.9$$

For details on setting the IPO clock cycle see
References: /FB/, G3, “Cycle times”

With a high feedrate (resulting from programmed feedrates and feedrate override) the maximum path velocity is limited to V_{max}.

This automatic feedrate limitation can lead to a drop in velocity over several blocks with programs generated by CAD systems with extremely short blocks.

Example:
IPO cycle = 12 ms
N10 G0 X0 Y0; [mm]
N20 G0 X100 Y100; [mm]
⇒ programmed path distance in block = 141.42 mm

⇒ V_{max} = (141.42 mm / 12 ms) * 0.9 = 10606.6 mm/s = 636.39 m/min

The following restriction applies to the minimum path or axis velocity:

$$V_{min} \geq \frac{10^{-3}}{\textit{Calculation Resolution } [\frac{\textit{incr.}}{\textit{mm or deg.}}] * \textit{IPO clock cycle [s]}}$$

The calculation resolution is defined in MD 10200: INT_INCR_PER_MM (calculation resolution for linear positions) or MD 10210: INT_INCR_PER_DEG (calculation resolution for angular positions). It is described in more detail on the following pages.

© Siemens AG, 2003. All rights reserved

If the velocity drops below V_{min} no traverse movement takes place.

Example: MD 10200: INT_INCR_PER_MM = 1000 [incr. / mm] ;
IPO cycle = 12 ms;
$\Rightarrow V_{min} = 10^{-3} / (1000\ ^{incr}/_{mm} \times 12\ ms) = 0.005\ ^{mm}/_{min}$;

The value range of the feedrates depends on the calculation resolution selected. With the default settings of MD 10200: INT_INCR_PER_MM (calculation resolution for linear positions) (1000 incr./mm) or MD 10210: INT_INCR_PER_DEG (calculation resolution for angular positions) (1000 incr./degrees), the following value range can be programmed with the specified resolution:

Value range for path feed F for geometry axes:
Metric system:
$0.001 \leq F \leq 999{,}999.999$ [mm/min, mm/rev, degree/min, degree/rev]
Inch system:
$0.001 \leq F \leq 399{,}999.999$ [inch/min, inch/rev]

Value range for feed for positioning axes:
Metric system:
$0.001 \leq FA \leq 999{,}999.999$ [mm/min, mm/rev, degree/min, degree/rev]
Inch system:
$0.001 \leq FA \leq 399{,}999.999$ [inch/min, inch/rev]

Value range for spindle speed S:
$0.001 \leq S \leq 999{,}999.999$ [rev/min]

If the calculation resolution is increased/decreased by a factor of 10, the value ranges change accordingly.

2.1.2 Traversing ranges

The value range of the traversing area depends on the calculation resolution selected.

With the default settings of MD 10200: INT_INCR_PER_MM (calculation resolution for linear positions) (1000 incr./mm) or MD 10210: INT_INCR_PER_DEG (calculation resolution for angular positions) (1000 incr./degrees), the following value range can be programmed with the specified resolution:

Table 2-1 Traversing ranges of axes

	G71 [mm, degrees]	G70 [inch, degrees]
	Range	Range
Linear axes X, Y, Z, ...	± 999,999.999	± 399,999.999
Rotary axes A, B, C, ...	± 999,999.999	± 999,999.999
Interpolation parameters I, J, K	± 999,999.999	± 399,999.999

The unit of measurement of rotary axes is always degrees.

If the calculation resolution is increased/decreased by a factor of 10, the value ranges change accordingly.

© Siemens AG, 2003. All rights reserved

The value range can be limited by software limit switches and working areas.

References: /FB/, A3, “Axis Monitoring Functions, Protection Zones”

The traverse range for rotary axes can be limited via machine data.

References: /FB/, R2, “Rotary Axes”

Special points to be noted with respect to wide traversing range for linear and rotary axes with SW 4 and higher, see:

References: /FB/, R1, “Reference Point Approach”

2.1.3 Positioning accuracy of the control system

The positioning accuracy of the control system depends on the actual value resolution (=encoder increments / (mm or degrees)) and the calculation resolution (=internal increments / (mm or degrees)).

The coarse resolution of these two values determines the positioning accuracy of the control.

The choice of input resolution, interpolator and position control cycle have no effect on this accuracy.

2.1.4 Block diagram of resolutions and normalization values

The following diagram shows how input values are converted into internal units.

It also shows the following conversion to internal increments / (mm or degrees) which can cause loss of decimal places if the calculation resolution was selected to be coarser than the input resolution.

The calculation of the actual value resolution shown is described in Subsection 2.2.3.

Moreover, it serves as an overview of the following topics described in this Section:

1. Selection of dimension system (metric / inch)
2. Normalization of physical quantities of machine and setting data
3. Conversion of basic system
4. Setting of calculation resolution

An example is given showing how a physical quantity
(MD 36110: POS_LIMIT_PLUS) is converted as a function of machine data parameter settings (MD 10230: SCALING_FACTORS_USER_DEF, MD 10220: SCALING_USER_DEF_MASK).

© Siemens AG, 2003. All rights reserved

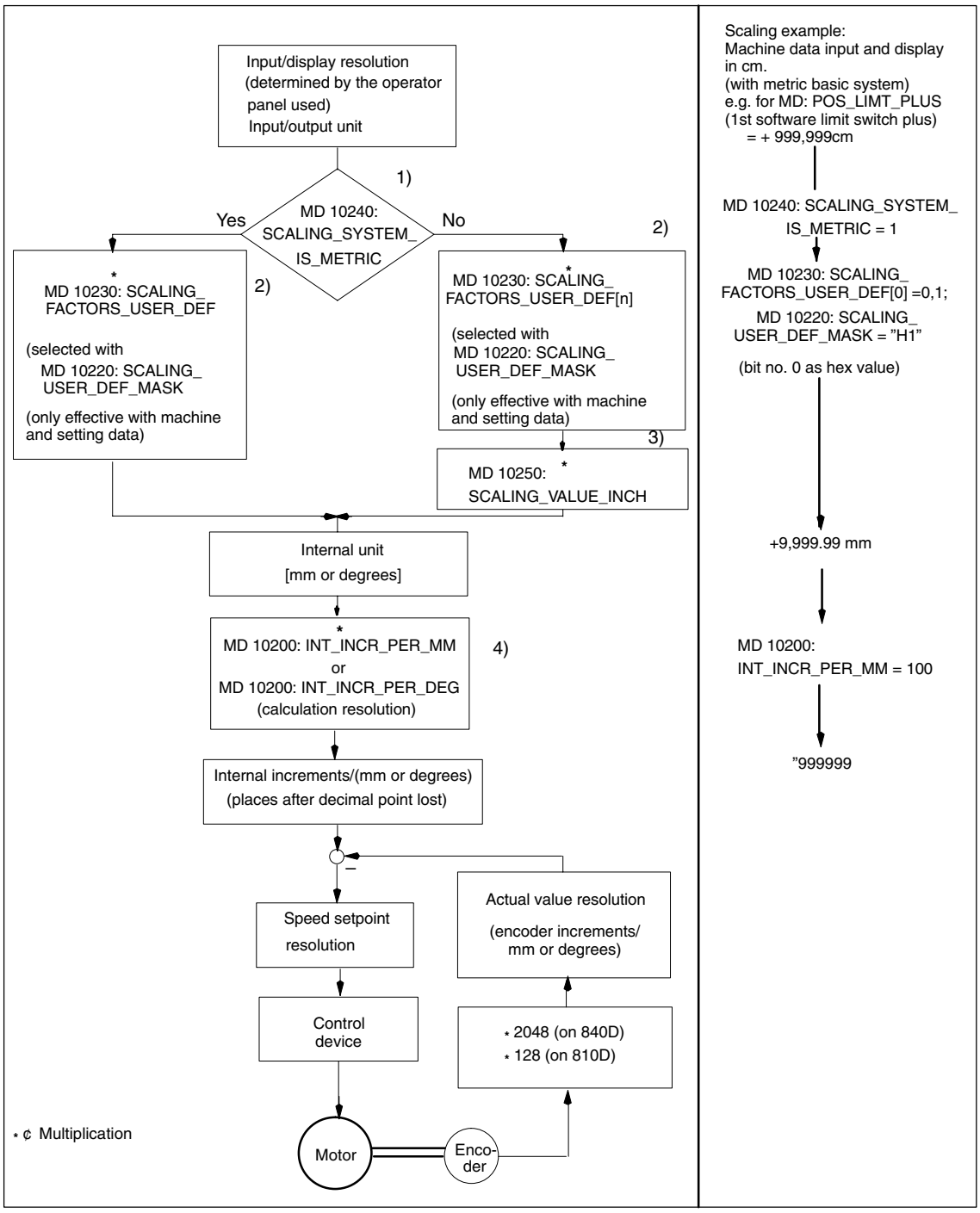


Fig. 2-1 Block diagram of units and resolutions

© Siemens AG, 2003. All rights reserved

2.1.5 Input/display resolution, calculation resolution

The following types of resolution, e.g. resolution of linear and angular positions, velocities, accelerations and jerk, must be differentiated as follows:

- The input resolution, i.e. the input of data via the operator panel or by means of parts programs.
- The display resolution, i.e. display of data via the operator panel.
- The calculation resolution, i.e. the internal representation of data entered via the operator panel or parts program.

The input and display resolutions are determined by the operator panel used, the display resolution for position values being determined by the setting in MD 9004: DISPLAY_RESOLUTION (display resolution).

In SW 5 and higher, MD 9011: DISPLAY_RESOLUTION_INCH can be used to configure the display resolution for position values in inches. This allows you to display up to six decimal places with the inch setting.

For the programming of parts programs, the input resolutions listed in the Programming Guide apply.

The desired calculation resolution is defined in MD 10200: INT_INCR_PER_MM (calculation resolution for linear positions) and MD 10210: INT_INCR_PER_DEG (calculation resolution for angular positions). It is independent of the input/display resolution but should use the same resolution.

The max. number of places after the decimal point for position values, velocities etc. in the parts program and the number of places after the decimal point for tool offsets, zero offsets etc. (and therefore also for the maximum possible accuracy) is defined with the calculation resolution.

The accuracy for entering angular and linear positions is limited to the calculation resolution because the product of the programmed value and the calculation resolution is rounded up to the next integer.
In order to keep the rounding easily reproducible, it is advisable to use powers of 10 for the calculation resolution.

Example for rounding:

Calculation resolution: 1000 increments / mm
Programmed path: 97.3786 mm
Effective value = 97.379mm

Example of programming in the $^1/_{10}$ – µm range:

- All linear axes of a machine should be programmed and traversed in the value range 0.1 ... 1000 µm.

⇒ In order to position to 0.1µm accuracy, the calculation resolution must be set to $\geq 10^4$ incr. /mm.

⇒ MD 10200: INT_INCR_PER_MM = 10000 [incr. /mm]:

⇒ Example of a parts program:

```
N20  G0  X 1.0000  Y 1.0000 ;     Axes move to the position
                                  X=1.0000mm, Y=1.0000mm;
N25  G0  X 5.0002  Y 2.0003 ;     Axes move to the position
                                  X=5.0002mm, Y=2.0003mm
```

© Siemens AG, 2003. All rights reserved

2.1.6 Standardizing physical quantities of machine and setting data

Machine and setting data that possess a physical quantity are interpreted in the input/output units below depending on whether the metric or inch basic system is selected:

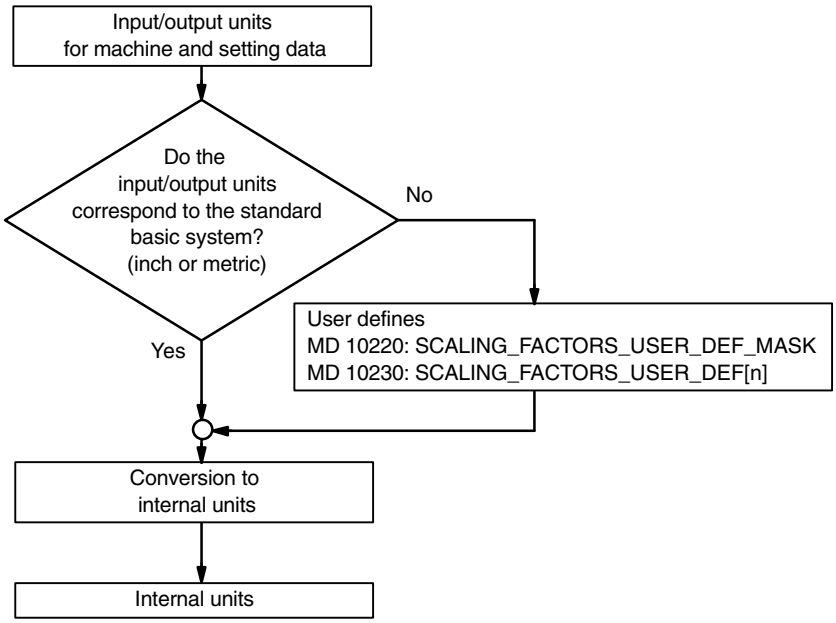
Physical quantity:	**Input/output units for standard basic system:**	
	Metric	**Inch**
Linear position	1 mm	1 inch
Angular position	1 degree	1 degree
Linear velocity	1 mm/min	1 inch/min
Angular velocity	1 rpm	1 rpm
Linear acceleration	1 m/s^2	1 $inch/s^2$
Angular acceleration	1 $rev./s^2$	1 $rev./s^2$
Linear jerk	1 m/s^3	1 $inch/s^3$
Angular jerk	1 $rev./s^3$	1 $rev./s^3$
Time	1 s	1 s
Position controller servo gain	1/s	1/s
Revolutional feedrate	1 mm/rev.	1 inch/rev.
Compensation value linear position	1 mm	1 inch
Compensation value angular position	1 degree	1 degree

The units listed below are used for storage. The control always uses these units internally irrespective of the basic system selected.

Physical quantity:	**Internal Unit:**
Linear position	1 mm
Angular position	1 degree
Linear velocity	1 mm/s
Angular velocity	1 degree/sec
Linear acceleration	1 mm/s^2
Angular acceleration	1 $degree/s^2$
Linear jerk	1 mm/s^3
Angular jerk	1 $degree/s^3$
Time	1 s
Position controller servo gain	1/s
Revolutional feedrate	1 mm/degree
Compensation value linear position	1 mm
Compensation value angular position	1 degree

© Siemens AG, 2003. All rights reserved

The user can define other input/output units for the machine and setting data. For this purpose, the newly selected input/output units and the internal units must be matched via MD 10220: SCALING_USER_DEF_MASK (activation of normalization factors) and MD 10230: SCALING_FACTORS_USER_DEF[n] (normalization factors of physical quantities).



The following applies:

Selected input/output unit =

MD : *SCALING_FACTORS_USER_DEF*[*n*] * *internal unit*

Enter the selected input/output unit in MD 10230: SCALING_FACTORS_USER_DEF[n] expressed in the internal units 1 mm, 1 degree, and 1 s.

Index [n] is explained in Section 4.1.

Example 1:
Machine data input/output of the linear velocities is to be in m/min instead of mm/min (initial setting). (The internal unit is mm/s)

⇒ The scaling factor for the linear velocities is to differ from the standard setting. This is achieved by setting Bit 2 in MD 10220: SCALING_USER_DEF_MASK.
  ⇒ MD: SCALING_USER_DEF_MASK = 'H4';
  (Bit no. 2 as hex value)

⇒ The scaling factor is calculated acc. to the following formula:

© Siemens AG, 2003. All rights reserved

$$MD: SCALING_FACTORS_USER_DEF[n] = \frac{Selected\ input/output\ unit}{internal\ unit}$$

$$MD: SCALING_FACTORS_USER_DEF[n] = \frac{1\frac{m}{min}}{1\frac{mm}{s}} = \frac{\frac{1000mm}{60s}}{1\frac{mm}{s}} = \frac{1000}{60} = 16.667;$$

$$\Rightarrow MD: SCALING_FACTORS_USER_DEF[2] = 16.667$$

Index 2 defines the “linear acceleration” in the list “Scaling factors of physical quantities”.

Example 2:
In addition to the modification made in example 1, the machine data input/output of linear velocities is to be in ft/s^2 instead of m/s^2 (initial setting) (the internal unit is mm/s^2).

$\Rightarrow$ *MD*: *SCALING_USER_DEF_MASK* = *'H*14*'*; (*bit no.* 4 ***and*** *bit no.* 2 *of example* 1 *as hexa value*)

$$\Rightarrow MD: SCALING_FACTORS_USER_DEF[n] = \frac{1\frac{ft}{s^2}}{1\frac{mm}{s^2}} = \frac{12 * 25,4\frac{mm}{s^2}}{1\frac{mm}{s^2}} = 304.8;$$

$$\Rightarrow MD: SCALING_FACTORS_USER_DEF[4] = 304.8$$

Index 4 defines the “linear acceleration” in the list “Scaling factors of physical quantities”.

© Siemens AG, 2003. All rights reserved

2.2 Metric/inch measuring system

The control system can operate with the inch or the metric system of measurement. The initial setting is defined in MD 10240: SCALING_SYSTEM_IS_METRIC (basic system metric). Depending on the setting in the MD, all geometric values are interpreted either as metric or inch values. All manual settings also refer to this basic setting (e.g. handwheel, INC, JOG feedrate), as do zero offsets, tool offsets, FRAMES etc. and the associated displays.

Extensions in SW 5 and higher

The setting of MD 10260: CONVERT_SCALING_SYSTEM=1 makes it much easier to change the system of measurement in SW 5 and higher.

- An MMC soft key is available in the “MACHINE” operating area for changing the system of measurement.
- Active NC data are converted automatically on a measurement system - switchover.
- Data backup is performed with the current measurement system identifier.
- The scope of MD 10240: SCALING_SYSTEM_IS_METRIC is Reset.
- The system of measurement for sag compensation is set in MD 32711: CEC_SCALING_SYSTEM_METRIC (see References /K3/).
- The system of measurement for positioning values of division axis tables and the switching points for software cams are set in MD 10270: POS_TAB_SCALING_SYSTEM (see References /T1, N3/).

2.2.1 Conversion of basic system by parts program

The conversion factor for metric to inch system for data input/output is entered in MD 10250: SCALING_VALUE_INCH (conversion factor for switchover to the inch system). MD 10250 is not visible without the Siemens password; default: 25.4). By changing the default value, the control can be adapted to a customerspecific measuring system.

When programming, it is possible to switch between measuring systems for some workpiece-related specifications with G70/G71; and also with G700/G710 from SW 5. The data influenced by G70/G71/G700/G710 are described in the Programming Guide.

The initial setting G70/G71/G700/G710 can be defined channel-specifically via the initial setting (Reset) of G groups in MD 20150: GCODE_RESET_VALUES[12].

When changing the system of measurement via MMC soft key (SW 5 and higher), these initial settings are automatically initialized for the new system of measurement with G700 or G710.

The axis-specific MD 31200: SCALING_FACTOR_G70_G71 (factor for converting values when G70/G71/G700/G710 is active), the conversion factor for conversion between the metric and inch system with G70/G71/G700/G710 programming can be chosen freely (default: 25.4). The control can be adapted to a customer-specific system of measurement by changing the default setting. The factor only takes effect if parts program programming differs from the initial setting.

© Siemens AG, 2003. All rights reserved

Example:

– Initial setting: Inch; (MD: SCALING_SYSTEM_IS_METRIC=0)

– Parts program:
  N15 G70; Factor has no effect because setting does not deviate from the initial setting.
  N20 G71; Factor has an effect.

The current setting (selected with G70/G71) and the initial setting can be the same or differ at any given time. The current setting is channel-specific, the initial setting applies to all channels.

Application:
With this function it is possible, for example, with a metric basic system, to machine an inch thread in a metric parts program. Tool offsets, zero offsets and feedrates remain metric.

Machine data are output in the basic system selected in
MD 10240: SCALING_SYSTEM_IS_METRIC (basic system metric).

Machine coordinates, tool data and zero offsets are always displayed in the initial setting, workpiece coordinate displays are in the current setting.

If programs incl. data records (zero offsets, tool offset) programmed in a unit system other than the basic system are read in from an external source, then first the initial setting must be programmed via the
MD 10240: SCALING_SYSTEM_IS_METRIC.

Data exchange with the PLC of interface signals containing dimension information, e.g. feedrate for path and positioning axes, is always carried out in the selected basic system.

Extensions in SW 5 and higher

G700/G710 extends the functionality of G70/G71 as follows:

1. The feed is interpreted in the programmed system of measurement:
   - G700: Length parameters [inch]; feedrates [inch/min]
   - G710: Length parameters [mm]; feedrates [mm/min]

   The programmed feed is modal and therefore remains active after subsequent G70/G71/G700/G710 commands. If the feedrate is to apply in the new G70/G71/G700/G710 context, it must be reprogrammed.
2. System variables and machine data specifying lengths in the parts program are read and written in the programmed system of measurement.

This allows you to implement parts programs that are independent of the default system of measurement.

Comparison of the effect of G70/G71 and G700/G710 on machine data and system variables in the parts program:

- with G70/G71: Reading/writing in the basic system
- with G700/G710: Reading/writing in the programmed system of measurement

© Siemens AG, 2003. All rights reserved

Examples:

(Both parts programs are implemented with a metric setting with
MD 1240: SCALING_SYSTEM_IS_METRIC=1)

```
N100 R1=0 R2=0
N120 G01 G70 X1 F1000
N130 $MA_LUBRICATION_DIST[X]=10
N140 NEWCONF
N150 IF ($AA_IW[X]>$MA_LUBRICATION_DIST[X])
N160 R1=1
N170 ENDIF
N180 IF ($AA_IW[X]>10)
N190 R2=1
N200 ENDIF
N210 IF ( (R1<>0) OR (R2<>0))
N220 SETAL(61000)
N230 ENDIF
N240 M30
```

Note

Alarm 61000 does not occur if G70 is replaced by G700 in block N120.

Synchronized actions

In order to prevent the current parts program context from changing the positioning behavior of a synchronized action arbitrarily in response to asynchronous trigger conditions, the system of measurement must be defined at the time of interpretation. This is the only way to achieve a defined and reproducible positioning behavior of a synchronized action.

Example 1:

```
N100 R1=0
N110 G0 X0 Z0
N120 WAITP(X)
N130 ID=1 WHENEVER $R1==1 DO POS[X]=10
N140 R1=1
N150 G71 Z10 F10        ;Z=10mm     X=10mm
N160 G70 Z10 F10        :Z=254mm    X=254mm
N170 G71 Z10 F10        ;Z=10mm     X=10mm
N180 M30
```

Example 2: The definition is made here by programming G71 in the synchronized action.

```
N100 R1=0
N110 G0 X0 Z0
N120 WAITP(X)
N130 ID=1 WHENEVER $R1==1 DO G71 POS[X]=10
N140 R1=1
N150 G71 Z10 F10        ;Z=10mm     X=10mm
N160 G70 Z10 F10        :Z=254mm    X=10mm (X always posit. to 10mm)
N170 G71 Z10 F10        ;Z=10mm     X=10mm
N180 M30
```

© Siemens AG, 2003. All rights reserved

Comparison: G70/G71– G700/G710

Meanings:
- P: Reading/writing of data in the programmed system of measurement
- G: Reading/writing of data in the basic system of (MD 10240: SCALING_SYSTEM_IS_METRIC)
- R/W: Read/Write

Table 2-2 Comparison

Range	G70/G71	G700/G710
	Parts programs	Parts programs
	R / W	R / W
Display, decimal places (WCS)	P / P	P / P
Display, decimal places (MCS)	G / G	G / G
Feedrates	G / G	P / P
Positional data X, Y, Z	P / P	P / P
Interpolation parameters I, J, K	P / P	P / P
Circle radius (CR)	P / P	P / P
Polar radius (RP)	P / P	P / P
Thread lead	P / P	P / P
Programmable FRAME	P / P	P / P
Settable FRAMES	G / G	P / P
Basic frames	G / G	P / P
Work offsets external	G / G	P / P
Axial preset offset	G / G	P / P
Working area limitations (G25/G26)	G / G	P / P
Protection zones	P / P	P / P
Tool offsets	G / G	P / P
Length-related machine data	G / G	P / P
Length-related setting data	G / G	P / P
Length-related system variables	G / G	P / P
GUD	G / G	G / G
LUD	G / G	G / G
PUD	G / G	G / G
R parameters	G / G	G / G
Siemens cycles	P / P	P / P
Jog/handwheel increment factor	G / G	G / G

References: /PG/, Chapter 12 “List of Addresses”

© Siemens AG, 2003. All rights reserved

2.2.2 Manual switchover of the basic system (SW 5 and higher)

General

The system of measurement of the complete machine is changed by activating an MMC soft key in the “MACHINE” operating area. The switchover is only accepted if:

- MD 10260: CONVERT_SCALING_SYSTEM=1.
- Bit 0 of MD 20110: RESET_MODE_MASK is enabled in every channel.
- All channels are in the Reset state.
- Axes are not traversing with JOG, DRF or PLC control.
- Constant grinding wheel peripheral speed (GWPS) is not active.

Actions such as parts program start or mode change are disabled for the duration of the switchover.

If the switchover cannot be performed, this is indicated by a message in the user interface. These measures ensure that a consistent set of data is always used for a running program with reference to the system of measurement.

The actual switchover of the system of measurement is performed internally by writing all the necessary machine data and subsequently activating them with a Reset.

MD 10240: SCALING_SYSTEM_IS_METRIC and the corresponding G70/G71/G700/G710 settings in MD 20150: GCODE_RESET_VALUES are automatically switched over consistently for all configured channels.

The value of MD 20150: GCODE_RESET_VALUES[12] alternates between G700 and G710.

This process takes place independently of the protection level currently set.

Note

The availability of the soft key and its functionality can be configured in the compatibility machine data MD 10260: CONVERT_SCALING_SYSTEM.

Important

If several NCUs are linked by NCU Link, the switchover available in SW 5 and higher has the same effect on all NCUs linked. If the prerequisites for a switchover are not fulfilled on one of the NCUs linked, no switchover will take place on none of the NCUs. It is assumed that interpolations between several NCUs will take place on the existing NCUs whereby the interpolations can provide correct results only if the same unit systems are used.
References: /FB/, B3, Distributed Systems

© Siemens AG, 2003. All rights reserved

System data

When the system of measurement is changed, all length-related parameters are automatically converted to the new system of measurement from the perspective of the operator. This includes:

- Positions
- Feedrates
- Acceleration rates
- Jerk
- Tool offsets
- Programmable, settable and work offsets external and DRF offsets
- Compensation values
- Protection zones
- Machine data
- Jog and handwheel factors

After the switchover, all of the above data are available in the physical magnitudes described in Subsection 2.1.6.

Data for which no unique physical units are defined, such as:

- R parameters
- GUD (**G**lobal **U**ser **D**ata)
- LUD (**L**ocal **U**ser **D**ata)
- PUD (**P**rogram global **U**ser **D**ata)
- Analog I/Os
- Data exchange via FC21

are not converted automatically. In this case, the user must allow for the current system of measurement in MD 10240: SCALING_SYSTEM_IS_METRIC.

The current system of measurement setting can be read at the PLC interface via the “inch system” signal DB10.DBX107.7. DB10.DBB71 can be used to read out the “system of measurement change counter”.

User tool data

For the user-defined tool data
MD 18094: MM_NUM_CC_TDA_PARAM and tool edge data MD18096: MM_NUM_CC_TOA_PARAM additional machine data sets:

- MD 10290: CC_TDA_PARAM_UNIT [MM_NUM_CC_TDA_PARAM]
- MD 10292: CC_TOA_PARAM_UNIT [MM_NUM_CC_TOA_PARAM].

have been introduced. These MD can be used to configure a physical unit (see the machine data description in Chapter 4). All length-related user-defined tool data are automatically converted to the new system of measurement according to the input on switchover.

Reference point

The reference point is retained. It is not necessary to repeat homing.

© Siemens AG, 2003. All rights reserved

Input resolution and calculation resolution

The input/calculation resolution is set via
MD 10200: INT_INCR_PER_MM. The default setting for a metric system is 1000 (0.001mm). 0.0001 inches is required for an inch system.

Example:

1 inch = 25.4mm ⇒ 0.0001 inch = 0.00254mm = 2.54μm

In order to program and represent the last 40 nm, a value of 100000 must be entered in MD 10200.

Only with this identical setting for both systems of measurement is it possible to change the system of measurement without a significant loss of accuracy. Once MD 10200 has been set as above, it no longer has to be changed for each measurement system switchover.

JOG and handwheel evaluation

MD 31090: JOG_INCR_WEIGHT consists, in SW 5 and higher, of two values containing axial increment factors for each of the two systems of measurement. The control automatically selects the correct value depending on the setting in MD 10240: SCALING_SYSTEM_IS_METRIC.

The user defines the two increment factors, e.g. for the first axis, during the installation and startup phase.

- Metric: MD 31090: JOG_INCR_WEIGHT[0;AX1]=0.001 mm
- Inch: MD 31090: JOG_INCR_WEIGHT[1;AX1]=0.00254 mm ≙ 0.0001 inch

MD 31090 no longer has to be changed for each inch/metric switchover.

Remaining distances are not accumulated during incremental traversing with JOG when the system of measurement is changed, since all internal positions always refer to mm.

Data saving

Data sets which can be read out separately from the control and which contain data relating to the system of measurement are given an INCH or METRIC identifier corresponding to MD 10240: SCALING_SYSTEM_IS_METRIC during the read process, depending on the setting in MD 10260: CONVERT_SCALING_SYSTEM. This records the system of measurement in which the data were originally read out.

This information is intended to prevent data sets from being read into the control system with a system of measurement which is different from the active system. In this case, alarm 15030 is triggered and the write process is interrupted.

Since the language instruction is also evaluated in parts programs, these can also be “protected” against operator errors as described above. You can therefore prevent parts programs containing e.g. only metric data from running on an inch system of measurement.

Archives and machine data are downwards compatible with the setting
MD 11220: INI_FILE_MODE = 2.

Note

The INCH/METRIC instruction is only generated if compatibility machine data MD10260: CONVERT_SCALING_SYSTEM is enabled.

© Siemens AG, 2003. All rights reserved

Rounding machine data

In order to prevent rounding problems, all length-related machine data are rounded to 1pm when writing in the inch system of measurement (MD 10240: SCALING_SYSTEM_IS_METRIC=0 and MD 10260: CONVERT_SCALING_SYSTEM=1).

The disturbing loss of accuracy which occurs as a result of conversion to ASCII when reading out a data backup in the inch system of measurement is corrected by this procedure when the data are read back into the system.

© Siemens AG, 2003. All rights reserved

2.2.3 FGROUP and FGREF (SW 5 and higher)

It should be possible to program the effective machining feedrate in the usual way as a path feed via the F value in machining operations where the tool or the workpiece or both are moved by a rotary axis (e.g. laser machining of rotating tubes).

In order to achieve this, it is necessary to specify an effective radius (reference radius) for each of the rotary axes involved. You can do this by programming the modal NC address

FGREF[<axis name>] = reference radius

The unit of the reference radius depends on the G70/G71/G700/G710 setting.

In order to include the axes in the calculation of the path feed, they must all be specified in the FGROUP command.

In order to ensure compatibility with the behavior with no FGREF programming, the factor 1 degree = 1 mm is activated on system power-up and RESET.

This corresponds to a reference radius of

FGREF = 360mm / (2π) = 57.296mm.

This default is independent of the active basic system MD 10240: SCALING_SYSTEM_IS_METRIC and of the currently active inch/metric G code.

Special features of the feedrate weighting for rotary axes in FGROUP:

With the following code:

```
N100 FGROUP(X,Y,Z,A)
N110 G1 G91 A10 F100
N120 G1 G91 A10 X0.001 F100
```

the programmed F value in block N110 is evaluated as a rotary axis feed in degrees/min, while the feedrate weighting in block N120 is either 100 inch/min or 100 mm/min, depending on the current inch/metric setting.

The time required to execute the two blocks can be very different!

Note

The FGREF weighting is effective even if only rotary axes are programmed in the block. The usual F value interpretation in degrees/min only applies in this case if the radius reference has been set in accordance with the FGREF default, i.e.

for G71/G710): FGREF[A]=57.296
or
for G70/G700): FGREF[A]=57.296/25.4

© Siemens AG, 2003. All rights reserved

Example:

The following example illustrates the effect of FGROUP on the path distance and path feed.

```
N100 R1=0
N110 FGROUP(X,A)
N120 G91 G1 G710 F100      ;feed=100 mm/min or 100 degrees/min
N130 DO $R1=$AC_TIME
N140 X10                   ;feed=100 mm/min          path dist.=10mm
                           ;R1=approx. 6s
N150 DO $R2=$AC_TIME
N160 X10 A10               ;feed=100 mm/min          path dist.=14.14mm
                           ;R2=approx. 8s
N170 DO $R3=$AC_TIME
N180 A10                   ;feed=100 degrees/min     path dist.=10 degr.
                           ;R3=approx. 6s
N190 DO $R4=$AC_TIME
N200 X0.001 A10            ;feed=100 mm/min          path dist.=10mm
                           ;R4=approx. 6s
N210 G700 F100             ;feed=2540 mm/min or 100 degrees/min
N220 DO $R5=$AC_TIME
N230 X10                   ;feed=2540 mm/min         path dist.=254mm
                           ;R5=approx. 6s
N240 DO $R6=$AC_TIME
N250 X10 A10               ,feed=2540 mm/min         path dist.=254.2mm
                           ;R6=approx. 6s
N260 DO $R7=$AC_TIME
N270 A10                   ;feed=100 degrees/min     path dist.=10 degr.
                           ;R7=approx. 6s
N280 DO $R8=$AC_TIME
N290 X0.001 A10            ;feed=2540 mm/min         path dist.=10mm
                           :R8=approx. 0.288s
N300 FGREF[A]=360/(2*$PI)  ;1 degree=1 inch
                           ;set via effective radius
N310 DO $R9=$AC_TIME
N320 X0.001 A10            ;feed=2540 mm/min         path dist.=254mm
                           ;R9=approx. 6s
N330 M30
```

Note

The variable $AC_TIME contains the time since the start of block in seconds. It can only be used in synchronized actions.

© Siemens AG, 2003. All rights reserved

2.3 Setpoint/Actual value system

2.3.1 General

Block diagram

A control loop with the following structure can be configured for every closed-loop controlled axis/spindle:

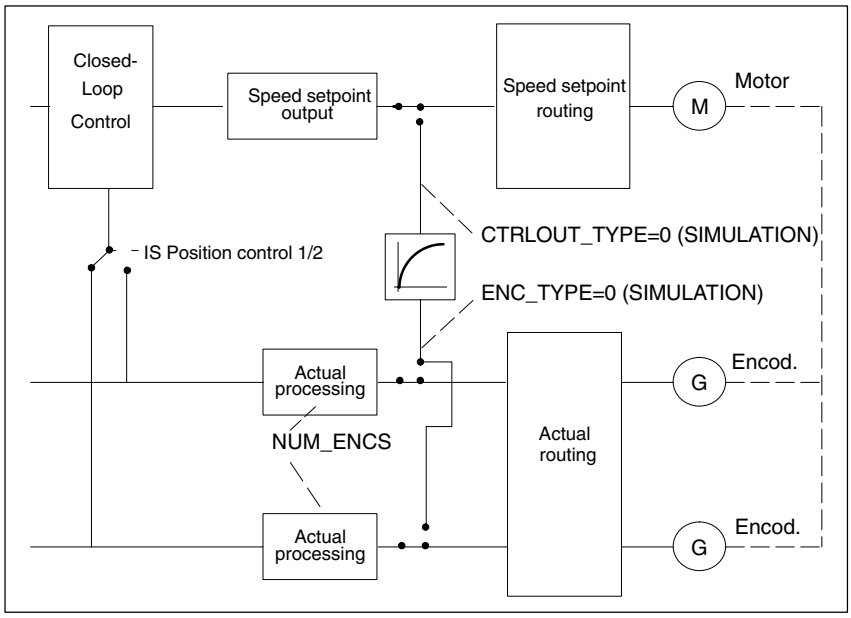


Fig. 2-2 Block diagram of a control loop

Setpoint output

A setpoint can be output for each axis/spindle. Setpoints are output digitally to the actuator on SINUMERIK 840D/810D.

Actual value sensing

A maximum of two measuring systems can be connected for each axis/spindle, e.g. a direct measuring system for machining processes with later accuracy requirements and an indirect measuring system for highspeed positioning tasks.

The number of encoders used is entered in MD 30200: NUM_ENCS (number of encoders). In the case of two actual value branches, the actual value is acquired for both branches.

The active measuring system is always used for position control, absolute value calculation and display. If both measuring systems are activated at the same time by the PLC interface, positioning measuring system 1 is chosen internally by the control.

Reference point approach is executed by the selected measuring system. Every position measuring system must be referenced separately.

For an explanation of the compensation functions for actual value acquisition see:
References: /FB/, K3, “Compensations”

For an explanation of the encoder monitoring functions see:
References: /FB/, A3, “Axis Monitoring Functions, Protection Zones”

© Siemens AG, 2003. All rights reserved

Switch over measuring system

It is possible to switch over between the two measuring systems via PLC interface signals “Position measuring system 1/2”, see:
References: /FB/, A2, “Various Interface Signals”

It is possible to switch over measuring systems at any time, the axes do not have to be stationary to do this. Switchover only takes place if a permissible deviation between the actual values and the two measuring systems has not been violated. The associated tolerance band is set in MD 36500: ENC_CHANGE_TOL (maximum tolerance for position actual value switchover). On switchover the actual difference between position measuring system 1 and 2 is traversed immediately.

If the actual value deviation is too large, alarm 25100, “Measuring system switchover not possible” is output and the measuring system is not changed.

Enter the actual value corresponding to the approached position in MD 36510: ENC_DIFF_TOL. This tolerance setting must not be exceeded during cyclical comparison of the two measuring systems or an error message will be generated. The associated monitoring function is not active when MD 35510=0, in cases where 2 measuring systems are not active/nonexistent in the axis or if the axis itself is not referenced (at least act. closed-loop control measuring system).

Types of actual value sensing

The encoder type used must be programmed in MD 30240: ENC_TYPE (type of actual value sensing (position actual value)):

Simulation axes

The speed control loop of an axis can be simulated for test purposes. The axis “traverses” with a following error, similarly to a real axis.

A simulation axis can be defined by setting the two machine data MD 30130: CTRLOUT_TYPE[n] (setpoint output mode) and MD 30240: ENC_TYPE[n] (actual value sensing type) to “0”.

As soon as the standard machine data have been loaded the axes become simulation axes.

The setpoint and actual value can be set to the reference point value with reference point approach.

The MD 30350: SIMU_AX_VDI_OUTPUT (output of axis signals for simulation axes) can be set to define whether axis-specific IS are output to the PLC during the simulation process.

Actual value correction

If actual value corrections carried out by the NC on the encoder selected for position control are not to influence the actual value of a second encoder defined on the same axis, the encoder is to be declared “independent” with MD 30242: ENC_IS_INDEPENDENT.

Actual value corrections include the following:

- Modulo treatment
- Reference point approach
- Measuring system adjustment
- PRESET

© Siemens AG, 2003. All rights reserved

2.3.2 Speed setpoint and actual value routing

General

For the purpose of speed setpoint and actual value routing, the following must be defined

for every axis/spindle:

- Assignment of 1st control loop
- Assignment of 2nd control loop if available
- Assignment of setpoint branch

Multiple assignment is also possible, e.g. using a control loop for position actual value acquisition for the alternating control of several axes/spindles.

Note

When a SIMODRIVE 611 universal is operated via the PROFIBUS DP, various machine data that need to be parameterized for digital and analog drives are not assigned.

Prerequisite for routing

All NC machine axes have to be clearly defined in
MD 10000: AXCONF_MACHAX_NAME_TAB[n] (machine axis name).
This name must be the one used throughout the system (all mode groups and channels).

References: /FB/, K2, “Axes, Coordinate Systems, Frames, Actual Value System for Workpiece, Reset Behavior.
/FB/, K1, ”Mode Group, Channels, Program Operation”

Speed setpoint routing

For speed setpoint routing, the following setpoint assignments for parameterizing the relevant machine data must be made:

• **Setpoint assignment**	Number
– Drive type:	Of bus segment
– Drive number/module number:	Of module within a bus segment
– Output to drive module:	Of setpoint output
– Setpoint output method:	Type of speed setpoint output
– Setpoint output is unipolar (operative only on PROFIBUS DP)	Polarity of output driver for speed setpoint output

For further information about PROFIBUS DP configuration, please see:
References: /FB1/, K4 Communication “SINUMERK 840D with PROFIBUS”

Index of MD for speed setpoint routing

The index [n] of the machine data for setpoint routing is coded with 0 for setpoint assignment with default setting 1

© Siemens AG, 2003. All rights reserved

Speed setpoint routing

The following machine data need to be parameterized for each setpoint branch:

- MD 30100: CTRLOUT_SEGMENT_NR[n] (setpoint assignment bus segment)

  The number of the bus segment via which the output is addressed is entered here. Depending on the SINUMERIK version, certain bus segments are preassigned.

  – Local bus (analog. to e.g. SINUMERIK FMNC) = 0,
  – 611D bus (1st DCM e.g. SINUMERIK 810D) = 1
  – Local P bus = 2
  – 611D bus (2nd DCM) = 3
  – Reserved for virtual buses = 4
  – PROFIBUS DP (NCU 573.2 and higher) = 5
    (PROFIBUS line for ProfiSafe at PLC end)
  – PROFIBUS DP link module
    (NCU 573.2 and higher) = 6

- MD 30110: CTRLOUT_MODULE_NR[n]
  (Setpoint assignment: drive number/module number):

  The number of the module in the bus segment via which the output is to be addressed is entered here. The logical drive number of the axis module can be set via MD 13010: DRIVE_LOGIC_NR[n] for:

  – SINUMERIK 810D Value range 0–15
  – SINUMERIK 840D as 611 digital drive no. Value range 0–31
  – SINUMERIK 840D PROFIBUS on link module Value range 0–125

- MD 30120: CTRLOUT_NR[n])
  (Setpoint assignment: Setpoint output on drive module/module):

  The number of the setpoint output must be entered here (always output 1 for SINUMERIK 840D/810D).

- MD 30130: CTRLOUT_TYPE[n] (setpoint output type):

  The speed setpoint output type is entered here.

  0 : Simulation (no HW required)
  1 : Default (differentiated via HW configuration)
  2 : Stepper motor and 3: FM module (for SINUMERIK FM NC only)

- MD 30134: IS_UNIPOLAR_OUTPUT[n] (setpoint output is unipolar):

  The unipolar speed setpoint output works only in conjunction with PROFIBUS DP.

Actual value routing

For actual value routing, the following actual value assignments for parameterizing the associated machine data must be made:

- **Actual value assignment** Number:
  – Drive type: Of bus segment
  – Drive number/module number: Of module within a bus segment
  – Input to drive module/measuring circuit card: Of setpoint input
  – Type of actual value acquisition (pos. actual value): Encoder type used
  – Independent / independently set encoder The encoder i “independent” Actual value corrections from NC

© Siemens AG, 2003. All rights reserved

Machine data actual value routing

The following machine data must be parameterized for each actual value branch:

- MD 30210: ENC_SEGMENT_NR[n] (actual value assignment bus segment):

  The number of the bus segment via which the encoder is addressed is entered here. Depending on the SINUMERIK version, certain bus segments are preassigned.

  – Local bus (analog. to e.g. SINUMERIK FMNC) = 0,
  – 611D bus (1st DCM e.g. SINUMERIK 810D) = 1
  – Local P bus = 2
  – 611D bus (2nd DCM) = 3
  – Reserved for virtual buses = 4
  – PROFIBUS DP (NCU 573.2 and higher) = 5
    (PROFIBUS line for ProfiSafe at PLC end)
  – PROFIBUS DP link module
    (NCU 573.2 and higher) = 6

- MD 30220: ENC_MODULE_NR[n]) (actual value assignment:
  (Actual value assignment: Drive module number / measuring circuit number):

  The number of the module in the bus segment via which the encoder is addressed is entered here. The logical drive number of the axis module can be set via MD 13010: DRIVE_LOGIC_NR[n] for:

  – SINUMERIK 810D Value range 0–15
  – SINUMERIK 840D as 611 digital drive no. Value range 0–31
  – SINUMERIK 840D PROFIBUS on link module Value range 0–125

- MD 30230: ENC_INPUT_NR[n]) (actual value assignment: input to drive module / control loop module):

  The number of the input to which the position actual value encoder is connected is entered here. When

  – SINUMERIK 840D/810D = 1 or 2 (counting from top to bottom),
  – SINUMERIK FM NC = 1–4 depending on input X3 – X6 selected.

- MD 30240: ENC_TYPE[n] (type of actual value acquisition):

  The encoder type used must be entered here (see Section 4.2).

- MD 30242: ENC_IS_INDEPENDENT[n] (encoder is independent):

  To prevent actual value corrections from influencing the actual value of an encoder defined in the same axis, the latter must be declared independent.

  0 : Encoder is independent
  1 : Encoder is dependent

MD index for actual value routing

The coding of the machine data index [n] for actual value routing is [encoder no.] 0 for the first encoder or 1 for the 2nd encoder.

© Siemens AG, 2003. All rights reserved

Examples of Setpoint/actual value routing

On SINUMERIK 840D/810D with SIMODRIVE 611 digital

For machine axis “X1”, digital setpoint output and actual value sensing are to be assigned to drive module 4 (4th slot = index [3]).
In this example, the “Logical drive number” of this module is 7.

Encoder number: 1, 2
Actual values are therefore sensed via a direct and an indirect measuring system.

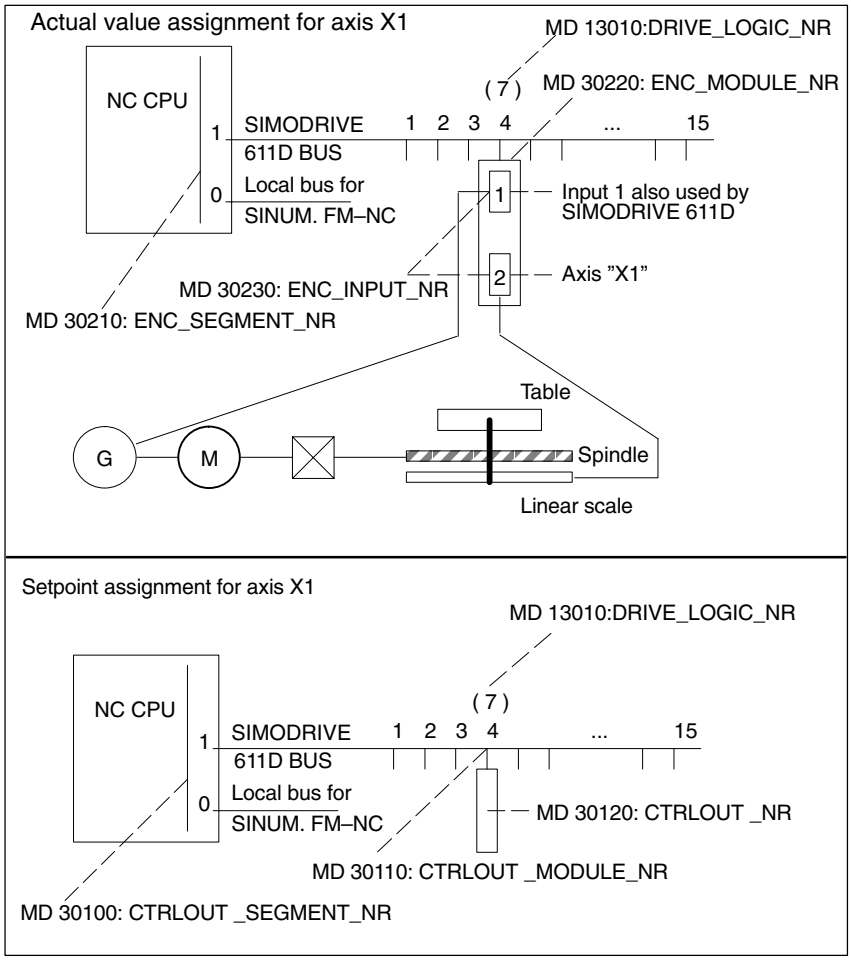


Fig. 2-3 Example of setpoint/actual value routing

Special points to be noted

SINUMERIK 840D/810D with SIMODRIVE 611 digital:

MD 30110: CTRLOUT_MODULE_NR[n] and MD 30220: ENC_MODULE_NR[n] for one machine axis always have the same logical drive number when indirect measuring systems are connected or when the motor encoder at the NC end needs to be evaluated.

When direct measuring systems are installed, you can also configure encoders that are connected to other drives.

© Siemens AG, 2003. All rights reserved

Setpoint/actual value routing

On a SINUMERIK 810D with axis expansion interface

Integrated drive no. 4

(motor measuring channel X414
pulse interface X304)

and the measuring channel no. 6 (X416 as direct measuring system) must be assigned to machine axis X1. The logical number of the drive is set automatically to 4.

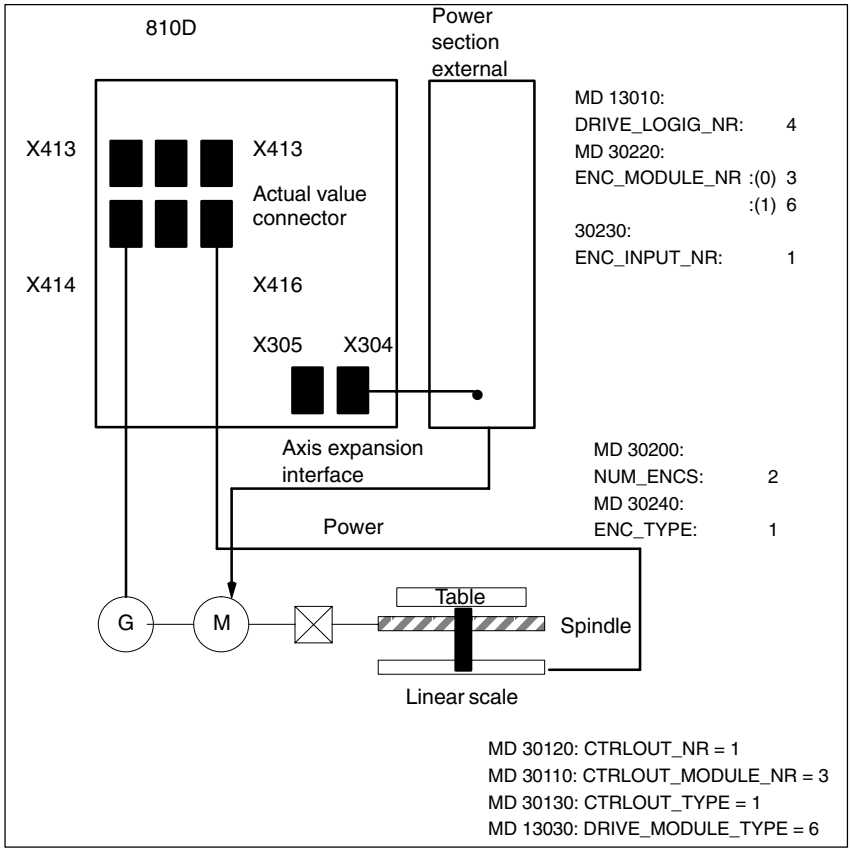


Fig. 2-4 Example of setpoint/actual value routing with axis expansion interface

Actual value assignment for Fig. 2–4	**Machine data parameterization for 810D**
1st axis actual value from motor encoder (X414)	MD 30220: ENC_MODULE_NR[0] = 4 MD 30230: ENC_INPUT_NR[0] = 1
2nd axis actual value from linear scale (X416)	MD 30220: ENC_MODULE_NR[1] = 3 MD 30230: ENC_INPUT_NR[1] = 1
Number of encoders:	MD 30200: NUM_ENCS = 2
Type of actual value acquisition 1:	MD 30240: ENC_TYPE[0] = 1
Type of actual value acquisition 2:	MD 30240: ENC_TYPE[1] = 1
Setpoint assignment for Fig. 2–4	MD 30110: CTRLOUT_MODULE_NR[0] = 4 MD 30120: CTRLOUT_NR[0] = 1 MD 13010: DRIVE_LOGIC_NR[3] = 4 MD 30130: CTRLOUT_TYPE[0] = 1

© Siemens AG, 2003. All rights reserved

2.3.3 Configuration of drives

SINUMERIK 840D/ 810D with SIMODRIVE 611 digital

SINUMERIK 840D/810D with 611D drive bus

You can configure the drive in the “Diagnostics” operating area on the operator panel (Human Machine Interface HMI). The following machine data are automatically parameterized for each real drive:

- MD 13010: DRIVE_LOGIC_NR[n] (logical drive number)
- MD 13000: DRIVE_IS_ACTIVE[n] (activate SIMODRIVE 611 digital drive)
- MD 13030: DRIVE_MODULE_TYPE[n] (module identification)
- MD 13040: DRIVE_TYPE[n] (drive type identification)
- MD 13020: DRIVE_INVERTER_CODE[n] (power section code of drive module)

The index [n] used in the machine data is the slot number of the real drives. The number is automatically assigned by the NC for all connected drive modules on POWER ON. The index is numbered starting with “0” at the beginning of the drive bus (1st real drive available) and continues in ascending order to the end. Multi-axis modules are assigned consecutive, physical drive numbers (counted from left to right). SINUMERIK 810D uses the first 6 slots (indices 0–5).
MD 13010: DRIVE_LOGIC_NR (logical drive number) can also be used to create wildcards for modules which do not yet exist.

SINUMERIK 840Di with SIMODRIVE 611 universal

SINUMERIK 840Di with PROFIBUS DP

When a SINUMERIK 840D is operated with the PROFIBUS DP drive 611 universal, the following MD are **not** used.

- MD 13000: DRIVE_IS_ACTIVE[n] (activate SIMODRIVE 611 digital drive)
- MD 13010: DRIVE_LOGIC_NR[n] (logical drive number)
- MD 13020: DRIVE_INVERTER_CODE[n] (power section code of drive module)
- MD 13030: DRIVE_MODULE_TYPE[n] (module identification)
- MD 13040: DRIVE_TYPE[n] (drive type identification)

Instead of MD 13000 to MD 13040**, the following are used in SW 5.2** and higher:

- MD 13050: DRIVE_LOGIC_ADDRESS[n] (drive address)
- MD 13060: DRIVE_TELEGRAM_TYPE[n] (message frame type for the drives connected to PROFIBUS DP)
- MD 13070: DRIVE_FUNCTION_MASK[n] (DP functions used for drives on PROFIBUS DP). Allows adaptation of certain non-standardized PROFIBUS control bits of the SIMODRIVE 611 universal.
- MD 13080: DRIVE_TYPE_DP[n] (drive type PROFIBUS DP with SW 6.4 and higher). Selection of external slaves, synchronous, asynchronous or linear drives.

© Siemens AG, 2003. All rights reserved

2.3.4 Adapting the motor/load ratios

Overview

The following gear types are available for adapting the meachincal ratios:

Gear type	Activation	Adaptation	Installation location
Motor/load gear box	Parameter set	Fixed configuration	Gearbox
Encoder measuring gearbox	Power ON	Sensor-dependent	Sensor-side
Load intermediate gearbox	NewConfig	Load-dependent	Tool-side

Local position of gearbox / encoder

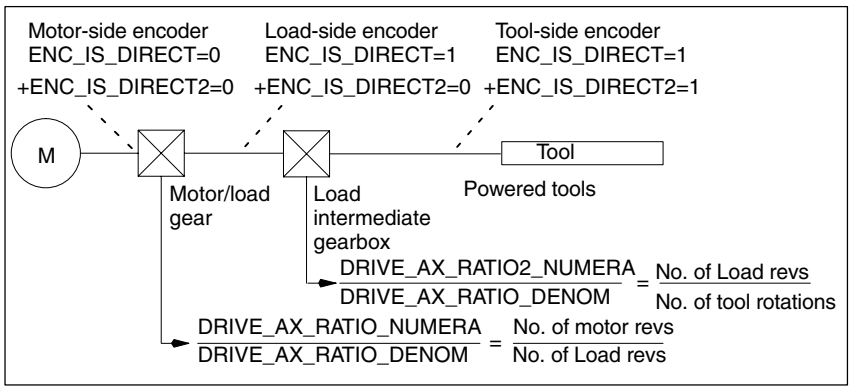


Fig. 2-5 Gearbox types and encoder locations

Motor/load gear

The motor/load gearbox supported by SINUMERIK is configured via

MD 31060: DRIVE_AX_RATIO_NUMERA “numerator load gearbox”
MD 31050: DRIVE_AX_RATIO_DENOM “denominator load gearbox”

projektiert. The gearbox transmission is obtained from the numerator/ denominator ration of both machine data. The associated parameter sets are used automatically per default to synchronize the position controller of the control to the relevant tranmission ratios.

Since a gear stage change is not always carried out automatically, and there are also several ways to request a gear stage change, the position controller is not always incoporated via parameter sets.

Note

For further information about parameter sets and gear stage change, please refer to:
References: /FB/, S1, Spindles

© Siemens AG, 2003. All rights reserved

Intermediate gears SW 6.4 and higher

In SW 6.4 and higher, the control also supports configurable load intermediate gears.

MD 31066: DRIVE_AX_RATIO_NUMERA “numerator intermediate gearbox”
MD 31064: DRIVE_AX_RATIO_DENOM “denominator intermediate gearbox”

Power tools generally have their own intermediate gearbox. Such variable mechanics can be configured by multiplying the active Vorsatz gearbox and the motor/load gearbox.



Caution

In contrast to the motor/load gearbox, the intermediate gear has no parameter set and therefore no way of controlling the time-synchronized switchover to parts program or PLC (VDI interface. Parts programming during gearchange is therefore ruled out.

It remains the task of the user to match the synchronization of the relevant changed machine data to the corresponding mechanical switchover and activate it. On switchover during a movement, compensations **cannot** be ruled out due to jumps in the scaling factors. These are not monitoried for violation of the maximum acceleration.

Encoder directly at tool
For intermediate gears, a further connection option is possible for an

“encoder on tool-side”

by configuring MD 31044: ENC_IS_DIRECT2.

Encoder not directly at tool
For a gear change of the intermediate gear in position-controlled mode, the following supplementary conditions apply:

- The gear ratio to be changed is incorporated in a re-scaling of the encoder information in this case.

In this case, the following applies for axes/spindles in position mode:

- A **non-abrupt** gear change is **only possible at zero speed**.

  To do this, the tool-side position before and after a gear change are set equal for a change in the ratio, since the mechanical position does not (or hardly) changes during a gear stage change.

  Recommendation: To avoid 21612 “Controller enable reset during motion”, changeover should be carried out “only at zero speed”. It is still permissible and expedient to switch the axis or spindle to speed-control or follow-up mode before or during a gear change.

Supplementary conditions

If the **encoder** to be used for position control **is connected directly at the tool**, the gear stage change only affects the physical variables at the speed interface between the NC and the drive of the motor/load gear. The internal parameter sets are not changed.

© Siemens AG, 2003. All rights reserved

Reference point and position reference

In the case of gear changes, it is not possible to make a statement about the effect of the reference point or machine position referrence on the encoder scaling. In such cases, the control partially cancels the status “Axis reference/synchroniyed”.

If the position reference to the machine, tool, et. has been lost, it must be restored through appropriate adjustment or referencing. This is especially important for the functions Travel to fixed stop, Referencing to Bero, Cam and Zero marker.



Caution

The control cannot detect all possible situations that can lead to loss of the machine position reference.

It is therefore the responsibility of the commissioning engineer or user to to initiate explicit referencing of zero marker synchronization in such cases.

Note

To enable re-referencing without interupts through reset, the machine data MD 34080: of MD: REFP_MOVE_DIST and MD 34090: REFP_MOVE_DIST_CORR in SW 6.4 and higher are active with NewConfig.

For further explanations on referencing, please refer to:
References: /FB/, R1, “Reference Point Approach”

© Siemens AG, 2003. All rights reserved

2.3.5 Speed setpoint output and actual value processing

Control direction and traverse direction of the feed axes

You must determine the travel direction of the feed axis before starting work.

Control direction

Before the position control is started up, the speed controller and current controller of the drive must be started up and optimized.

Traversing direction

The MD 32100: AX_MOTION_DIR (traverse direction) without affecting the feedback polarity of the position control.

Speed setpoint adjustment

SINUMERIK 840D/810D

In the case of speed setpoint comparison, the NC is informed which speed setpoint corresponds to which motor speed in the drive for parameterizing the axial control and monitoring. This comparison is carried out automatically.

SINUMERIK 840D with PROFIBUS-DP (SW 6.3 to SW 6.4)
For PROFIBUs DP drives with automatic adjustment of speed setpoint scaling, MD 32250: RATED_OUTVAL[n] must be modified from 80% to 0% (setting “0”). This value must also be set in the case of adjustment in the NC. Alternatively for PROFIBUS DP drives, manual speed setpoint comparison is also possible.

Manual comparison
In MD 32250: RATED_OUTVAL, enter a value not equal to zero. For further explanations on the speed setpoint comparison, please refer to:
References: /HBI/, SINUMERIK 840Di, “Axes and Spindles”

SINUMERIK 840Di with SIMODRIVE 611 universal

The speed setpoint comparison for SINUMERIK 840Di with SIMODRIVE 611 universal drives can be performed automatically or manually.

Automatic comparison
The configuration values for setpoint scaling are compared automatically, provided that MD 32250: RATED_OUTVAL[n] = 0. The speed setpoint comparison through acyclic services at PROFIBUS-DP can be performed automatically.

SINUMERIK 840D/810D with SIMODRIVE digital

Velocity adjustment and maximum speed setpoint

Note

The velocity does not need to be adjusted on a SINUMERIK 840D/810D due to automatic speed setpoint comparison!

© Siemens AG, 2003. All rights reserved

Maximum speed setpoint

On the SINUMERIK 840D/810D, the maximum speed setpoint is the highest value that can be output to the SIMODRIVE 611 digital drive as a result of the maximum speed parameterized in the drive machine data

MD 1401/2401: MOTOR_MAX_SPEED (maximum motor speed)

that can be output at SIMODRIVE 611 digital drive.

For spindle drives, MD 1401 corresponds to the maximum motor speed. The desired speed at the spindle is reached via the mechanical gear stage.

The output of the spindle speed is implemented in the NC for SINUMERIK 840D/840Di. Data for 5 gear stages are implemented in the control. The gear stages are defined by a minimum and maximum speed for the gear stage and a minimum speed and a maximum speed for the automatic gear stage change. A new set gear stage is output only if the new programmed speed cannot be traversed in the current gear stage.

MD 36210: Can be defined to limit the speed setpoint as a percentage of the maximum speed setpoint. Values up to 200 % are possible.
If this limit is exceeded, an alarm is output.

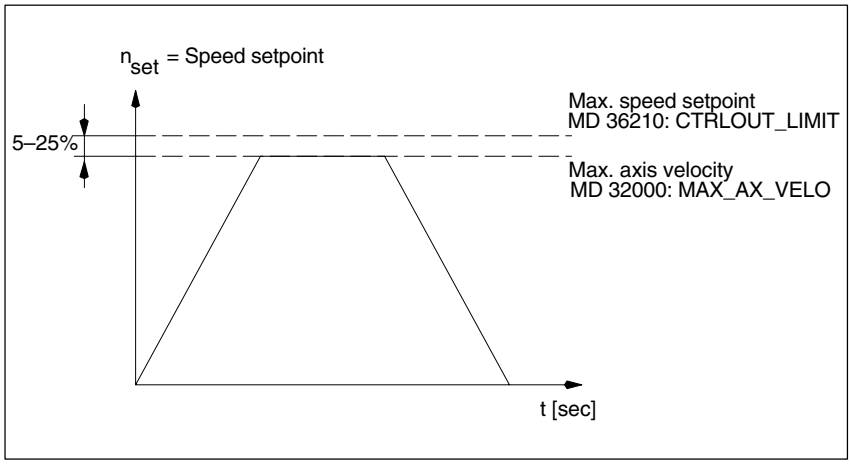


Fig. 2-6 Maximum speed setpoint

However, owing to control processes, the axes should be able to reach their maximum velocity (MD 32000: MAX_AX_VELO) before the 100% speed setpoint is reached, i.e. at between 80% and 95%.

The default value (default 80%) of the MD 32000: MAX_AX_VELO can be accepted for axes which reach maximum velocity at about 80% of the speed setpoint range.

On SINUMERIK 840D/810, MD 36210: CTRLOUT_LIMIT[n] and MD 1405: MOTOR_SPEED_LIMIT (monitoring speed motor) should tally.

Note

For further details of setpoint adaption for SIMODRIVE digital drives, refer to:
References: /IAD/, Installation and StartUp Guide, “Axis and Spindles”

Explanations for setpoint normalization for SIMODRIVE analog drives, see:
References: /FB/, B2, “Analog Axis”

© Siemens AG, 2003. All rights reserved

Actual value processing

Actual value resolution

In order to be able to create a properly closed position closed-loop control, the control system must be informed of the valid actual value resolution. The following axis-specific machine data serve this purpose (see Figs. 2-8 to 2-12).

The control calculates the actual value resolution from the settings made in the MD. The control parameter set of the position controller are referred to as servo parameter sets.
The machining process of the machine provides the basis for actual position recording. A direct measuring system (DM) must be configured directly at the machine: Load-side encoder. Indirect measuring system (IM) must be configured indirectly at the motor: Motor-side encoder.

Depending on the type of axis (linear axis, rotary axis) and the type of actual value acquisition (directly at the machine, indirectly at the motor), the following machine data must be parameterized to calculate the actual value resolution:

Machine data	Linear axis	Linear axis		Rotary axis	
	Linear scale/ or as direct measuring system	Encoder at motor	Encoder at machine and/or tool	Encoder at motor	Encoder at machine and/or tool
MD 30300: IS_ROT_AX	0	0	0	1	1
MD 31000: ENC_IS_LINEAR[n]	1	0	0	0	0
MD 31010: ENC_GRID_POINT_DIST[n]	Spacing	–	–	–	–
MD 34320: ENC_INVERS[n]	inverted	–	–	–	–
MD 31040: ENC_IS_DIRECT[n]	– / (1)	0	1	0	1
MD 31044: ENC_IS_DIRECT2[n]	– / (1)	0	1	0	1
MD 31020: ENC_RESOL[n]	–	Marks/rev	Marks/rev	Marks/rev	Marks/rev
MD 31030: LEADSCREW_PITCH	–	mm/rev.	mm/rev.	–	–
MD 31050: DRIVE_AX_RATIO_DENOM[n]	–	Load revs	–	Load revs	see Note
MD 31060: DRIVE_AX_RATIO_NUMERA[n]	–	Motor rpm if infeed present	–	Motor revs	see Note
MD 31070: DRIVE_ENC_RATIO_DENOM[n]	–	Encoder revs	Encoder revs	Encoder revs	Encoder revs
MD 31080: DRIVE_ENC_RATIO_NUMERA[n]	–	Motor-side encoder*	Motor revs	Motor revs	Load revs

– = Does not apply to this combination

* The encode on the motor side in a built-in encoder and therefore does **not have** a measuring gear. The gear ratio is always 1:1.

Note

These machine data are not required for encoder matching (path evaluation). However, they must be entered correctly for the setpoint calculation! Otherwise the required servo gain (K_V) factor will not be set.
Load revolutions are entered in MD 31050: DRIVE_AX_RATIO_DENOM, and motor revolutions in MD 31060: DRIVE_AX_RATIO_NUMERA.

© Siemens AG, 2003. All rights reserved

Coding the machine data

The index of the following machine data are coded at the encoder [Encodernr.] : encoder 0 or 1

Encoder-dependent machine data	Meaning
MD 31070: DRIVE_ENC_RATIO_DENOM[n]	(Denominator measuring gearbox)
MD 31080: DRIVE_ENC_RATIO_NUMERA[n]	(Numerator measuring gearbox)
MD 31000: ENC_IS_LINEAR[n]	(Direct measuring system linear scale)
MD 31010: ENC_GRID_POINT_DIST[n]	(Distance between reference marks on linear scales)
MD 31020: ENC_RESOL[n]	(Encoder pulses per resolution) for rotating encoders
MD 31040: ENC_IS_DIRECT[n]	(Encoder is connected directly at the machine)
MD 34320: ENC_INVERS[n]	(Linear measurement system is inverted)
Further machine data without index	
MD 30200: NUM_ENCS	(Number of encoders)
MD 30300: IS_ROT_AX	(Rotary axis)
MD 31030: LEADSCREW_PITCH	(Leadscrew pitch)

Index [n] of the following machine data depend on the servo parameters of the position controller with which the actual value resolution was calculated automatically in the controll:

MD: DRIVE_AX_...[Servo parameter set No.] : 0–5

Parameter-set-dependent machine data	Meaning
MD 31050: DRIVE_AX_RATIO_DENOM[n]	(Denominator load gearbox)
MD 31060: DRIVE_AX_RATIO_NUMERA[n]	(Numerator load gearbox)

For the following machine data, the control does not consider any parameter set nor any indices for coded encoders.

NewConfig-dependent machine data	Meaning
MD 31064: DRIVE_AX_RATIO_DENOM	(Encoder on intermediate gear SW 6.4 and higher)
MD 31066: DRIVE_AX_RATIO2_NUMERA	(Encoder on intermediate gear SW 6.4 and higher)
MD 31044: ENC_IS_DIRECT2	(Encoder on intermediate gear SW 6.4 and higher)
MD 32000: MAX_AX_VELO	(Maximum axis velocity)
MD 34080: REFP_MOVE_DIST	(Reference point distance)
MD 34090: REFP_MOVE_DIST_CORR	(Reference point offset)

Note

These machine data cannot be activated in parts programs with the command NEWCONF or via the HMI operator panel using a soft key.

Variants of actual value sensing

The relevant machine data and relational calculations for the different methods of actual value sensing are described in the following.

© Siemens AG, 2003. All rights reserved

2.3.6 Adjustments to actual-value resolution

Calculation of the ratio is obtained from the associated machine dat and is defined for incremental encoders as follows:

$$\frac{\textit{Calculation resolution}}{\textit{Actual value resolution}} = \frac{\textit{Internal increments/(mm)}}{\textit{Encoder increments/(mm)}}$$

For incremental encoders with rotary axis, the following applies:

$$\frac{\textit{Calculation resolution}}{\textit{Actual value resolution}} = \frac{\textit{Internal increments/(deg.)}}{\textit{Encoder increments/(deg.)}}$$

The internal pulse multiplication factor provided by the measuring system logic module is

- 2048 for raw signal generators on 840D with SIMODRIVE 611 digital
- 128 for raw signal generators with 810D

Linear axis with linear scale

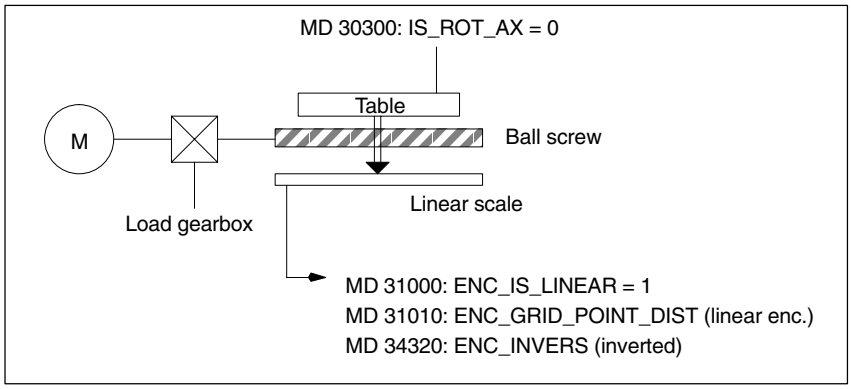


Fig. 2-7 Linear axis with linear scale

In order to adapt the actual value resolution to the calculation resolution, the control calculates the quotients from the “internal increments/mm” and the “encoder increments/mm” as follows:

$$\frac{\textit{Internal increments/mm}}{\textit{Encoder increments/mm}} = \frac{\textit{ENC_GRID_POINT_DIST}[n] * \textit{INT_INCR_PER_MM}}{\textit{Internal multiplication}}$$

The distance for linear encoder is based in the increments.

© Siemens AG, 2003. All rights reserved

Linear axis with rotary encoder at motor

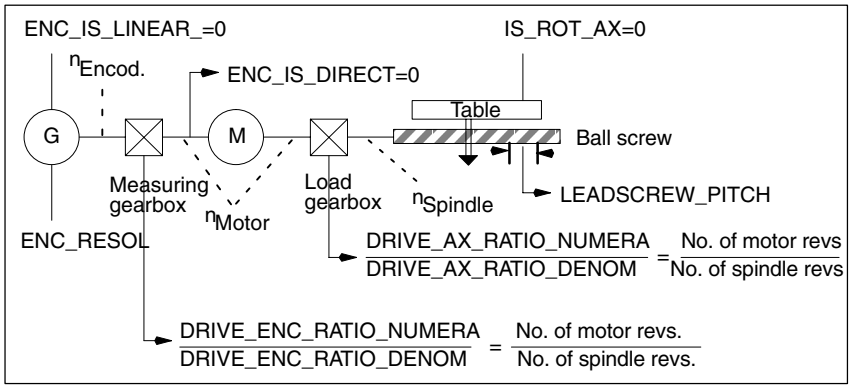


Fig. 2-8 Linear axis with rotary encoder on motor

In order to adapt the actual value resolution to the calculation resolution, the control calculates the quotients from the “internal increments/mm” and the “encoder increments/mm” as follows:

Example of SINUMERIK

Linear axis with rotary encoder (2048 pulses) **on motor**; internal multiplication (2048),
Gearing: Motor/leadscrew 5,
Pitch 10 mm,
10000 increments per mm

$$\frac{\textit{Internal Increments/mm}}{\textit{Encoder increments/mm}} = \frac{1}{\textit{ENC_RESOL}[n] * \textit{Internal multiplication}} * \frac{\textit{DRIVE_ENC_RATIO_NUMERA}[n]}{\textit{DRIVE_ENC_RATIO_DENOM}[n]} * \frac{\textit{DRIVE_AX_RATIO_DENOM}[n]}{\textit{DRIVE_AX_RATIO_NUMERA}[n]} * \textit{LEADSCREW_PITCH} * \textit{INT_INCR_PER_MM}$$

⇒ MD 30300: IS_ROT_AX = 0
MD 31000: ENC_IS_LINEAR[0] = 0
MD 31040: ENC_IS_DIRECT[0] = 0
MD 31020: ENC_RESOL[0]=2048
MD 31030: LEADSCREW_PITCH = 10
MD 31080: DRIVE_ENC_RATIO_NUMERA[0] = 1
MD 31070: DRIVE_ENC_RATIO_DENOM[0] = 1
MD 31060: DRIVE_AX_RATIO_NUMERA[0] = 5
MD 31050: DRIVE_AX_RATIO_DENOM[0] = 1
MD 10200: INT_INCR_PER_MM = 10000

$$\Rightarrow \frac{\textit{Internal increments/mm}}{\textit{Encoder increments/mm}} = \frac{1}{2048 * 2048} * \frac{1}{1} * \frac{1}{5} * 10\ \textit{mm} * 10000\ \textit{Incr./mm} = 0.004768$$

Result: 1 encoder increment corresponds to 0.004768 increments of the internal unit. In practise, the available encoder resolution should not be finer than the internal calculation resolution.

© Siemens AG, 2003. All rights reserved

Linear axis with rotary encoder on the machine

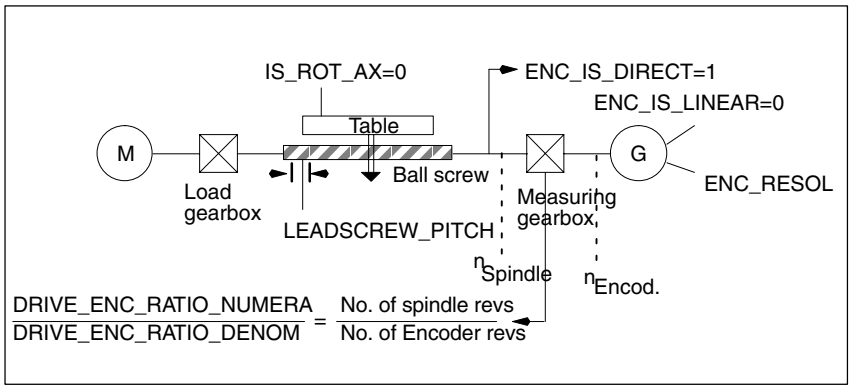


Fig. 2-9 Linear axis with rotary encoder on the machine

In order to adapt the actual value resolution to the calculation resolution, the control calculates the quotients from the “internal increments/mm” and the “encoder increments/mm” as follows:

$$\frac{\textit{Internal increments/mm}}{\textit{Encoder increments/mm}} = \frac{1}{\textit{ENC_RESOL}[n] * \textit{Internal multiplication}}$$

$$* \frac{\textit{DRIVE_ENC_RATIO_NUMERA}[n]}{\textit{DRIVE_ENC_RATIO_DENOM}[n]}$$

$$* \textit{LEADSCREW_PITCH}$$

$$* \textit{INT_INCR_PER_MM}$$

© Siemens AG, 2003. All rights reserved

Rotary axis with rotary encoder at motor

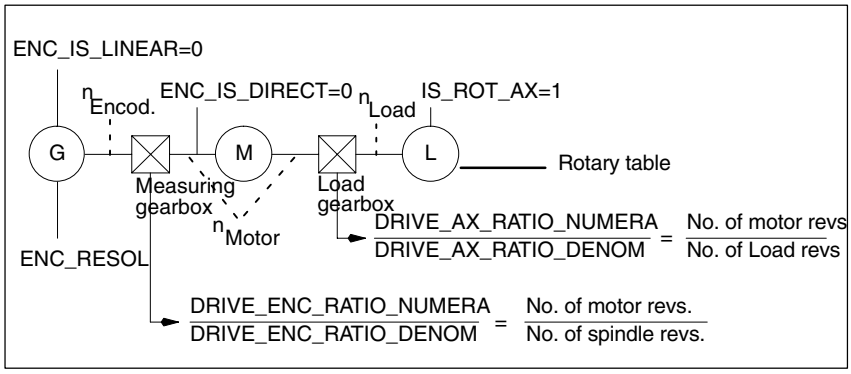


Fig. 2-10 Rotary axis with rotary encoder on motor

In order to adapt the actual value resolution to the calculation resolution, the control calculates the quotients from the “internal increments/degree” and the “encoder increments/degree” as follows:

Example for rotary axis with encoder on motor

Rotary axis with rotary encoder (2048 pulses) **on motor**; internal multiplication (2048),
Gearing: Motor/rotary axis 5,
1000 increments per degree.

$$\frac{\textit{Internal increments/deg}}{\textit{Encoder increments/deg}} = \frac{360\ \textit{deg}}{\textit{ENC_RESOL}[n] * \textit{Internal multiplication}} * \frac{\textit{DRIVE_ENC_RATIO_NUMERA}[n]}{\textit{DRIVE_ENC_RATIO_DENOM}[n]} * \frac{\textit{DRIVE_AX_RATIO_DENOM}[n]}{\textit{DRIVE_AX_RATIO_NUMERA}[n]} * \textit{INT_INCR_PER_DEG}$$

⇒ MD 30300: IS_ROT_AX = 1
MD 31000: ENC_IS_LINEAR[0] = 0
MD 31040: ENC_IS_DIRECT[0] = 0
MD 31020: ENC_RESOL[0]=2048
MD 31080: DRIVE_ENC_RATIO_NUMERA[0] = 1
MD 31070: DRIVE_ENC_RATIO_DENOM[0] = 1
MD 31060: DRIVE_AX_RATIO_NUMERA[0] = 5
MD 31050: DRIVE_AX_RATIO_DENOM[0] = 1
MD 10210: INT_INCR_PER_DEG = 1000

$$\Rightarrow \frac{\textit{Internal increments/deg}}{\textit{Encoder increments/deg}} = \frac{360\ \text{deg}}{2048\ *\ 2048} * \frac{1}{1} * \frac{1}{5} * 1000\ \textit{Incr./deg} = 0.017166$$

Result: 1 encoder increment corresponds to 0.017166 increments in the internal unit. So the encoder resolution is coarser than the computation resolution by a factor of 58.

© Siemens AG, 2003. All rights reserved

Rotary axis with rotary encoder on the machine

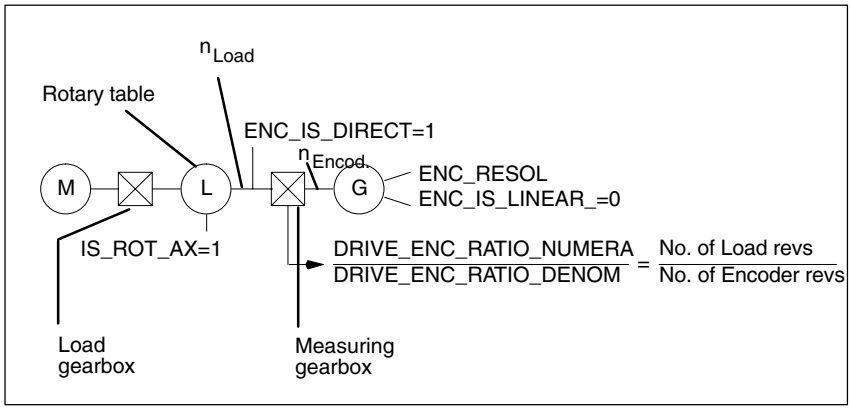


Fig. 2-11 Rotary axis with rotary encoder on the machine

In order to adapt the actual value resolution to the calculation resolution, the control calculates the quotients from the “internal increments/degree” and the “encoder increments/degree” as follows:

$$\frac{\textit{Internal increments/deg}}{\textit{Encoder increments/deg}} = \frac{360\ \textit{deg}}{\textit{ENC_RESOL}[n] * \textit{Internal multiplication}} * \frac{\textit{DRIVE_ENC_RATIO_NUMERA}[n]}{\textit{DRIVE_ENC_RATIO_DENOM}[n]} * \textit{INT_INCR_PER_DEG}$$

Encoder on intermediate gear (SW 6.4 and higher)

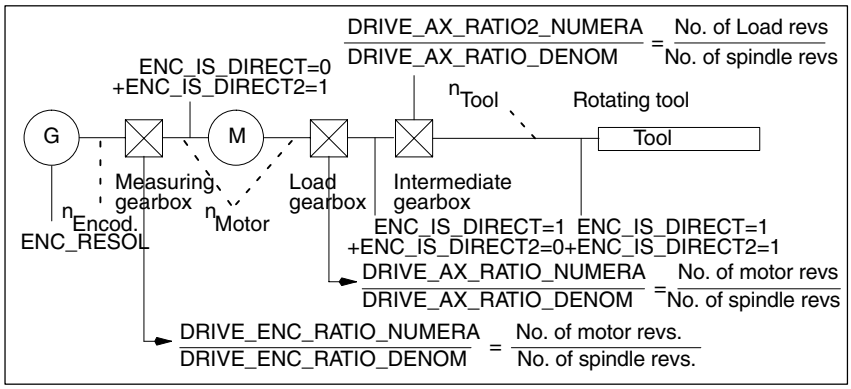


Fig. 2-12 Intermediate gear with encoder directly on the rotating tool

In order to adapt the actual value resolution to the calculation resolution, the control calculates the quotients from the “internal increments/mm” and the “encoder increments/mm” as follows:

$$\frac{\textit{Internal increments/deg}}{\textit{Encoder increments/deg}} = \frac{360\ \textit{deg}}{\textit{ENC_RESOL}[n] * \textit{Internal multiplication}} * \frac{\textit{DRIVE_ENC_RATIO_NUMERA}[n]}{\textit{DRIVE_ENC_RATIO_DENOM}[n]} * \textit{INT_INCR_PER_DEG}$$

© Siemens AG, 2003. All rights reserved

2.4 Closed-loop control

General

The closed-loop control of an axis consists of the current and speed control loop of the drive plus a later-level position control loop in the NC.

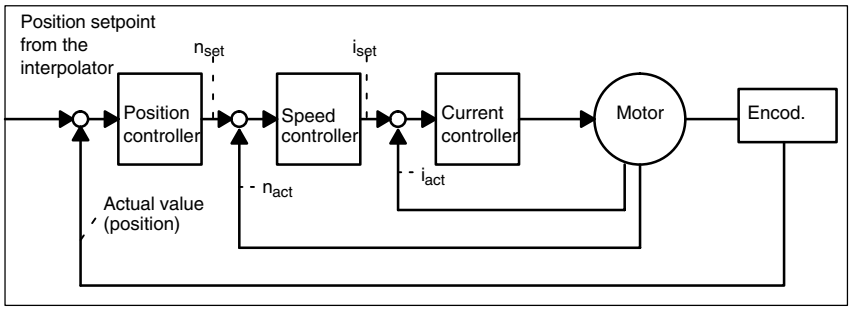


Fig. 2-13 Block diagram of axis closed-loop control

The speed and current closed-loop controls for SIMODRIVE 611 are explained in:

References: /IAD/, “Installation and StartUp Guide” SINUMERIK 840D/611D
/IAC/, “Installation and StartUp Guide” SINUMERIK 810D
/PJU/, “Planning Guide” Converters

The basic structure of an axis/spindle closed-loop control is illustrated below:

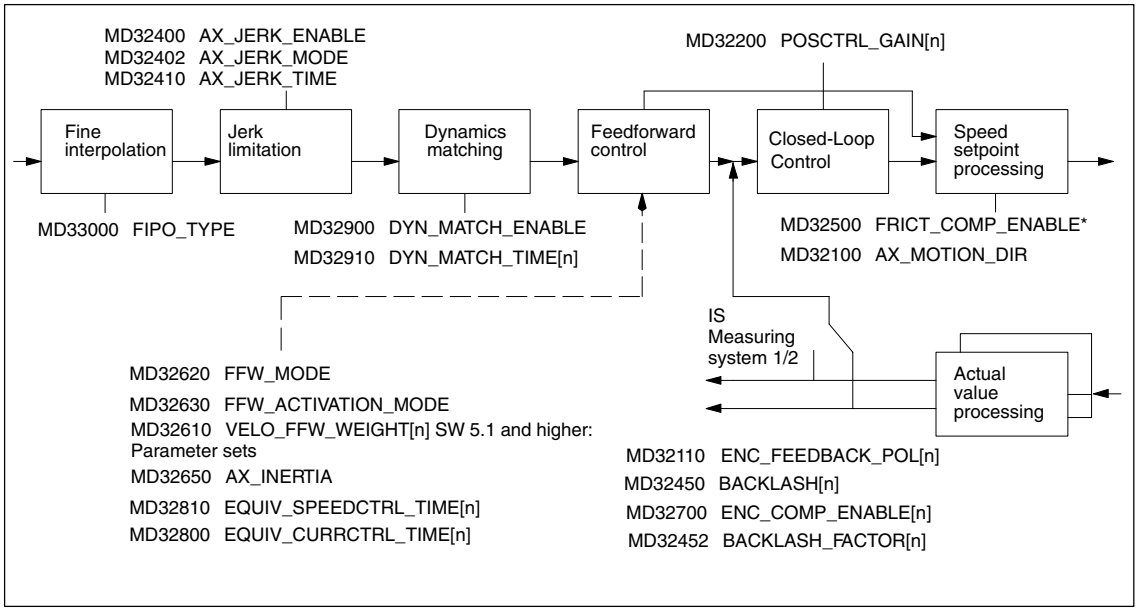


Fig. 2-14 Additional servo parameter set for position control

For a description of the feedforward control, backlash, friction compensation with *further machine data and leadscrew error compensation, see:

References: /FB/, K3, “Compensations”

For a description of jerk limitation see:

References: /FB/, B2, “Acceleration”

© Siemens AG, 2003. All rights reserved

Fine interpolation

The fine interpolator (FIPO) is used to adjust the setpoint of the (generally lower) interpolator control cycle to the later position control cycle.

Fine interpolation further improves the quality of the contour (decreasing the step effect of the speed setpoint).

There are three types of FIPOs:
1: Differential FIPO
2: Cubic FIPO
3: Cubic FIPO, optimized for operation with feedforward control

The method of fine interpolation can be selected in MD 33000: FIPO_TYPE (fine interpolator type).

A differential FIPO not only performs cycle matching but also calculates a mean value (smoothing) from an IPO cycle. The cubic FIPO type 3 supplies the best contour accuracy in additon to the cycle adaption.

Servo gain factor

A high servo gain factor setting in MD 32200: POSCTRL_GAIN[n] is required to prevent the occurrence of significant contour deviations.

However, if the servo gain factor is too high, instability, overshooting and possible impermissibly high loads on the machine will result.

The maximum permissible servo gain factor depends on the following:

- Design and dynamics of the drive
  (rise time, acceleration and deceleration capabilities)
- Quality of the machine (elasticity, vibration suppression)
- Position control cycle

The servo gain factor is defined as follows:

$$\textit{Servo gain} = \frac{\textit{Speed}}{\textit{Following error}} ; \frac{[m/min]}{[mm]}$$

$\frac{[m/min]}{[mm]}$ is the unit of the servo gain factor according to VDI standard

Servo gain factor setting for SINUMERIK 840D/810D

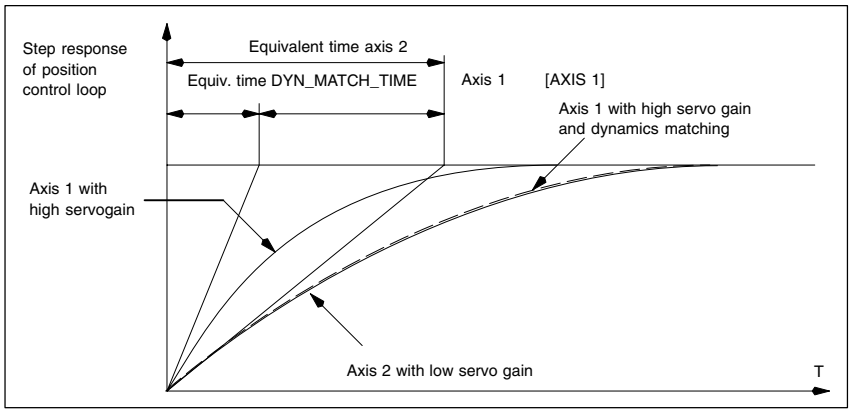


Fig. 2-15 Dynamic response adaptation

© Siemens AG, 2003. All rights reserved

Dynamic response adaptation

The purpose of dynamic response adaptation is to set an identical following error for axes with different servo gain factors. The optimum contour precision for axes interpolating with each other can thus be achieved without reduced control quality. A high servo gain factor for an axis can be maintained, which guarantees optimum disturbance suppression of the axis.

The function is activated in MD 32900: DYN_MATCH_ENABLE (dynamics matching).

The axes are matched in MD 32910: DYN_MATCH_TIME[n] (time constant for dynamics matching). With this MD, the equivalent time constant of the position control loop of the axes with a higher servo gain factor is matched to the axis with the lowest servo gain factor.

Bits set to 1 in MD 32910: DYN_MATCH_TIME[n] is the difference between the equivalent time constants of the slowest closed-loop control circuit for the relevant axis.

Example of dynamics matching of axes 1, 2 and 3 (without speed feedforward control):

The equivalent time constant of the position control loop is as follows:

- Axis 1 : 30ms
- Axis 2 : 20ms
- Axis 3 : 24ms

⇒ Axis 1 is dynamically the slowest axis

⇒ The value settings for MD 32910: DYN_MATCH_TIME[n] (time constant for dynamics matching) are therefore as follows:

- Axis 1 : 0ms
- Axis 2 : 10ms
- Axis 3 : 6ms

The equivalent time constant of the position control loop of an axis is calculated according to the following formula:

Approximation formula

No feedforward control active

$$T_{equivalent} \approx \frac{1}{\text{MD 32200: POSCTRL_GAIN [1 / s]}} \quad (1)$$

Speed feedforward control

$$T_{equivalent} \approx \text{MD 32810: EQUIV_SPEEDCTRL_TIME} \quad (2)$$

Torque feedforward control

$$T_{equivalent} \approx \text{MD 32800: EQUIV_CURRCTRL_TIME} \quad (3)$$

Note

If dynamics matching is executed for one geometry axis, dynamics matching must also be activated for all further geometry axes by setting MD 32900: DYN_MATCH_ENABLE= 1.

References: /IAD/, “Installation and StartUp Guide”, SINUMERIK 840D / 611D
/IAC/, SINUMERIK 810D “Installation and StartUp Guide”

© Siemens AG, 2003. All rights reserved

2.4.1 Parameter sets of the position controller

Six different parameters sets

The position control can operate with 6 different servo parameter sets. This information is used for

– fast adaptation of the position control to altered machine characteristics during operation, e.g. a gear change of the spindle.

– matching the dynamics of an axis to another axis, e.g. during tapping.

Switch over parameter set

The following machine data can be changed by switching over the parameter set.

- Denominator load gearbox MD 31050: DRIVE_AX_RATIO_DENOM[n]
- Numerator load gearbox MD 31060: DRIVE_AX_RATIO_NUMERA[n]
- Servo gain factor MD 32200: POSCTRL_GAIN [n]
- Backlash compensation (SW 5 and higher)
  MD 32452: BACKLASH_FACTOR[n]
- Feedforward control factor (SW 5 and higher)
  MD 32610: VELO_FFW_WEIGHT[n]
- Exact stop limits (SW 5 and higher)
  MD 36012: STOP_LIMIT_FACTOR[n]
  MD 36000: STOP_LIMIT_COARSE
  MD 36010: STOP_LIMIT_FINE
  and standstill window MD 36030: STANDSTILL_POS_TOL
- Equivalent time constant MD 32800: EQUIV_CURRCTRL_TIME [n]
  Current control loop for torque feedforward control
- Equivalent time constant MD 32810: EQUIV_SPEEDCTRL_TIME[n]
  Speed control circuit for speed feedforward control
- Time constant of MD 32910: DYN_MATCH_TIME [n]
  Dynamic response adaptation
- Threshold value for MD 36200: AX_VELO_LIMIT [n]
  Velocity monitoring

For further information on machine data by switching over the parameter sets, please refer to Section 2.5 “Optimizing the control”.
Parameter set extensions in SW 5.1 and higher are described in Subsection 2.4.2.

Tapping or thread cutting

The following applies to **parameter sets for axes**:

- for machine axes **not** involved in tapping or thread cutting, parameter set 1 (index=0) is always used.
  The further parameter sets need not be considered.
- for machine axes involved in tapping or thread cutting, the same parameter set number as for the current gear stage of the spindle is activated.
  All parameter sets correspond to the gear stages and must therefore be parameterized.

The current parameter set is displayed in operating area “Diagnostics” in the “Service axis” display. The parameter sets for gear stages 1 to 5 are referred to as interpolation parameters.

© Siemens AG, 2003. All rights reserved

Parameter sets for gear stage change

Interpolation parameter sets for m gear stage change:

In the case of spindles, each gear stage is assigned its own parameter set. As a function of the the start of IS “Actual gear stage” (DB31, ... DBX16.0 to 16.2), the corresponding parameter set is activated.

IS “Actual gear stage”		Active parameter set
000	1st gear step	2 (Index=1)
001	1st gear step	2 (Index=1)
010	2nd gear step	3 (Index=2)
011	3rd gear step	4 (Index=3)
100	4th gear step	5 (Index=4)
101 110 111	5th gear step	6 (Index=5)

For furtherinformation about gear stages for spindles, refer to:
References: /FB/, S1, “Spindles”

2.4.2 Extension of the parameter set (SW 5.1 and higher)

Application

Some machines use the same drive for moving various machine parts, what in view of considerably varying speeds results in a gear stage change. With each gear stage change, the corresponding parameter set is also switched over.

Now several parameter sets are provided for further practical applications and for setting the feedforward control of the servo loop.

Functionality

To optimize control during start-up, these codable parameter sets support practice-oriented start-up by substantially reducing configuration expenditure with regard to the new functions, such as backlash compensation, feedforward control factor, exact stop limits and zero speed window.

© Siemens AG, 2003. All rights reserved

New parameter sets

Until now it has been possible to change the gear ratio and other servo loop parameters (such as achievable control loop gain) by switching over the servo parameter set. The already existing machine data with parameter set coding are extended as follows:

Parameter set coding (modifiable as a function of parameter set)

• Backlash compensation	MD 32450: BACKLASH[n] Max. no. of encod.
• Feedforward control factor	MD 32610: VELO_FFW_WEIGHT[n]
• Exact stop limits	MD 36000: STOP_LIMIT_COARSE MD 36010: STOP_LIMIT_FINE
• Standstill window	MD 36030: STANDSTILL_POS_TOL

Weighting factor for parameter set changeover

• Weighting factor for:	**MD 32452: BACKLASH_FACTOR[n]:**
1. Backlash compensation	MD 32450: BACKLASH[n]
• Weighting factor for:	**MD 36012: STOP_LIMIT_FACTOR[n]:**
1. Exact stop limits	MD 36000: STOP_LIMIT_COARSE MD 36010: STOP_LIMIT_FINE
2. Standstill window	MD 36030: STANDSTILL_POS_TOL

Machine data tried and tested to date

Further machine data with parameter set coding

The following existing machine data can be coded using parameter sets and have already been tried and tested during the start-up of the NC:

Denominator load gearbox	MD 31050: DRIVE_AX_RATIO_DENOM
Numerator load gearbox	MD 31060: DRIVE_AX_RATIO_NUMERO
Equivalent time const. Current control loop	MD 32800: EQUIV_CURRCTRL_TIME
Equivalent time const. Speed control loop	MD 32810: EQUIV_SPEEDCTRL_TIME
Time constant dynamic adaption	MD 32910: DYN_MATCH_TIME
Threshold value for velocity monitoring	MD 36200: AX_VELO_LIMIT

Benefits

The indirect switchover **of a single** function: (e.g. MD 32452 unequal 1).
The parameterset dependent function is only really relevant as it is needed.

The indirect switchover of **several** functions: (e.g. MD 36012 unequal 1).
The common weighting factor causes the ratios of all parameter set dependent functions in the different machine data to be identical another to one. To switch over several functions, one single machine data can be sufficient.

In this way, configuration errors are avoided and input work reduced.

© Siemens AG, 2003. All rights reserved

Activating the parameter set coding

Presetting without parameter set coding
If the following machine data:

1. MD 32452: BACKLASH_FACTOR = 1
2. MD 32610: VELO_FFW_WEIGHT = 1 and
3. MD 36012: STOP_LIMIT_FACTOR = 1

remain set to (1), the control system is compatible with previous software versions.

Activating the parameter set coding:
If the default settings of the following machine data:

1. MD 32452: BACKLASH_FACTOR is changed to a value unequal to (1.0), then the backlash compensation is also changed depending on the parameter set.
2. MD 32610: VELO_FFW_WEIGHT is changed to a value unequal to (1.0), then the feedforward control factor is changed depending on the parameter set.
3. MD 36012: STOP_LIMIT_FACTOR is changed to a value unequal to (1.0), then the exact stop limits and the zero speed window are changed, depending on the parameter set.

When loading old archives (data backed up from previous software versions), MD 32610: VELO_FFW_WEIGHT is automatically set to the same value for all indices so that the control system's response is compatible.

The machine data MD 32452: BACKLASH_FACTOR and MD 36012: STOP_LIMIT_FACTOR do not exist at all in old archives. In this case, the default setting remains automatically active. The response is compatible also in this case.

Example

Effects of various parameter sets with backlash compensation

MD 32450: BACKLASH[AX1] = 0.01

MD 32452: BACKLASH_FACTOR[0,AX1] = 1.0	Parameter set 1
MD 32452: BACKLASH_FACTOR[1,AX1] = 2.0	Parameter set 2
MD 32452: BACKLASH_FACTOR[2,AX1] = 3.0	Parameter set 3
MD 32452: BACKLASH_FACTOR[3,AX1] = 4.0	Parameter set 4
MD 32452: BACKLASH_FACTOR[4,AX1] = 5.0	Parameter set 5
MD 32452: BACKLASH_FACTOR[5,AX1] = 6.0	Parameter set 6

In parameter set 1 (index 0) of the first axis (AX1), a backlash compensation factor with the value 1.0 has the following effect:

1.0 * MD 32450: BACKLASH = 0.01mm (or inches or degrees).

In parameter set 2, the backlash compensation is twice as large and in parameter set 3, it is triple as large, etc. The maximum value is 100.

Supplementary conditions

With SW 5.1 and higher, the function extension is available for all control variants.

Control response with POWER ON, RESET, REPOS etc.

The new or modified data are activated via the soft key "Enable machine data" or by a RESET or POWER ON. Mode switchover, block search or REPOS have no influence.

© Siemens AG, 2003. All rights reserved

2.5 Optimization of the control (SW 5.0 and higher)

2.5.1 Position controller: Positional deviation injection (SW 5.1 and higher)

Application

The stability and positioning response of axes with a low natural frequency (up to approx. 20Hz) and a mechanical design capable of generating oscillations is improved by active vibration damping with simultaneous use of the feedforward control.

In order to accomplish this, the difference in the position of two measuring systems is formed and injected as an additional current setpoint for feedforward control after an appropriate weighting of MD 32950: POSCTRL_DAMPING. The function is used predominantly for axes with strong tendency to vibrate.

Functionality

The injection of the positional deviation is carried out in the NC during the position controller cycle of the higher-level position control loop in the NC. The difference in position between a direct measuring system and an indirect measuring system is formed and injected as an additional current setpoint depending on the weighting of the machine data MD 32950: POSCTRL_DAMPING.

- Direct measuring system: MD 31040: ENC_IS_DIRECT[1]=1, the encoder for actual position detection is mounted directly on the machine (load encoder).
- Indirect measuring system: MD 31040: ENC_IS_DIRECT[0] = 0, the encoder for actual position detection is mounted on the motor (motor encoder).

MD 32950

The function is activated via data MD 32950: POSCTRL_DAMPING = 1. It is possible to enter both positive and negative values which will then serve to normalize the injection of the positional deviation.

MD 32950: POSCTRL_DAMPING = 0 is entered by default. The injection of the positional deviation is active in this case.

Note

The weighting of MD 32950 can be set, e.g. using step responses. If the control approaches the stability limit (vibration inclination increases), then the parameter is too large.

Boundary condition

- The function extension is available for all control variants with SW 5.1 and higher which use SIMODRIVE 611 digital drives.
- The function can only be used in the case of axes with two encoders MD 30200: NUM_ENCS = 2 whereby one of the encoders must be parameterized as an indirect measuring system and the other encoder as a direct measuring system.

If these conditions are not fulfilled, the function is acknowledged with the reset alarm 26016 when you try to activate MD 32950: POSCTRL_DAMPING = 1. The function remains deactivated internally when this alarm is active.

© Siemens AG, 2003. All rights reserved

2.5.2 Position controller position setpoint filter: New balancing filter (SW 5.0 and higher)

For speed and torque feedforward control

Application

With feedforward control active, the position setpoint is sent through a so-called balancing filter before it reaches the controller itself. It is thus possible to feedforward control the speed setpoint at 100%, without resulting in overshoots when positioning.

Functionality

Until now, it was possible to compensate the negative effects of the balancing filter (a low pass with settable time constant), which has remained unchanged since SW 1 and which has admitted instead of an overshoot also undesired undershoots of some 10 micrometers, as follows:

- Setting a compromise of overshoots and undershoots or
- setting the positioning slightly overshooting and reducing the amplitude of these overshoots by way of position setpoint signal smoothing and jerk limitation, or
- setting a speed controller with reference model instead of PI behavior (only possible with 840D), or
- setting the filter time to zero and setting the feedforward control factor to a value less than 100%, or
- to sacrifice of the feedforward control and to bring the machine to a very high position controller gain using the dynamic stiffness control. This measure requires a stiff machine.

With SW 5 and higher, a second, improved balancing filter is therefore available. The existing filter is still installed and is active with the same function when transferring existing archives (e.g. in the case of upgrades).

Advantages

The new balancing filter provides the following improvements:

- An axis with feedforward control has a considerably lower inclination to undershoots when positioning.
- Achieves a higher accuracy at bent contours (can be measured, e.g. using the circularity test) and can be set more easily.
- A part of the setting is carried out by the control system automatically.

Filter activation with MD 32620

The new filter only becomes active by changing the machine data MD 32620: FFW_MODE and selecting the values 3 and 4.

The desired active variant of the feedforward control with new balancing is selected as follows using the machine data MD 32620: FFW_MODE:

3 = Speed feedforward control with new balancing

4 = Torque feedforward control (only possible with SINUMERIK 840D) with new balancing.

© Siemens AG, 2003. All rights reserved

For reasons of a compatible response of archives that contain only changes compared with the default settings, no default values can be set for these new values 3 and 4 in the new software versions.

Activation of feedforward control

The feedforward control for all axes can be activated and deactivated by the parts program by the commands FFWON and FFWOF when the machine data MD 32630: FFW_ACTIVATION_MODE remain unchanged.

Control response with POWER ON, RESET, REPOS etc.

In the case of POWER ON and RESET, as well as with “Enable machine data”, the setting data of the feedforward control are read in anew (see the appropriate values of the machine data). Mode change, block search and repositioning have **no influence** on the feedforward control.

Supplementary conditions

With SW 5.1 and higher, the function extension is available for all control variants.

New setting rule for MD 32810 and MD32800

If the new filter is active, the setting rule for the data in the machine data MD 32810: EQUIV_SPEEDCTRL_TIME a machine data MD 32800: EQUIV_CURRCTRL_TIME will change.
This means that if the old balancing filter has been active until now and you want to change to the new one, the following actions must be done:

Setting the equivalent time constant with speed feedforward control

- If MD 32620: FFW_MODE = 1 has been set until now:
  MD 32620: Set FFW_MODE = 3,
  MD 32610: Set VELO_FFW_WEIGHT = 1 and
  MD 32810: Set EQUIV_SPEEDCTRL_TIME to another value.

Setting the equivalent time constant with torque feedforward control

- If MD 32620: FFW_MODE = 2 has been set until now:
  MD 32620: Set FFW_MODE = 4,
  MD 32610: Set VELO_FFW_WEIGHT = 1 and
  MD 32800: Set EQUIV_CURRCTRL_TIME to another value.

This also applies when you load an older **archive** into the control system, e.g. from a previous version. When changing, e.g. MD 32620: FFW_MODE from 1 to 3, MD 32810: EQUIV_SPEEDCTRL_TIME may **not** simply be left at its old value, but must be **set anew**. Otherwise, no improvement will be achieved, but rather a worsening of the positioning response.

Recommended setting in case of recommissioning

In case of recommissioning or if standard values have been loaded (switch position 1 at the start-up switch on POWER ON),
MD 32620: FFW_MODE = 1 and MD 32610: VELO_FFW_WEIGHT = 1 are set by default. This is **not** the recommended setting, but has only been selected from compatibility reasons.

In the case of recommissioning, it is recommended to enter MD 32620: FW_MODE = 3. In this case, only the balancing time MD 32810: EQUIV_SPEEDCTRL_TIME must be adapted accordingly for the speed feedforward control.

© Siemens AG, 2003. All rights reserved

Setting the equivalent time constant of the speed control loop

MD 32810 speed feedforward control

It is advisable to let the axis move back and forth using an automatic program to observe its approach to the target position, i.e. the actual position value of the active measuring system using the Servo Trace (MMC 102 or MMC 103 or, with SW 6.1 or higher, HMI Advanced or PG required).
It is also possible to output the actual position value to the digital-to-analog converters of the drive module and to use an oscilloscope for monitoring.

The starting value for setting is the time constant of the speed control loop. This can be read from the reference frequency characteristic of the speed control loop. In the frequently occurring case PI controller with special value smoothing, the equivalent time can be approximately read from the drive machine data 1500–1503.
Another possibility is to record set speed and actual speed value at constant acceleration using an oscilloscope and to measure the time lag of the actual speed value.

Now, enter this starting value (e.g. 1.5ms):
MD 32810: EQUIV_SPEEDCTRL_TIME = 0.0015.
Then traverse the axis to and fro and watch the course of the actual position value at the target position strongly enlarged.

The following rule applies to the manual fine adjustment:

- Overshoots noticed:
  MD 32810: **Increase** EQUIV_SPEEDCTRL_TIME
- Too slow approximation noticed:
  MD 32810: **Reduce** EQUIV_SPEEDCTRL_TIME

Increasing MD 32810

Increasing MD 32810: EQUIV_SPEEDCTRL_TIME causes the axis getting slower and the geometric contour error at bends a bit larger. It has a similar effect as reducing the position controller gain MD 32200: POSCTRL_GAIN. This can also be watched in the Diagnostics area in the screen form “Service Axis” based on the servo gain value calculated.

Reducing MD 32810

Reducing MD 32810 EQUIV_SPEEDCTRL_TIME makes the axis faster. In other words, it is tried to dimension MD 32810: EQUIV_SPEEDCTRL_TIME as low as possible whereby the limits are set by the overshoots when positioning.

Fine adjustment of MD 32810

Experience has shown that the initial value is changed only to a very small degree during the fine adjustment, typically by 0.25ms up or down.
With an initial value of, e.g. 1.5ms, the manually found optimum is normally in the range between 1.25ms and 1.75ms.

In the case of axes equipped with direct measuring systems (load encoders) and strong elasticity, you may possibly accept little overshoots of some micrometers. These can be reduced using the position setpoint filters for

- dynamic response adaptation (MD 32910: DYN_MATCH_TIME) and
- jerk (MD 32410: AX_JERK_TIME),

making the axis more slowly again.

© Siemens AG, 2003. All rights reserved

Equal axis data in an interpolation compound

Set all axes of an interpolation compound in the data
MD 32200: POSCTRL_GAIN, MD 32620: FFW_MODE,
MD 32610: VELO_FFW_WEIGHT, MD 32810: EQUIV_SPEEDCTRL_TIME (or MD 32800: EQUIV_CURRCTRL_TIME), MD 32400: AX_JERK_ENABLE, MD 32402: AX_JERK_MODE, MD 32410: AX_JERK_TIME

to the same value. For checking, the servo gain display in the Service Axis screen form is used. If the same values are not possible for a/m data, an adaptation can be made using MD 32910: DYN_MATCH_TIME. It is thus possible to display the same servo gain value, what, however, is seldom. Normally, different servo gain values are a sign that one of the following cases have occurred:

- The gear ratio factors do not match in one or several axes,
- the setting data of the feedforward control are not correct, and/or
- the setting data of the jerk filter are anywhere not correct.

Automatic switchover when changing the position control cycle

If the position control cycle (MD 10050: SYSCLOCK_CYCLE_TIME) is changed or the acceptance time of the speed setpoints is changed in order to increase the Kv (servo gain factor) (MD 10082: CTRLOUT_LEAD_TIME) or is switched over to Dynamic Stiffness Control (MD 32640: STIFFNESS_CONTROL_ENABLE), the adjustment of MD 32810 EQUIV_SPEEDCTRL_TIME had to be repeated to date, since the optimum value substantially changed.

With MD 32620: FFW_MODE = 3 or 4
the control system will now take into account the above mentioned changes automatically so that the MD 32810: EQUIV_SPEEDCTRL_TIME now needs no longer to be readjusted. After major changes, however, you should nevertheless check the positioning behavior (via Servo Trace).

Example of speed feedforward control

Programming sample of a selection of the speed feedforward control with new balancing and default setting: FFWON and FFWOF active.

MD 32620: FFW_MODE[X1] = 3 ; New speed feedforward control mode
MD 32630: FFW_ACTIVATION_MODE[X1] = 1 ; FFWON and FFWOF are active in the NC program.

Now FFWON enables the speed feedforward control in the program (with all axes of the channel with the same settings as X1);
FFWOF will disable them again.
MD 20150: GCODE_RESET_VALUES[23] can also be used to set FFWON as the default for each channel.

MD 32620: FFW_MODE[X1] = 3 ; New speed feedforward control mode
MD 32630: FFW_ACTIVATION_MODE[X1] = 0 ; FFWON and FFWOF are active in the NC program.

In this case, the speed feedforward control with X1 is enabled continuously, also in JOG mode. MD 20150: GCODE_RESET_VALUES[ ], FFWON and FFWOF do not act on X1. This can be reasonable if the machine may only run with feedforward control, e.g. for reasons of accuracy, or if you want to test the feedforward control also without a program during start-up.

Note

The setting of the feedforward control must be the same for all axes of an interpolation compound.

© Siemens AG, 2003. All rights reserved

Setting the equivalent time constant of the current control loop

MD 32800 Torque feedforward control as an option

For the setting of the time constant of the current controller, the same rules and recommendations apply as with the speed feedforward control. To set the time constant with MD 32800: EQUIV_CURRCTRL_TIME, however, the activation of the filter for the feedforward control must be enabled both as previously with MD 32620: FFW_MODE = 4 and, **in addition with the AND option** in the drive.

As with the activation of the existing filter with MD 32620: FFW_MODE = 2, all further machine data must also be set taking into account the appropriate elasticity limits of the machine.

Limitation to stiff machines

Experience has shown that this expenditure is only worth in the case of very stiff machines and requires appropriate experience. The elasticities of the machine are often excited due to the injection of the torque so strong that the existing vibrations neutralize the gain in contour accuracy. In this case it is worth to try out the use of Dynamic Stiffness Control (DSC), MD 32640: STIFFNESS_CONTROL_ENABLE as an alternative.

Example for torque feedforward control

Programming sample of a selection of the speed feedforward control with new balancing and default setting: FFWON and FFWOF active.

MD 32620: FFW_MODE[X1] = 4 ; New torque feedforward control mode.
MD 32630: FFW_ACTIVATION_MODE[X1] = 1 ; FFWON and FFWOF active.

Now FFWON enables the torque feedforward control in the program
(with all axes of the channel with the same settings as X1);
FFWOF will disable them again.
MD 20150: GCODE_RESET_VALUES[23] can be used to set FFWON also for each program by default.

MD 32620: FFW_MODE[X1] = 4 ; New torque feedforward control mode.
MD 32630: FFW_ACTIVATION_MODE[X1] = 0 ; FFWON and FFWOF active.

In this case, the torque feedforward control with X1 is enabled continuously, also in JOG mode. MD 20150: GCODE_RESET_VALUES[ ], FFWON and FFWOF do not act on X1. This can be reasonable if the machine may only run with feedforward control, e.g. for reasons of accuracy, or if you want to test the feedforward control also without a program during start-up.

For further information on the method of operation of the feedforward control with regard to the position setpoints, speed and torque, please refer to:

References: /FB2/, K3, “Following Error Compensation (Feedforward Control)” and “Reference Manual of the Machine Data”.

Note

The setting of the feedforward control must be the same for all axes of an interpolation compound.

© Siemens AG, 2003. All rights reserved

2.5.3 Position controller position setpoint filter: New jerk filter (SW 5.1 and higher)

Application

In some applications, such as when milling sculptured surfaces, it can be advantageous to smooth the position setpoint curves to obtain better surfaces thanks due to less excitations of machine vibrations.

Functionality

The filtering effect of the position setpoints must be as strong as possible, but without having an inadmissible negative effects on the contour accuracy. Furthermore, the filter must have a smoothing response as balanced as possible, i.e. when you traverse the same contour forward and reverse, the characteristic rounded by the filter should be as similar as possible in both directions.

The effect of the filter can be watched using the display of the effective servo gain factor in the Axes service screen. The filtering effect slightly rounds the position setpoints, thus reducing the path accuracy so that with increasing filter time a smaller effective servo gain factor is displayed.

Benefits

The filter which has already been existing since SW 1 and which is set with MD 32400: AX_JERK_ENABLE = 1 and MD 32410: AX_JERK_TIME = filer time fulfills the conditions of a strong filtering effect with a smoothing response as balanced as possible for small time constants up to approx. 10 ms.

A considerably better result is achieved using this new filter type for axial jerk limitation with floating averaging. In this case, filter time constants of approx. 20 ... 40 ms can be used, depending on the particular machine.

Thanks to the floating averaging, the new filter achieves lower contour errors with the same smoothing effect, compared with the previous 2nd degree filter type. The contour of the smoothing behavior is to a large degree balanced.

Enabling the filter using MD 32402

Machine data MD 32402: AX_JERK_ENABLE can be used to enable the new position setpoint filter acc. to the following settings:

- **MD 32402: AX_JERK_MODE = 2 Filter mode, select new jerk filter**
- MD 32410: AX_JERK_TIME = 0.02 Set filter time in seconds (20 ms)
- MD 32400: AX_JERK_ENABLE = 1 Enable filter calculation

If no filter mode MD 32402: AX_JERK_MODE = 2 has been activated beforehand, POWER ON must be initiated once. Otherwise, it is sufficient to “Enable machine data” or to provide RESET on the machine control panel.

For reasons of compatibility, MD 32402: AX_JERK_MODE = 1 is set by default.

Note

In the case of new machines, the new filter is generally recommended with MD 32402: AX_JERK_MODE = 2. Only if very high filter times are needed and the contour accuracy play a minor part, what sometimes occurs in the case of positioning axes, the old filter can be more advantageous.

© Siemens AG, 2003. All rights reserved

Fine adjustment

The fine adjustment of the position setpoint filter is carried out as follows:

- Assess the traversing response of the axis (e.g. based on positioning processes with Servo Trace).
- Change filter time in MD 32410: AX_JERK_TIME.
- Activate the changed time via “Enable machine data” or RESET on the machine control panel.

Disabling

Disabling the position setpoint filter:

- Disabling the filter calculation: MD 32410: AX_JERK_ENABLE = 0
- Activate the interlock via “Enable machine data” or RESET on the machine control panel.

Supplementary conditions

The position setpoint filter is available in all control system variants as follows:

- In SW 5.1 and higher, the effective filter times are limited in the range between a minimum of 1 position control cycle up to a maximum of 16 position control cycles (15 position control cycles are available).
- In SW 5.2 and higher, the effective filter times are limited in the range between a minimum of 1 position control cycle up to a maximum of 32 position control cycles (31 position control cycles are available).

Further supplementary conditions regarding the filter effect:

- The display of the calculated servo gain factor in the Axes service screen form displays smaller values than it would be appropriate based on the filter effect.
- The path accuracy is better than it could be expected from the displayed servo gain factor. Therefore, when changing over
  MD 32400: AX_JERK_MODE = 1
  to MD 32400: AX_JERK_MODE = 2
  it is possible that the displayed servo gain factor is reduced with the filtering time remaining unchanged although the path accuracy gets better.

Axes interpolating with each other must be set identically.
If you have found an optimum for such axes, the axis with the largest filter time amongst these settings must be entered for all axes of the interpolation compound.

For further information on jerk limitation on the interpolator level, please refer to:

References: /FB/, B2, “Axis-related jerk limitation” and “Axis-Specific Machine Data”.

© Siemens AG, 2003. All rights reserved

2.5.4 Position feedback loop with PI controller (SW 6.4 and higher)

Function

For standard applications up to and including SW 6.3, the core of the position controller is a P controller with the manipulation options connected in series upstream described above. The option of connecting in an integral component for special applications (e.g. electronic gear) has been added in SW 6.4. The resulting PI controller then corrects the error between setpoint and actual positions down to zero in a finite, settable time period when the appropriate machine data are set accordingly.



Caution

When the PI controller is active, overshoots in the actual position occur. If this instance, you must decide whether this effect is admissible or acceptable for the application in question. Utilization of this function will require specialist control knowledge and measurements with a Servotrace.

If the relevant machine data are set incorrectly, the consequent instability may cause damage to the machine.

Background

The transfer function of the PI controller is as follows:

$$G_R(s) = K_R + \frac{1}{T_n} * s$$

Where:

K_R	Proportional component gain
T_n	Integrator time

Procedure

Optimize the position feedback loop as a P controller first using the tools described in the previous subsections.

Increase of the tolerances of the following machine data while measurements are being taken to determine the quality of the feedback loop with PI position controller:

MD 36020: POSITIONING_TIME
MD 36030: STANDSTILL_POS_TOL
MD 36040: STANDSTILL_DELAY_TIME
MD 36400: CONTOUR_TOL

Activate the position feedback loop as a PI controller by setting the following machine data:

MD 32220: POSCTRL_INTEGR_ENABLE ; Set value 1

MD 32210: POSCTRL_INTEGR_TIME ; Integrator time [s]

Effect of integrator time:

- $T_n \rightarrow 0$ The control error is corrected quickly, but the control loop can still become unstable.
- $T_n \rightarrow \infty$ The control error is corrected more slowly.

© Siemens AG, 2003. All rights reserved

Search for the right compromise for the application between these two extreme cases T_n. T_n must not be selected too close to the instability limit as the machine may sustain damage if the loop becomes unstable.

Use the servo trace (see /IAD/, Installation & StartUp Guide, Measurement of Position Feedback Loop and Trace Function) to record the approach to a target position using an automatic program for reciprocating motion.

Set the servo trace to display the following:

- Following error
- Actual velocity
- Actual position
- Set position

Set the tolerance values in MD 36020, 36030, 36040 and 36400 back to the required settings once you have found the optimum setting for T_n.

Example

Setting result after several iterative processes for K_R and T_n.

Each of the following quantities – following error, actual velocity, actual position and position setpoint – have been recorded by servo trace. On traversal in JOG mode, the characteristic of the individual quantities shown in the following Fig. was then drawn.
Machine data settings:
$MA_POSCTRL_INTEGR_ENABLE = 1
$MA_POSCTRL_INTEGR_TIME = 0.003
$MA_POSCTRL_GAIN[1] = 5.0
Parameter set selection 0

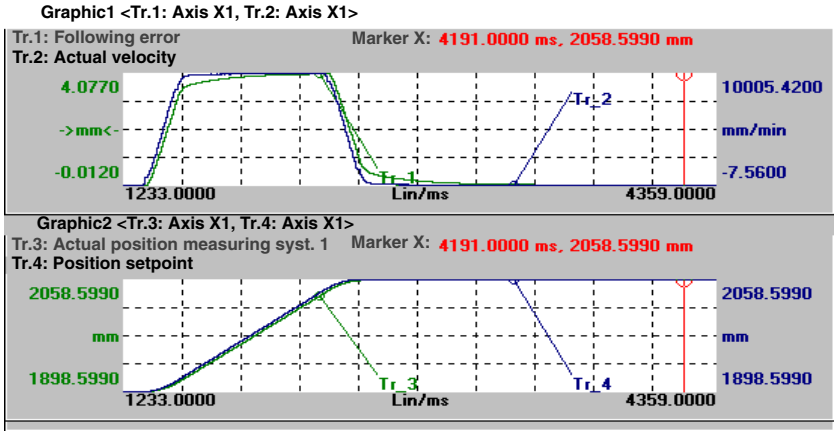


Fig. 2-16 Following error (1), act. veloc. (2), actual pos. (3), position setp. (4)

© Siemens AG, 2003. All rights reserved

2.5.5 System variable for the status of pulse enable (SW 5.1 and higher)

Application

For all applications that must quickly react to pulse enabling, the status of the pulse enable is imaged to a new system variable in order to accelerate the brake control.

This system variable is preferably evaluated in synchronized actions. Using the synchronized action, either a direct output to an NCK output can be carried out or a faster transfer to the PLC.

Functionality

Since 611D digital drives have no integrated brake control, the brakes are normally controlled from the PLC. The brake can be closed again by deleting the pulse enable in the PLC.

If the pulse enable is deleted due to external events (611D interface, terminal 663 to the PLC) or due to drive or axis errors, the PLC can close the brake only with a delay, since the transport of the pulse enable signal via servo and interpolator requires 2 ... 3 interpolator cycles. In the worst case, the PLC needs another two PLC cycles. With hanging axes and linear motors, this is often too slow.

System variable for enabling the drive power

Since the function must be available for all kinds of drives in the same form (also for non-electrical drives), the variable is given the name “Drive power enable”.

Description	NCK variables
Status of the power enable (pulse enable) of a machine axis. Defined range of values: FALSE: No power enable TRUE: Power enable exists	$VA_DPE[machine axis]

Activation

The variable is predefined and can be used at any time acc. to the data type BOOL with FALSE or TRUE. Until the communication with the drive is not yet established, the value 0 (FALSE) is supplied.

The synchronized actions in which the variable is evaluated should preferably be carried out already when the control powers up, e.g. by starting an ASUB from the PLC. To make sure that the synchronized actions remain continuously active and are not effected by the mode change, it is recommended to program them as a “static synchronized action” (using IDS as the vocabulary word).

Example

Output of the pulse enable of machine axis X1 to the first digital NCK output in all modes.

IDS = 1 DO $A_OUT[1] = $VA_DPE[X1]

© Siemens AG, 2003. All rights reserved

Supplementary conditions

With SW 5.1 and higher, the function extension is available for digital drives in all control variants providing synchronized actions.

Control response with POWER ON, RESET, REPOS etc.

After POWERON, the value 0 is supplied until the communication with the drive is not yet established. Then, the system variable $VA_DPE always specifies the pulse enable value acquired at the beginning of the interpolation cycle. The mode-independent evaluation or retransmission should be carried out using static synchronized actions (vocabulary word IDS).

2.5.6 Extensions for “deceleration axes” (SW 5.3 and higher)

Application

In the case of design-conditioned non-linearities and elasticities as it often occurs in the conveying and highbay racking technology, it is often necessary to sacrifice of the position control due to the unstable position control loop. The axes are therefore traversed closed-loop controlled and not open-loop controlled. To this aim, the WF723 module offers the special function of the “deceleration axes”.

Functionality

This “deceleration axes” function blocks any path movements, but allows positioning with approach to the target position by reducing the velocity step by step. The response corresponds to that of a multipoint controller.

For reasons relating to position feedback loop stability, the position controller algorithm is extended instead of the special positioning function “deceleration axes”. It is thus also possible to operate drives that might cause problems using the position control.

Benefits

Compared with the existing method “deceleration axes”, the extension of the position controller algorithm provides the following advantages:

- The structure of the interpolation and control is extended only slightly. This axis has thus automatically all properties of a normal axis.
- Traversing on the path also becomes possible and it can be considerably earlier foreseen which path, e.g. a highbay racking cage will take at different velocities. A possible collision can thus be avoided.
- The function can be configured considerably easier.
- In particular, with feedforward control enabled, the positioning processes are faster.

Low-pass filter for analog drives

A low-pass filter connected at the position controller output now also allows the position controller gain to be reduced quickly with increasing frequency for analog drives (axes). This suppresses the effect of strong natural frequencies in the form of settable speed setpoint filters.

This functionality is provided as standard for digital drives.

MD 32930
MD 32940

The low-pass filter is activated via
MD 32930: POSCTRL_OUT_FILTER_ENABLE = 1.

© Siemens AG, 2003. All rights reserved

The filter time constant is entered via
MD 32940: POSCTRL_OUT_FILTER_TIME

"Dead zone" MD 32960

Non-linearities close to zero speed, such as those which can occur when simple frequency converters are used, are inhibited by a "dead zone" in the controller. The threshold for the control deviation below which a zero speed setpoint is output can be set in machine data MD 32960: POSCTRL_ZERO_ZONE.

The "dead zone" is set in MD 32960: POSCTRL_ZERO_ZONE for each individual encoder.

Stability risk

The extended position controller algorithm will reduce the risk that a stable control admits only a rather poor setting of the gain or that the position control loop does not remain stable despite the effective extensions to a minimum.

Supplementary conditions

With SW 5.3 and higher, the function extension is available for all control variants.

The activation of the low-pass filter becomes only into effect with inactive dynamic stiffness control with MD 32640: STIFFNESS_CONTROL_EN–ABLE = 0.

With the active low-pass filter, the simulated following error in acceleration phases is greater than usually. It can therefore be necessary to increase MD 36400: CONTOUR_TOL compared with the default value to avoid a response of the axial contour monitoring (alarm 25050).

If the settable "dead zone" of the position controller remains in the default setting

- MD 32960: POSCTRL_ZERO_ZONE = 0,

then this corresponds to an input value whose size corresponds to the current encoder fine resolution.
The default setting is therefore compatible with previous software versions.

If the "dead zone"

- MD 32960: POSCTRL_ZERO_ZONE is configured to a value greater than the

zero speed tolerance MD 36030: STANDSTILL_POS_TOL,
then it is possible that the zero speed control responses on the completion of a positioning operation (alarm 25040).

In the case of a "dead zone"

- MD 32960: POSCTRL_ZERO_ZONE is configured to a value greater than the

exact stop limits MD 36000: STOP_LIMIT_COARSE,
no exact limit signals are output. This can result in a response of the positioning monitoring (alarm 25080) and block the block change.

■

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

None

4 Data Descriptions (MD, SD)

4.1 General machine data

10200 MD number	**INT_INCR_PER_MM** Calculation resolution for linear positions		
Default setting: 1000	Minimum input limit: 0.000001	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	With MD: INT_INCR_PER_MM the number of internal increments per millimeter is set. The precision of the linear position input is limited to the calculation resolution by rounding the product of the programmed value and the calculation resolution to an integer value. In order to keep the rounding easily reproducible, it is advisable to use powers of 10 for the calculation resolution.		
Application example(s)	The calculation resolution can be increased to > 1000 incr./mm for linear axes operating to high accuracy requirements.		

10210 MD number	**INT_INCR_PER_DEG** Calculation resolution for angular positions		
Default setting: 1000	Minimum input limit: 0.000001	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The number of internal increments per degree is defined in INT_INCR_PER_DEG. The precision of the angular position input is limited to the calculation resolution by rounding the product of the programmed value and the calculation resolution to an integer value. To make the rounding clear, powers of 10 should be used for the calculation resolution.		
Application example(s)	The calculation resolution can be changed to > 1000 incr./degrees for a high-resolution rotary axis.		

© Siemens AG, 2003. All rights reserved

10220 MD number	**SCALING_USER_DEF_MASK** Activation of scaling factors		
Default setting: 200	Minimum input limit: 0	Maximum input limit: 17FF	
Changes effective after POWER ON	Protection level: 2	Unit: Hex	
Data type: DWORD	Applies from SW 1.1		

Significance:

Machine and setting data with a physical quantity are interpreted in the predefined units listed below depending on the basic system (metric/inch)
If other input/output units are used for the individual physical quantities, the associated normalization factors are activated with this machine data (– entered in MD 10230: SCALING_FACTORS_USER_DEF[n]–).

Bit set:

Data of the associated physical quantity (see list) are normalized to the unit defined by MD: SCALING_FACTORS_USER_DEF[n] control.

Bit not set:

Data of the associated physical quantity are scaled to the predefined unit listed below:

Assigned physical quantity	**Predefined units for:** MD 10240: SCALING_SYSTEM_IS_METRIC	
Bit no.	1	0
(Given as hexadecimal value)		
0 Linear position	1 mm	1 inch
1 Angular position	1 degree	1 degree
2 Linear velocity	1 mm/min	1 inch/min
3 Angular velocity	1 rpm	1 rpm
4 Linear acceleration	1 m/s^2	$inch/s^2$
5 Angular acceleration	1 $rev./s^2$	1 $rev./s^2$
6 Linear jerk	1 m/s^3	$inch/s^3$
7 Angular jerk	1 $rev./s^3$	1 $rev./s^3$
8 Time	1 s	1 s
9 Position-controlled servo gain	1/s	1/s
10 Revolutional feedrate	1 mm/rev.	1 inch/rev.
11 Compensation value linear axis	1 mm	1 inch
12 Compensation value angular axis	1 degree	1 degree

Example:
SCALING_USER_DEF_MASK ='H3'; (bit no. 0 and 1 as hex value)
⇒ For linear and angular positions, the scaling factor specified in the relevant MD: SCALING_FACTORS_USER_DEF[n] is activated.

If this machine data is changed, a start-up is required because otherwise the relevant machine data in those physical units would be wrongly scaled.

Proceed as follows:

- MD change by manual input
  ⇒ First perform start-up and then enter the physical units in the related machine data.
- MD change via machine data file
  ⇒ First perform start-up and then reload the machine data file so that the new physical units are considered.

If the machine data are altered, alarm 4070 “Scaling machine data altered” is output.

Application example(s):
Input/output of linear velocities must be made in cm/min:
SCALING_USER_DEF_MASK = 'H4' (bit no. 2 as hex value)
SCALING_FACTORS_USER_DEF[2] = 0.1666666667 (10/60)

Related to ....:
MD 10230: SCALING_FACTORS_USER_DEF[n] (scaling factors of the physical quantities)

© Siemens AG, 2003. All rights reserved

10220 MD number	**SCALING_USER_DEF_MASK** Activation of scaling factors		
10230 MD number	**SCALING_FACTORS_USER_DEF[n]** Scaling factors of physical quantities		
Default setting: 1, 1, 1,1, 1, 1, 1, 1, 1, 1000/60, 1,1, 1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		

Significance:

The normalization factor of a physical quantity that has a unit other than the default unit setting (set bit in MD 10220: SCALING_FACTORS_USER_DEF_MASK) is entered in this MD. The factor must refer to the unit used internally for the physical quantity in question.

Index[n]	Assigned physical quantity	Internal unit
0	Linear position	1 mm
1	Angular position	1 degree
2	Linear velocity	1 mm/s
3	Angular velocity	1 degree/s
4	Linear acceleration	1 mm/s^2
5	Angular acceleration	1 $degree/s^2$
6	Linear jerk	1 mm/s^3
7	Angular jerk	1 $degree/s^3$
8	Time	1 s
9	Position-controlled servo gain	1/s
10	Revolutional feedrate	1 mm/degree
11	Compensation value linear position	1 mm
12	Compensation value angular position	1 degree

The scaling factor is assigned to the physical quantity using the index [0 ...12].
If this machine data is changed, a start-up is required because otherwise the relevant machine data in those physical units would be wrongly scaled.

Proceed as follows:

- MD change by manual input
  ⇒ Perform a start-up and then enter the associated machine data with the physical units.
- MD change via machine data file
  ⇒ First perform start-up and then reload the machine data file so that the new physical units are activated.

If the machine data are altered, alarm 4070 “Scaling machine data altered” is output.

Application example(s)

Input/output of angular velocities is to be in new degrees/min:
SCALING_FACTORS_USER_DEF_MASK = ’H8’; (bit no. 3 as hex value)
SCALING_FACTORS_USER_DEF[3] = 0.01851852; (400/360/60)
[3]: Index for angular velocity.

Related to ....

MD 10220: SCALING_USER_DEF_MASK (activation of scaling factors)

© Siemens AG, 2003. All rights reserved

10240 MD number	**SCALING_SYSTEM_IS_METRIC** Basic system metric		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	The MD specifies the basic system used by the control for scaling length-dependent physical variables for data input/output. All relevant data are stored internally in the basic units 1 mm, 1 degree and 1 sec. When accessing a parts program via the operator panel or from an external device scaling is in the following units: SCALING_SYSTEM_IS_METRIC = 1: scaled in: mm, mm/min, m/s^2, m/s^3 mm/rev. SCALING_SYSTEM_IS_METRIC = 0: scaled in: inch, inch/min, $inch/s^2$ $inch/s^3$ inch/rev. The basic system selected also determines the way a programmed F value for linear axes is interpreted: G94 mm/min inch/min G95 mm/rev. inch/rev. If this machine data is changed, a start-up is required because otherwise the relevant machine data in those physical units would be wrongly scaled. Proceed as follows: • MD change by manual input ⇒ Perform a start-up and then enter the associated machine data with the physical units. • MD change via machine data file ⇒ First perform start-up and then reload the machine data file so that the new physical units are activated. If the machine data are altered, alarm 4070 “Scaling machine data altered” is output.		
Application example(s)	Installation in the metric system and then conversion to inch system.		
Special cases, errors, ...	The factor for converting 1 mm to 1 inch can be altered by setting MD 10250: SCALING_VALUE_INCH.		

© Siemens AG, 2003. All rights reserved

10250 MD number	**SCALING_VALUE_INCH** Conversion factor for switching over to INCH system		
Default setting: 25.4	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 1	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The conversion factor for metric to inch system for data input/output is entered in this MD. This factor is active only for the selection of the non-metric basic system (MD 10240: SCALING_SYSTEM_IS_METRIC = 0) for the following conversions: – Programmed F values for linear axes – Input/output of lengths and linear data (e.g. machine data, zero offsets) Programmed geometry axis positions are converted with this factor if the dimension system programmed with G70/G71 differs from the selected basic system (SCALING_SYSTEM_IS_METRIC). Programmed synchronous axis positions are converted via the appropriate axial factors (MD 31200: SCALING_FACTOR_G70_G71) if the dimension system programmed with G70/G71 differs from the selected basic system (SCALING_SYSTEM_IS_METRIC). Settings that deviated from the default setting 25.4 should only be made in exceptional cases because correct display of the unit on the operator interface is only possible with this value. If this machine data is changed, a start-up is required because otherwise the relevant machine data in those physical units would be wrongly scaled. Proceed as follows: • MD change by manual input ⇒ First perform start-up and then enter the physical units in the related machine data. • MD change via machine data file ⇒ First perform start-up and then reload the machine data file so that the new physical units are activated. If the machine data are altered, alarm 4070 “Scaling machine data altered” is output.		
Application example(s)	This conversion factor is used if the dimension system is changed from metric to inch or to a customer-specific dimension system after installation. All machine data etc. that are entered are then converted with this factor. The converted value is then output at the next readout or on the operator interface.		
Related to	MD 10240: SCALING_SYSTEM_IS_METRIC		

© Siemens AG, 2003. All rights reserved

10260 MD number	**CONVERT_SCALING_SYSTEM** Basic system switchover active		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 5		
Significance:	This machine data defines the handling of MD 10240: SCALING_SYSTEM_IS_METRIC. MD 10260=0: Inch/metric response compatible with SW 1 to SW 4.4 MD 10260=1: Inch/metric response for SW 5 and higher Inch/metric functionality for SW 5 and higher: • Availability of MMC soft key • Automatic conversion of NC data on an inch/metric switchover • INCH/METRIC instruction in the data backup • The scope of MD 10240: SCALING_SYSTEM_IS_METRIC is RESET • Functionality of MD 32711: CEC_SCALING_SYSTEM_METRIC (sag compensation)		
Application example(s)	This machine data is used to configure the availability of the new inch/metric functionality (SW 5 and higher).		
Related to	MD 10240: SCALING_SYSTEM_IS_METRIC		

10290 MD number	**CC_TDA_PARAM_UNIT** Physical units of the tool data for compile cycles		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 16	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 5		
Significance:	Physical units for the user-defined tool-specific data: 0, // No unit 1, // Linear position [mm ; inch] 2, // Angular position [degrees ; degrees] 3, // Linear velocity [mm/min ; inch/min] 4, // Angular velocity [rev/min ; rev/min] 5, // Linear acceleration [m/s^2 ; $inch/s^2$] 6, // Angular acceleration [rev/s^2 ; rev/s^2] 7, // Linear jerk [m/s^3 ; $inch/s^3$] 8, // Angular jerk [rev/s^3 ; rev/s^3] 9, // Revolutional feedrate [mm/rev ; inch/rev]		
Related to	MD 18094: MM_NUM_CC_TDA_PARAM		

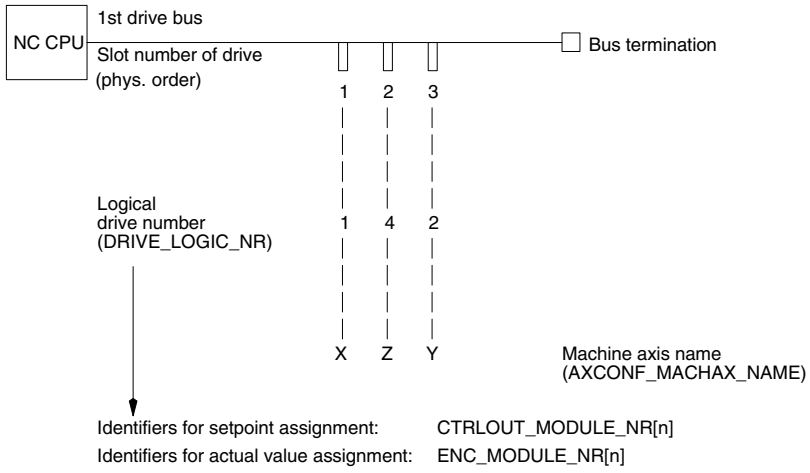
10292 MD number	**CC_TOA_PARAM_UNIT** Physical units of the cutting edge data for compile cycles		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 16	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 5		
Significance:	Physical units for the user-defined tool edge data: 0, // No unit 1, // Linear position [mm ; inch] 2, // Angular position [degrees ; degrees] 3, // Linear velocity [mm/min ; inch/min] 4, // Angular velocity [rev/min ; rev/min] 5, // Linear acceleration [m/s^2 ; $inch/s^2$] 6, // Angular acceleration [rev/s^2 ; rev/s^2] 7, // Linear jerk [m/s^3 ; $inch/s^3$] 8, // Angular jerk [rev/s^3 ; rev/s^3] 9, // Revolutional feedrate [mm/rev ; inch/rev]		
Related to	MD 18096: MM_NUM_CC_TOA_PARAM		

© Siemens AG, 2003. All rights reserved

13000 MD number	**DRIVE_IS_ACTIVE[n]** Drive activation (SIMODRIVE 611D)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Only applies to drives with SIMODRIVE 611D! A drive is activated/deactivated with this MD. 1: Drive is active 0: Drive is **not** active. The drive is not supplied with setpoints and is not monitored. No input (actual value) or output (setpoint) on this module can be configured. This module is only taken into account during basic initialization of the drive bus. MD 13020: DRIVE_INVERTER_CODE, MD 13010: DRIVE_LOGIC_NR, MD 13030: DRIVE_MODULE_TYPE and MD 13040: DRIVE_TYPE must be parameterized. The slot number of a real drive [slot no.] must be used as the index [n]. The index is numbered starting with “0” at the beginning of the drive bus (≙ 1st real drive available) and continues in ascending order to the end. 810D is assigned the slot nos. 0–5.		
Application example(s)	Deactivating an existing drive for test purposes.		

13010 MD number	**DRIVE_LOGIC_NR[n]** Logical drive number		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 15 (810D) Maximum input limit: 31 (840D)	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Only applies to drives with SIMODRIVE 611D! A “logical drive number” must be assigned to every real drive. This parameterization defines how the actual order of the modules connected to one drive bus relate to the logical drive numbers. A non-existent drive module is assigned the logical drive number “0”. In this way it is possible to create positions for modules that do not yet exist but are to be inserted at this point in the drive bus at a later time. The logical drive number is used in assigning actual values and setpoints (MD 30220: ENC_MODULE_NR[n], MD 30110: CTRLOUT_MODULE_NR[n]). The slot number of a real drive [slot no.] must be used as the index [n]. The index is numbered starting with “0” at the beginning of the drive bus (≙ 1st real drive available) and continues in ascending order to the end. 810D is assigned the slot nos. 0–5. Each drive must only be assigned one logical drive number. The following alarms may be output if an error occurs: – Alarm 300001 “Drive number not possible” – Alarm 300002 “Drive number assigned several times” – Alarm 300005 “At least 1 module too many at drive bus” – Alarm 300006 “At least 1 module too few at drive bus”		

© Siemens AG, 2003. All rights reserved

13010 MD number	**DRIVE_LOGIC_NR[n]** Logical drive number
Application example(s)	Example of routing:  Related machine data: DRIVE_LOGIC_NR[0] = 1 DRIVE_LOGIC_NR[1] = 4 DRIVE_LOGIC_NR[2] = 2 DRIVE_LOGIC_NR[3] = 0 DRIVE_LOGIC_NR[4] = 0 DRIVE_LOGIC_NR[5] = 0 . . . ENC_MODULE_NR[0,X] = 1 (1st encoder) ENC_MODULE_NR[0,Y] = 2 ENC_MODULE_NR[0,Z] = 4 CTRLOUT_MODULE_NR[0, X] = 1 (Speed setpoint) CTRLOUT_MODULE_NR[0, Y] = 2 CTRLOUT_MODULE_NR[0, Z] = 4 Note: MD 30220: ENC_MODULE_NR[n] and MD 30110: CTRLOUT_MODULE_NR[n] of **one** machine axis have to have the same logical drive number.

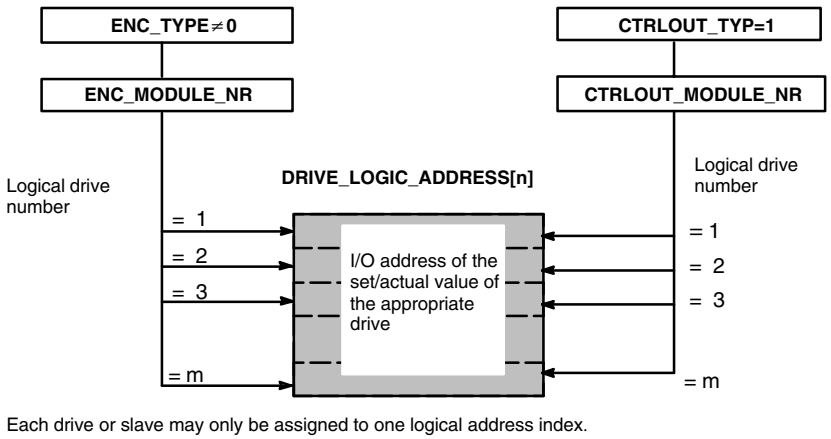
© Siemens AG, 2003. All rights reserved

13020 MD number	DRIVE_INVERTER_CODE[n] Power section code of drive module		
Default setting: 0	Minimum input limit: –	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2	Unit: Hex	
Data type: DWORD	Applies from SW 1.1		
Significance:	Only applies to drives with SIMODRIVE 611 digital or SINUMERIK 810D! The power section code of the MSD or FDD module being used is entered (as hex value) in the MD. **Power section code for FDD modules:** Code number (hex) / Power section order no. / Current 00 / no power section specified / ——— 11 / 6SN11301Dx1x0HA0 / 3/6 A 12 / 6SN11301Dx1x0AA0 / 5/10 A 13 / 6FC54471Dx1x0AA0 / 6/12 A (810D) 14 / 6SN11301Dx1x0BA0 / 9/18 A 16 / 6SN11301Dx1x0CA0 / 18/36 A 17 / 6SN11301Dx1x0DA0 / 28/56 A 19 / 6SN11301Dx1x0EA0 / 56/112 A 1A / 6SN11301Dx1x0FA0 / 70/140 A 1E / 6FC54471Dx1x0AA0 / 18/36 A (810D) **Power section code for MSD modules:** Code number (hex) / Power section order no. / Current 06 / 6SN11351DA1x0CA0 / 24/32/36 A 07 / 6SN11351DA1x0DA0 / 30/40/56 A 08 / 6SN11351DA1x0GA0 / 45/60/80 A 09 / 6SN11351DA1x0EA0 / 60/80/106 A 0A / 6SN11351DA1x0FA0 / 85/110/140 A 0E / 6FC54471DA1x0AA0 / 24/32 A (810D) The slot number [slot no.] of a real drive must be used as the index [n]. The index is numbered starting with “0” at the beginning of the drive bus (≙ 1st real drive available) and continues in ascending order to the end.		

13030 MD number	DRIVE_MODULE_TYPE[n] Module identifier		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 10	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Only applies to drives with SIMODRIVE 611 digital! An identifier for the module type used is entered in this MD. 0: No module or module has been removed (SW 6.3 and higher) 1: 1 axis module 2: 2 axis module 6: SINUMERIK 810D 9: Terminal block for digital/analog inputs/outputs 10: Bit bus interface The slot number [slot no.] of a real drive must be used as the index [n]. The index is numbered starting with “0” at the beginning of the drive bus (≙ 1st real drive available) and continues in ascending order to the end. If the wrong module type is assigned to an axis module, alarm 300003 “Wrong module type (1, 2 axis)” is output.		

© Siemens AG, 2003. All rights reserved

13040 MD number	**DRIVE_TYPE[n]** Identifier of drive type		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Only applies to drives with SIMODRIVE 611 digital! A drive type identifier must be entered in this MD. Drive type: 0 = No drive (possible I/O module) 1 = FDD drive 2 = MSD drive 3 = Linear drive 4 = Analog drive (spec. hydraulic) 5 = Hydraulic drive The slot number [slot no.] of a real drive must be used as the index [n]. The index is numbered starting with "0" at the beginning of the drive bus (≒ 1st real drive available) and continues in ascending order to the end. If the wrong drive type is programmed for a drive, alarm 300004, "Incorrect drive type (FDD, MSD, Linear drive, Hydraulic drive)" is generated.		
Related to	MD 13080: DRIVE_TYPE_DP[n] (drive type PROFIBUS DP)		

13050 MD number	**DRIVE_LOGIC_ADDRESS[n]** Logic address index	
Default setting: 272, 292, 312, 332, ...	Minimum input limit: 258	Maximum input limit: 1023
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: BYTE	Applies from SW 5.2	
Significance:	Only applicable to PROFIBUS DP! This MD represents the joining element to describe the PROFIBUS configuration PROFIBUS SDB. PROFIBUS SDB defines the logic I/O address of the drives connected to PROFIBUS. One address is assigned to a drive or a slave. The address index is used for actual and setvalue assignment (MD 30220: ENC_MODULE_NR[n] and MD 30110: CTRLOUT_MODULE_NR[n]). **Note:** MD 30220: ENC_MODULE_NR[0] and MD 30110: CTRLOUT_MODULE_NR[0] on **one** machine axis are to be assigned to the same drive (I/O address). Index [n] of the machine data is coded as follows: [drive index]: n = 0 for drive 1, n = 1 for drive 2 etc...	
Fig.	 Each drive or slave may only be assigned to one logical address index.	

© Siemens AG, 2003. All rights reserved

13050 MD number	**DRIVE_LOGIC_ADDRESS[n]** Logic address index
Application example(s)	The value of this MD 13050: DRIVE_LOGIC_ADDRESS[n] is the address index, i.e. the logical I/O address of the drive which is assigned using HW Config (SIMATIC Manager S7). **Example:** MD 13050: DRIVE_LOGIC_ADDRESS[1] = 272 Basic address 272 is assigned to drive 1.

13060 MD number	**DRIVE_TELEGRAM_TYPE[n]** Drive message frame type for drives connected to PROFIBUS DP		
Default setting: 102	Minimum input limit: 0	Maximum input limit: $2^{31}-1$	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 5.2, types 5 and 6 in SW 6.3 and higher		
Significance:	Only applicable to PROFIBUS DP! The message frame type must be specified for each drive connected to PROFIBUS DP. Standard message frame type for PROFIBUS axes: 0 = No standard type, user-defined simulation 1 = standard type 1 2 = standard type 2 3 = standard type 3 4 = standard type 4 5 = standard type 5 6 = standard type 6 101 = 611 universal type 1 102 = 611 universal type 2 103 = 611 universal type 3 104 = 611 universal type 4 105 = 611 universal type 5 106 = 611 universal type 6 The index[n] of the machine data has the following code: [drive index]: n = 0 for drive 1, n = 1 for drive 2 etc...		
Application example(s)	Message frame type 102 for 611 universal with an encoder on the drive		

13070 MD number	**DRIVE_FUNCTION_MASK[n]** DP functions used		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 5.3, bits 4 to 8 in SW 6.3 and higher		
Significance:	Bit-coded mask for selecting expected NC scope of functionality for PROFIBUS DP axes. Significance of set bits for SIMORIVE 611 universal drive: Bit 0 = 1: Disable drive alarm image specific to 611 universal. Bit 1 = 1: Disable drive type detection specific to 611 universal. Bit 2 = 1: Disable parameter access encoder driver specific to 611 universal. Bit 3 = 1: Disable parameter access output driver specific to 166 universal. Bit 4 = 1: Activate external drive: DSC bits **SW 6.3 and higher** (CTW1.12/STA1.12) Bit 5 = 1: Disable drive parking specific to 611 universal (CTW2.7 / STA2.7) Bit 6 = 1: Disable travel to fixed stop specific to 611 universal (CTW2.8 / STA2.8) Bit 7 = 1: Disable motor switchover specific to 611 universal int. (STA2.9 to 2.11) Bit 8 = 1: Disable ramp block specific to 611 universal (CTW1.13) By configuring the new bits 4 to 8 provided in SW 6.3 and higher, you can adapt certain non-standardized PROFIBUS control and/or status bits for SIMODRIVE 611 universal included in the Profidrive profile. When set to their defaults, these bits 4 to 8 may have a different significance for external drives.		

© Siemens AG, 2003. All rights reserved

13080 MD number	**DRIVE_TYPE_DP[n]** Drive type PROFIBUS DP		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	Relevant for drives on PROFIBUS DP: The message frame type must be specified for each drive connected to PROFIBUS DP: Drive type: 0 = No drive or unknown drive type (default), treated internally like: 1 = FDD (SRM: Synchronous rotating drive) 2 = MSD (ARM: Asynchronous rotating drive) 3 = Linear drive **Note:** The drive type is generally entered automatically for Siemens drives as soon as these are started. In contrast, the value must be entered manually for non-Siemens drives (at least for linear drives) if they cannot be identified automatically.		
Related to	MD 13040: DRIVE_TYPE[n] (drive type identification)		

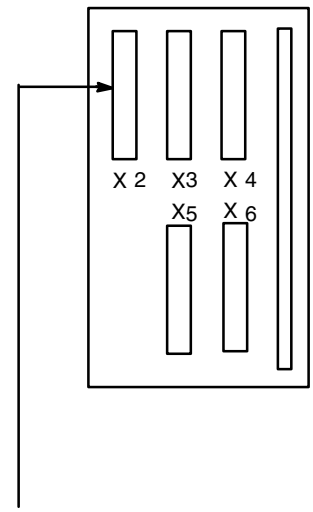
© Siemens AG, 2003. All rights reserved

4.2 Axis-specific machine data

30100 MD number	**CTRLOUT_SEGMENT_NR[n]** Setpoint assignment: Drive type		
Default setting: 0 for FMNC 1 for 840D/810D	Minimum input limit: 0	Maximum input limit: 6	
Changes effective after POWER ON	Protection level: 0	Unit: –	
Data type: Byte	Applies from SW 1.1, expanded to 6 with SW 6.4 and higher		
Significance:	The number of the bus segment via which the output is addressed is entered in this MD. The bus segments have a fixed assignment in the control systems. 0: Local bus (analogous, e.g. to SINUMERIK FMNC up to SW 5.3) 1: 611D bus (1st DCM e.g. drive bus for SINUMERIK 840/810D fixed up to NCU 572.3) 2: Local P bus (e.g. for FMNC with FM 354) 3: 611D bus (2nd DCM) 4: Reserved (virtual buses) 5: PROFIBUS DP (e.g. SINUMERIK 840Di) 6: PROFIBUS DP link module (e.g. SINUMERIK 840D, NCU 573.2 and higher) Index [n] of the machine data is coded as follows: [setpoint branch]: 0		

30110 MD number	**CTRLOUT_MODULE_NR[n]** Setpoint assignment: Drive number/module number		
Default setting: 1,2,3,4,5,6,7,8	Minimum input limit: 1	Maximum input limit: 15	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The number of the module within a bus segment via which the output is addressed is entered in this MD. – For SINUMERIK 840D/810D the logical drive number must be entered. (via MD 13010: DRIVE_LOGIC_NR[n]). – For SINUMERIK FMNC the modules (e.g. FMNC, FM position) are numbered consecutively starting with “1”. (counting from left to right) The index[n] of the machine data has the following code: [setpoint branch]: 0		

© Siemens AG, 2003. All rights reserved

30120 MD number	**CTRLOUT_NR[n]** Setpoint assignment: Setpoint output on drive module/module		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 4	
Changes effective after POWER ON	Protection level: 0	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	With FM-NC, the number of the output on the module, via which the setpoint output is addressed, must be entered in the MD: 1 to 4 according to selected output on socket X2. With SIN 840D/810D, the value is always 1. The index[n] of the machine data has the following code: [setpoint branch]: 0		
Fig.	FM-NC  CTRLOUT_NR[n] (for FM-NC = 1 ...4 depending on the output selected on socket X2)		

30130 MD number	**CTRLOUT_TYPE[n]** Output type of setpoint	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: BYTE	Applies from SW 1.1	
Significance:	The speed setpoint output type is entered in the MD: 0: Simulation (no HW required) 1: Standard (HW configuration differentiation) 2: Stepper motor (for SINUMERIK FM-NC only) 3: Stepper motor with direct setpoint output (SW 4) 4: No longer permissible as from SW 4!!! Virtual axis. (No HW exists for simulation) It can be interpolated in follow-up mode. (Technology of electronic transfer. Virtual and real master value.) Note: As from SW4 the MD 30132: IS_VIRTUAL_AX should be used. The index[n] of the machine data has the following code: [setpoint branch]: 0	
Application example(s)	Simulation: Machine functions can be simulated even if there is no drive system.	

© Siemens AG, 2003. All rights reserved

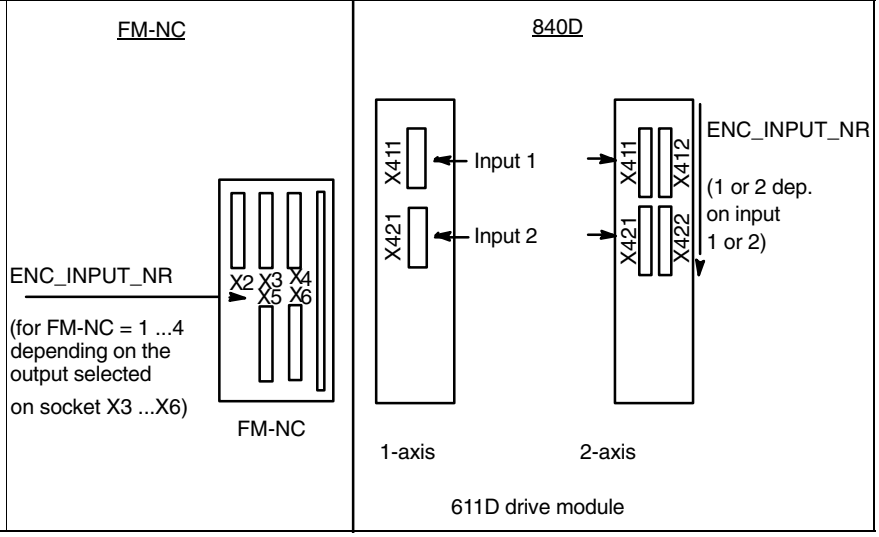
30134 MD number	**IS_UNIPOLAR_OUTPUT[n]** Setpoint output is unipolar		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 5.3		
Significance:	Unipolar output drive (for unipolar, analog drive actuator): All speed setpoints to the drive are positive, the speed setpoint sign is output separately in a special digital control signal. Input value: 0: Bipolar output with pos./neg. speed setpoint (normal application) 1: 0. Digital bit = servo enable 1. Digital bit = neg. direction of travel 2: (gating of enabling and travel direction signals): 0. Digital bit = servo enable in pos. direction of travel 1. Digital bit = servo enable in neg. direction of travel		

30200 MD number	**NUM_ENCS** Number of encoders	
Default setting: 1	Minimum input limit: 0	Maximum input limit: 2 (FMNC: 1)
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: BYTE	Applies from SW 1.1	
Significance:	The number of encoders for the axis or spindle for actual value acquisition is entered in this MD. If the actual position must be acquired with a direct measuring system: NUM_ENCS = 2 NUM_ENCS = 1 must be parameterized for referencing simulation axes/spindles.	

30210 MD number	**ENC_SEGMENT_NR[n]** Actual value assignment: Drive type	
Default setting: 0 for FMNC 1 for 840D/810D	Minimum input limit: 0	Maximum input limit: 6
Changes effective after POWER ON	Protection level: 0	Unit: –
Data type: BYTE	Applies from SW 1.1, expanded to 6 with SW 6.4 and higher	
Significance:	Number of the bus segment via which the encoder is addressed. The bus segments are permanently assigned in the control systems SINUMERIK FMNC and SINUMERIK 840D/810D. 0: Local bus (analogous, e.g. to SINUMERIK FM-NC up to SW 5.3) 1: 611D bus (1st DCM e.g. drive bus for SINUMERIK 840/810D fixed up to NCU 572.3) 2: Local P bus (e.g. for FM-NC with FM 354) 3: 611D bus (2nd DCM) 4: Reserved (virtual buses) 5: PROFIBUS DP (e.g. SINUMERIK 840Di) 6: PROFIBUS DP link module (e.g. SINUMERIK 840D, NCU 573.2 and higher) No. 2 and 3 are reserved for expansion purposes. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1	

© Siemens AG, 2003. All rights reserved

30220 MD number	**ENC_MODULE_NR[n]** Actual value assignment: Drive module number/control loop number		
Default setting: 1, 2, 3, 4, 5, 6, 7, 8	Minimum input limit: 1	Maximum input limit: 15	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Number of the module in the bus segment (MD: ENC_SEGMENT_NR[n]) via which the encoder is addressed. – For SINUMERIK 840D/810D the logical drive number must be entered (via MD: DRIVE_LOGIC_NR[n]). – For SINUMERIK FM-NC the modules (e.g. FMNC, FM position) are numbered consecutively starting with "1" (counting from left to right). The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1		
Related to	MD 30110: CTRLOUT_MODULE_NR[n] (Setpoint assignment: Drive number/module number)		

30230 MD number	**ENC_MODULE_NR[n]** Actual value assignment: Input on drive module/control loop module		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The number of the input on a module via which the encoder is addressed is entered here. This defines the input used for the actual value acquisition: – For SINUMERIK 840D = 1 or 2 (counting from top to bottom). – For SINUMERIK FMNC =1–4 depending on the input socket X3–X6 selected. – For SINUMERIK 810D measuring channels: 1 The index[n] of the machine data has the following code: [encoder no.]: 0 or 1 If an output to which no encoder is connected is selected, alarm 300008 "Control loop does not exist on drive" is triggered. (applies to SINUMERIK 840D/810D only).		
Fig.			

© Siemens AG, 2003. All rights reserved

30240 MD number	**ENC_TYPE[n]** Type of actual value acquisition (position actual value)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 5	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The encoder type used is entered in the MD: 0: Simulation 1: Raw signal encoder, (SINUMERIK 840D/810D) 2. Rectangular signal encoder, high-resolution standard (quadruplication of the increments), (for SINUMERIK FM-NC) 3: Encoder for stepper motors (for SINUMERIK FM-NC) 4: Absolute encoder with EnDat interface 5: Absolute encoder with SSI interface (SINUMERIK FM–NC) The index[n] of the machine data has the following code: [encoder no.]: 0 or 1 If the wrong encoder type is defined, alarm 300009, “Wrong control loop type drive [number], control loop [number]” is output.		
Application example(s)	Simulation: Machine functions can be simulated even if there is no measuring system.		

30242 MD number	**ENC_IS_INDEPENDENT[n]** Encoder is independent		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 4.1		
Significance:	If actual value corrections performed by the NC on the encoder selected for position control are not to influence the actual value of any other encoder defined in the same axis, then the position control encoder must be declared to be “independent”. ENC_IS_INDEPENDENT = 0: Encoder is dependent ENC_IS_INDEPENDENT = 1: Encoder is independent Actual value corrections include the following: Modulo treatment, reference point approach, measuring system calibration, PRESET Extension as of SW 5: ENC_IS_INDEPENDENT = 2: The passive encoder is dependent. The encoder actual value is modified by the active encoder. In combination with MD35102: REFP_SYNC_ENCS = 1 the passive encoder is adjusted to the active encoder during reference point approach, but not referenced. In reference mode MD 34200: ENC_REFP_MODE = 3 (distance-coded reference markers), the passive encoder is automatically referenced to the next traversing movement after it passes the zero marker distance. This happens independently of the current mode set. ENC_IS_INDEPENDENT = 3: Unlike totally independent encoders (value=1), independent encoders are permitted to use modulo correction.		
Application example(s)	MD 30200: NUM_ENCS[AX1] = 2 MD 30242: ENC_IS_INDEPENDENT[0, AX1] = 0 MD 30242: ENC_IS_INDEPENDENT[1, AX1] = 1 If the first encoder is selected for position control via the VDI interface, then the actual value corrections listed above are performed only on this encoder. If the second encoder is selected for position control via the VDI interface, then the actual value corrections listed above are performed on both encoders. This machine data therefore applies only to the encoder that is not currently selected for position control via the VDI interface!		

© Siemens AG, 2003. All rights reserved

30350 MD number	**SIMU_AX_VDI_OUTPUT** Output of axis signals for simulation axes		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	This machine data defines whether axis-specific interface signals are output to the PLC during simulation of an axis. 1: The axis-specific IS signals of a simulated axis are output to the PLC. In this way the PLC program can be tested without the drives. 0: The axis-specific IS signals of a simulated axis are not output to the PLC. All axis-specific IS signals are reset.		
MD irrelevant for	MD 30130: CTRLOUT_TYPE (output type of setpoint value) = 1		
Application example(s)	MD: SIMU_AX_VDI_OUTPUT = 0 PLC user program: . . U DBX61.5 ; (position controller active) U DBX61.6 ; (speed controller active) = A62.0 ; (brake released) . . This prevents the brake from being released while an axis is being simulated.		

31000 MD number	**ENC_IS_LINEAR[n]** Direct measuring system (linear scale)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	MD = 1: Encoder for position actual value acquisition is linear (linear scale). MD = 0: Encoder for position actual value acquisition is rotary. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1		
References	Subsection 2.2.3		

31010 MD number	**ENC_GRID_POINT_DIST[n]** Distance between reference marks on linear scales		
Default setting: 0.01	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: mm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The distance between the reference marks on the linear scale is entered in this MD. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1		
References	Subsection 2.2.3		

© Siemens AG, 2003. All rights reserved

31020 MD number	**ENC_RESOL[n]** Encoder pulses per revolution		
Default setting: 2048	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	The number of encoder pulses per revolution are entered in this MD. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1		
References	Subsection 2.2.3		

31030 MD number	**LEADSCREW_PITCH** Leadscrew pitch		
Default setting: 10	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: mm/rev.	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The pitch of the leadscrew is entered in this MD.		
References	Subsection 2.2.3		

31040 MD number	**ENC_IS_DIRECT[n]** Encoder is connected directly to the machine		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	MD = 1: Encoder for actual value acquisition is connected directly to the machine. MD = 0: Encoder for actual value acquisition is connected to the motor. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1		
Special cases, errors, ...	Entering the wrong information can cause an incorrect encoder resolution, because, for example, the wrong gear ratios are used for calculation.		
References	Subsection 2.2.3		

31050 MD number	**DRIVE_AX_RATIO_DENOM[n]** Denominator load gearbox		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 2 147 000 000	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	The load gearbox denominator is entered in this MD. The index[n] of the machine data has the following code: [closedloop control set of parameters set no.]: 0–5		
References	Subsection 2.2.3		

© Siemens AG, 2003. All rights reserved

31060 MD number	**DRIVE_AX_RATIO_NUMERA[n]** Numerator load gearbox		
Default setting: 1	Minimum input limit: –2 147 000 000	Maximum input limit: 2 147 000 000	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	The load gearbox numerator is entered in this MD. The index[n] of the machine data has the following code: [closedloop control set of parameters set no.]: 0–5 The MD is an element of the servo parameter set. It is not possible to enter the value 0. In this case, alarm 17095 is output.		
References	Subsection 2.2.3		

31070 MD number	**DRIVE_ENC_RATIO_DENOM[n]** Denominator measuring gearbox		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 2147000000	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	The measuring gearbox denominator is entered in this MD. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1 SW 5 and higher: By changing the default value 1:1, MD acts on a different value even when linear measuring systems are used.		
References	Subsection 2.2.3		

31080 MD number	**DRIVE_ENC_RATIO_NUMERA[n]** Numerator measuring gearbox		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 2147000000	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	The measuring gearbox numerator is entered in this MD. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1 SW 5 and higher: By changing the default value 1:1, MD acts on a different value even when linear measuring systems are used.		
References	Subsection 2.2.3		

© Siemens AG, 2003. All rights reserved

31200 MD number	**SCALING_FACTOR_G70_G71** Factor for converting values when G70/G71 is active		
Default setting: 25.4	Minimum input limit: –	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The conversion factor for inch/metric conversion by which the programmed geometry of an axis (position, polynomial coefficients, radius for circular programming, ...) is multiplied when the programmed value for G code group G70/G71 differs from the initial setting value (set in MD: GCODE_RESET_VALUES[n]) is entered in this MD. The factor can be set for each axis individually so that pure positioning axes are not dependent on G70/G71. It is not expedient to select the factor within three geometry axes. The data influenced by G70/G71 are described in the Programming Guide.		
Application example(s)	Basic system metric: As soon as G71 (inch) has been programmed in the parts program and is active all subsequent positioning values are divided by the factor entered in this MD and then converted to the internal metric system. Basic system inch: As soon as G70 (metric) has been programmed in the part program and is active all subsequent positioning values are divided by the factor entered in this MD and then converted to the internal inch system.		
Related to	MD: GCODE_RESET_VALUES[n] (G group initial setting).		

32000 MD number	**MAX_AX_VELO** Maximum axis velocity		
Default setting: 10000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2	Unit: mm/min, rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The limit velocity to which the axis can accelerate (rapid traverse limitation) is entered in this MD. This velocity is used if rapid traverse is programmed. The maximum linear and rotary axis speed is entered in this MD depending on the setting in MD: IS_ROT_AX. The maximum permissible axis velocity depends on the machine and drive dynamics and the limit frequency of the actual value acquisition.		

32100 MD number	**AX_MOTION_DIR** Traversing direction		
Default setting: 1	Minimum input limit: –1	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The direction of traverse of the machine can be reversed with this MD. Feedback polarity is not reversed, i.e. closed-loop control remains stable. MD = +1: Normal direction MD = –1: Direction reversed MD = 0: Normal direction		

© Siemens AG, 2003. All rights reserved

32110 MD number	**ENC_FEEDBACK_POL[n]** Sign actual value (feedback polarity)		
Default setting: 1	Minimum input limit: – 1	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The direction of evaluation of the rotary encoder signals is entered in this MD. –1: Direction reversed 0, 1: Direction not reversed If the direction is reversed, the control direction is also reversed if the encoder is used for position control. If input 1 is used for SINUMERIK 840D/810D. ⇒ MD: ENC_FEEDBACK_POL[n] always 1. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1		
Special cases, errors, ...	If the wrong control direction is entered the axis can be crashed. Depending on the limit values set, the following alarms will be triggered: Alarm 25040 “Zero point monitoring” Alarm 25050 “Contour monitoring” Alarm 25060 “Speed setpoint limitation” The associated limit values are described in: **References:** /FB/, A3, “Axis Monitoring Functions” The control direction selected might be incorrect if an uncontrolled setpoint jump occurs when a drive is connected.		

© Siemens AG, 2003. All rights reserved

32200 MD number	**POSCTRL_GAIN[n]** Servo gain factor		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 2000	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: 1/s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Position controller gain, or servo gain factor. The input/output unit for the user is [(m/min)/mm]. I.e. POSCTRL_GAIN[n] = 1 corresponds to 1 mm following error at V = 1m/min. The following machine data have default settings for adapting the standard selected input/ output unit to the internal unit [rev/s]. • MD 10230: SCALING_FACTORS_USER_DEF[9] = 16.666667 • MD 10220: SCALING_USER_DEF_MASK = 'H200'; (bit no 9 as hex value). If the value "0" is entered the position controller is opened. When entering the servo gain factor it is important to check that the gain factor of the whole position control loop is still dependent on other parameters of the controlled system. A distinction should be made between a "desired servo gain factor" acceleration during oscillation (MD: (MD: POSCTRL_GAIN) and the "actual servo gain" (produced by the machine). Only when all the parameters of the control loop are matched will these servo gain factors be the same. These factors are: – Speed setpoint adjustment (MD 32260: RATED_VELO MD 32250: RATED_OUTVAL) – Tacho compensation at speed encoder – Tacho generator on drive **Note:** Axes which interpolate with one another must either have the same gain setting (e.g. at the same velocity and an identical following error = 45° slope) or they must be matched via MD 32910: DYN_MATCH_TIME. The actual servo gain factor can be checked by means of the following error (in the service display). The drift compensation must be checked first (in the case of SINUMERIK FM–NC). The index[n] of the machine data has the following code: [closedloop control set of parameters set no.]: 0–5		

32210 MD number	**POSCTRL_INTEGR_TIME** Integrator time position controller		
Default setting: 1.0	Minimum input limit: 0.001	Maximum input limit: 10.0	
Changes effective after NEW_CONF	Protection level: 2 / 7	Unit: s	
Data type: DOUBLE	Applies from SW 6.4		
Significance:	Position controller integrator time for integral component in seconds This MD is active only if MD 32220: POSCTRL_INTEGR_ENABLE = 1 is set.		
MD irrelevant for	Position controller without integral component		
Related to	MD 32220: POSCTRL_INTEGR_ENABLE MD 32200: POSCTRL_GAIN		
References	/IAD/, Installation and StartUp Guide		

© Siemens AG, 2003. All rights reserved

32220 MD number	**POSCTRL_INTEGR_ENABLE** Activation of integral component of position controller		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.4		
Significance:	Activation of position controller integral component which converts P controller to PI controller Position overshoots may occur when I component is programmed. This function should be used in special cases only. It requires specialist control knowledge.		
Related to	MD 32210: POSCTRL_INTEGR_TIME MD 32200: POSCTRL_GAIN		
References	/IAD/, Installation and StartUp Guide		

32250 MD number	**RATED_OUTVAL[n]** Rated output voltage		
Default setting: 100	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 0	Unit: %	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Only applies to analog drives and PROFIBUS drives! The speed setpoint as a percentage of the max. speed setpoint at which the motor speed defined in MD: RATED_VELO[n] is reached is entered in this MD.		
Application example(s)	**1st Example:** At a voltage of 5V, the drive reaches a speed of 1875 rev/min. ⇒ RATED_OUTVAL = 50 %, RATED_VELO = 1875 [U/min] **2nd example:** At a voltage of 8V, the drive reaches a speed of 3000 rev/min. ⇒ RATED_OUTVAL = 80 %, RATED_VELO = 3000 [U/min] **3rd Example:** At a voltage of 1.5V, the drive reaches a speed of 562.5 rev/min. ⇒ RATED_OUTVAL = 15 %, RATED_VELO = 562.5 [U/min] All three examples are possible with one and the same drive/converter. The ratio of the two values is decisive and this is the same in all three examples. The index[n] of the machine data has the following code: [setpoint branch]: 0		
Related to	MD: RATED_OUTVAL[n] is only sensible in combination with MD: RATED_VELO[n].		

32260 MD number	**RATED_VELO[n]** Rated motor speed		
Default setting: 3000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 0	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Applies only to analog drives and PROFIBUS drives! The speed of the drive (scaled on the drive side!) which is attained with the speed setpoint in percent entered in MD: RATED_OUTVAL[n] is entered in this MD. RATED_OUTVAL[n] is entered in this MD. The index[n] of the machine data has the following code: [setpoint branch]: 0		
Related to	MD: RATED_VELO[n] is only sensible in combination with MD:RATED_OUTVAL[n].		

© Siemens AG, 2003. All rights reserved

32900 MD number	**DYN_MATCH_ENABLE** Dynamic response adaptation		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after NEW_CONF	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Axes with different servo gain factors can be set to the same following error in MD: DYN_MATCH_TIME. 1: Dynamics matching is active. 0: Dynamics matching is not active.		
Application example(s)	See Section 2.3		
Related to	MD 32900: DYN_MATCH_TIME[n] (time constant for dynamic response sensing)		

32910 MD number	**DYN_MATCH_TIME[n]** Time constant for dynamic matching		
Default setting: 0.01	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The time constant of the dynamics matching of an axis is entered in this MD. The difference between the equivalent time constant of the “slowest” control loop and the axis concerned is entered as the time constant for the dynamics matching. The MD is only active if MD: DYN_MATCH_ENABLE = 1. The index[n] of the machine data has the following code: [closed-loop control set of parameters set no.]: 0–5		
Application example(s)	See Section 2.3		
Related to	MD 32900: DYN_MATCH_ENABLE (dynamics matching).		

32930 MD number	**POSCTRL_OUT_FILTER_ENABLE** Activation of the low pass filter at the position controller output		
Default setting: FALSE (0)	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 5.3		
Significance:	Activation of the low pass filter at the position controller output using MD 32930: POSCTRL_OUT_FILTER_ENABLE = 1 The low pass filter only comes into effect with the dynamic stiffness control disabled via MD 32640: STIFFNESS_CONTROL_ENABLE = 0.		
Related to	MD 32950: POSCTRL_DAMPING		

32940 MD number	**POSCTRL_OUT_FILTER_TIME** Time constant of the low pass filter at the position controller output		
Default setting: 0.0	Minimum input limit: 0.0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	The corresponding filter time constant of the low pass filter at the position controller output can be activated via MD 32930: POSCTRL_OUT_FILTER_ENABLE = 1.		
Related to	MD 32950: POSCTRL_DAMPING		

© Siemens AG, 2003. All rights reserved

32950 MD number	**POSCTRL_DAMPING** Factor for additional damping of the position control loop		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: %	
Data type: DOUBLE	Applies from SW 1.1 (in SW 5.1 and higher extended)		
Significance:	Factor for injection of the positional deviation (with SIMODRIVE_611D drives). An input value of 100% means: An additional torque acc. to the drive MD1725 is injected • with linear drives, a positional deviation of 1mm exists; • with rotary axes, a positional deviation of 360 degrees exists for the two encoders. • with linear drives (rot. drive), a positional deviation exists acc. to MD 31030: LEADSCREWPITCH (e.g. default 10mm). Depending on the machine configuration, the best setting value can be beyond the range –100% .. 100%, e.g. at 800%. It can be found out, e.g. by trying and testing step by step. The extension of function is available for all control variants with SW 5.1 and higher, which use SIMODRIVE_611D drives. Conditions for using the function: In the case of an input value = 0, the alarm 26016 is triggered for this MD if the axis concerned is not equipped with two encoders: MD 30200: NUM_ENCS = 2. In this case, one encoder (load encoder) must be connected directly, and the other (motor encoder) indirectly: MD 31040: ENC_IS_DIRECT.		
Related to	MD 32930: POSCTRL_OUT_FILTER_ENABLE (activation of low pass filter)		

32960 MD number	**POSCTRL_ZERO_ZONE[n]** Dead zone of the position controller	
Default setting: 0.0	Minimum input limit: 0.0	Maximum input limit: plus
Changes effective after NEW_CONF	Protection level: 2/7	Unit: mm, inches, degrees
Data type: DOUBLE	Applies from SW 5.3	
Significance:	If the amount of the controller deviation is less than $MA_POSCTRL_ZERO_ZONE, the controller output is set to zero. The default setting is MD 32960: POSCTRL_ZERO_ZONE = 0 and corresponds to the current encoder fine resolution (compatible with the software versions in which this machine data did not yet exist). Index n = 0 refers to the 1st encoder of the axis. Index n = 2 refers to the 2nd encoder of the axis.	
Related to	MD 32950: POSCTRL_DAMPING	

© Siemens AG, 2003. All rights reserved

33000 MD number	**FIPO_TYPE** Fine interpolator type		
Default setting: 2	Minimum input limit: 1	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The fine interpolator type is entered in the MD: 1: Differential FIPO 2: Cubic FIPO 3: Cubic FIPO, optimized for operation with feedforward control Computing time and contour quality increase as the type if interpolator increases. – Cubic FIPO is standard. – If no feedforward or position control loop are used, a differential FIPO will give a computing time saving with a slightly higher contour error. – If the position control cycle and interpolator cycle are identical, both fine interpolator types have the same effect. ⇒ No fine interpolation takes place.		

34320 MD number	**ENC_INVERS[n]** Linear measuring system is reversed		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Incremental measuring system – MD has no meaning Distancecoded measuring system: – MD = 1: The measuring system is reversed compared with the machine system. – MD = 0: The measuring system same direction as machine system. The index[n] of the machine data has the following code: [encoder no.]: 0 oder 1		
Application example(s)	See Description of Functions Reference Point Approach R1.		
References	Subsection 2.2.3		

36210 MD number	**CTRLOUT_LIMIT[n]** Maximum speed setpoint		
Default setting: 100 for FMNC 110 for 840D/810D	Minimum input limit: 0	Maximum input limit: 200	
Changes effective after NEW_CONF	Protection level: 0 / 0	Unit: %	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The maximum speed setpoint is defined as a percentage in this MD. 100 % corresponds to the maximum speed setpoint (10 V with analog interface, maximum speed on SIMODRIVE 611D (settable with 611D–MD 1401: MOTOR_MAX_SPEED)). The maximum speed setpoint is determined by any existing setpoint limits in the speed and current controller. If this limit is violated, an alarm is triggered ad the axis is stopped. A limit should be chosen for which the maximum speed (rapid traverse) can be reached and the relevant control reserves maintained. The index[n] of the machine data has the following code: [setpoint branch]: 0		
Application example(s)	Reducing the speed setpoint during installation.		

© Siemens AG, 2003. All rights reserved

36500 MD number	**ENC_CHANGE_TOL** Maximum tolerance for position actual value switchover		
Default setting: 0.1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The permissible deviation between the actual values of the two measuring systems is entered in the MD. This tolerance must not be violated when switching from one measuring system to the other for closed loop control in order to avoid too large compensatory processes. Otherwise error message 25100, Axis %1 “Measuring system switchover not possible” is generated and switchover between the two systems does not take place. **SW 5.3 and higher** This MD is used to manage large backlash compensation values. It ensures that the backlash is not switched through to the actual value all at once, but in n steps with an increment size as set in MD 36500: ENC_CHANGE_TOL. Calculation of the backlash thus takes n servo cycles. If it takes too long to complete computation of the backlash, zero speed monitoring alarms may be generated. The original method of injecting the backlash compensation value is used if MD 36500: ENC_CHANGE_TOL is set higher than MD 32450: BACKLASH.		
MD irrelevant for	This MD is irrelevant for MD 30200: NUM_ENCS = 0 or 1.		
Application example(s)	To avoid too large compensatory processes when switching measuring systems.		
Related to	MD 32450: BACKLASH		

36510 MD number	**ENC_DIFF_TOL** Measuring system synchronism tolerance		
Default setting: 0.0	Minimum input limit: 0.0	Maximum input limit: plus	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 4.2		
Significance:	Permissible deviation between the actual values of the two measuring systems. This tolerance setting must not be exceeded during cyclical comparison of the two measuring systems or an error message will be generated. The associated monitoring function is not active with an MD setting = 0, in cases where 2 measuring systems are not active/non-existent in the axis or if the axes themselves are not referenced (at least act. closed-loop control measuring system). If this machine data is exceeded in cyclic transmission, fault message 25105 “Axis %1 measuring systems misaligned will appear”.		

■

© Siemens AG, 2003. All rights reserved

5 Signal Descriptions

None

6 Example

None

7 Data Fields, Lists

7.1 Machine data

Number	Identifier	Name	Refer-ence
Operator-panel-specific($MM_ ...)			
9004	DISPLAY_RESOLUTION	Display resolution	A2
9010	SPIND_DISPLAY_RESOLUTION	Display resolution for spindles	A2
9011	DISPLAY_RESOLUTION_INCH	Display resolution for INCH system of measurement	A2
General ($MN_ ...)			
10000	AXCONF_MACHAX_NAME_TAB[n]	Machine axis name	K2
10050	SYSCLOCK_CYCLE_TIME	System basic cycle	G3

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Reference
10070	IPO_SYSCLOCK_TIME_RATIO	Factor for interpolator cycle	G3
10060	POSCTRL_SYSCLOCK_TIME_RATIO	Factor for position control cycle	G3
10080	SYSCLOCK_SAMPL_TIME_RATIO	Division factor of position control cycle for actual value acquisition	G3
10200	INT_INCR_PER_MM	Calculation resolution for linear positions	
10210	INT_INCR_PER_DEG	Calculation resolution for angular positions	
10220	SCALING_USER_DEF_MASK	Activation of scaling factors	
10230	SCALING_FACTORS_USER_DEF[n]	Scaling factors of physical quantities	
10240	SCALING_SYSTEM_IS_METRIC	Basic system metric	
10250	SCALING_VALUE_INCH	Conversion factor for switchover to inch system	
10260	CONVERT_SCALING_SYSTEM	Basic system switchover active	
10270	POS_TAB_SCALING_SYSTEM	System of measurement of position tables	T1
10290	CC_TDA_PARAM_UNIT	Physical units of the tool data for CC	
10292	CC_TOA_PARAM_UNIT	Physical units of the tool edge data for CC	
13000	DRIVE_IS_ACTIVE[n]	Drive activation	
13010	DRIVE_LOGIC_NR[n]	Logical drive number	
13020	DRIVE_INVERTER_CODE[n]	Power section code of drive module	
13030	DRIVE_MODULE_TYPE[n]	Module identifier	
13040	DRIVE_TYPE[n]	Identifier of drive type	
13050	DRIVE_LOGIC_ADDRESS[n]	Logical drive addresses (SW 5.2 and higher)	
13060	DRIVE_TELEGRAM_TYPE[n]	Standard message frame type for PROFIBUS DP	
13070	DRIVE_FUNCTION_MASK[n]	DP function used (SW 5.3 and higher)	
13080	DRIVE_TYPE_DP[n]	Drive type PROFIBUS DP (SW 6.3 and higher)	
Channel-specific ($MC_ ...)			
20150	GCODE_RESET_VALUES[n]	Reset G groups	K1
Axis-specific ($MA_ ...)			
30100	CTRLOUT_SEGMENT_NR[n]	Setpoint assignment: Drive type	
30110	CTRLOUT_MODULE_NR[n]	Setpoint assignment: Drive number/module number	
30120	CTRLOUT_NR[n]	Setpoint assignment: Setpoint output on drive module/module	
30130	CTRLOUT_TYPE[n]	Output type of setpoint	
30134	IS_UNIPOLAR_OUTPUT[n]	Setpoint output is unipolar (SW 5.3 and higher)	
30200	NUM_ENCS	Number of encoders	
30210	ENC_SEGMENT_NR[n]	Actual value assignment: Drive type	
30220	ENC_MODULE_NR[n]	Actual value assignment: Drive module no./control loop no.	
30230	ENC_MODULE_NR[n]	Actual value assignment: Input on drive module/control loop module	
30240	ENC_TYPE[n]	Type of actual value acquisition (position actual val.)	
30242	ENC_IS_INDEPENDENT	Encoder is independent	
30300	IS_ROT_AX	Rotary axis	R2
30350	SIMU_AX_VDI_OUTPUT	Output of axis signals for simulation axes	
31000	ENC_IS_LINEAR[n]	Direct measuring system (linear scale)	
31010	ENC_GRID_POINT_DIST[n]	Distance between reference marks on linear scales	
31020	ENC_RESOL[n]	Encoder pulses per revolution	
31030	LEADSCREW_PITCH	Leadscrew pitch	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Reference
31040	ENC_IS_DIRECT[n]	Encoder is connected directly to the machine	
31044	ENC_IS_DIRECT2	Encoder on intermediate gear (SW 6.4 and higher)	S1
31050	DRIVE_AX_RATIO_DENOM[n]	Denominator load gearbox	
31060	DRIVE_AX_RATIO_NUMERA[n]	Numerator load gearbox	
31064	DRIVE_AX_RATIO_DENOM	Denominator on inter. gear gearbox (SW 6.4 and higher)	S1
31066	DRIVE_AX_RATIO2_NUMERA	Numerator on intermediate gear (SW 6.4 and higher)	S1
31070	DRIVE_ENC_RATIO_DENOM[n]	Denominator measuring gearbox	
31080	DRIVE_ENC_RATIO_NUMERA[n]	Numerator measuring gearbox	
31090	JOG_INCR_WEIGHT	Weighting of increment for INC/handwheel	H1
31200	SCALING_FACTOR_G70_G71	Factor for convert. values when G70/G71 is active	
32000	MAX_AX_VELO	Maximum axis velocity	
32100	AX_MOTION_DIR	Traversing direction	
32110	ENC_FEEDBACK_POL[n]	Sign actual value (feedback polarity)	
32200	POSCTRL_GAIN [n]	Servo gain factor	
32210	POSCTRL_INTEGR_TIME	Integrator time position controller	
32220	POSCTRL_INTEGR_ENABLE	Activation of integral component of position controll.	
32250	RATED_OUTVAL[n]	Rated output voltage	
32260	RATED_VELO[n]	Rated motor speed	
32450	BACKLASH[n]	Backlash	K3
32500	FRICT_COMP_ENABLE	Friction compensation active	K3
32610	VELO_FFW_WEIGHT	Feedforward control factor for speed feedfor. control	K3
32620	FFW_MODE	Feedforward control mode	K3
32630	FFW_ACTIVATION_MODE	Activate feedforward control from program	K3
32650	AX_INERTIA	Moment of inertia for torque feedforward control	K3
32711	CEC_SCALING_SYSTEM_METRIC	System of measurement of sag compensation	K3
32800	EQUIV_CURRCTRL_TIME [n]	Equivalent time constant current control loop for feedforward control	K3
32810	EQUIV_SPEEDCTRL_TIME[n]	Equivalent time constant speed control loop for feedforward control	K3
32900	DYN_MATCH_ENABLE	Dynamics matching	
32910	DYN_MATCH_TIME [n]	Time constant for dynamic matching	
32930	POSCTRL_OUT_FILTER_ENABLE	Activation of the low pass filter at the position controller output (SW 5.3 and higher)	
32940	POSCTRL_OUT_FILTER_TIME	Time constant of the low pass filter at the position controller output (SW 5.3 and higher)	
32950	POSCTRL_DAMPING	Factor for additional damping of the position control loop (SW 5.3 and higher)	
32960	POSCTRL_ZERO_ZONE[n]	Dead zone of position controller (SW 5.3 and high.)	
33000	FIPO_TYPE	Fine interpolator type	
34320	ENC_INVERS[n]	Linear measuring system is reversed	
35100	SPIND_VELO_LIMIT	Maximum spindle speed	S1
36200	AX_VELO_LIMIT [n]	Threshold value for velocity monitoring	A3
36210	CTRLOUT_LIMIT[n]	Maximum speed setpoint	
36400	AX_JERK_ENABLE	Axial jerk limitation	B2
36410	AX_JERK_TIME	Time constant for axial jerk filter	B2
36500	ENC_CHANGE_TOL	Max. tolerance for position actual value switchover	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
36510	ENC_DIFF_TOL	Measuring system synchronism tolerance	
36700	ENC_COMP_ENABLE[n]	Interpolatory compensation	K3

7.2 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Auxiliary Function Output to PLC (H2)

1 Brief Description 1/H2/1-3

2 Detailed Description 1/H2/2-5

2.1 Programming of auxiliary functions 1/H2/2-5

2.2 Predefined auxiliary functions 1/H2/2-8
2.2.1 Output specifications for predefined auxiliary functions 1/H2/2-10
2.2.2 Associated auxiliary functions for M0, M1 (SW 6.3 and higher) ... 1/H2/2-16

2.3 User-defined auxiliary functions 1/H2/2-18
2.3.1 Output specifications for user-defined auxiliary functions 1/H2/2-18

2.4 Grouping of auxiliary functions 1/H2/2-21

2.5 Acknowledgment of auxiliary functions by the PLC 1/H2/2-23
2.5.1 Normal acknowledgment with delayed block change 1/H2/2-24
2.5.2 Highspeed acknowledgment with delayed block change 1/H2/2-25
2.5.3 High-speed acknowledgment without delayed block change 1/H2/2-27

2.6 M function with implicit preprocessing stop (SW 6.3 and higher) .. 1/H2/2-28

2.7 Block search response 1/H2/2-29

2.8 Response to overstore 1/H2/2-30

2.9 Applications of auxiliary functions 1/H2/2-31

3 Supplementary Conditions 1/H2/3-33

4 Data Descriptions (MD, SD) 1/H2/4-35

4.1 General machine data 1/H2/4-35

4.2 Channel-specific machine data 1/H2/4-37

5 Signal Descriptions 1/H2/5-47

5.1 Channel-specific signals 1/H2/5-47
5.1.1 Signals to channel 1/H2/5-47
5.1.2 Signals from channel 1/H2/5-47

5.2 Axis-specific signals 1/H2/5-51

6 Example 1/H2/6-53

7 Data Fields, Lists 1/H2/7-57

7.1 Interface signals 1/H2/7-57

7.2 Machine data 1/H2/7-60

© Siemens AG, 2003. All rights reserved

7.3 Alarms 1/H2/7-61

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

Auxiliary functions

For the purpose of workpiece machining operations, it is possible to program process-related functions (feedrate, spindle speed, gear stage) and functions for controlling additional devices on the machine tool (sleeve forward, gripper open, clamp chuck) in the parts program in addition to axis positions and interpolation methods. These process-related functions are programmed by means of auxiliary functions.

The following different types of auxiliary function are available:

- Miscellaneous function (M)
- Spindle function (S)
- Auxiliary function (H)
- Tool number (T)
- Tool offset D, DL (SW 5.2)
- Feedrate F.

These functions are activated and/or output to the PLC:

– at defined moments in time referred to the execution of parts program blocks or

– when programmed conditions are fulfilled.

Output of auxiliary functions to PLC

Through auxiliary function outputs in the parts program, auxiliary function parameters are transferred to the PLC interface. Here, they are made available to the PLC user program through facilities belonging to the basic PLC program

References: /FB/, P3, “Basic PLC Program”

so that the appropriate reactions can be initiated on the machine tool.

Output of auxiliary functions to PLC properties

Common properties can be assigned to the above auxiliary function types through appropriate configuring.

Auxiliary function groups

Auxiliary functions can be combined to form channel-specific groups.

© Siemens AG, 2003. All rights reserved

SW 6.4

This software version can be used to configure:

- Associated auxiliary functions for M0 and M1
- Output specifications for individual auxiliary functions
  - Predefined (system functions)
  - User-defined auxiliary functions.

PLC acknowledgments

There are two classes of auxiliary function which depend on the selected PLC acknowledgment response in the parts program:

- Normal auxiliary functions
- High-speed auxiliary functions
- High-speed auxiliary functions without block change delay (SW 5)

Application of auxiliary function outputs

Auxiliary functions are used

- in normal parts program blocks
- in action parts of synchronized actions/technology cycles.

References: /FB/, S5, “Synchronized Actions”
/FBSY/, Synchronized Actions, SW 4.2 and higher

■

© Siemens AG, 2003. All rights reserved

Detailed Description 2

Introduction

When output to the PLC, auxiliary functions activate a system function or send information from the parts program to the PLC.

The following types of auxiliary function are available:

- Predefined auxiliary functions, see 2.2
- User-defined auxiliary functions, see 2.3

The following section describes the various methods of configuring and programming auxiliary functions as well as their operating principles.

2.1 Programming of auxiliary functions

General structure of an auxiliary function

ID letter[address extension]=value

The permissible ID letters for auxiliary functions are:
M, S, H, T, D, DL, F.

The address extension must be an integer. The address extension and value are defined differently for individual auxiliary functions depending on their meaning and value range. The square brackets can be omitted when an address extension is specified directly as a numeric value. For details, please see Table 2-1.

No address extension is permitted for D and DL.

Shortened structure of auxiliary functions

ID letterValue

Specification of address extension and value by variables

Address extension and value in auxiliary functions can be specified by variables. (Not applicable to FM-NC).

Example

```
SPINDLE_NO=1
DIRECTIONOFROTATION=3
...
M[SPINDLE_NO]=DIRECTIONOFROTATION
....
corresponds to: M1=3
```

© Siemens AG, 2003. All rights reserved

Address extensions of the character string type (axis name for F functions) are not transferred to the PLC interface, but only the assigned numerical values.

Table 2-1 Overview of auxiliary functions, programming

Function	**Address extension** (integer)		**Value**			**Explanations**	**Number per block** (Note 7)
	Meaning	**Range**	**Range**	**Type**	**Meaning**		
M (Note 5)	–	implicit 0	max. 8 digits	INT	Function		5
	Spindle no.	1 – 12	1 – 99		Function	M3, M4, M5, M19, M70 with address extension for spindle no. e.g. M5 for spindle 2: M2=5. Master spindle is used where no spindle is specified.	
	Optional	0 – 99	0 – 2147483647 (max. INT value)		Function	User M function	
S	Spindle no. Note: 5	0 – 12	0–±3.4028 ex 38 see Note 3	REAL	Speed	Without spindle no. for master spindle	3
H	Optional	0 – 99	–2147483648 +2147483647 ±3.4028 ex 38 Notes: 2, 3 and 4	INT (SW 5) REAL	Optional	Functions have no effect in the NCK, they are implemented solely by the PLC	3
T (Notes 5, 6)	Spindle no. (for active tool management)	1 – 12	0 – 32000 (also tool names for active tool management)	INT	Tool selection	Tool names are not sent to the PLC interface. See note 1	1
D			0 – 9	INT	Tool offset selection	D0 selection, default D1	1
DL	–	–	0 – 6	INT	See tool fine offset selection /FBW/	Refers to previously selected D number	1
F	Path feed	0	0,001– 999 999,999	REAL	Path feeds		6
(FA)	Axis no.	1–31	0,001– 999 999,999		Axis feeds		

Special points to be noted

The following applies for active tool management:

- T and Mk (k: tool change code, MD 22560: $MC_TOOL_CHANGE_M_CODE; Default=6) are not output as auxiliary functions.
- If no address extension is given, this is treated as the number of the master spindle (MD 20090: $MC_SPIND_DEF_MASTER_SPIND or command SETMS) or the master tool holder (MD 20124: $MC_TOOL_MANAGEMENT_TOOLHOLDER or command SETMTH).

© Siemens AG, 2003. All rights reserved

Note

1. When tool management is active, neither a T change signal nor a T word are output to the interface (channel).
2. The type for the values can be selected by the user via MD 22110: AUXFU_H_TYPE_INT. (SW 5 and higher.)
3. Due to the limited display possibilities on the screens of the control panels, the values of type REAL are actually restricted to –999999999.9999 to 999999999.9999. Calculations in the NC kernel are performed with full REAL accuracy.
4. The REAL values offered for MD 22110: AUXFU_H_TYPE_INT= 1 are rounded and forwarded to the PLC. The PLC program must interpret the value according to the setting in the MD.
5. If tool management is used (see /FBW/), the meaning of the address expansion can be controlled by an MD in SW 5 and higher.
   Address extension = 0 means that the NCK value is to be replaced by that of the master spindle number; this effectively corresponds to non-programming of the address extension.
   Spindle programming collecting during block search with the spindle auxiliary function with M19 is not output with SW 5.3 or higher.
6. M6: Without tool management: Addr. ext. range 0–99
   With tool management: Addr. ext. range 0 – max. spindle no.
   0: Replace with value of the master spindle no. or the master tool holder
7. With active tool management the tool change M-code (M6) can only be present once in the NC block, irrespective of which address extensions have been programmed.

A maximum total of 10 auxiliary functions may be programmed in one block. Alarm 14770 “Auxiliary function incorrectly programmed” is output when the specified length for address extension of value is exceeded or when the wrong data type is used. The following table shows some programming examples for H functions.

Table 2-2 Programming examples for H functions

Programming	Output of H function to the PLC
(coolant=12, lubricant=130)	
H[coolant]=lubricant	H12=130
H=coolant	H0=12
H5	H0=5
H=5.379	H0=5.379
H17=3.5	H17=3.5
H[coolant]=13.8	H12=13.8
H=’HFF13’	H0=65299
H=’B1110’	H0=14
H5.3=21	Error, alarm 14770

© Siemens AG, 2003. All rights reserved

PLC interface

Auxiliary functions are transferred to the PLC interface with 3 components:

- Address extension
- Value
- Signal indicating the signal validity.

The position of the components in the PLC interface is described:

References: /LIS/, Lists

The procedures for accessing the PLC interface are described in:

References: /FB/, P3, “Basic PLC Program”

2.2 Predefined auxiliary functions

Predefined auxiliary functions activate a **system function** when they are output to the PLC.

In SW 6.4, index 6–17 can be used to set individual output times and spindle actions for the predefined functions based on the acknowledgment. If an individual setting has been defined, it takes priority over the default setting for that type of auxiliary function.

In addition to the default, a different group assignment can set for predefined auxiliary functions with index 11–17 (see 2.4).

© Siemens AG, 2003. All rights reserved

Table 2-3 Predefined auxiliary functions

Index	Type	Extension possible	Value	Group see MD 22040	System function	Output specification (SPEC), bit-coded see MD 22080							
						End of motion	During motion	Prior to motion	Spin. action after ack.	No output	Reserv.	High-speed acknowledgment	Normal ack.
0	M	0	0	1	Stop	1	0	0	0	0	0	0	1
1	M	0	1	1	Cond. stop	1	0	0	0	0	0	0	1
2	M	0	2	1	Subprogram motion	1	0	0	0	0	0	0	1
3	M	0	17	1		1	0	0	0	0	0	0	1
4	M	0	30	1		1	0	0	0	0	0	0	1
5	M	1	6	65	Tool change	0	0	1	0	0	0	0	1
6	M	1	3	2	Spindle right	**(0)**	**(0)**	**(1)**	**(0)**	0	0	0	1
7	M	1	4	2	Spindle left	**(0)**	**(0)**	**(1)**	**(0)**	0	0	0	1
8	M	1	5	2	Spindle stop	**(0)**	**(0)**	**(1)**	**(0)**	0	0	0	1
9	M	1	19	2	Position spindle	**(0)**	**(0)**	**(1)**	**(0)**	0	0	0	1
10	M	1	70 (*)	2	Axis mode	**(0)**	**(0)**	**(1)**	**(0)**	0	0	0	1
11	M	1	40	**(4)**	Automatic gear stage	0	0	1	**(0)**	0	0	0	1
12	M	1	41	**(4)**	Gear stage 1	0	0	1	**(0)**	0	0	0	1
13	M	1	42	**(4)**	Gear stage 2	0	0	1	**(0)**	0	0	0	1
14	M	1	43	**(4)**	Gear stage 3	0	0	1	**(0)**	0	0	0	1
15	M	1	44	**(4)**	Gear stage 4	0	0	1	**(0)**	0	0	0	1
16	M	1	45	**(4)**	Gear stage 5	0	0	1	**(0)**	0	0	0	1
17	S	1	–1	**(3)**	Spindle speed	**(0)**	**(1)**	**(0)**	**(0)**	0	0	0	1

The values in brackets can be set using MD. The values in the table are the default settings.

(*): A different value can be preset using MD 20095: EXTERN_RIGID_TAPPING_M_NR or MD 20094: SPIND_RIGID_TAPPING_M_NR. However, only the fixed value 70 is transferred to the PLC.

Block search

The group assignment (see 2.4) of an auxiliary function defines its behavior during a block search. Only the last auxiliary function within a group is accumulated and output on the first start after a block search. The auxiliary functions in the 1st group are not accumulated.

© Siemens AG, 2003. All rights reserved

Spindles

The predefined auxiliary functions M6, M3, .., M45 and S are always assigned to the 1st spindle.

Output options can be configured for other spindles using **user-defined** auxiliary functions by specifying the corresponding **extension** (see MD 22060). The output specification configuration should only contain the permitted modifications, as plausibility checks are not applied to userdefined auxiliary functions.

Note

As the predefined auxiliary functions are configured on a channel-specific basis, if a spindle is assigned to a different channel, you must make sure that the configuration in that channel is set accordingly.

2.2.1 Output specifications for predefined auxiliary functions

The point in time at which auxiliary functions are output can be configured using the **output specification** (MD 22080).

Up to SW 6.3, output times could only be defined on a group-specific basis. The machine data available in SW 6.4 and higher enables output times to be set explicitly for specific predefined auxiliary functions.

All predefined auxiliary functions are configured using the following machine data and their various components can be reconfigured:

MD 22040: AUXFU_PREDEF_GROUP[index] ; group assignment for predefined auxiliary function

MD 22050: AUXFU_PREDEF_TYPE[index] ; ID letter (M,S,H,T,D,DL,F)

MD 22060: AUXFU_PREDEF_EXTENSION[index] ; address extension

MD 22070: AUXFU_PREDEF_VALUE[index] ; auxiliary function value

MD 22080: AUXFU_PREDEF_SPEC[index] ; **Output specification**

The index array “index” corresponds to the index in column 1 of the predefined auxiliary functions table.

The output specification for predefined auxiliary functions is set as follows using MD 22080:

MD 22080 AUXFU_ PREDEF_SPEC	**Bit7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Output specification	Output at block end	Output during motion	Output prior to motion	Spindle response following acknowledgment	No output to interface	Reserved	High-speed acknowledgment with OB40	Normal acknowledgment after OB1 cycle

© Siemens AG, 2003. All rights reserved

Spindle response

Output specification bit 4 (spindle response after acknowledgment) can be used to configure whether the spindle responds after or before the PLC acknowledgment.

Motion

Bits 5 to 7 control the output of the auxiliary function in terms of single-axis or path motion. If the auxiliary function has been programming without a traversing block, bits 5 to 7 are insignificant.

Priority

The output specifications have higher priority than the properties defined for the group in MD 11110 and these properties in turn have higher priority than the definitions for the type in MD 22200 22252:

MD 22080: AUXFU_PREDEF_SPEC[index]	; priority: high
MD 11110: AUXFU_GROUP_SPEC[grpindex]	; priority: average (see 2.4)
MD 22210: AUXFU_M_SYNC_TYPE	; priority: low
MD 22220: AUXFU_T_SYNC_TYPE	; priority: low
MD 22230: AUXFU_H_SYNC_TYPE	; priority: low
MD 22240: AUXFU_F_SYNC_TYPE	; priority: low
MD 22250: AUXFU_D_SYNC_TYPE	; priority: low
MD 22252: AUXFU_DL_SYNC_TYPE	; priority: low

If auxiliary functions have been predefined the predefined output specifications are always valid.

Priority: progr. high-speed acknowledgment

The programming of high-speed auxiliary functions, such as m0=qu(100), in the parts program and in synchronized action takes priority over configuration.

Synchronized actions

For the purpose of auxiliary function output via synchronized actions, all output specifications except quick and normal are ignored due to the fact that synchronized actions switch entirely asynchronously and the auxiliary functions are output to the PLC immediately as soon as the synchronized action condition is true.

Special cases relating to M commands

The PLC output of M commands M2, M17, M30 in subprograms can be determined selectively in MD 20800: SPF_END_TO_VDI:

1:	The M functions for subprograms (M17 or M2/M30) are transferred to the PLC interface
0:	The M functions for subprograms (M17 or M2/M30) are not transferred to the PLC interface

DL auxiliary function for sum offset (SW 5.2 and higher)

1 DL auxiliary function ca be programmed per block and output to the PLC. An extended address notation is not permitted for the DL command. Range of values: 0–32000 (programming: DL=n).
DL values cannot be output to the PLC via synchronized actions.
You can use MD 22252 AUXFU_DL_SYNC_TYPE to specify the time for output of the DL function.

© Siemens AG, 2003. All rights reserved

Configuration

Only the group assignments and output specifications in brackets in Table 2-3 can be modified.

Any attempts to modify other data will be rejected during ramp-up with the error message:
“Channel %1 illegal predefined auxiliary function in %2 %3, MD”.

The machine data will be reset to the fixed preset values. The **predefined** auxiliary functions cannot be overwritten by configuring user-defined auxiliary functions (see 2.3). If an attempt is made to do this, alarm “4185 channel K1 illegal configuration of an auxiliary function SM I1 S5” will be generated.

The configurable output time of predefined auxiliary functions can be used to determine e.g. whether a spindle function is initiated prior to path or single-axis motion, during motion or at the end of the block.

For example:

```
x10 m3          ; spindle turning according to configuration, during or
                ; after motion x10
```

Thread cutting

For the purposes of thread cutting G33, G34 and G35 auxiliary functions M3 and M4 are always output during motion and without block change delay. Spindle stop M5 is always output at the end of the block. The block containing M5 is always approached with exact stop. Continuous-path mode may be interrupted under certain circumstances.

The table below contains a list of predefined auxiliary functions whose group assignment and output specification **cannot be changed**. These auxiliary functions can only be configured using certain machine data.

Type	Extension possible	Value	Group	System function	Output specification (SPEC), bitcoded							
					End of motion	During motion	Prior to motion	Res.	No output	Res.	High-speed ack–nowledgment	Normal ack.
M	0	(*1)	1	Assoc. stop	1	0	0	0	0	0	0	1
M	0	(*2)	1	Assoc. cond. stop	1	0	0	0	0	0	0	1
T	SpNo.	–1	64	Tool selection	0	0	1	0	0	0	0	1
M	0	(*3)	1	Subprogram motion	1	0	0	0	0	0	0	1
M	0	(*4)	10	Nibbling	1	0	0	0	0	0	0	1
M	0	(*4)	10	Nibbling	1	0	0	0	0	0	0	1
M	0	(*4)	11	Nibbling	1	0	0	0	0	0	0	1
M	0	(*4)	11	Nibbling	1	0	0	0	0	0	0	1
M	0	(*4)	11	Nibbling	1	0	0	0	0	0	0	1
M	0	(*4)	11	Nibbling	1	0	0	0	0	0	0	1
M	0	(*4)	12	Nibbling	1	0	0	0	0	0	0	1
M	0	(*4)	12	Nibbling	1	0	0	0	0	0	0	1

© Siemens AG, 2003. All rights reserved

(*1): The value is set using MD 22254: AUXFU_ASSOC_M0_VALUE.

(*2): The value is set using MD 22256: AUXFU_ASSOC_M1_VALUE.

(*3): The value is set using MD 10714: M_NO_FCT_EOP.

(*4): The value is set using MD 26008: NIBBLE_PUNCH_CODE.

The diagrams below illustrate the effect of the various output specifications.

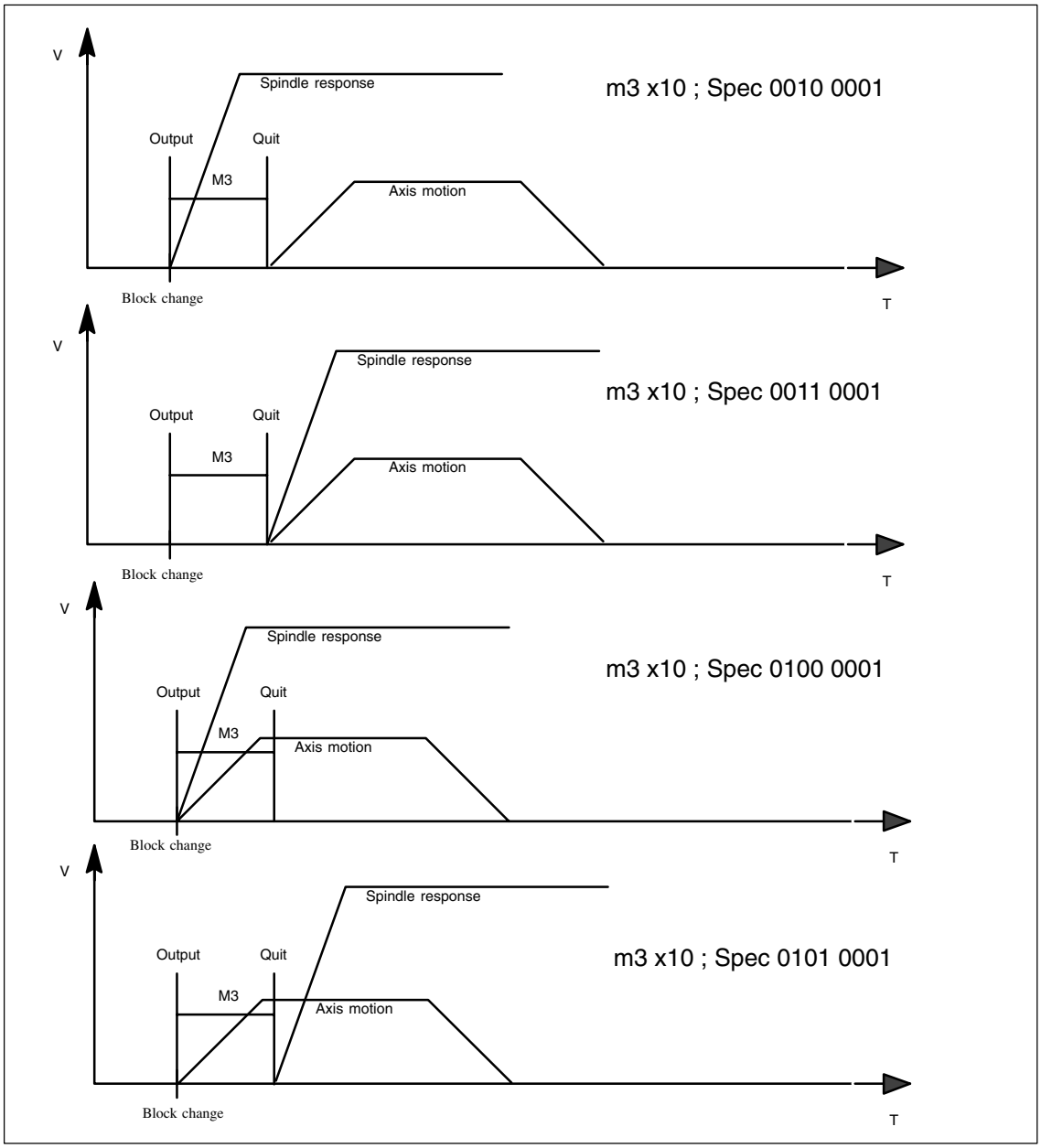


Fig. 2-1 Possible output specifications (1.)

© Siemens AG, 2003. All rights reserved

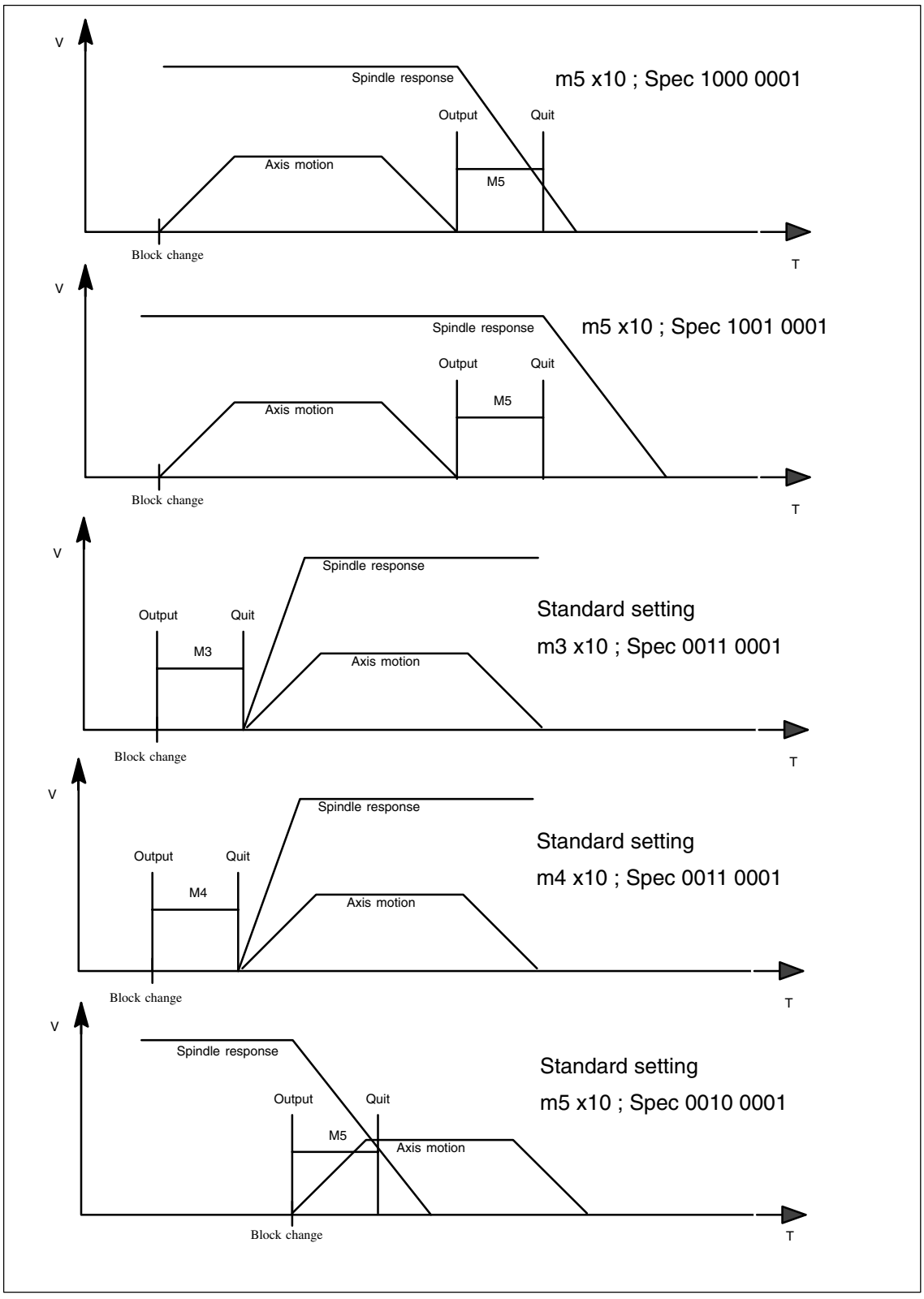


Fig. 2-2 Possible output specifications (2.)

© Siemens AG, 2003. All rights reserved

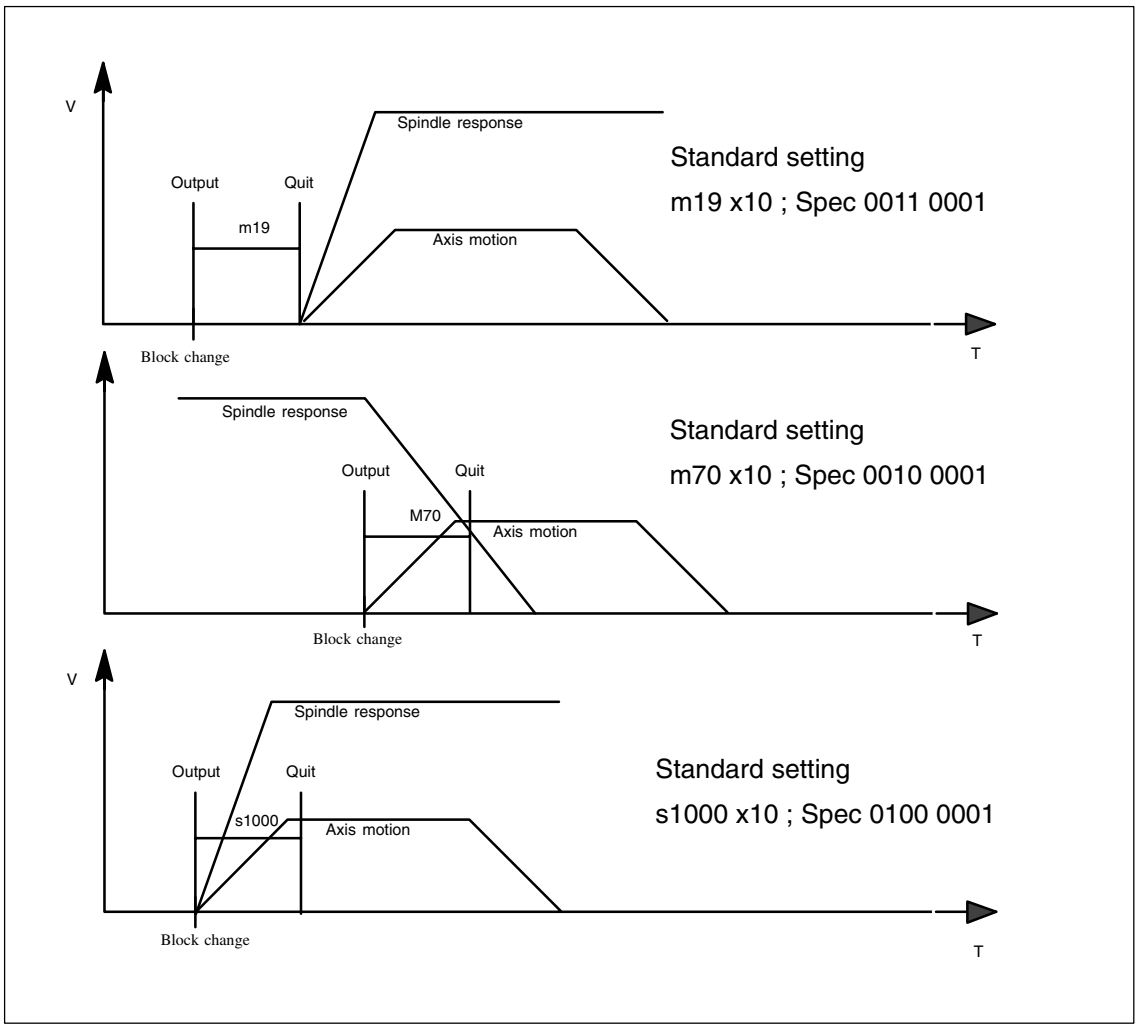


Fig. 2-3 Possible output specifications (3.)

© Siemens AG, 2003. All rights reserved

2.2.2 Associated auxiliary functions for M0, M1 (SW 6.3 and higher)

Auxiliary functions

```
M0          ; Parts program stop
M1          ; Conditional stop
```

have the components and meanings defined in 2.2. An associated auxiliary function with the same functionality can be declared for each of these auxiliary functions in the user program.

Precondition

The PLC user program implements processing of associated auxiliary functions in the same way as for original function M0 or M1.

Procedure

The MD 22254: AUXFU_ASSOC_M0_VALUE must contain the value of an auxiliary function number, e.g. 134.

M134 then initiates the same function as performed by M0.

The MD 22256: AUXFU_ASSOC_M1_VALUE must contain the value of an auxiliary function number, e.g. 173.

M173 then initiates the same function as performed by M1.

Neither machine data permits predefined auxiliary function numbers with fixed meaning.
Alarm 4181 is output in response to an error.

The auxiliary function numbers in the two machine data must be **different**. It is also possible to declare just one of the two associated auxiliary functions.

Auxiliary function groups

The group assignment for associated M functions is identical to that for the corresponding original auxiliary functions. M0, M1 and their associated auxiliary functions are preassigned to group 1. See 2.4.

Application

It is possible to make particular distinctions in the programming, e.g.

```
M1    ; Conditional stop in parts program and
M173  ; Conditional stop in user cycle.
```

Conclusive distinctions are taken care of in the user PLC program.

Use

Associated auxiliary functions may be used in main programs, subroutines and cycles, but **not** in **synchronized actions**.

References: /FBSY/, Synchronized Actions.

© Siemens AG, 2003. All rights reserved

Note

Modifications to machine data:

MD 22254: AUXFU_ASSOC_M0_VALUE and

MD 22256: AUXFU_ASSOC_M1_VALUE

require the user PLC program to be adapted.

PLC/NCK

The associated M00 and M01 auxiliary functions use the same bits in the PLC interface as M0 and M1. The function value output to the interface is, however, the configured value from the machine data.

The following extra signals are provided for associated auxiliary function M1

DB 21–30, DBB318 Bit 5: Associated M00/M01 active (NCK→PLC) and

DBB30 bit 5: Activate associated M01 (PLC→NCK).

© Siemens AG, 2003. All rights reserved

2.3 User-defined auxiliary functions

Term

All auxiliary functions listed not denoted as predefined auxiliary functions in Table 2-3 are user-defined auxiliary functions.

2.3.1 Output specifications for user-defined auxiliary functions

In addition to the options described in 2.4 for assigning an auxiliary function to a group and defining its output specification for the group, this section also explains how specific output options can be assigned to individual auxiliary functions within a group.

MD 22035: AUXFU_ASSIGN_SPEC is used to specify the definitions for the individual auxiliary functions.

Overview of output specification options (see also Figures 2-1 to 2-3)

Once the required response has been selected, the output specification is set in MD 22035: AUXFU_ASSIGN_SPEC as follows:

0: “Normal” acknowledgment following an OB1 cycle
1: “Quick” acknowledgment with OB40
3: No output to the VDI
4: Spindle response following acknowledgment
5: Output prior to motion
6: Output during motion
7: Output at block end

Sequence

The following machine data define settings for auxiliary function output options.

Output specifications for user-defined auxiliary functions are set according to the priority of the following machine data:

$MC_AUXFU_ASSIGN_SPEC[index]	; highest
$MN_AUXFU_GROUP_SPEC[index]	; average (see 2.4)
$MC_AUXFU_(M,S,T,H,F,D,DL)_SYNC_TYPE	; lowest

Default

MD 22035: AUXFU_ASSIGN_SPEC[index] is 0. At a value of 0, the group output specification is valid.

For example, if there are two spindles in the system, the response of the master spindle takes priority over the **predefined** auxiliary functions. The response of the auxiliary functions for the second spindle can be defined by assigning the auxiliary functions for the second spindle to a separate group.

Examples

The 2nd spindle is assigned to group 5.

```
m2 = 3          ; Spindle 2 is to be turned clockwise by the PLC
                ; prior to acknowledgment (see 2.5)
```

© Siemens AG, 2003. All rights reserved

```
m2 = 4          ; Spindle 2 is to be turned counterclockwise by the PLC
                ; during path motion and after acknowledgment.
m2 = 5          ; Spindle 2 is to be stopped by the PLC
                ; after path motion and after acknowledgment.
```

```
;m2 = 3
$MC_AUXFU_ASSIGN_GROUP[0]=5
$MC_AUXFU_ASSIGN_TYPE[ 0 ] = 'M'
$MC_AUXFU_ASSIGN_EXTENSION[0]=2
$MC_AUXFU_ASSIGN_VALUE[0]=3
$MC_AUXFU_ASSIGN_SPEC[ 0 ] = 'H21'
```

```
;m2 = 4
$MC_AUXFU_ASSIGN_GROUP[1]=5
$MC_AUXFU_ASSIGN_TYPE[ 1 ] = 'M'
$MC_AUXFU_ASSIGN_EXTENSION[1]=2
$MC_AUXFU_ASSIGN_VALUE[1]=4
$MC_AUXFU_ASSIGN_SPEC[ 1 ] = 'H51'
```

```
;m2 = 5
$MC_AUXFU_ASSIGN_GROUP[2]=5
$MC_AUXFU_ASSIGN_TYPE[ 2 ] = 'M'
$MC_AUXFU_ASSIGN_EXTENSION[2]=2
$MC_AUXFU_ASSIGN_VALUE[2]=5
$MC_AUXFU_ASSIGN_SPEC[ 2 ] = 'H91'
```

Output times of various types of auxiliary function

Machine data can be programmed to define the timing for transmission of auxiliary function parameters to the PLC for each **type** of auxiliary function.

MD 22200: AUXFU_M_SYNC_TYPE

MD 22210: AUXFU_S_SYNC_TYPE

MD 22220: AUXFU_T_SYNC_TYPE

MD 22230: AUXFU_H_SYNC_TYPE

MD 22240: AUXFU_F_SYNC_TYPE

MD 22250: AUXFU_D_SYNC_TYPE

MD 22252: AUXFU_DL_SYNC_TYPE (SW 5.2 and higher)

Assignment of machine data:

0:	Output prior to motion
1:	Output during motion
2:	Output at block end
3:	**No** output to the PLC

In the case of auxiliary functions which are output at the end of a block, the parameters are not transferred until all path and positioning axes have finished interpolation.

If several auxiliary functions with different output types (prior, during, at end of motion) are programmed in one motion block, then they are output individually according to their output type.

© Siemens AG, 2003. All rights reserved

In a block without a path motion (even those with positioning axes and spindles), the auxiliary functions are all output immediately in a block.

If auxiliary function M17 or M2/M30 is programmed as the only auxiliary function in a block where an axis is still in motion, then its parameters are not output to the PLC until the axis has reached standstill.

Sequence

If definitions for the output time of the auxiliary function are made using MD

$MC_AUXFU_ASSIGN_SPEC[index] ; highest (see 2.3.1)

$MN_AUXFU_GROUP_SPEC[index] ; average (see 2.4)

definitions made using AUXFU_(M, S, T, H, F, D, DL)_SYNC_TYPE will not be taken into account.

Example of auxiliary function output

The NC block has 3 modalities which can be defined by MD:

N10 G01 X100 M07 H5 T5

The machine data settings are as follows:

MD 22220: AUXFU_T_SYNC_TYPE =0	Output time for T functions **prior to** motion
MD 22200: AUXFU_M_SYNC_TYPE =1	Output time for M functions **during** motion
MD 22230: AUXFU_H_SYNC_TYPE =2	Output time for H functions **at the end** of the block

The chronological sequence of the auxiliary function outputs is then as follows:

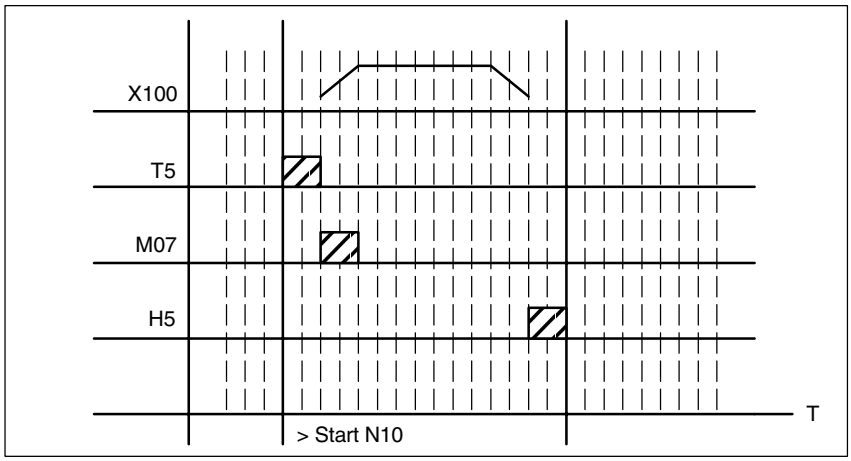


Fig. 2-4 Example of auxiliary function output

Continuous-path mode

A path movement can only remain continuous if auxiliary function output takes place **during the movement** and is acknowledged by the PLC before the path end is reached.

References: /FB/, B1, “Continuous-Path Mode, Exact Stop, Look Ahead”
See also 2.5 Acknowledgment of auxiliary functions by the PLC

© Siemens AG, 2003. All rights reserved

2.4 Grouping of auxiliary functions

Introduction

Auxiliary functions can be grouped through the programming of channel-specific machine data.
MD 11110: AUXFU_GROUP_SPEC[k]
can be set to assign identical properties in terms of transfer timing to the PLC and mode of PLC acknowledgment to all the auxiliary functions in a group.

Note

A group assignment of type DL is not supported.

Control of group properties

MD 11110: AUXFU_GROUP_SPEC[k]

Set bit:

0:	Auxiliary function acknowledgment (see 2.5)
1:	High-speed auxiliary function acknowledgment (see 2.5)
2:	Reserved
3:	**No** output to the PLC
4:	Spindle response following acknowledgment by the PLC
5:	Output before block processing commences
6:	Output while block processing is in progress
7:	Output at block end

If a programmed auxiliary function is assigned to a group, then only the corresponding group specifications are valid.

The specifications apply to all auxiliary functions assigned to the group.
Index k stands for the number of the group.

Properties without group assignment

Auxiliary functions that are not assigned to a group are treated as auxiliary functions with “normal acknowledgment”. See 2.5.

Number of groups (SW 5.2 and higher)

A maximum of 15 or 64 (SW 5.2 and higher, see Chapter 3) auxiliary function groups can be defined.

Number of auxiliary function groups per channel (SW 5.2 and higher)

A maximum of 50 or 255 (SW 5.2 and higher, see Chapter 3) auxiliary functions can be assigned to the 15 or 64 (SW 5.2 and higher) auxiliary function groups. The number actually used can be set in MD 11100: AUXFU_MAXNUM_GROUP_ASSIGN. This number does not include auxiliary functions that are preassigned as standard. See “Default group properties”, “Default auxiliary function assignments”.

© Siemens AG, 2003. All rights reserved

Assignment of groups

An auxiliary function is configured uniquely using 5 MDs:

MD 22000: AUXFU_ASSIGN_GROUP[n]= k — Auxiliary function group
MD 22010: AUXFU_ASSIGN_TYPE[n]="Address letter" — Type of auxiliary function (ID letter)
MD 22020: AUXFU_ASSIGN_EXTENSION[n]= address extension — Auxiliary address extension
MD 22030: AUXFU_ASSIGN_VALUE[n]= value — Auxiliary function value
MD 22035: AUXFU_ASSIGN_SPEC[n]='H51' — Output specification

The maximum number of auxiliary functions that can be defined is set using MD 11100: AUXFU_MAXNUM_GROUP_ASSIGN.

The index to be used in the above machine data is:
0 – (MD 11100 – 1)

Example

The spindle speed of the 2nd spindle is to be assigned to group 11. The MDs for the 17th auxiliary function are then:
⇒ MD 22010: AUXFU_ASSIGN_TYPE[17] = "S"
MD 22020: AUXFU_ASSIGN_EXTENSION[17] = 2
MD 22030: AUXFU_ASSIGN_VALUE[17] = –1
MD 22000: AUXFU_ASSIGN_GROUP[17] = 11
MD 22035: AUXFU_ASSIGN_SPEC[17]='H21'

Special points to be noted

The auxiliary functions assigned to a group **ignore** the set default values in the following machine data (output times of the M, S, T, H, F, D and DL functions):
MD 22200: AUXFU_M_SYNC_TYPE,
MD 22210:AUXFU_S_SYNC_TYPE,
MD 22220:AUXFU_T_SYNC_TYPE,
MD 22230:AUXFU_H_SYNC_TYPE,
MD 22240:AUXFU_F_SYNC_TYPE,
MD 22250:AUXFU_D_SYNC_TYPE and
MD 22252:AUXFU_DL_SYNC_TYPE (SW 5.2 and higher).

If MD 22030: AUXFU_ASSIGN_VALUE[n] (auxiliary function value) for speed auxiliary functions is set to a **value of <0**, **all** auxiliary functions of the configured type with the same address extension are assigned to this group. This feature has been utilized in the above example to assign all S values of a spindle to one group.

Note

The group assignment for predefined auxiliary functions can be found in Table 2-4.

© Siemens AG, 2003. All rights reserved

Default group properties

If auxiliary functions are not defined in MD 11110: AUXFU_GROUP_SPEC[n], they have the following properties:

- Normal acknowledgment by PLC
- Transfer time:
  - At end of block for group 1
  - Prior to motion for group 2
  - During motion for group 3.

The user can change this setting for groups 2 to 15, but not for group 1.

2.5 Acknowledgment of auxiliary functions by the PLC

Introduction

Section 2.4 describes how the mode in which the PLC acknowledges grouped auxiliary functions can be programmed. The acknowledgment response (higher priority) can be defined by means of output specifications for individual auxiliary functions. This Section describes the response characteristics for:

- Normal acknowledgment
- High-speed acknowledgment.

With SW 5 and higher, it is also possible to control the block change behavior:

- with block change delay
- without block change delay.

The block change is controlled by MD 22100: AUXFU_QUICK_BLOCKCHANGE.

Value of the MD	Meaning
0	with block change delay
1	without block change delay

The possible combinations of these specifications are described in the Section below.

The following table specifies the block change behavior and the velocity reduction depending on the specified time of auxiliary function output, the machine data setting, and the type of acknowledgment.

© Siemens AG, 2003. All rights reserved

2.5.1 Normal acknowledgment with delayed block change

Auxiliary functions with normal acknowledgment are transferred to the PLC interface on commencement of the OB1 call by the basic PLC program. Interface modification signals indicate when the values are valid.
The basic PLC program acknowledges auxiliary functions, as soon as the PLC has run through the complete OB1 once (one full PLC user cycle).

Normal auxiliary functions are output **block-synchronously**.

If block processing is finished before the acknowledgment, the block change is delayed until the acknowledgment is received. Consequently: a path movement with auxiliary functions with normal acknowledgment can only remain continuous if auxiliary function output takes place during the movement and is acknowledged by the PLC before the block end is reached.

The MD 11110: AUXFU_GROUP_SPEC[n],
can be programmed to define all the auxiliary functions belonging to an auxiliary function group (see Section 2.4 Auxiliary function groups) as auxiliary functions with normal acknowledgment.

If an auxiliary function is not assigned to any auxiliary function group or an output specification has not been defined, it is always treated like an auxiliary function with “normal acknowledgment” provided that the time of output to the MD 22200–22250:

AUXFU_x_SYNC_TYPE ; x stands for the auxiliary function type M, S, H, T,

is not set to 0 (setting 3).

With block change delay With velocity reduction

Normal acknowledgment of auxiliary functions
MD 22100: AUXFU_QUICK_BLOCKCHANGE = 0

Table 2-5

Auxiliary function output	Behavior
before the movement	The movement stops at the end of the preceding block. The auxiliary functions are output in the first interpolation cycle of the block. The movement is not continued until the auxiliary functions have been acknowledged by the PLC after **one OB1 PLC cycle**.
during the movement	The current **tool path velocity** is **reduced** until the time to the end of the block is greater than the maximum PLC acknowledgment time (time of one OB1 cycle). This ensures that the PLC can acknowledge all auxiliary functions before the next block is executed.
after the movement	The movement stops at the end of the block. The auxiliary functions are output at the end of the block. The next block is not executed until the **OB1 acknowledgment** is received.

© Siemens AG, 2003. All rights reserved

SW 6.4 and higher

If an output specification has been configured for the auxiliary function, the following priority is valid for the acknowledgment response defined:

$MC_AUXFU_ASSIGN_SPEC[index]	; highest
$MN_AUXFU_GROUP_SPEC[index]	; average
$MC_AUXFU_(M,S,T,H,F,D,DL)_SYNC_TYPE	; lowest

2.5.2 Highspeed acknowledgment with delayed block change

Application

Prompt initiation of parallel activities in the PLC, subsequent scanning for acknowledgment if necessary for continued processing.

Auxiliary functions with high-speed acknowledgment are transferred to the PLC interface at the beginning of the next OB1 call by the basic PLC program.

They are characterized by the fact that they are immediately acknowledged by the basic PLC program (in OB40) if the PLC has recognized the transferred auxiliary functions of a block.

This transport acknowledgment does not provide information about the execution of the auxiliary functions on the PLC, but merely confirms that the PLC has received the auxiliary functions.

However, the next auxiliary function block cannot be transferred until the OB1 has been fully executed once. This produces a noticeable delay in cases where auxiliary functions with high-speed acknowledgment are output in several NC blocks in a sequence.

In the case of auxiliary functions with high-speed acknowledgment, it **cannot** be guaranteed that the PLC user program will react **block-synchronously**.

The MD 11110: AUXFU_GROUP_SPEC[n],
can be programmed to define all the auxiliary functions belonging to an auxiliary function group (see Section 2.4 Auxiliary function groups) as auxiliary functions with high-speed acknowledgment.

Auxiliary function evaluation in process alarm OB40

Through appropriate parameterization of the basic PLC program, it is possible to program direct evaluation of auxiliary functions T and H by the PLC user program in process alarm OB40 (SW 3.5 and higher).

References: /FB/, P3, “Basic PLC Program”

Explicit, high-speed acknowledgement

M, S, H, T, DL and D auxiliary functions which are not assigned to an auxiliary function group can be identified as high-speed auxiliary functions in the parts program.

Example: M = QU(7); High-speed output for M7,
Identification as high-speed auxiliary function:
QU(value)

© Siemens AG, 2003. All rights reserved

Note

A programmed explicit, high-speed acknowledgment is ignored if the auxiliary function concerned is assigned to an auxiliary function group with normal acknowledgment by the PLC. The properties assigned to an auxiliary function group have priority.
The block limits clearly defined in the parts program are not defined in the PLC user program.

Example of time properties

Parts program:

N10 G94 G01 X50 M100	MD 22200: AUXFU_M_SYNC_TYPE = 1, Output of M100 during motion
N20 Y5 M100 M200	MD 22200: AUXFU_M_SYNC_TYPE = 0, Output of M200 prior to motion.
N30 Y0 M=QU(100) M=QU(200)	Output as high-speed auxiliary functions
N40 X0	
N50 M100 M200 M17	No axis motions, immediate output

The following diagram shows the chronological sequence of normal and high-speed auxiliary function outputs (with different modes of synchronization with the programmed motions). Note the different block processing times of N20 and N30.

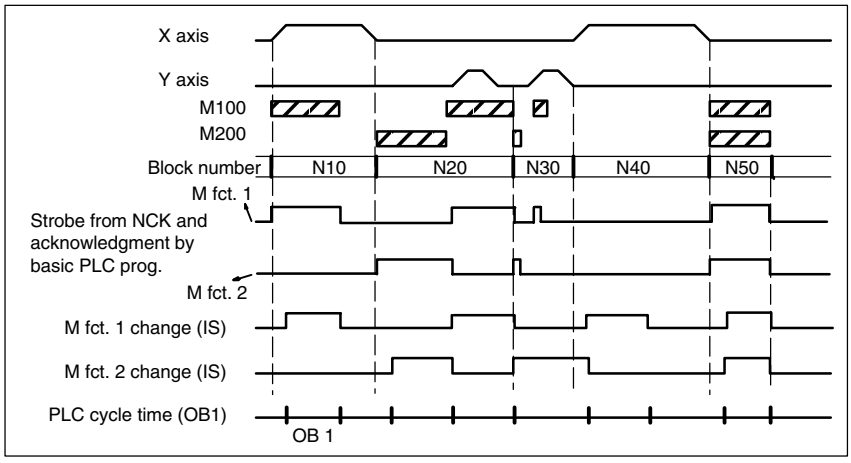


Fig. 2-5 Normal and highspeed auxiliary function outputs

With block change delay
With velocity reduction

High-speed acknowledgment of auxiliary functions
MD 22100: AUXFU_QUICK_BLOCKCHANGE = 0

© Siemens AG, 2003. All rights reserved

Table 2-6

Auxiliary function output	Behavior
before the movement	The movement stops at the end of the preceding block. The auxiliary functions are output in the first interpolation cycle of the block. The movement is not continued until the auxiliary functions have been acknowledged by the PLC after **one OB40 PLC cycle**.
during the movement	The current **tool path velocity** is **reduced** until the time to the end of the block is greater than the maximum PLC acknowledgment time (time of one OB1 cycle). This ensures that the PLC can acknowledge all auxiliary functions before the next block is executed.
after the movement	The movement stops at the end of the block. The auxiliary functions are output at the end of the block. The next block is not executed until the **OB40 acknowledgment** is received.

2.5.3 High-speed acknowledgment without delayed block change

The **block synchronism** of the transport from NCK to PLC **cannot** be **guaranteed** with this configuration, and, in a worst case scenario, may not occur until one OB1 cycle time after the block change. The highspeed acknowledgment does not ensure block synchronism because the auxiliary functions are not executed until the OB1 cycle.

A **path movement** remains **continuous** with this auxiliary function output.

Without block change delay Without velocity reduction

High-speed acknowledgment of auxiliary functions
MD 22100: AUXFU_QUICK_BLOCKCHANGE = 1

Table 2-7

Auxiliary function output	Behavior
before the movement	The block transition between blocks with highspeed auxiliary functions occurs **without** interruption and **without** a reduction in velocity. The auxiliary functions are output in the first interpolation cycle of the block. The next block is executed without an acknowledgment delay.
during the movement	The block transition between blocks with highspeed auxiliary functions occurs **without** interruption and **without** a reduction in velocity. The auxiliary functions are output during the block. The next block is executed without an acknowledgment delay.
after the movement	The movement stops at the end of the block. The auxiliary functions are output at the end of the block. The next block is executed without an acknowledgment delay.

© Siemens AG, 2003. All rights reserved

Example of dyn. response from SW 5 and higher

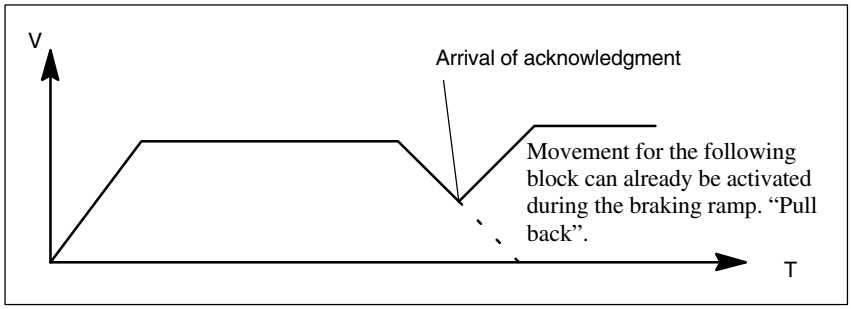


Fig. 2-6 Anticipated acknowledgment received during the deceleration phase

2.6 M function with implicit preprocessing stop (SW 6.3 and higher)

Sequence

The machine data MD 10713: M_NO_FCT_STOPRE can be programmed to define 15 M functions that execute an implicit preprocessing stop.

If an M function configured via MD 10713: M_NO_FCT_STOPRE is programmed, interpretation of the following parts program line is delayed until the block containing the M function has been executed fully, i.e. acknowledged by the PLC.

Example

Implicit preprocessing stop by M88

MD 10713: M_NO_FCT_STOPRE [0] = 88

```
N100 G0 X10 M88  ; Implicit preprocessing stop by M88
N110 Y=R1        ; N110 s not interpreted until N100 has
                 ; finished executing
```

Supplementary conditions

The implicit preprocessing stop is subject to the following restrictions:

- The subroutine call by M function using
  MD 10715: M_NO_FCT_CYCLE or
  M98 in ISO dialect T / ISO dialect M
  is **always executed without** a preprocessing stop.

© Siemens AG, 2003. All rights reserved

2.7 Block search response

Block search with calculation

With block searches involving calculation, auxiliary functions are evaluated and accumulated on a group-specific basis. The last auxiliary function of each group is output after NC start and prior to the actual reentry block.

All accumulated auxiliary functions are output in a separate block

- as normal auxiliary functions and
- prior to the motion

after NC start and on completion of the block search.

If auxiliary functions must be accumulated during block searches, they must be assigned to an auxiliary function group (see Section 2.4). An example illustrating how to assign functions to a group is also given in this Section.

Exception:
The auxiliary functions in the 1st group are not accumulated.

Overstoring after a block search

In addition to the auxiliary functions accumulated during block searches, new auxiliary functions can be output with the “Overstore” function prior to NC start. See 2.8.

Programmed spindle positioning

After a block search with calculation, the spindle is always positioned as programmed last even if other spindle functions are programmed before the block search target.

Note

The spindle enable must be derived from the motion commands (DB3[spindle] DBB64) since, under certain circumstances, the spindle-related auxiliary functions M3, M4, M5 may not be output to the PLC until the spindle has been positioned.

For further information on block search and spindle auxiliary functions:
References: /FB/, K1 Mode Group, Channels, Program Operation,

References: /FB/, S1 Spindles, Spindle auxiliary functions, Chapter 2 (SW 5.3 and higher)

© Siemens AG, 2003. All rights reserved

2.8 Response to overstore

Application

The auxiliary functions can be altered by means of “Overstore” prior to program start or in the event of a program interruption. It may therefore be necessary, for example, for the user to enter the current M, S, T and H values manually during a program trial run or after a block search.

Output of auxiliary functions to PLC types

The following auxiliary function types can be altered by Overstore:

- Tool number (T)
- Tool offset number D
- Sum offset DL (SW 5.2 and higher)
- Spindle speed S
- Auxiliary function (H)
- Miscellaneous function (M)
- Feedrate F.

Activation/ deactivation

The auxiliary function is output with the associated transition signal on the interface only when NC start is initiated and thus remains valid for further processing of the parts program until it is deselected by another “Overstore” operation or by a programmed auxiliary function of the same type.

Operation

References: /BA/, Operator’s Guide

© Siemens AG, 2003. All rights reserved

2.9 Applications of auxiliary functions

Function	Application	Special cases
M *)	Activate switching operations on the machine via parts program.	The following M functions have a **predefined** meaning (see Programming Guide): M0, M1, M2, M17, M30, each without address extension! M3, M4, M5, M6, M19, M70, M40, M41, M42, M43, M44, M45.
		For each auxiliary function M00–M99 there is a dyn. signal at the PLC interface which displays the arrival of a new M function. A further 64 signals can also be assigned for user-defined M functions. /FB/, P3, “Basic PLC Program”
		For M17, M2, and M30, MD 20800: SPF_END_TO_VDI can be used to define whether output to the PLC takes place.
		The MD 22560: TOOL_CHANGE_M_CODE initiates tool changing. In DIN 66025 this tool change is programmed with the function M06.
S *)	Transfer of spindle speed to the PLC interface	S functions are assigned as standard to group 3. See 2.4 A function without address extension is referred to the master spindle.
H	Switching, auxiliary and motion functions which are not controlled by the NCK, but exclusively by the PLC.	The data type in which the value of the H auxiliary functions is interpreted can be defined in MD 22110: AUXFU_H_TYPE_INT (SW 5 and higher) The following are possible: INT from –2 147 483 648 to 2 147 483 647 or REAL from –3.4028 ex 38 to 3.4028 ex 38
T *)	Selection of tool	Identification of tools by either tool number or location number. /FBW/, Tool management /FB/, W1, “Tool offset” When T0 is selected, the current tool is removed from the tool holder but not replaced by a new tool. (Default setting).
D	The D function is used to select a tool offset stored in the NC memory that was defined for a previously active tool. The D number transferred to the PLC can be evaluated for the purpose of implementing switching operations and protective functions.	D1 applies if no D function is activated. MD 20270: CUTTING_EDGE_DEFAULT determines the default offset after tool changes. D0 deselects the offset for the tool.
		The system response to RESET, program start and power-up is determined by MD 20110: RESET_MODE_MASK and MD 20120: START_MODE_MASK in combination with other machine data. /FB/, K2, Workpiece-Related Actual Value System
DL	The DL function selects a sum offset of a previously activated tool offset D from the NC memory. The DL number passed to the PLC can be evaluated to initiate switching actions and protective functions.	The parameter refers to a D number which has already been transferred. DL values cannot be output to the PLC via synchronized actions. If no DL function is activated, DL=0 is assumed. MD 20272: SUMCORR_DEFAULT determines the default offset after tool changing. DL=0 deselects the sum offset for the active D offset.
F	The feedrates programmed in the parts program are transferred to the PLC with the F function. The PLC can derive switching operations and protective functions from the feedrate data.	No address extension: Path feed FA[address extension] : Feedrate of a positioning axis with machine axis number as address extension. Feedrates can be transferred with: G93, G94, G95, G96, G33, G63, G331, G332, but not with dwell times programmed with G4. **Note:** The axis velocity may drop in continuouspath mode during output of F functions to the PLC. This function should be used only if unavoidable.

*) If tool management is installed (see /FBW/), the meaning of the address extension can be controlled by an MD in SW 5 and higher.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

Availability

The Output of auxiliary functions to PLC function is available with SW 1 and higher.

SW 5

The following function extensions have been added:

- High-speed acknowledgment without delayed block change
- Selectable data type for H auxiliary functions and extended range for integer type
- Variable meaning of address extension for M, S, T auxiliary functions in association with tool management (> SW 5.1).
- Transfer of DL sum offsets to PLC (SW 5.2 and higher)
- Expansion to 64 auxiliary function groups and 64 auxiliary functions (SW 5.2 and higher)

SW 6

SW 6.4 also provides:

- Associated auxiliary functions M0, M1
- Predefined auxiliary functions with output specifications
- User-defined auxiliary functions with output specifications.

Note

A tight configuration of the block memory may mean that the maximum number 64 is not reached (abort with error message). A more generous configuration is then recommended.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

4.1 General machine data

10713 MD number	**M_NO_FCT_STOPRE** M function with preprocessing stop		
Default setting: –1	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	The M functions defined via machine data MD 10713: M_NO_FCT_STOPRE execute an implicit preprocessing stop, i.e. interpretation of the next parts program line is delayed until the block containing the M function defined in this MD has finished running (acknowledgment from PLC, motion, etc.).		
Application example(s)	See Section 2.2		

11100 MD number	**AUXFU_MAXNUM_GROUP_ASSIGN** Number of auxiliary functions distributed among the AUXFU groups		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 50 /64 (SW 5.2)	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1 or 5.2		
Significance:	Maximum number of auxiliary functions, which can be assigned to one group using AUXFU_ASSIGN_TYPE, AUXFU_ASSIGN_EXTENSION, AUXFU_ASSIGN_VALUE and AUXFU_ASSIGN_GROUP. This number only includes the user-defined auxiliary functions (see 2.3) and not the predefined auxiliary functions (see Section 2.2).		
Application example(s)	See Chapter 6		
Related to	MD 22010: AUXFU_ASSIGN_TYPE[n]		

© Siemens AG, 2003. All rights reserved

11110 MD number	**AUXFU_GROUP_SPEC[n],** Auxiliary function group specification		
Default setting: see below	Minimum input limit: –	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2/7	Unit: HEX	
Data type: BYTE	Applies from SW 1.1		
Significance:	This MD defines the output modes for the auxiliary functions belonging to a group. Coding: Bit 0=1—> Output duration 1 OB1 pass (normal auxiliary function) Bit 1=1—> Output duration 1 OB40 pass, alarm-controlled (high-speed auxiliary function) Bit 2=1—> reserved Bit 3=1—> No output to PLC (may only be set as the only bit) Bit 4=1—> Spindle response after acknowledgment by the PLC Bit 5=1—> Output before the motion Bit 6=1—> Output during the motion Bit 7=1—> Output at the end of the block The assignment of the individual aux. functions is defined in the channel-specific machine data (AUXFU_ASSIGN_TYPE, AUXFU_ASSIGN_EXTENSION, AUXFU_ASSIGN_VALUE, AUXFU_ASSIGN_GROUP). M0, M1, M2, M17 and M30 are assigned as standard to group 1. The specification of this group (0x81: output duration 1 OB1 pass, output at end of block) may not be changed. All spindle-specific auxiliary functions (M3, M4, M5, M19, M70) are assigned as standard to group 2. If several auxiliary functions with different output types (prior, during, at end of motion) are programmed in one motion block, then they are output individually according to their output type. In a block without motion, all auxiliary functions are output at the same time. The MD must be defined for every existing auxiliary function group. The index [n] corresponds to the number of the function group: 0...14 or 0...63 SW 5.2 and higher [0] ≙ 1st auxiliary function group, [1] ≙ 2nd auxiliary function group ... Default setting: AUXFU_GROUP_SPEC0]=81H AUXFU_GROUP_SPEC[1]=21H AUXFU_GROUP_SPEC[2]=41H … AUXFU_GROUP_SPEC[63]=41H		
Application example(s)	See Chapter 6		

© Siemens AG, 2003. All rights reserved

4.2 Channel-specific machine data

20800 MD number	**SPF_END_TO_VDI** Subprogram end to PLC		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Bit 0 = 1: M17 and M2/M30 are output to the VDI interface. Bit 0 = 0: M17 and M2/M30 are not output to the VDI interface. No unnecessary blocks are generated. Path-controlled mode is not interrupted on end of subprogram. Bit 1 = 0: M01:Checkpoint program stop is always output to PLC, irrespective of whether signal M01 is active or not. High-speed auxiliary function output M=QU(1) is not active as M01 is assigned to the 1stspecial function group and is thus always output at the end of the block. Bit 1 = 1: M01 special function: Conditional program stop is only output to the PLC if M01 is also active. This enables the execution of the parts program to be optimized in terms of runtime. With high-speed auxiliary function output M=QU(1) M1 is output during movement and thus allows operation of blocks with programmed M01 in continuous-path mode as long as M01 is not active. The M01 signal is no longer queried at the end of the block on M=QU(1); rather, during motion. Example of a subprogram: G64 F2000 G91 Y10 X10 X10 Z10 M17		
Application example(s)	• Continuous-path mode remains active with a setting of SPF_END_TO_VDI=0. • Drops in feedrate may occur in continuous-path mode with a setting of SPF_END_TO_VDI=1.		

22000 MD number	**AUXFU_ASSIGN_GROUP[n]** Auxiliary function group [aux. func. no. in channel]: 0...49 or 63 (SW 5.2 and higher)		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 15	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	See MD: AUXFU_ASSIGN_TYPE [n] (auxiliary function type)		
Application example(s)	See Chapter 6		

© Siemens AG, 2003. All rights reserved

22010 MD number	**AUXFU_ASSIGN_TYPE[n]** Auxiliary function type [aux. func. no. in channel]: 0...49 or 63 (SW 5.2 and higher)		
Default setting: –	Minimum input limit: –	Maximum input limit: 16 characters	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: STRING	Applies from SW 1.1		
Significance:	Machine data AUXFU_ASSIGN_TYPE[n] (auxiliary function type), AUXFU_ASSIGN_EXTENSION[n] (auxiliary function extension), AUXFU_ASSIGN_VALUE[n] (auxiliary function value) and AUXFU_ASSIGN_GROUP[n] (auxiliary function group) are defined to assign an auxiliary function type (M,S,H,T,D), the associated extension and the auxiliary function value to an auxiliary function group. Example: M 0 = 100 => Group 5 (corr. to M100) Type of auxiliary function extension Auxiliary function extension Auxiliary function value Auxiliary function group ⇒ MD: AUXFU_ASSIGN_TYPE[0] = “M” MD: AUXFU_ASSIGN_EXTENSION[0] = 0 MD: AUXFU_ASSIGN_VALUE[0] = 100 MD: AUXFU_ASSIGN_GROUP[0] = 5; (≒ 5th group) Auxiliary functions M00, M01, M02, M17 and M30 are assigned as standard to group 1. M3, M4, M5 and M70 of the master spindle are assigned as standard to group 2. The S functions of the master spindle are assigned as standard to group 3. When assigning an auxiliary function to a group, the set synchronization with respect to the PLC interface and a simultaneously programmed motion can be found in MD: AUXFU_GROUP_SPEC [n] (auxiliary function group specification). The settings defined in machine data MD: AUXFU_[M,S,H,T,D,F]_SYNC_TYPE (output timing for [M,S,H,T,D,F] functions) are ignored for the selected auxiliary functions. A programmed high-speed auxiliary function (e.g. M=QU(100)) is also ignored. The machine data index [n] indicates the auxiliary function number All auxiliary functions that are assigned to auxiliary function groups should be numbered in ascending order. [0] ≒ 1st auxiliary function [1] ≒ 2nd ,, . . The same index [n] must be set for each of the machine data used to assign an auxiliary function to an auxiliary function group. **Note:** It is not possible to assign the type DL.		
Application example(s)	See Chapter 6		
Special cases, errors, ...	If the auxiliary function value of an auxiliary function is less than 0, then all auxiliary functions of this type and extension are assigned to a group. Example: S2 = –1 => Group 9 (all S values of the 2nd spindle are assigned to group 9)		
Related to	MD 11100: AUXFU_MAXNUM_GROUP_ASSIGN		

© Siemens AG, 2003. All rights reserved

22020 MD number	**AUXFU_ASSIGN_EXTENSION[n]** Auxiliary function extension [aux. func. no. in channel]: 0...49 or 63 (SW 5.2 and higher)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 99	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	See MD: AUXFU_ASSIGN_TYPE [n] (auxiliary function type)		
Application example(s)	See Chapter 6		
Special cases, errors, ...	The auxiliary function extensions 1 to 4 are reserved for spindle functions with S and M functions.		

22030 MD number	**AUXFU_ASSIGN_VALUE[n]** Auxiliary function value [aux. func. no. in channel]: 0...49 or 63 (SW 5.2 and higher)		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	See MD: AUXFU_ASSIGN_TYPE [n] (auxiliary function type)		
Application example(s)	See Chapter 6		

22035 MD number	**AUXFU_ASSIGN_SPEC[n]** Output specification[AuxFNo. in channel] 0...49 or 63 (SW 5.2 and higher)	
Default setting: 0	Minimum input limit: 0	Maximum input limit: –
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –
Data type: DWORD	Applies from SW 6.4	
Significance:	Specifies the output option for the userdefined auxiliary functions If the bit is set, this indicates: 0: “Normal” acknowledgment following an OB1 cycle 1: “Quick” acknowledgment with OB40 3: No output to NCK PLC interface 4: Spindle response following acknowledgment 5: Output prior to motion 6: Output during motion 7: Output at block end	
Related to	MD 11110: AUXFU_GROUP_SPEC MD 22000: AUXFU_ASSIGN_GROUP MD 22010: AUXFU_ASSIGN_TYPE MD 22020: AUXFU_ASSIGN_EXTENSION MD 22030: AUXFU_ASSIGN_VALUE	

© Siemens AG, 2003. All rights reserved

22040 MD number	**AUXFU_PREDEF_GROUP** Predefined auxiliary function groups		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 64	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: Byte	Applies from SW 6.4		
Significance:	Predefined auxiliary function groups: The predefined auxiliary functions are as follows: M0, M1, M2, M17, M30, M6, M3, M4, M5, M19, M70, M40, M41, M42, M43, M44, M45, S The predefined groups cannot be modified for M0, M1, M2, M17, M30 and M6.		
Related to	MD 22050: AUXFU_PREDEF_TYPE MD 22060: AUXFU_PREDEF_EXTENSION MD 22070: AUXFU_PREDEF_VALUE MD 22080: AUXFU_PREDEF_SPEC		

22050 MD number	**AUXFU_PREDEF_TYPE** Predefined auxiliary function type		
Default setting: "M"	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: STRING	Applies from SW 6.4		
Significance:	The address codes for the predefined auxiliary functions are preset and cannot be changed. The predefined auxiliary functions are as follows: M0, M1, M2, M17, M30, M6, M3, M4, M5, M19, M70, M40, M41, M42, M43, M44, M45, S The setting for these cannot be changed.		
Related to	MD 22040: AUXFU_PREDEF_GROUP MD 22060: AUXFU_PREDEF_EXTENSION MD 22070: AUXFU_PREDEF_VALUE MD 22080: AUXFU_PREDEF_SPEC		

22060 MD number	**AUXFU_PREDEF_EXTENSION** Predefined auxiliary function extension		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 6.4		
Significance:	Predefined auxiliary function extension 1: Address extension permitted 0: No address extension The predefined auxiliary functions are as follows: M0, M1, M2, M17, M30, M6, M3, M4, M5, M19, M70, M40, M41, M42, M43, M44, M45, S The setting for these cannot be changed.		
Related to	MD 22040: AUXFU_PREDEF_GROUP MD 22050: AUXFU_PREDEF_TYPE MD 22070: AUXFU_PREDEF_VALUE MD 22080: AUXFU_PREDEF_SPEC		

© Siemens AG, 2003. All rights reserved

22070 MD number	**AUXFU_PREDEF_VALUE** Predefined auxiliary function value		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Predefined auxiliary function value The predefined auxiliary functions are as follows: M0, M1, M2, M17, M30, M6, M3, M4, M5, M19, M70, M40, M41, M42, M43, M44, M45, S The setting for these cannot be changed.		
Related to	MD 22040: AUXFU_PREDEF_GROUP MD 22050: AUXFU_PREDEF_TYPE MD 22060: AUXFU_PREDEF_EXTENSION MD 22080: AUXFU_PREDEF_SPEC		

22080 MD number	**AUXFU_PREDEF_SPEC** Output specification		
Default setting: 0x81, ...	Minimum input limit: –	Maximum input limit: –	
Change effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Specifies the output option for the predefined auxiliary functions The predefined auxiliary functions are as follows: M0, M1, M2, M17, M30, M6, M3, M4, M5, M19, M70, M40, M41, M42, M43,M44, M45, S The setting for M0, M1, M2, M17, M30 and M6 cannot be modified! Significance of set bit: 0: “Normal” acknowledgment following an OB1 cycle 1: “Quick” acknowledgment with OB40 3: No output to the VDI 4: Spindle response following acknowledgment 5: Output prior to motion 6: Output during motion 7:Output at block end		
Related to	MD 22040: AUXFU_PREDEF_GROUP MD 22050: AUXFU_PREDEF_TYPE MD 22060: AUXFU_PREDEF_EXTENSION MD 22070: AUXFU_PREDEF_VALUE		

22100 MD number	**AUXFU_QUICK_BLOCKCHANGE** Block change without delay		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: INT	Applies from SW 5		
Significance:	Block change is not delayed with high-speed auxiliary functions 0: With high-speed auxiliary function output, the block change is delayed until the acknowledgement is received from the PLC (OB40). 1: With high-speed auxiliary function output to the PLC, the block change is not delayed.		
MD irrelevant for ...	Auxiliary functions with normal acknowledgment		
References	/FBSY/, Synchronized Actions		

© Siemens AG, 2003. All rights reserved

22110 MD number	**AUXFU_H_TYPE_INT** Type of H auxiliary functions		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: INT	Applies from SW 5		
Significance:	0: The values of H auxiliary functions are presented in floating point format. The maximum value range is +/–3.4028 ex 38. 1: The value of H auxiliary functions is rounded and converted to integer format. The basic program in the PLC must interpret the value as an integer. The value range is –2147483648 to 2147483647. With SW 5.2 and higher, the H function can be entered with ten decades. MD 22110 has the value 0 (compatible with the old software version). Up to 5 decades, the result is identical, and if more than five decades are programmed, the response is different. The function is represented as an exponential function.		

22200 MD number	**AUXFU_M_SYNC_TYPE** Output timing of M functions		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Synchronization of M auxiliary functions with respect to a simultaneously programmed axis motion. 0 = Output before the movement 1 = Output during the movement (the movement may have not yet begun depending on the time of the PLC fetch) 2 = Output at the end of the block 3 = No output to the PLC (hence no block change delay) Caution: The group assignment has a higher priority!		
Application example(s)	All M functions must be output to the PLC before the motion commences ⇒MD: AUXFU_M_SYNC_TYPE=0		

22210 MD number	**AUXFU_S_SYNC_TYPE** Output timing of S functions		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Synchronization of S auxiliary functions with respect to a simultaneously programmed axis motion. 0 = Output before the movement 1 = Output during motion 2 = Output at the end of the block 3 = No output to the PLC (hence no block change delay) Caution: The group assignment has a higher priority!		
Application example(s)	All S functions must be output to the PLC before the motion commences ⇒MD: AUXFU_S_SYNC_TYPE=0		

© Siemens AG, 2003. All rights reserved

22220 MD number	**AUXFU_T_SYNC_TYPE** Output timing of T functions		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Synchronization of T auxiliary functions with respect to a simultaneously programmed axis motion. 0 = Output before the movement 1 = Output during motion 2 = Output at the end of the block 3 = No output to the PLC (hence no block change delay) Caution: The group assignment has a higher priority!		
Application example(s)	All T functions must be output to the PLC before the motion commences ⇒MD: AUXFU_T_SYNC_TYPE=0		

© Siemens AG, 2003. All rights reserved

22230 MD number	**AUXFU_H_SYNC_TYPE** Output timing of H functions		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Synchronization of H auxiliary functions with respect to a simultaneously programmed axis motion. 0 = Output before the movement 1 = Output during motion 2 = Output at the end of the block 3 = No output to the PLC (hence no block change delay) Caution: The group assignment has a higher priority!		
Application example(s)	All H functions must be output to the PLC while the axis/spindle is moving ⇒MD: AUXFU_H_SYNC_TYPE=1		

22240 MD number	**AUXFU_F_SYNC_TYPE** Output timing of F functions		
Default setting: 3	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Synchronization of F auxiliary functions with respect to a simultaneously programmed axis motion. 0 = Output before the movement 1 = Output during motion 2 = Output at the end of the block 3 = No output to the PLC (hence no block change delay) Caution: The group assignment has a higher priority!		
Application example(s)	All F functions must be output to the PLC at the end of the block. ⇒MD: AUXFU_F_SYNC_TYPE=2		

22250 MD number	**AUXFU_D_SYNC_TYPE** Output timing of D functions		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Synchronization of D auxiliary functions with respect to a simultaneously programmed axis motion. 0 = Output before the movement 1 = Output during motion 2 = Output at the end of the block 3 = No output to the PLC (hence no block change delay) Caution: The group assignment has a higher priority!		
Application example(s)	None of the D functions must be output to the PLC. ⇒MD: AUXFU_D_SYNC_TYPE=3		

© Siemens AG, 2003. All rights reserved

22252 MD number	**AUXFU_DL_SYNC_TYPE** Output time of DL functions		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 5.2		
Significance:	Synchronization of D auxiliary functions with respect to a simultaneously programmed axis motion. 0 = Output before the movement 1 = Output during motion 2 = Output at the end of the block 3 = No output to the PLC (hence no block change delay) Caution: The synchronization type of the group, to which individual auxiliary functions can be assigned via the configuration, has higher priority!		
Application example(s)	None of the DL functions are to be output to the PLC. ⇒MD: AUXFU_DL_SYNC_TYPE=3		

22254 MD number	**AUXFU_ASSOC_M0_VALUE** Additional M fct. for program stop	
Default setting: –1	Minimum input limit: 6	Maximum input limit: 0x7FFF
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: DWORD	Applies from SW 6.4	
Significance:	This machine data specifies an additional, predefined M function which behaves in the same way as M0. The value of the machine data corresponds to the M auxiliary function number. The MD must not be set to any of the following values: 0, 1, 2, 3, 4, 5, 17, 19, 30, 40, 41, 42, 43, 44, 45, 70 It must also have a different setting to the value in MD 22256. Restriction: see MD 10715: M_NO_FCT_CYCLE	
Related to	MD 22256: AUXFU_ASSOC_M1_VALUE	
References	/FB/, K1, Mode Group, Channel, Program “Operation, Subroutine Call with M/T Functions”	

22256 MD number	**AUXFU_ASSOC_M1_VALUE** Additional M fct. for conditional stop	
Default setting: –1	Minimum input limit: 6	Maximum input limit: 0x7FFF
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: DWORD	Applies from SW 6.4	
Significance:	This machine data specifies an additional, predefined M function which behaves in the same way as M1. The value of the machine data corresponds to the M auxiliary function number. The MD must not be set to any of the following values: 0, 1, 2, 3, 4, 5, 17, 19, 30, 40, 41, 42, 43, 44, 45, 70 It must also have a different setting to the value in MD 22254. Restriction: see MD 10715: M_NO_FCT_CYCLE	
Related to	MD 22254: AUXFU_ASSOC_M0_VALUE	
References	/FB/, K1, Mode Group, Channel, Program “Operation, Subroutine Call with M/T Functions”	

© Siemens AG, 2003. All rights reserved

22300 MD number	**AUXFU_AT_BLOCK_SEARCH_END** Output of auxiliary functions after block search		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1 **to 3.2**		
Significance:	This machine data defines the output mode of auxiliary functions accumulated during a block search. 1: Unconditional output of the accumulated auxiliary functions at the end of a block search. 0: Output of accumulated auxiliary functions after actuation of NC start at the end of a block search.		

■

© Siemens AG, 2003. All rights reserved

5 Signal Descriptions

5.1 Channel-specific signals

5.1.1 Signals to channel

DB 21–30 **DBX30.5** Data block	**Activate associated M01** **Signal from PLC to channel (PLC—> NCK)**		
Edge evaluation: no		Signal(s) updated:	Signal(s) valid from SW: 6.4
Signal state 1 or signal transition 0 ——> 1	PLC signals to NCK that associated M01 (auxiliary function) must be activated.		
Signal state 0 or signal transition 1 ——> 0	Deactivate associated M01 (auxiliary function).		
Related to	DB 21–30, DBX 318.5		

5.1.2 Signals from channel

DB21–30 **DBB58,60–65** **Data block**	**M, S, T, D, H, F functions change** **Signal(s) from channel (NCK –> PLC)**		
Edge evaluation: no		Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	An M, S, T, D, H or F function has been output to the interface with a new value together with the associated change signal at the beginning of an OB1 cycle. In this case, the change signal indicates that the appropriate value is valid.		
Signal state 0 or signal transition 1 ——> 0	The change signals are reset by the basic PLC program at the beginning of the next OB1 cycle. The value of the data concerned is not valid.		

DB21–30 **DBB60–64** **DBB66–67** **Data block**	**M, S, T, D, H, F functions Additional info “Quick” (high-speed acknowledgment)** **Signal(s) from channel (NCK –> PLC)**		
Edge evaluation: no		Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 4.1
Signal state 1 or signal transition 0 ——> 1	An M, S, T, D, H or F function has been output to the interface with a new value together with the associated change signal at the beginning of an OB1 cycle. In this case, the additional info “Quick” indicates that the auxiliary function is programmed for high-speed acknowledgment.		
Signal state 0 or signal transition 1 ——> 0	The change signals are reset by the basic PLC program at the beginning of the next OB1 cycle. The value of the data concerned is not valid.		

© Siemens AG, 2003. All rights reserved

DB21–30 **DBX59.0** **to DBX59.4** **Data block**	**M Fct. 1–5 not included in list** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	M function is higher than 99 (with extended address = 0) or not included in the decoding list with an extended address > 0. This signal together with the associated M change signal is applied to the interface at the beginning of an OB1 cycle. Cause: • Incorrect M function programmed. • M function not configured in the decoding list of the PLC. Remedy, e.g. • PLC sets Read-in disable • Output of a PLC alarm	
Signal state 0 or signal transition 1 ——> 0	M function is lower than 99 (with extended address = 0) or included in the decoding list with an extended address > 0.	

DB21–30 **DBB68–97** **Data block**	**M functions 1 to 5** **Extended address M functions 1 to 5** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Up to 5 M functions programmed in an NC block are made available here simultaneously as soon as the M change signals are applied. Value range of M functions: 0 to 9999 9999; integer Value range of extended address: 0 to 99; integer The M functions remain valid until they are overwritten by new M functions.	
Signal state 0 or signal transition 1 ——> 0	• After PLC power-up. • All auxiliary functions are deleted before a new function is entered.	
Application example(s)	Decoding or evaluation of M functions that are not decoded as standard or via a list. The extended address allows the M function to be assigned to a channel other than the one in which the program is running.	
Special cases, errors, ...	The extended address for M00 to M99 = 0.	

DB21–30 **DBB98–115** **Data block**	**S functions 1 to 3** **Extended address S functions 1 to 3** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW 1.1
Signal state 1 or signal transition 0 ——> 1	Up to 3 S functions programmed in an NC block are made available here simultaneously as soon as the S change signals are applied. Value range of spindle speed: 0 to 999,999; integer Value range of extended address: 0 to 6; integer The S functions remain valid until they are overwritten by new S functions.	
Signal state 0 or signal transition 1 ——> 0	• After PLC power-up. • All auxiliary functions are deleted before a new function is entered.	
Application example(s)	Spindle speed control by the PLC. The extended address is used to program the spindle for which the S word is valid. E.g. S2=500	

© Siemens AG, 2003. All rights reserved

DB21–30 **DBB 119** **Data block**	**T function 1** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The T function programmed in an NC block is made available here as soon as the T change signal is applied. Value range of T function: 0–99,999,999; integer The T function remains valid until it is overwritten by a new T function.	
Signal state 0 or signal transition 1 ——> 0	• After PLC power-up. • All auxiliary functions are deleted before a new function is entered.	
Application example(s)	Control of automatic tool selection.	
Special cases, errors, ...	When T0 is selected, the current tool is removed from the tool holder but not replaced by a new tool (default configuration by machine manufacturer).	

DB21–30 **DBB129** **Data block**	**D function 1** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The D function programmed in an NC block is made available here as soon as the D change signal is applied. Value range of D function: 0–999; integer The D function remains valid until it is overwritten by a new D function.	
Signal state 0 or signal transition 1 ——> 0	• After PLC power-up. • All auxiliary functions are deleted before a new function is entered.	
Application example(s)	Implementation of protective functions.	
Special cases, errors, ...	D0 is reserved for deselecting the current tool offset.	

DB21–30 **DBB140–157** **Data block**	**H functions 1 to 3** **Extended address H functions 1 to 3** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Up to 3 H functions programmed in an NC block are made available here simultaneously as soon as the H change signals are applied. Value range of H function: Floating point (according to MC5+format) Value range of extended address: 0 to 99; integer The H functions remain valid until they are overwritten by new H functions.	
Signal state 0 or signal transition 1 ——> 0	• After PLC power-up. • All auxiliary functions are deleted before a new function is entered.	
Application example(s)	Switching functions on the machine.	

© Siemens AG, 2003. All rights reserved

DB21–30 **DBB158–193** **Data block**	**F functions 1 to 6** **Extended address F functions 1 to 6** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Up to 6 F functions (one path feed and up to 5 axis-specific feeds for positioning axes) are made available here simultaneously as soon as the F change signals are applied. • Value range of F function: Floating point (according to MC5+format) • Value range of extended address: 0 to 18; integer • The extended address of the F function is generated from the feed mode (path feed or axis-specific feed) and the axis name. It is coded as follows: 0: Path feed; e.g.: F=1000 1 to 18: Machine axis number of positioning axis with axis-specific feedrate; e.g. FA[X1]=500 The F functions remain valid until they are overwritten by new F functions.	
Signal state 0 or signal transition 1 ——> 0	• After PLC powerup. • All auxiliary functions are deleted before a new function is entered.	
Application example(s)	Control of programmed F word by the PLC, e.g. through overwriting of the set feedrate override.	
Related to	MD: AUXFU_F_SYNC_TYPE Output time of the F functions	

DB21–30 **DBB194–206** **Data block**	**Dynamic M functions: M0 – M99** **Signal(s) from channel (NCK –> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled by NCK	Signal(s) valid from SW 1.1
Signal state 1 or signal transition 0 ——> 1	The dynamic M signal bits are set by decoded M functions.	
Signal state 0 or signal transition 1 ——> 0	With a general auxiliary function output, dynamic M signal bits are acknowledged by the basic PLC program once the complete OB1 routine has been executed once. In the case of high-speed auxiliary function outputs, they are acknowledged within the same OB40 cycle once the PLC has recognized the auxiliary functions.	
Application example(s)	Spindle clockwise/counterclockwise rotation, switch coolant ON/OFF	

DB 21–30 **DBX318.5** **Data block**	**Associated M01/M00 active** **Signal from channel (NCK ——>PLC)**	
Edge evaluation: no	Signal(s) updated:	Signal(s) valid from SW: 6.4
Signal state 1 or signal transition 0 ——> 1	This bit indicates that an M00 or M01 auxiliary function is active if these associated (auxiliary) functions have been enabled/activated beforehand.	
Signal state 0 or signal transition 1 ——> 0	No associated M00/M01 auxiliary functions active.	
Related to	DB 21–30, DBX30.5	

© Siemens AG, 2003. All rights reserved

5.2 Axis-specific signals

DB31–61 **DBD 78** **Data block**	**Value of F auxiliary function** Signal(s) **from axis/spindle (NCK ——> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled	Signal(s) valid from SW: 1.1
	The values of the F auxiliary functions for positioning axes are stored here. The axis to which each value applies is determined by the extended address.	

DB31–61 **DBD86** **Data block**	**Value of M auxiliary function** Signal(s) **from axis/spindle (NCK ——> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled	Signal(s) valid from SW: 1.1
	The values for the M3, M4 and M5 auxiliary functions are sent to the associated interface for the addressed spindle.	

DB31–61 **DBD88** **Data block**	**Value of S auxiliary function** Signal(s) **from axis/spindle (NCK ——> PLC)**	
Edge evaluation: no	Signal(s) updated: Job-controlled	Signal(s) valid from SW: 1.1
	The values for the S auxiliary functions are sent to the associated interface for the addressed spindle.	

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

6 Example

Machine configuration

The machine data of the necessary auxiliary functions must be defined for a machine tool with the following configuration:

- Machine tool with 2 spindles
  Spindle 1 = master spindle
  Spindle 2 = second spindle
- Gear stages
  Stages 1–5 for master spindle
- Cooling water
  No. 1 “On” = M50
  No. 1 “Off” = M51
  No. 2 “On” = M52
  No. 2 “Off” = M53

Requirements and sensible assignments

Spindle 1 (master spindle):

- M3, M4, M5, M70 and M1=3, M1=4, M1=5, M1=70 are assigned as standard to auxiliary function group 2.
- All S and SI values of the master spindle are assigned as standard to group 3.
- Requirement: The last gear stage to be programmed must be output after the block search.
  ⇒ M40, M41, M42, M43, M44, M45 and M1=40, M1=41, M1=42, M1=43, M1=44, M1=45 are assigned, for example, to group 9.

Spindle 2:

- Requirement: Only one M function for directional reversal may be programmed in one block. The last rotational direction to be programmed must be output after the block search.
  ⇒ M2=3, M2=4, M2=5, M2=70 are assigned to a group, e.g. to group 10.
- Requirement: All S2 values must be assigned to a group, e.g. to group 11.

Cooling water:

- Requirement: It is not permissible to switch the cooling water on and off in one block. The cooling water must be switched on or off after the block search.
  ⇒ Assign M50 and M51 to a group, e.g. to group 12.
  ⇒ Assign M52 and M53 to a group, e.g. to group 13.

© Siemens AG, 2003. All rights reserved

Output of auxiliary functions

The output period for auxiliary functions and the method by which they are synchronized with a programmed motion are determined by the following requirements:

Spindle 1 (master spindle):

- The reversal in rotational direction of the master spindle is an internal function.
  ⇒ M3, M4, M5, M70 and M1=3, M1=4, M1=5, M1=70 (group 2) are output as high-speed functions prior to the motion.
  ⇒ S and S1 values (group 3) are output as high-speed auxiliary functions prior to the motion.
- Gear changes are controlled by the PLC.
  ⇒ M40 to M45, M1=40 to M1=45 (group 9) are output as normal auxiliary functions prior to the motion.

Spindle 2:

- Reversal of the rotational direction of the 2nd spindle is an internal function.
  ⇒ M2=3, M2=4, M2=5, M2=70 (group 10) are output as high-speed functions prior to the motion.
  ⇒ S2 values (group 11) are output as high-speed auxiliary functions prior to the motion.

Cooling water:

- The cooling water is switched on and off by the PLC.
  ⇒ M50, M51 (group 12) and M52, M53 (group 13) are output as normal auxiliary functions prior to the motion.

Parts program for configuring

General:
A total of 21 auxiliary functions are distributed among all groups.

```
$MN_AUXFU_MAXNUM_GROUP_ASSIGN=21
```

Spindle 1:
Specification of group 2 deviating from defaults:

```
$MN_AUXFU_GROUP_SPEC[1]='H22'; (coding as hex value)
```

Specification of group 3 deviating from defaults:

```
$MN_AUXFU_GROUP_SPEC[2]='H22'; (coding as hex value)
```

Specification of group 9:

```
$MN_AUXFU_GROUP_SPEC[8]='H21'; (coding as hex value)
```

© Siemens AG, 2003. All rights reserved

Description of 1st auxiliary function: M40 ... Description of 6th auxiliary function: M45

```
$MC_AUXFU_ASSIGN_TYPE[0]="M"
$MC_AUXFU_ASSIGN_EXTENSION[0]=0
$MC_AUXFU_ASSIGN_VALUE[0]=40
$MC_AUXFU_ASSIGN_GROUP[0]=9
                :
$MC_AUXFU_ASSIGN_TYPE[5]="M"
$MC_AUXFU_ASSIGN_EXTENSION[5]=0
$MC_AUXFU_ASSIGN_VALUE[5]=45
$MC_AUXFU_ASSIGN_GROUP[5]=9
```

Description of 7th auxiliary function: M1=40 ... Description of 12th auxiliary function: M1=45

```
$MC_AUXFU_ASSIGN_TYPE[6]="M"
$MC_AUXFU_ASSIGN_EXTENSION[6]=1
$MC_AUXFU_ASSIGN_VALUE[6]=40
$MC_AUXFU_ASSIGN_GROUP[6]=9
                :
$MC_AUXFU_ASSIGN_TYPE[11]="M"
$MC_AUXFU_ASSIGN_EXTENSION[11]=1
$MC_AUXFU_ASSIGN_VALUE[11]=45
$MC_AUXFU_ASSIGN_GROUP[11]=9
```

Spindle 2: Specification of group 10:

```
$MN_AUXFU_GROUP_SPEC[9]='H22'; (coding as hex value)
```

Description of 13th auxiliary function: M2=3

```
$MC_AUXFU_ASSIGN_TYPE[12]="M"
$MC_AUXFU_ASSIGN_EXTENSION[12]=2
$MC_AUXFU_ASSIGN_VALUE[12]=3
$MC_AUXFU_ASSIGN_GROUP[12]=10
```

Description of 14th auxiliary function: M2=4

```
$MC_AUXFU_ASSIGN_TYPE[13]="M"
$MC_AUXFU_ASSIGN_EXTENSION[13]=2
$MC_AUXFU_ASSIGN_VALUE[13]=4
$MC_AUXFU_ASSIGN_GROUP[13]=10
```

Description of 15th auxiliary function: M2=5

```
$MC_AUXFU_ASSIGN_TYPE[14]="M"
$MC_AUXFU_ASSIGN_EXTENSION[14]=2
$MC_AUXFU_ASSIGN_VALUE[14]=5
$MC_AUXFU_ASSIGN_GROUP[14]=10
```

Description of 16th auxiliary function: M2=70

```
$MC_AUXFU_ASSIGN_TYPE[15]="M"
$MC_AUXFU_ASSIGN_EXTENSION[15]=2
$MC_AUXFU_ASSIGN_VALUE[15]=70
$MC_AUXFU_ASSIGN_GROUP[15]=10
```

© Siemens AG, 2003. All rights reserved

Specification of group 11:

$MN_AUXFU_GROUP_SPEC[10]='H22'; (coding as hex value)

Description of 17th auxiliary function: S2= ....

$MC_AUXFU_ASSIGN_TYPE[16]="S"
$MC_AUXFU_ASSIGN_EXTENSION[16]=2
$MC_AUXFU_ASSIGN_VALUE[16]=–1
$MC_AUXFU_ASSIGN_GROUP[16]=11

Cooling water: Specification of group 12:

$MN_AUXFU_GROUP_SPEC[11]='H21'; (coding as hex value)

Description of 18th auxiliary function: M50

$MC_AUXFU_ASSIGN_TYPE[17]="M"
$MC_AUXFU_ASSIGN_EXTENSION[17]=0
$MC_AUXFU_ASSIGN_VALUE[17]=50
$MC_AUXFU_ASSIGN_GROUP[17]=12

Description of 19th auxiliary function: M51

$MC_AUXFU_ASSIGN_TYPE[18]="M"
$MC_AUXFU_ASSIGN_EXTENSION[18]=0
$MC_AUXFU_ASSIGN_VALUE[18]=51
$MC_AUXFU_ASSIGN_GROUP[18]=12

Specification of group 13:

$MN_AUXFU_GROUP_SPEC[12]='H21'; (coding as hex value)

Description of 20th auxiliary function: M52

$MC_AUXFU_ASSIGN_TYPE[19]="M"
$MC_AUXFU_ASSIGN_EXTENSION[19]=0
$MC_AUXFU_ASSIGN_VALUE[19]=52
$MC_AUXFU_ASSIGN_GROUP[19]=13

Description of 21th auxiliary function: M53

$MC_AUXFU_ASSIGN_TYPE[20]="M"
$MC_AUXFU_ASSIGN_EXTENSION[20]=0
$MC_AUXFU_ASSIGN_VALUE[20]=53
$MC_AUXFU_ASSIGN_GROUP[20]=13

■

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

7

7.1 Interface signals

DB number	Byte.Bit	Name	Refer-ence
Channel-specific			
21–30	30.5	Activate associated M01	
21–30	58.0	M function 1 change	
21–30	58.1	M function 2 change	
21–30	58.2	M function 3 change	
21–30	58.3	M function 4 change	
21–30	58.4	M function 5 change	
21–30	59.0	M function 1 not included in list	
21–30	59.1	M function 2 not included in list	
21–30	59.2	M function 3 not included in list	
21–30	59.3	M function 4 not included in list	
21–30	59.4	M function 5 not included in list	
21–30	60.0	S function 1 change	
21–30	60.1	S function 2 change	
21–30	60.2	S function 3 change	
21–30	60.4	S function 1 quick	
21–30	605	S function 2 quick	
21–30	60.6	S function 3 quick	
21–30	61.0	T function 1 change	
21–30	61.1	T function 2 change	
21–30	61.2	T function 3 change	
21–30	61.4	T function 1 quick	
21–30	61.5	T function 2 quick	
21–30	61.6	T function 3 quick	
21–30	62.0	D function 1 change	
21–30	62.1	D function 2 change	
21–30	62.2	D function 3 change	
21–30	62.4	D function 1 quick	
21–30	62.5	D function 2 quick	
21–30	62.6	D function 3 quick	
21–30	63.0	E function 1 change	

© Siemens AG, 2003. All rights reserved

DB number	Byte.Bit	Name	Refer-ence
Channel-specific			
21–30	63.4	E function 1 quick	
21–30	64.0	H function 1 change	
21–30	64.1	H function 2 change	
21–30	64.2	H function 3 change	
21–30	64.4	H function 1 quick	
21–30	64.5	H function 2 quick	
21–30	64.6	H function 3 quick	
21–30	65.0	F function 1 change	
21–30	65.1	F function 2 change	
21–30	65.2	F function 3 change	
21–30	65.3	F function 4 change	
21–30	65.4	F function 5 change	
21–30	65.5	F function 6 change	
21–30	66.0	M function 1 quick	
21–30	66.1	M function 2 quick	
21–30	66.2	M function 3 quick	
21–30	66.3	M function 4 quick	
21–30	66.4	M function 5 quick	
21–30	67.0	F function 1 quick	
21–30	67.1	F function 2 quick	
21–30	67.2	F function 3 quick	
21–30	67.3	F function 4 quick	
21–30	67.4	F function 5 quick	
21–30	67.5	F function 6 quick	
21–30	68–69	Extended address of M function 1 (dual)	
21–30	70–73	M function 1 (DInt)	
21–30	74–75	Extended address of M function 2 (dual)	
21–30	76–79	M function 2 (DInt)	
21–30	80–81	Extended address of M function 3 (dual)	
21–30	82–85	M function 3 (DInt)	
21–30	86–87	Extended address of M function 4 (dual)	
21–30	88–91	M function 4 (DInt)	
21–30	92–93	Extended address of M function 5 (dual)	
21–30	94–97	M function 5 (DInt)	
21–30	98–99	Extended address of S function 1 (dual)	
21–30	100–103	S function 1 (Real)	
21–30	104–105	Extended address of S function 2 (dual)	
21–30	106–109	S function 2 (Real)	
21–30	110–111	Extended address of S function 3 (dual)	
21–30	112–115	S function 3 (Real)	
21–30	116–117	Extended address of T function 1 (dual)	

© Siemens AG, 2003. All rights reserved

DB number	Byte.Bit	Name	Refer-ence
Channel-specific			
21–30	118–119	T function 1 (Int)	
21–30	120–121	Extended address of T function 2 (dual)	
21–30	122–123	T function 2 (Int)	
21–30	124–125	Extended address of T function 3 (dual)	
21–30	126–127	T function 3 (Int)	
21–30	128	Extended address of D function 1 (dual)	
21–30	129	D function 1 (dual)	
21–30	130	Extended address of D function 2 (dual)	
21–30	131	D function 2 (dual)	
21–30	132	Extended address of D function 3 (dual)	
21–30	133	D function 3 (dual)	
21–30	134–135	Extended address of E function (dual)	
21–30	136	E function (REAL)	
21–30	140	Extended address of H function 1 (dual)	
21–30	142–145	H function 1 (Real format)	
21–30	146	Extended address of H function 2 (dual)	
21–30	148–151	H function 2 (Real format)	
21–30	152	Extended address of H function 3 (dual)	
21–30	154–157	H function 3 (Real format)	
21–30	158	Extended address of F function 1 (dual)	
21–30	160–163	F function 1 (Real format)	
21–30	164	Extended address of F function 2 (dual)	
21–30	166–169	F function 2 (Real format)	
21–30	170	Extended address of F function 3 (dual)	
21–30	172–175	F function 3 (Real format)	
21–30	176	Extended address of F function 4 (dual)	
21–30	178–181	F function 4 (Real format)	
21–30	182	Extended address of F function 5 (dual)	
21–30	184–187	F function 5 (Real format)	
21–30	188	Extended address of F function 6 (dual)	
21–30	190–193	F function 6 (Real format)	
21–30	194.0–194.7	Dynamic M function: M00–M07	
21–30	195.0–195.7	Dynamic M function: M08–M15	
21–30	196.0–196.7	Dynamic M function: M16–M23	
21–30	197.0–197.7	Dynamic M function: M24–M31	
21–30	198.0–198.7	Dynamic M function: M32–M39	
21–30	199.0–199.7	Dynamic M function: M40–M47	
21–30	200.0–200.7	Dynamic M function: M48–M55	
21–30	201.0–201.7	Dynamic M function: M56–M63	
21–30	202.0–202.7	Dynamic M function: M64–M71	
21–30	203.0–203.7	Dynamic M function: M72–M79	

© Siemens AG, 2003. All rights reserved

DB number	Byte.Bit	Name	Refer-ence
Channel-specific			
21–30	204.0–204.7	Dynamic M function: M80–M87	
21–30	205.0–205.7	Dynamic M function: M88–M95	
21–30	206.0–206.3	Dynamic M function: M96–M99	
21–30	318.5	Associated M00/M01 active	

DB number		Name	Refer-ence
Axis-specific			
31–61	DBD 78	Value of F function	
31–61	DBD 86	Value of M function	
31–61	DBD 88	Value of S function	

7.2 Machine data

Number	Identifier	Name	Refer-ence
General ($MN_ ...)			
10713	M_NO_FCT_STOPRE	M function with preprocessing stop (≥ SW 6.3)	
10714	M_NO_FCT_EOP	M function for spindle active after RESET	S1
11100	AUXFU_MAXNUM_GROUP_ASSIGN	Number of auxiliary functions distributed among the AUXFU groups	
11110	AUXFU_GROUP_SPEC[n],	Auxiliary function group specification	
Channel-specific ($MC_ ...)			
20110	RESET_MODE_MASK	Def. of control initial setting after part program start	K2
20112	START_MODE_MASK	Definition of control initial setting after power-up and on RESET or at parts program end	K2
20270	CUTTING_EDGE_DEFAULT	Basic setting of tool cutting edge without programming	W1
20800	SPF_END_TO_VDI	Subprogram end to PLC	
22000	AUXFU_ASSIGN_GROUP[n]	Auxiliary function group	
22010	AUXFU_ASSIGN_TYPE[n]	Type of auxiliary function	
22020	AUXFU_ASSIGN_EXTENSION[n]	Auxiliary function extension	
22030	AUXFU_ASSIGN_VALUE[n]	Auxiliary function value	
22035	AUXFU_ASSIGN_SPEC	Output specification (user-def. auxiliary function)	
22040	AUXFU_PREDEF_GROUP	Predefined auxiliary function groups	
22050	AUXFU_PREDEF_TYPE	Predefined auxiliary function type	
22060	AUXFU_PREDEF_EXTENSION	Predefined auxiliary function extension	
22070	AUXFU_PREDEF_VALUE	Predefined auxiliary function value	
22080	AUXFU_PREDEF_SPEC	Output specification (predefined auxiliary function)	
22100	AUXFU_QUICK_BLOCKCHANGE	Block change without delay	
22110	AUXFU_H_TYPE_INT	Type of H auxiliary functions	

© Siemens AG, 2003. All rights reserved

Channel-specific ($MC_ ...)			
22200	AUXFU_M_SYNC_TYPE	Output time for M functions	
22210	AUXFU_S_SYNC_TYPE	Output timing of S functions	
22220	AUXFU_T_SYNC_TYPE	Output timing of T functions	
22230	AUXFU_H_SYNC_TYPE	Output time for H functions	
22240	AUXFU_F_SYNC_TYPE	Output timing of F functions	
22250	AUXFU_D_SYNC_TYPE	Output timing of D functions	
22252	AUXFU_DL_SYNC_TYPE	Output time of DL functions	
22254	AUXFU_ASSOC_M0_VALUE	Additional M function for program stop	
22256	AUXFU_ASSOC_M1_VALUE	Additional M function for conditional stop	
22300	AUXFU_AT_BLOCK_SEARCH_END	Output of auxiliary functions after block search (not relevant with SW 4 and higher)	

7.3 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Mode Group, Channel, Program Operation, Reset Response (K1)

1	**Brief Description**		**1/K1/1-5**
2	**Detailed Description**		**1/K1/2-9**
	2.1	Mode group	1/K1/2-9
	2.1.1	Mode group stop	1/K1/2-11
	2.1.2	Mode group reset	1/K1/2-12
	2.2	Channel	1/K1/2-13
	2.2.1	Channel-dependent technological applications	1/K1/2-15
	2.3	Mode groups	1/K1/2-16
	2.3.1	Monitoring functions and interlocks of the individual modes	1/K1/2-18
	2.3.2	Mode change	1/K1/2-19
	2.4	Program test	1/K1/2-20
	2.4.1	Program execution without setpoint output	1/K1/2-20
	2.4.2	Program execution in single block mode	1/K1/2-22
	2.4.3	Program execution with dry run feedrate	1/K1/2-24
	2.4.4	Skip part program blocks	1/K1/2-25
	2.5	Block search	1/K1/2-26
	2.5.1	Block search functions	1/K1/2-27
	2.5.2	Sequence for block searches of type 1, 2 and 4 (SW 4.3 and later)	1/K1/2-28
	2.5.3	Automatic start of an ASUB following block search (SW 6.1 and higher)	1/K1/2-31
	2.5.4	Cascaded block search (SW 6.1 and higher)	1/K1/2-31
	2.5.5	Examples of block search with calculation	1/K1/2-33
	2.5.6	Block search in Program Test mode, SERUPRO	1/K1/2-36
	2.5.7	Stop at any point in the program for SERUPRO (SW 6.4 and higher)	1/K1/2-41
	2.5.8	Multiple starts with SERUPRO in different channels	1/K1/2-42
	2.5.9	Carrying out SERUPRO in a group of channels (SW 6.4 and higher)	1/K1/2-44
	2.5.10	SERUPRO sequence without defining a search target beforehand	1/K1/2-46
	2.5.11	System variables updated by the SERUPRO sequence (SW 6.1 and higher)	1/K1/2-47

© Siemens AG, 2003. All rights reserved

2.5.12	NC functions supported in SERUPRO (SW 6.3 and higher)	1/K1/2-50
2.5.14	SERUPRO ASUB (SW 6.3 and higher)	1/K1/2-58
2.5.15	Continue machining after SERUPRO search target found	1/K1/2-61
2.5.16	Reapproach contour with controlled REPOS (SW 6.3 and higher)	1/K1/2-67
2.6	Program operation mode	1/K1/2-69
2.6.1	Initial settings	1/K1/2-69
2.6.2	Part program selection	1/K1/2-70
2.6.3	Start of part program or part program block	1/K1/2-70
2.6.4	Parts program interruption	1/K1/2-71
2.6.5	RESET command	1/K1/2-73
2.6.6	Program status	1/K1/2-73
2.6.7	Channel status	1/K1/2-75
2.6.8	Response to operator and program actions	1/K1/2-76
2.6.9	Start part program	1/K1/2-77
2.6.10	Time diagram example for a program run	1/K1/2-78
2.6.11	Repeating program sections (SW 4 and higher)	1/K1/2-79
2.6.12	Event-driven program calls (SW 6.1 and higher)	1/K1/2-80
2.6.13	Asynchronous subroutines (ASUBs), interrupt routines	1/K1/2-86
2.6.14	Calling the ASUB outside program operation (SW 4 and higher)	1/K1/2-90
2.6.15	User-defined system ASUBs (SW 4 and higher)	1/K1/2-94
2.7	Block	1/K1/2-97
2.7.1	Decoding single block SBL2 with implicit preprocessing stop (SW 6.3 and higher)	1/K1/2-97
2.7.2	Single block suppression via SBLOF (SW 4.2 and higher)	1/K1/2-98
2.7.3	Preventing single block suppression (SW 4.2 and higher)	1/K1/2-100
2.8	Program control	1/K1/2-101
2.8.1	Control via the operator interface and the PLC	1/K1/2-101
2.8.2	Multiple skip levels (SW 5 and later)	1/K1/2-102
2.8.3	Use of a reduced interpolation buffer	1/K1/2-103
2.8.4	Basic block display (SW 6.4 and higher)	1/K1/2-104
2.8.5	Execution from external source	1/K1/2-109
2.8.6	Execute external subroutine (SW 4 and higher)	1/K1/2-110
2.9	Reset response	1/K1/2-113
2.9.1	System settings for power-up, RESET/end of parts program, and parts program start	1/K1/2-113
2.10	Subroutine call with M and T functions	1/K1/2-121
2.10.1	M function replacement	1/K1/2-122
2.10.2	T function replacement	1/K1/2-124
2.10.3	Examples for M/T function replacement for tool change	1/K1/2-125
2.11	Program runtime/workpiece counter (SW 5.2 and higher)	1/K1/2-127
2.11.1	Program runtime	1/K1/2-127
2.11.2	Workpiece counter	1/K1/2-129
3	**Supplementary Conditions**	**1/K1/4-131**
4	**Data Descriptions (MD, SD)**	**1/K1/4-131**
4.1	Machine data for operator panel	1/K1/4-131
4.2	General machine data	1/K1/4-133
4.3	Machine data for processing programs from external (SW 4 and later)	1/K1/4-143

© Siemens AG, 2003. All rights reserved

4.4 Channel-specific machine data 1/K1/4-145
4.4.1 Block search 1/K1/4-150
4.4.2 ASUB 1/K1/4-153
4.4.3 Reset and starting response 1/K1/4-154
4.4.4 Basic block display 1/K1/4-164
4.5 Axis/spindle-specific machine data 1/K1/4-165
4.6 Channel-specific setting data 1/K1/4-166
4.7 Setting data for processing programs from external (SW 4 and later) 1/K1/4-167
4.8 Setting data: Enable basic block display (SW 6.4 and later) 1/K1/4-167

5 Signal Descriptions 1/K1/5-169

5.1 Mode group-specific signals 1/K1/5-169
5.2 Channel-specific signals 1/K1/5-175
5.3 Axis-specific signals 1/K1/5-192

6 Example 1/K1/7-195

7 Data Fields, Lists 1/K1/7-195

7.1 General machine data 1/K1/7-195
7.1.1 General machine data for operator panel 1/K1/7-195
7.1.2 General machine data mode group/program operation 1/K1/7-196
7.2 Channel-specific machine data 1/K1/7-197
7.2.1 Basic machine data of channel 1/K1/7-197
7.2.2 Block search 1/K1/7-198
7.2.3 Reset response 1/K1/7-198
7.2.4 Auxiliary function settings of the channel 1/K1/7-199
7.2.5 Transformation definitions in channel 1/K1/7-200
7.2.6 Channel-specific memory settings 1/K1/7-201
7.2.7 Program runtime and workpiece counter (SW 5.2 and higher) 1/K1/7-202
7.3 Axis/spindle-specific machine data 1/K1/7-202
7.4 Channel-specific setting data 1/K1/7-203
7.5 Interface signals 1/K1/7-204
7.5.1 Analog inputs and outputs 1/K1/7-204
7.5.2 DB11, ... Mode group signals 1/K1/7-204
7.5.3 DB21, ... Channel signals 1/K1/7-205
7.5.4 DB 31, ... Axis-specific signals 1/K1/7-207
7.6 Alarms 1/K1/7-207

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

1 Brief Description

Mode group

A mode group is a collection of machine axes, spindles and channels which are programmed to form a unit.

In principle, a single mode group equates to an independent NC control (with several channels). A mode group:

- is made up of those channels that always have to operate simultaneously in the same mode.
- is either in **AUTOMATIC**, **JOG** or in **MDA** mode

Channel

Every channel has its own program decoding, block preparation and interpolation functions. A channel can process a parts program independently.

Note

One mode group and one channel are configured as standard.

Channel gaps (SW 4 and higher)

When configuring the channels, placeholder channels can be provided in order to keep the machines of a series as uniform as possible. Only the channels which are actually used are then activated.

Program testing

To test or trial run a new parts program, there is a variety of options in the test phase for significantly reducing critical situations:

- Program execution without setpoint outputs
- Program processing in single-block mode
- Program processing with dry run feedrate
- Skip parts program blocks
- Block search with or without calculation.

Block search (SW 4.3 and higher)

Block search provides the following program simulation functions for searching for particular program points:

Type 1	without calculation at the contour
Type 2	with calculation at the contour
Type 4	with calculation at block end point
Type 5	automatic start at selected program point with calculation of all required data from history

© Siemens AG, 2003. All rights reserved

Block search (SW 6.1 and higher)

Automatic start of an ASUB after block search
Cascaded block search
Cross-channel block search in Program Testing mode

Program operation mode

The term Program operation mode refers to the state where parts programs or parts program blocks are processed in AUTOMATIC or MDA mode. Program processing can be modified with PLC interface signals and commands.

Initial settings can be programmed in channel-specific machine data for each channel. These initial settings affect, for example, G groups and auxiliary function output.

A **parts program can be selected** only if the relevant channel is in the Reset state.

Furthermore, all further program runs are handled by PLC interface signals and the corresponding commands.

- Start of parts program or parts program block
- Parts program calculation and program control
- RESET command, program status and channel status
- Response to operator and program actions
- Event-driven program calls (SW 6.1 and higher)

Asynchronous subprograms (ASUB), interrupt routines

SW 4 and higher
Interrupt inputs allow the NC to interrupt the current processing operation in the NC so that it can react to more urgent events in interrupt routines or ASUBs.

Block

With the single block function, the user can process a parts program block by block. There are 3 types of setting for the single block function:

- SLB1 := IPO single block
- SLB2 := Decoding single block.
- SLB3 := Stop in cycle

Basic block display (SW 6.4 and higher)

A socalled basic block display showing all blocks which will generate an action on the machine can be called in parallel to the existing block display.

The actually approached end positions are shown as an absolute position. The positional data refer either to the workpiece coordinate system (WCS) or the settable zero system (SZS).

Processing programs from external, extension (SW 4 and higher)

In certain cases, the memory of the NC may be insufficient to machine a complex workpiece. The “Process from external” function (EXTCALL) allows subroutines from an external memory (e.g. from HMI Advanced hard disk) to be called (EXTCALL) and processed.

© Siemens AG, 2003. All rights reserved

Reset response

The control system response can be altered for G codes such as tool length compensation, transformation, coupled axis groupings, tangential follow-up and programmable synchronous spindle functions for certain system settings after

- Power up (power ON),
- Reset/parts program end and
- Parts program start

by means of machine data.

Subroutine call with M and T functions

For some applications it can be advantageous to replace M and/or T functions by a subroutine call. This can be used, for example, to call the tool change routine.

The relevant machine data can be used to define and control subroutines accordingly with M and/or T functions:

Program runtime/ workpiece counter

SW 5.2 and higher
Information on program runtime and workpiece counts is provided in order to support the machine tool operator.

The functions defined for this purpose are **not identical to the functions of tool management** and are intended primarily for NC systems without tool management.
(please refer to Section 2.11 for a detailed description)

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description

2

2.1 Mode group

Definition of a mode group

A mode group combines NC channels with axes and spindles to form a machining unit.

A mode group contains all those channels that always have to operate in the same mode.

Note

The following description is based on one mode group and one channel.

Functions requiring several channels (e.g. axis exchange) are described in

References: /FB/, K5, “Mode Groups, Channels, Axis Exchange”

The configuration of a mode group defines which channels are to be included in the group.

Mode group assignment

A mode group may therefore contain one or several channels. Axes and/or spindles are assigned to a channel. Machine data

MD10010: ASSIGN_CHAN_TO_MODE_GROUP

is parameterized to assign a channel to a mode group. If the same mode group is addressed in several channels, these together form a mode group.

Note

The control system does not know modegroup-specific data. It is, however, also possible to make some channel-specific settings pertaining to the mode group.

Channel-specific assignments

Axes can be assigned to different channels which, in turn, are allocated to different mode groups. The axes can then be swapped between these channels (axis replacement). Axis replacement functions independently of the mode group.

© Siemens AG, 2003. All rights reserved

Machine axes and/or spindles are assigned to a channel, with the following different operating characteristics:

- Geometry axes — can be operated within a path grouping. Using the master spindle, they can perform functions such as G96, G961, G331, G332 etc.
- Channel axes — that are not defined as geometry axes can traverse as path axes, synchronous axes, positioning axes, PLC axes and command axes.
- Special axes — have no geometric relationship with one another.
- Master spindle — Geometry axes can perform functions using the master spindle.
- Auxiliary spindles and auxiliary axes — are all other auxiliary axes, in addition to the master spindle spindles/axes in the channel.

The command Geo axis replacement can be programmed to declare a channel axis as a geometry axis and define its geometry axis number. Which spindle in the channel will be the master spindle is defined using SETMS.

With SW 4.4 and higher, it is possible to configure each axis existing in the channel as a spindle. The number of axes per channel depends on the control version. In order to optimize the performance utilization, the available channel configurations and associated axes depend on the hardware.

In the case of the SINUMERIK 840D, the various hardware/software versions allow up to 12 axes/spindles per channel.
Up to 31 axes or up to 20 spindles are allowed per NCU.

For further information about axis configurations, such as axis containers, link, reciprocating, main run, rotary, linear, leading and following axes, and details of the various versions, please refer to:
References: /FB/, K2, “Axes, Coordinate Sys., Frames, Workpieces IWS”
/FB/, S1, “Spindles” , /BU/, “ordering information, Catalog NC 60”

Modegroup-specific interface signals

Modegroup-specific signals to/from the mode group are transferred to DB11 on the user interface. In this way the mode group can be monitored by the PLC or NCK. The following table lists all modegroup-specific signals:

Modegroup signals (PLC => NCK)	Modegroup signals (NCK => PLC)
Mode group reset	Mode group strobe: JOG, MDA, AUTOMATIC
Mode group stop axes plus spindles	Machine function strobe: REF, REPOS, TEACH IN
Mode group stop	All channels (1 to 10 max.) in reset state
Mode change	Mode group ready
Mode: JOG, MDA, AUTOM.	Active mode: JOG, MDA, AUTOMATIC
Single block: Type A, Type B	Digitizing
Machine function: REF, REPOS, TEACH IN,	Active machine function: REF, REPOS, TEACH IN var. INC, 10000 INC ..... 1 INC

A description of all modegroup-specific signals can be found in Chapter 5.

© Siemens AG, 2003. All rights reserved

Modification of mode group

If the configuration of a mode group is modified with respect to its assigned channels, the modification becomes active only after a POWER ON.
Modifications are made in MD 10010: ASSIGN_CHAN_TO_MODE_GROUP.
Mode group numbers must be assigned contiguously starting with 1.

Machine data

There are no modegroup-specific machine data.

Channel gaps

Channels to which a mode group is assigned with MD10010: ASSIGN_CHAN_TO_MODE_GROUP are considered active.
In SW 4 and higher, the number 0 can be assigned to channels instead of a mode group. The result is as follows:

- The non-activated channel does not require memory in the control
- Series machines with similar designs can be kept uniform during configuration. Only the channels which can actually be used by the machine tool are activated.
  (Mode group number greater than 0)

Special case:
Channel 1 must always exist. If ASSIGN_CHAN_TO_MODE_GROUP[0] = 0 is specified, the control automatically sets
ASSIGN_CHAN_TO_MODE_GROUP[0] = 1 (mode group 1).

Example configuration:

```
ASSIGN_CHAN_TO_MODE_GROUP[0] = 1
ASSIGN_CHAN_TO_MODE_GROUP[1] = 2
                ...
ASSIGN_CHAN_TO_MODE_GROUP[3] = 0        ; gap
                ...
ASSIGN_CHAN_TO_MODE_GROUP[8] = 1
ASSIGN_CHAN_TO_MODE_GROUP[9] = 2
```

2.1.1 Mode group stop

Mode group stop (BAG-Stop)

IS “BAG-Stop” (DB11, ... DBX0.5), IS “BAG Stop axes plus spindles” (DB11, ... DBX0.6) can be used to issue a stop signal (comparable to NC stop) simultaneously for each channel of the mode group. Depending on the interface signal used, either only the axes or in addition the spindles of the channels are stopped.

© Siemens AG, 2003. All rights reserved

2.1.2 Mode group reset

Explanation of mode group reset

Activating interface signal IS “Mode group reset” (DB11, ... DBX0.7) effects a channel-specific reset for each channel in the mode group. The channels can be in different program states when the signals is set. For this reason the channels will execute this channel-specific Reset at different times. The channel-specific Resets are not signaled in the channel-specific interface.

A new channel-specific start command cannot be executed until the mode group Reset has been completed. This state is indicated by IS “All channels in reset state” (DB11, ... BX6.7).

Status of mode group

Although the channels within a mode group may be in different states, they are always reset by the mode group Reset command. Once the mode group resets have been executed, all channels of the mode group are in the reset state. A new channel-specific start command cannot be executed until the mode group Reset has been completed.

Execution of command

Activating interface signal IS “Mode group reset” (DB11, ... DBX0.7) aborts all active part programs in the channels of the mode group. The following actions are carried out when IS “Mode group Reset” is triggered:

- Parts program preparation in the individual channels is stopped immediately.
- The axes and spindles of the channels are decelerated along a configurable braking ramp.
  SW 6.2 and higher
  The block change time can be set for various motion end criteria in the braking ramp for single axis interpolation. For further information on block change in the braking ramp, please refer to:
  References: /FB/, P2, “Positioning Axes”, block change
- Any pending auxiliary functions from the individual channels or current block are not longer output.
- To this aim, position the preprocessing pointer on the interruption point; the block pointers of the channels will then be reset to the beginning of the appropriate parts programs.
- All Reset alarms are deleted from the display.
- The mode group reset is complete as soon as IS “All channels in Reset state” (DB11, ... DBX6.7) is set, or on the basic machine of the channel, for example, the following machine data fulfill this Reset condition:
  MD 20110: RESET_MODE_MASK bit 0 is set, or
  MD 20150: GCODE_RESET_VALUES[n] is set to its initial state, or
  MD 20152: GCODE_RESET_MODE[n] is set, or
  SW 6.1 and higher
  MD 20108: PROG_EVENT_MASK the user ASUB is activated for operator panel reset.
  See also Subsection 2.6.12 “Event-driven program calls”.

Mode group Ready

The ready-to-operate status of the mode group is indicated by IS “Mode group Ready” (DB11, ... DBX6.3) displayed.

© Siemens AG, 2003. All rights reserved

2.2 Channel

Axis

A channel (see Chapter 3) constitutes an “NC” in which a parts program can be executed at a time. Machine axes, geometry axes and positioning axes are assigned to the channels according to the machine configuration and the current program status (AXIS CHANGE, GEO AXIS CHANGE, SETMS).

Each channel is assigned its own path interpolator with program processing by the system. Each channel can run its own machining program which is controlled from the PLC.

The following channel-specific functions make it possible for the channels to process parts programs independently:

- Each channel has its own NC Start, NC Stop, Reset
- One feedrate and rapid traverse override per channel
- Dedicated interpreter for each channel
- Dedicated path interpolator for each channel
  which calculates the path points such that all the machining axes of the channel are controlled simultaneously by the path axes
- Selection and deselection of tool cutting edges and their length and radius compensation for a tool in a specific channel

  Further information about tool compensation
  References: /FB/, W1, “Tool offset”
- Channel-specific frames and frames active in the channel for transforming closed calculation rules into Cartesian coordinate systems. Translations, rotations, scales and mirrors are programmed in frames for geometry axes and special axes.

  Further information about frames
  References: /FB/, K2, “Axes, Coordinate Systems, Frames, Actual Value System for Workpiece, External Zero Offset”
- Display of channel-specific alarm reactions
- Display of current machining state (axis position, current G functions, auxiliary functions, current program block) of each channel
- Separate program control functions for each channel.

These functions (with the exception of the display functions) are controlled and checked by the PLC with interface signals.

Channels in the same mode group always have to be operated in the same mode (AUTOMATIC, JOG, MDA).

Channel configuration

Individual channel names can be assigned to the channels in MD 20000: CHAN_NAME (channel name). The various axes are then assigned to the available channels via machine data**.** For one axis/spindle, only one setpoint-determined channel can exist at a time. The axis/spindle actual value can be read by several channels at the same time. The axis/spindle must be registered with the relevant channel.

© Siemens AG, 2003. All rights reserved

The following channel-specific settings can also be made in the machine data:

- Initial settings or basic programming settings of G groups via MD 20150: GCODE_RESET_VALUES (initial setting of G groups)
- Composition and output time of auxiliary function groups, see Subsection 7.2.2 for relevant MD, Auxiliary function settings of a channel.
- Transformation conditions between machine axes and the geometry axes, see Subsection 7.2.3 for relevant MD, Transformation definitions in the channel.
- For other settings regarding processing of a parts program, see Subsection 7.2.1 for relevant MD, Basic machine data of a channel.

Alteration of channel assignments

The channel configuration cannot be changed online by reprogramming a parts program or altering the PLC user program.
Changes can only be made by via the machine data. These changes only take effect after a POWER ON.

Container axes and link axes

An axis container combines a group of axes in a container. These axes are referred to as container axes. This involves assigning a pointer to a container slot (ring buffer location within the relevant container) to a channel axis. One of the axes in the container is located temporarily in this slot.

Each machine axis in the axis container must be assigned at all times to exactly **one** channel axis.

Link axes can be assigned permanently to one channel or dynamically (by means of an axis container switch) to several channels of the local or other NCU. A link axis is nonlocal from the perspective of one of the channels belonging to the NCU to which the axis is not physically connected. The assignment between the link axes and a channel is implemented

- Via machine data for permanent assignments:
  Program a direct logical machine axis image pointing to the link axes.
- For dynamic assignment:
  Axis container slot program machine data pointing to the link axes.

For further information on link and container axes, please refer to
References: /FB/, B3, “Several Operator Panels and NCUs”

Machine data

The channel-specific machine data are listed in Chapter 7.

Interface signals

NCK channel 1 signals lie in DB21 of the user interface, those of channel 2 in DB22. The channel or channels can be monitored and controlled from by PLC or NCK. The channel signals are described in Chapter 5.

© Siemens AG, 2003. All rights reserved

2.2.1 Channel-dependent technological applications

Technology in channel (SW 4 and higher)

Machine data MD 27800: TECHNOLOGY_MODE. Among other functions, this information is evaluated for HMI, PLC and standard cycles.

Siemens supplies standard machine data for milling. If the machine tool is not a milling machine, but some other type, a different data/program block can be loaded by the HMI or PLC depending on the technology mode set in the machine data.

Spindle functions via PLC with channel acceptance

In SW 6.2 and higher, it is possible to control special spindle motions via an axial PLC interface as an alternative to FC18 and to start and stop them using VDI interface signals without executing a parts program.

This option is available only if the channel status (see Subsection 2.3 under operating modes) is “interrupted” or “Reset” and the program status (see 2.6.6) “interrupted” or “aborted”. This acceptance state will occur in case of Reset and in JOG mode.

A separate spindle start can be set for each spindle. The following spindle functions can be controlled by PLC via interface signals:

- Spindle stop (equivalent to M5)
- Spindle start clockwise (equivalent to M3)
- Spindle start counter-clockwise (equivalent to M4)
- Select gear stage
- Spindle positioning (corresponds to M19)

In the case of multi-channel operation, the spindle started by the PLC becomes active in the channel that is handling the spindle when the start command is received.

For further information about the special spindle interface:
References: /FB/, S1, “Spindles”

Autonomous single-axis operations

In SW 6.3 and higher, it is possible to decouple a particular axis/spindle in the main run from the channel behavior triggered by the NC program run. The PLC identifies the corresponding axis/spindle from the axial VDI signal

IS “PLC controlling axis” (DB31, ... DBB28.7) = 1 Assume control,
IS “PLC controlling axis” (DB31, ... DBB28.7) = 0 Return control

The following functions can be controlled from the PLC:

- Cancel axis/spindle sequence (equivalent to delete distance-to-go)
- Stop or interrupt an axis/spindle
- Resume an axis/spindle operation (continue the sequence of motions)
- Reset an axis/spindle to the initial state.

The exact functionality of independent single-axis operations is described in:
References: /FB/, P2, “Positioning Axes”
For further information about channel-specific signal exchange (PLC→NCK):
References: /FB/, P3, “Basic PLC Program”

© Siemens AG, 2003. All rights reserved

2.3 Mode groups

A mode group operates in one of the modes

AUTOMATIC, **JOG** or **MDA**

Multiple channels

Several channels of the same mode group can **not** be in different modes at the same time. If individual channels are assigned to different mode groups, a channel switchover activates the corresponding mode group. This allows mode changes to be initiated implicitly.

The following operating modes are available:

AUTOMATIC

- Automatic processing of parts programs
- Parts program test
- All channels of the mode group can be active at the same time

JOG

Jogging (manual axis traversal)

- The axes can be traversed manually with the handwheel or the traversing keys.
- Channel-specific signals and interlocks are effective with an active ASUB or when a movement is activated via IDS synchronized actions. Couplings are also effective.
- Every channel in the mode group can be active

MDA (MDI)

Manual Data Automatic (the NC blocks are entered on the operator panel)

- Restricted automatic execution of parts programs and sections of parts programs (can only be a block or a sequence of blocks).
- Parts program test.
- A maximum of 1 channel per mode group can be active (applies only to TEACH IN)
- Axes can be manually traversed only in subordinate machine functions such as JOG, REPOS or TEACH IN.

Valid for all operating modes

Modally active synchronized actions

Modal synchronized actions can be executed per IDS in all modes for the following functions in parallel to the channel:

- Command axis functions
- Spindle functions
- Technology cycles

© Siemens AG, 2003. All rights reserved

Selection

The user can select the desired operating mode by means of soft keys on the operator interface. This selection (AUTOMATIC, MDA or JOG) is passed via interface signals IS “Mode strobe” (DB11, ... DBX4.0 to DBX4.2) to the PLC, but not yet activated.

Activation and priorities

The chosen group operating mode is activated via the signals on interface “Mode” (DB11, ... DBX0.0 to DBX0.2). If several modes are selected at the same time the priority is as follows:

- JOG (1st priority, high) (DB11, ... DBX0.2)
- MDA (2nd priority, medium) (DB11, ... DBX0.1)
- AUTOMATIC (3rd priority, low) (DB11, ... DBX0.0)

Display

The current group operating mode is activated via the signals on interface “Active mode” (DB11, ... DBX6.0 to DBX6.2).

- JOG (DB11, ... DBX6.2)
- MDA (DB11, ... DBX6.1)
- AUTOMATIC (DB11, ... DBX6.0)

Global machine function for mode group

After mode selection, a machine function can be selected which is then valid globally for the whole mode group.

Within JOG mode
One of the following machine functions can be selected:

- REF, (reference point approach),
  References: /FB/, R1, “Reference Point Approach”
- REPOS, (repositioning),
  References: /BA/, “Operator’s Guide”

  Reapproach contour with controlled REPOS, see Subsection 2.5.16.

Within MDA mode
One of the following machine functions can be selected:

- REF, (reference point approach),
  References: /FB/, R1, “Reference Point Approach”
- REPOS, (repositioning),
  References: /BA/, “Operator’s Guide”
- TEACHIN, (teaching axis positions),
  References: /BA/, “Operator’s Guide”

TEACH IN, REPOS or REF

The selection of machine function
TEACH IN, REPOS or REF via the operator interface is stored in
IS “Machine function strobe” (DB11, ... DBX5.0 to DBX5.2).

The chosen machine function TEACH IN, REPOS or REF is activated on interface “Machine function” (DB11, ... DBX1.0 to DBX1.2).

The active machine function TEACH IN, REPOS or REF is displayed on interface “Active machine function” (DB11, ... DBX7.0 to DBX7.2).

© Siemens AG, 2003. All rights reserved

Operating states

The following three channel states can occur in each mode:

1. **Channel reset**
   The machine is in the initial state. This is defined by the machine manufacturer's PLC program, e.g. after POWER ON or at the end of the program.
2. **Channel active**
   A program has started; program execution or reference point approach is in progress.
3. **Channel interrupted**
   The running program or reference point approach has been interrupted.

Functions in the modes

User-specific functions are available within the modes. These functions are not related to any particular technology or machine. A subset of the available functions can be selected in each mode, depending on the operating state. These functions are categorized as follows:

- NCK-specific functions
- Modegroup-specific functions
- Channel-specific functions

The individual functions can be started and/or executed from the three individual channel states channel reset, channel active or channel interrupted. The channel states and program states can be checked on the operator panel.

2.3.1 Monitoring functions and interlocks of the individual modes

Monitoring functions in the modes

Different monitoring functions are active in individual operating modes. These monitoring functions are not related to any particular technology or machine.

In a particular mode only some of the monitoring functions are active depending on the operating status. The channel state determines which monitoring functions are active in which mode and operating state.

Monitoring functions in the modes

A variety of interlocks can be active in the different operating modes. These interlocks are not related to any particular technology machine.

Almost all the interlocks can be activated in every mode, depending on the operating status.

© Siemens AG, 2003. All rights reserved

2.3.2 Mode change

General

Switchover to another operating mode is requested and activated via the mode group interface (DB11, ...). A mode group will either be in AUTOMATIC, JOG or MDA mode, i.e. it is not possible for several channels in a mode group to accept different modes at the same time.
What mode transitions are possible and how these are executed is configured machine-specifically in the PLC program.

Note

The mode is not changed internally until the signal “Channel status active” is no longer pending. All channels must have entered a permissible operating mode before an error-free mode change can be performed.

The following table shows possible mode changes for one channel.

Table 2-1 Mode change

	from	AUTOMATIC		JOG			MDA (MDI)			
					AUTO	MDA (MDI)		JOG without handwheel		AUTO
After		Reset	Inter-rupt	Reset	Inter-rupt	Inter-rupt	Reset	Inter-rupt	Active	Inter-rupt
AUTOMATIC				X	X		X			
JOG		X	X				X	X		X
MDA (MDI)		X	X	X		X				

Possible mode changes are shown by an “X”.

Error on operating mode switchover

If an operating mode switchover request is rejected by the system, the error message “Operating mode cannot be changed until after NC stop” is output. This error message can be cleared without changing the channel status.

Mode switchover disable

IS “Operating mode switchover stop”(DB11, ... D0.4) can prevent switchover of the operating mode. This suppresses the operating mode change request.
The user must a configure a signal about the active disable for the operator. No signal is set by the system.

Mode change from MDA to JOG

If **all channels** of the mode group are in Reset after an operating mode switchover from MDA to JOB, the NC switches from JOG to AUTO. In this state, the parts program commands “START” or “INIT” can be executed.

If a channel of the mode group is no longer in the Reset state after an operating mode switchover, the part program command “START” is rejected in thiis situation and Alarm 16952 issued.

© Siemens AG, 2003. All rights reserved

2.4 Program test

Purpose

Several control functions are available for testing a new parts program. These functions are provided to reduce danger at the machine and time required for the test phase. Several program functions can be activated at the same time to achieve a better result.

Test options

The following test options are described below:

- Program execution without setpoint outputs
- Program processing in singleblock mode
- Program processing with dry run feedrate
- Skip parts program blocks
- Block search with or without calculation.

2.4.1 Program execution without setpoint output

Functionality

In the Program Testing status, a parts program is executed without the output of axis or spindle setpoints.

The part program can be started and processed via the interface signal IS “NC Start” (DB21, ... DBX7.1), i.e. using auxiliary functions, G function outputs, etc.. The safety functions, for example software limit switch, work are limits remain valid.

The only difference to normal program operation is that an internal axis disable is set for all axes (including spindles). The machine axes do not move, the actual values are generated internally from the setpoints that are not output. The programmed velocities remain unchanged. This means that the position and velocity information output on the operator interface is exactly the same as that output during normal parts program operation.

The position control is not interrupted when this function is active, so the axes do not have to be referenced when the function is switched off.

Advantages

The user can check the programmed axis positions and auxiliary function outputs of a parts program. This program simulation can also be used as an extended syntax check.

Selection

This function is selected via the operator interface in the menu Program control. IS “Program test selected” (DB21, ... DBX25.7) is set on selection of the function. This does not activate the function.

© Siemens AG, 2003. All rights reserved

Activation

This function is activated via IS “Activate program test” (DB21, ... DBX1.7)

Display

As a checkback for the active program test, the appropriate field on the operator interface is reversed and the IS “Program test active” (DB21, ... DBX33.7) set in the PLC.

Note

The signals for exact stop (DB 31–61, DBX60.6, DBX60.7) reflect the actual status on the machine. They are only canceled during program testing if the axis is pushed out of its set position (the set position remains constant during program testing).
Using the “Program testing active” signal (DB21, ... DBX33.7), both the PLC program and the parts program can decide via the $P_ISTEST variable how to react or branch in response to these signals during testing.

Program processing without axis motion can also be activated with the function dry run feedrate. With this function, parts program sections with a small programmed feedrate can be processed in a shorter time.

Tool management

Because of the axis disable, the assignment of a tool magazine is not changed during program testing. A PLC application must ensure that the integrity of the data in the tool management system and the magazine is not corrupted.
You will find an example in the basic PLC program on the toolbox diskettes.

© Siemens AG, 2003. All rights reserved

2.4.2 Program execution in single block mode

Functionality

The parts program can be started via IS “NC Start” (DB21, ... DBX7.1).

When the function “single block” is activated, the parts program stops after every program block during processing.
If cutter radius path compensation or tool nose radius compensation is active, processing stops after every intermediate block inserted by the control.

The program status switches to “Program status stopped”. The channel status remains active.

The next parts program block is processed on NC Start.

Advantages

The user can process a parts program block by block to check the individual machining steps. Once he has decided that a processed parts program block is correct, he can call the next block. This is done by triggering NC Start.

Single block type

The following different types of single block are provided:

- Decoding single block
  With this type of single block, all blocks in the parts program (even the pure computation blocks without traversing motions) are processed sequentially by “NC Start”.
- Single action block
  With this type, all blocks which initiate actions (traversing motions, auxiliary function outputs, etc.) are processed individually. Those blocks generated during decoding (e.g. cutter radius path compensation at acute angles) are also processed in single-block mode. Processing is however not stopped at calculation blocks as these do not trigger actions.

Single action block is an initial setting.

For selection of single-block mode see:
References: /BA/, “Operator’s Guide”



Caution

- With a series of G33/G34/G35 blocks, single block is operative only if “Dry run feedrate” is selected.
- Calculation blocks are not processed in single step mode (only if single decoding block is active).
- In SW 5.2 and higher, SBL2 is also ineffective with G33/G34/G35

© Siemens AG, 2003. All rights reserved

Selection

Single-block mode is selected via the operator interface in the menu Program control. IS “Single block selected” (DB21, ... DBX24.4) is set on selection of the function. This does not activate the function.

Activation

This function is activated via IS “Activate single block” (DB21, ... DBX0.4)

Display

The status signal that single block mode is active is shown by inversion of the relevant field on the operator interface. As soon as the parts program processing routine has processed a parts program block as a result of single block mode, the IS “Program status interrupted” (DB21, ... DBX35.3) set in the PLC.

Function extension with SW 4 and higher

With SW 4 and higher, continuous program execution may be needed for certain operations even when single block mode is selected.
MD 10702: IGNORE_SINGLEBLOCK_MASK specifies which routines must be processed without interruption.

- Internal ASUBs
- User ASUBs
- Intermediate blocks
- Block search group blocks (action blocks)
- Init blocks
- Subroutines with DISPLOF
- Non-reorganizable blocks
- Non-repositionable blocks
- Reposition blocks without travel information
- Tool approach block.

Assignments between operations and bits of the machine data can be found in Section 4.2.

For further information about action blocks in block searches with calculation, see Subsection 2.5.2 “Processing specific program sections”.

© Siemens AG, 2003. All rights reserved

2.4.3 Program execution with dry run feedrate

Functionality

The parts program can be started via IS “NC Start” (DB21, ... DBX7.1). When the function is active, the traversing velocities programmed in conjunction with G01, G02, G03, G33, G34 and G35 are replaced by the feed value stored in SD 42100: DRY_RUN_FEED. The dry run feedrate also replaces the programmed revolutional feedrate in program blocks with G95.



Danger

Workpieces must not be machined when dry run feedrate is active because the altered feedrates might cause the permissible tool cutting rates to be exceeded and the workpiece or machine tool could be damaged.

Selection

Dry run feedrate is selected via the operator interface in the Program control menu. IS “Dry run feedrate selected” (DB21, ... DBX24.6) is set on selection of the function. This does not activate the function. In addition, the required dry run feedrate must be entered in the menu Setting data/JOG data.

Activation

This function is activated via IS “Activate dry run feedrate” (DB21, ... DBX0.6)

Changing the dry run feedrate

In SW 6.2 and higher, the effect of SD 42100: DRY_RUN_FEED can be controlled using a further setting data SD 42101: DRY_RUN_FEED_MODE.

The following options are available for changing the dry run feedrate:

1. The maximum of the programmed feedrate and setting data SD 42101: DRY_RUN_FEED_MODE applies as the dry run feedrate.
2. The minimum of the programmed feedrate and setting data SD 42101: DRY_RUN_FEED_MODE applies as the dry run feedrate.
3. The value of SD 42101: DRY_RUN_FEED_MODE applies directly, irrespective of the programmed feedrate.

A dry run feed can be selected in the automatic modes and activated on interruption of an automatic mode or end of a block.

For further information on feedrate control, please see:
References: /FB/, V1, “Feedrates”

Display

The status signal that dry run feedrate is active is shown by inversion of the relevant field on the operator interface.

© Siemens AG, 2003. All rights reserved

2.4.4 Skip part program blocks

Functionality

When testing new programs it is useful to be able to disable or skip certain parts program blocks during program processing.

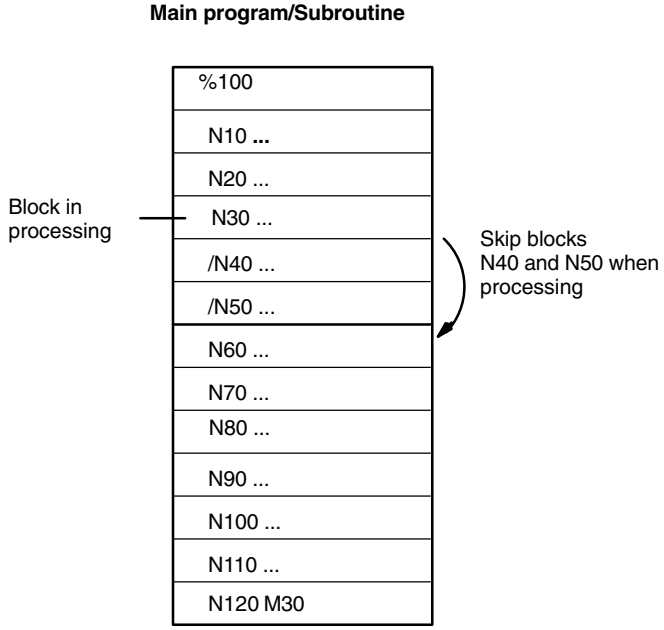


Fig. 2-1 Skipping parts program blocks

Selection

The skip function is selected via the operator interface in the menu Program control. IS “Skip block selected” (DB21, ... DBX26.0) is set on selection of the function. This does not activate the function. In addition, a slash “/” must be written before the blocks to be skipped (see Fig. 2-1).

Activation

This function is activated via IS “Activate skip block” (DB21, ... DBX2.0).

The “Skip parts programs” function remains active during block searches.

Display

The relevant field in the status line on the operator interface is shown in reverse video to indicate that the function “Skip block” has been activated.

© Siemens AG, 2003. All rights reserved

2.5 Block search

Functionality

To set the control system to a particular block of a subroutine, it is possible to simulate sections of the program using the function block Search. It can be selected whether or not the same calculations are to be performed during the block search as would be performed during normal program operation. The following types of program simulation are available:

- Locate particular points in the program:
  Type 1 without calculation
  Type 2 with calculation at the contour
  Type 4 with calculation at block end point (SW 4.3 and higher)
  Type 5 block search with execution in program testing mode (SERUPRO stands for search run by program test in SW 6.1 and higher)

Overview

The following options for applying block searches are available:

- Automatic start of an ASUB after block search for types 1 to 5
  In SW 6.1 and higher, a user program can be started as an ASUB when the last action block is loaded.
- Cascaded block search for types 1 to 5
  In SW 6.1 and higher, a further block search with a modified search target can be started from the “Search target found” state.
- Cross-channel block search in program testing mode SERUPRO Type 5.
  With SW 6.1 and higher, a channel can interact with synchronized actions, and several channels with one another, on the same NCU.

SERUPRO sequence without defining a search target (SW 6.1 and higher)

NC functions supported in SERUPRO (SW 6.3 and higher)

- Gear step change
- Setpoint and actual value couplings for drives such as

  Master-slave and electronic gear and axial master-value coupling
- Coupled motion in axis grouping and
- Gantry axes
- Tangential follow-up of individual axes
- Superimposed motion interpolation
- Travel to fixed stop
- Synchronous spindle grouping

NC functions supported with SERUPRO ASUB (SW 6.3 and higher)

- Reference point approach
- Tool management
- Detect spindle power-up and spindle status

Continue machining after SERUPRO search target found (REPOS offset)

Reapproach contour with controlled REPOS (SW 6.3 and later)

Parts program expansions for SERUPRO (SW 6.2 and higher)

© Siemens AG, 2003. All rights reserved

2.5.1 Block search functions

Block search functions

The processing of specific program sections using program simulations is referred to as a Block Search. Block searches can be activated on the HMI operator panel with the appropriate soft key for the following functions:

- Block search without calculation (type 1)

  Used for rapid searches in the main program.
  No calculation is performed. The internal controller values indicate the status valid before the search.

- Block search with calculation at the contour (type 2)

  Is used in any circumstances in order to approach the contour. On NC Start, the start position of the target block or the end position of the block before the target block is approached. This is traversed up to the end position. Processing is true to contour.

- Block search with calculation at block end point (type 4 from SW 4.3 and higher)

  Is used in any circumstances in order to approach a target position (e.g. tool change position). The end position of the target block or the next programmed position is approached using the type of interpolation valid in the target block. This is not true to contour. Only the axes programmed in the target block are moved. If necessary, a collision-free start position can be approached manually with JOGREPOS.

- Block search with execution in program test mode, SERUPRO (type 5)

  SERUPRO is a multi-channel block search with calculation of all required status data from the history. Single-channel applications are supported for parallel interaction with other functions. The NC will start the selected program in the Program Test mode automatically, will stop at the beginning of the target block and will then internally deselect the program test. A message such as waiting for axis replacement will then be displayed at the start of the “Search target found” target block.

Note

For further information about the block search function, please see
References: /FB/, H2, “Auxiliary Function Output to the PLC”, behavior with block search

---

Selection, activation and display

The block search is selected in AUTOMATIC mode on the operator interface HMI. The current block window appears with the target block.

The window is displayed only if the search target has been found.

Block search on active skip level

The following applies to block search types 1 to 5:

The active skip levels are evaluated when the search target is found. If the search target is located in a skipped block, it will not be located. In SW 6.3 and lower, the block search continues without error message.

In SW 6.4 and higher, the block search is halted and alarm 15350: “Search target not found” output.

© Siemens AG, 2003. All rights reserved

2.5.2 Sequence for block searches of type 1, 2 and 4 (SW 4.3 and later)

Chronological sequence

The chronology of block searches of type 1, 2 and 4 is as follows:

- Activate search via input in HMI Advanced or HMI Embedded
- Search target found, or alarm if target cannot be found
- NC Start for output of action blocks
- NC Start for program continuation.

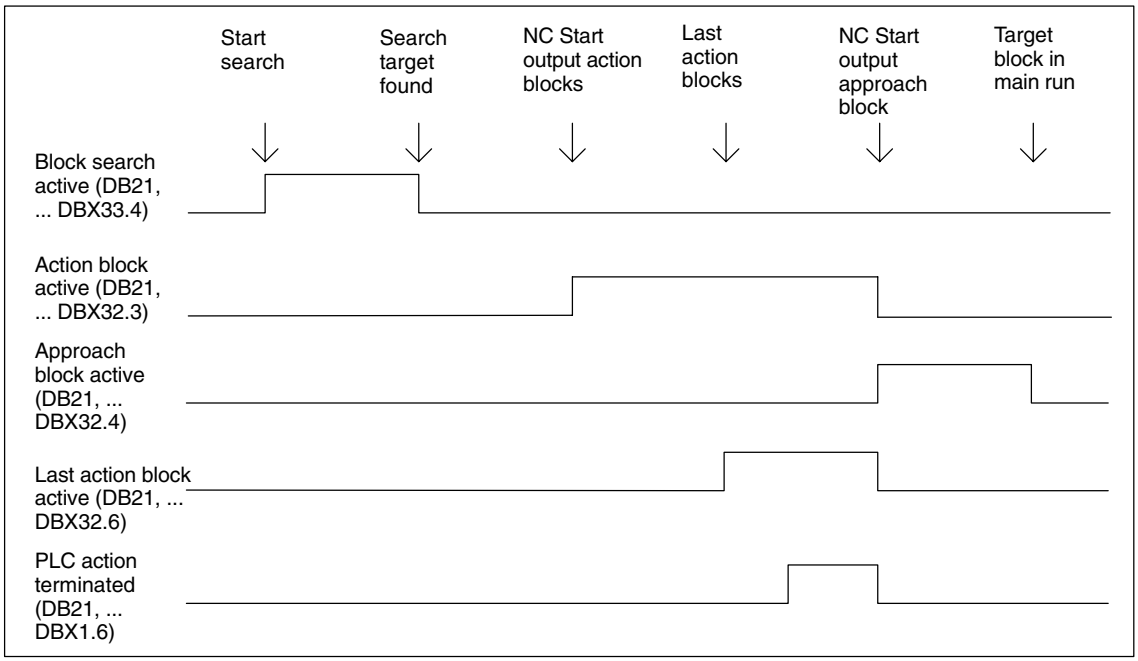


Fig. 2-2 Chronological order of interface signals

Interface signal

In the PLC, the following interface signals are set as shown in the sequence displayed in figure 2-2:

- IS “Block search active” (DB21, ... DBX33.4)
- IS “Action block active” (DB21, ... DBX32.3)
- IS “Action block active” (DB21, ... DBX32.4)
- IS “Last action block active” (DB21, ... DBX32.6)
- IS “PLC action terminated” (DB21, ... DBX1.6)

Continuation mode after block search

After the block search, the program can be restarted via IS “NC Start” DB21, ... DBX7.1).
If the axis is programmed for the first time after “Block search with calculation at end of block”, setting data SD 42444: TARGET_BLOCK_INCR_PROG of the incremental value can be added to the total value at the search target.

© Siemens AG, 2003. All rights reserved

Action blocks

Action blocks contain the actions accumulated during “block search with calculation”, e.g. auxiliary function outputs, and tool (T, D), spindle (S) and feed programming commands. During “block search with calculation” (contour or block end point), actions such as M function outputs are accumulated in socalled action blocks. These blocks are output on an NC start after “search target found”.

Note

With the action blocks, the accumulated spindle programming (S value, M3/M4/M5, SPOS) becomes also active. The PLC user program must ensure that the tool can be operated and, if necessary, the spindle programming is reset via the PLC signal “Spindle reset” (DB31, ... DBX2.2) or the programmed spindle values are not output.

Single block processing: MD 10702: IGNORE_SINGLEBLOCK_MASK ()

By setting bit 3 =1, it is possible to prevent a stop after every action block in single-block mode. For further information about “Single block mode”, see Section 2.7 and Subsection 2.4.2 “Program execution in single block mode”.

Supplementary conditions approach block/target block

Block search type 2
The IS “Approach block active” (DB21, ... DBX32.4) is **only** enabled with “Block search with calculation on contour ” because a separate approach block is not generated with “Block search with calculation at block end point ” (the approach block is the same as the target block).

Block search type 4
The approach movement “Search with calculation to block end point” is performed using the type of interpolation valid in the target block. This should be G0 or G1, as appropriate. With other types of interpolation, the approach movement can be aborted with an alarm (e.g. circle end point error on G2/G3).

ASUB after and during block search

If an ASUB is started after “Block search with calculation”, the positions in the interpreter are synchronized. $P_EP returns the position at which the axes are actually located to the ASUB.

The collective block search position can be scanned via system variable $AC_RETPOINT. This variable supplies the position as not modulo-converted and an existing number of rotations can be lost.

Block search type 2
The ASUB must be exited with command REPOSA with “Block search with calculation on contour”. The axes are then automatically moved to the collected block search position. $P_EP then also returns to this value.

Block search type 4
After “Block search with calculation at block end point”, automatic repositioning is not performed by the REPOS parts program command between “Last action block active” and continuation of parts program processing by NC Start. The start point of the approach movement is the current axis position on NC Start; the end point results from the processing of the parts program.

No approach motion is generated by the system with “Block search with calculation at block end point”. $P_EP thus returns the position at which the axes were located by the ASUB or in JOG after the ASUB has been exited.

© Siemens AG, 2003. All rights reserved

PLC actions after block search

To activate PLC actions after a block search, e.g. to permit starting of ASUBs or FCs at a defined position, SW4.3 and higher include an IS “Last action block active” DB21, ... DB32.6 The signal indicates that all actions have been executed and it is now possible to perform PLC actions (ASUB, FC) or operator actions (override, mode change with JOG/REPOS). This allows the PLC to perform another tool change, for example, before the start of the movement.

Alarm 10208 is output per default at this moment to notify the operator that an NC Start is needed to continue program execution.

MD 11450

The machine data MD 11450: SEARCH_RUN_MODE allows you to set a different response.

If machine data MD 11450: SEARCH_RUN_MODE=1,
alarm 10208 is not output until the PLC sets interface signal “PLC action ended” (DB21, ... DBX1.6).

Auxiliary spindle functions after block search

Control system response and output (SW 4.3 and higher)

The machine data MD 11450: SEARCH_RUN_MODE determines the control system response after the end of the block search via the following bits:

- Bit 0 = 0: NC Stop after output of last action block (default setting)
  Bit 0 = 1 : NC Stop after output of the last action block only if the PLC requests the latter with IS “Last action block active”.
- Bit 1 = 0: No automatic ASUB start after block search.
  Bit 1 = 1: Automatic start of user program (SW 6.1 and higher) /_N_CMA_DIR/_N_PROG_EVENT_SPF as ASUB.
- Bit 2 = 0: Output of auxiliary spindle functions (M3, M4, M5, M19, M70) takes place in the action blocks as before (SW 5.3 and higher).
  Bit 2 = 1 : Output of auxiliary spindle functions and spindle setpoints suppressed after block search (e.g. output in ASUB).
- Bit 3=0 : Cascaded block search is enabled (SW 6.1 and higher) (i.e. several search targets can be specified).
  Bit 3 = 1: Cascaded block search is disabled.

The programmed spindle values are collected in the new block search system variables during block searches irrespective of the configuration.

Spindle programming in ASUBs

The collective spindle programming can be output in an ASUB after

- a block search and
- output of IS “Last action block active” (DB21, ... DB32.6) = 1

It can thus be ensured that, independently of the block search target, the maximum admissible speed of the active tool is not exceeded.

For a further explanation of spindle auxiliary functions and a description of the new block search system variables, please see:
References: /FB/, S1, Spindles, Spindle auxiliary functions SSL.

© Siemens AG, 2003. All rights reserved

2.5.3 Automatic start of an ASUB following block search (SW 6.1 and higher)

Activation

The automatic ASUB start after block search is configured in the existing machine data MD 11450: SEARCH_RUN_MODE with bit 1 = 1 (TRUE).

- Bit 1 = 1: Automatic start from the user program /_N_CMA_DIR/_N_PROG_EVENT_SPF as an ASUB

With changing of the last action block, the user program **/_N_CMA_DIR/_N_PROG_EVENT_SPF** is started as an ASUB.

Start event

By which event this user program has been started can be determined by requesting the system variable $P_PROG_EVENT. If it is an ASUB start after a block search, the **value 5** can be requested using the system variable $P_PROG_EVENT (SW 6 and higher).

Note

For further information on how to parameterize certain events, please refer to

Section 2.6 Program operation “Event-controlled program calls”.

Example

Sequence for the automatic start of an ASUB after block search:

1. Start block search (with/without calculation, at contour, at end-of-block point)
2. Stop after “Search target found”
3. NC Start for output of action blocks
4. Last action block is changed;
5. Automatic start of /_N_CMA_DIR/_N_PROG_EVENT_SPF as an ASUB
6. The NC will stop after changing the last ASUB block (REPOSA command) and output the alarm 10208 depending on MD 11450: SEARCH_RUN_MODE with bit 0.

2.5.4 Cascaded block search (SW 6.1 and higher)

Activation

The cascaded block search is configured in the existing machine data MD 11450: SEARCH_RUN_MODE with bit 3 = 0 (FALSE).

- Bit 3=0 : Cascaded block search is enabled (SW 6.1 and higher) (i.e. several search targets can be specified).

For compatibility reasons, the cascaded block search can be disabled in MD 11450: SEARCH_RUN_MODE with bit 3 = 1 (TRUE). By default, the cascaded block search is set with bit 3 = 0.

© Siemens AG, 2003. All rights reserved

Functionality

The function “Cascaded block search” can be used to start another block search from the status “Search target found”. The cascading can be continued after each search target found as often as you want and can be used for the following block search functions:

- Block search with calculation to contour
- Block search with calculation to block end point
- Block search without calculation.

When the search target is reached, the program execution stops and the search target is displayed as a current block. Only if the search target has been found, another cascaded block search can be started from this status. After each search target found, a new block search can be repeated as often as you want. You can change search target specification and block search function prior to each block search start.

Example

Sequence with cascaded block search:

- Reset
- Block search up to search target 1
- Block search up to search target 2 –> “Cascaded block search”
- NC Start for output of the action blocks –> alarm 10208
- NC Start –> continue program execution.

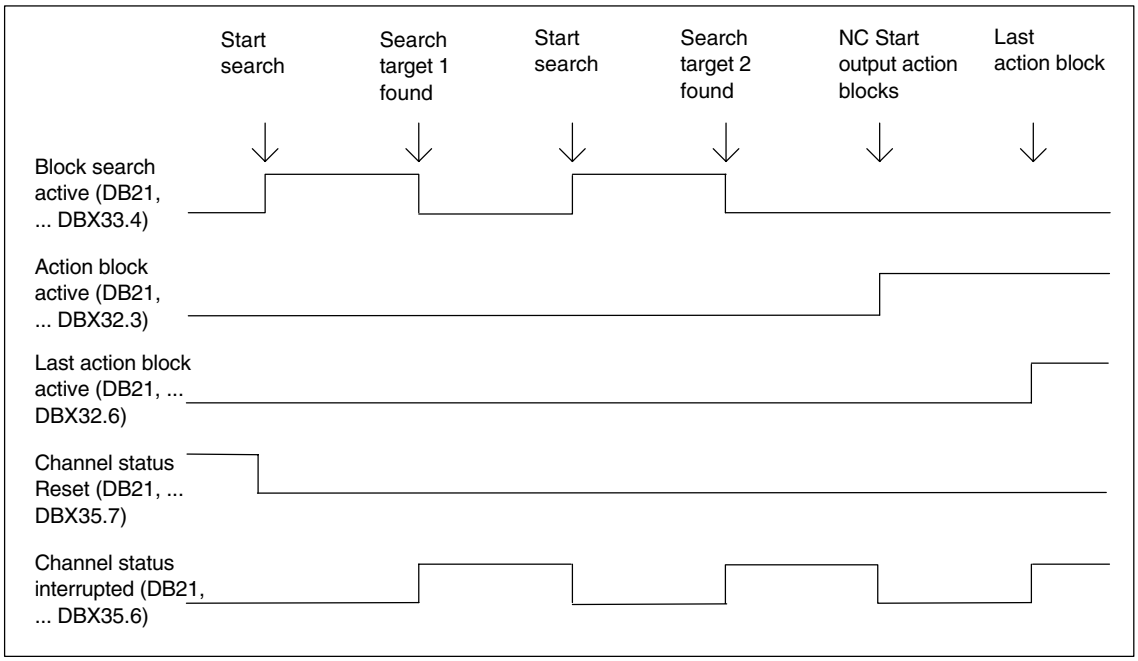


Fig. 2-3 Chronological order of interface signals

© Siemens AG, 2003. All rights reserved

2.5.5 Examples of block search with calculation

Block search type 4

Block search at block end point

Example with automatic tool change after block search with active tool management:

1. Set MD 11450: SEARCH_RUN_MODE to 1;
   Set machine data MD 11602: ASUP_START_MASK Bit 0 = 1 (start ASUB from stopped state)
2. Select ASUB “SUCHLAUF_ENDE” from PLC via FB4
   (see References /FB1/, P3 “Basic PLC Program”)
3. Load and select parts program “TOOL_1”
4. Search to block end point, block number N220
5. MMC reports “Search target found”
6. NC Start for output of action blocks
7. The PLC uses the PLC signal “Last action block active” to start the ASUB “END_OF_SEARCH” via FC9 (see References /FB/, P3, “Basic PLC Program”)
8. After the end of the ASUB (can be evaluated, e.g., via the M function M90 to be defined, see example for block N1110), the “PLC enables the PLC action terminated” signal.
   As of SW 5.3, the VDI interface signal “ASUB is halted” (DB 21–DB30 DBB318 Bit 0) can also be polled.
   This causes alarm 10208 to be displayed, i.e. further operator actions may now be performed.
9. Manual operator actions (JOG, JOGREPOS, override)
10. Continue parts program on NC Start.

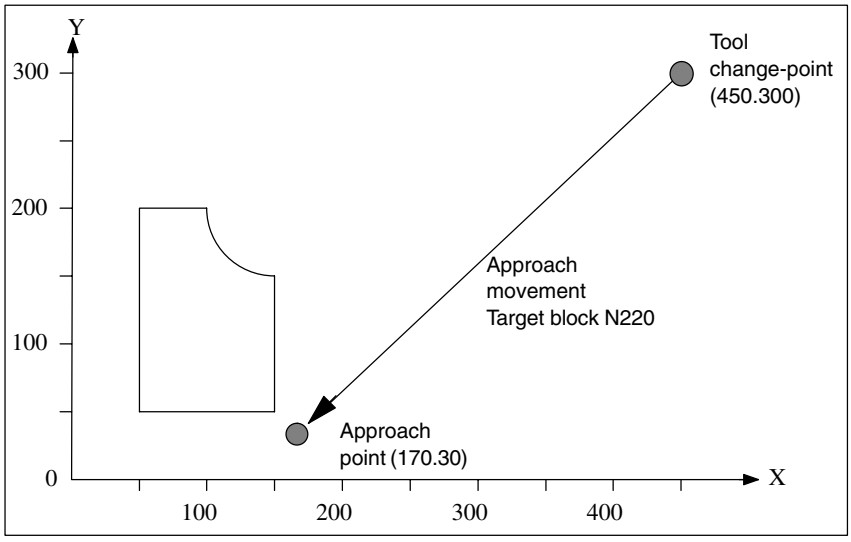


Fig. 2-4 Approach movement for search to block end point (target block N220)

Note

“Search to contour” with target block N220 would generate an approach movement at the tool change point (start point of the target block).

© Siemens AG, 2003. All rights reserved

Block search type 2

Block search on contour

Example with automatic tool change after block search with active tool management:

Points 1 to 3: as example for block search type 4
4. Search to contour, block number N260
Points 5 to 10: as example for block search type 4

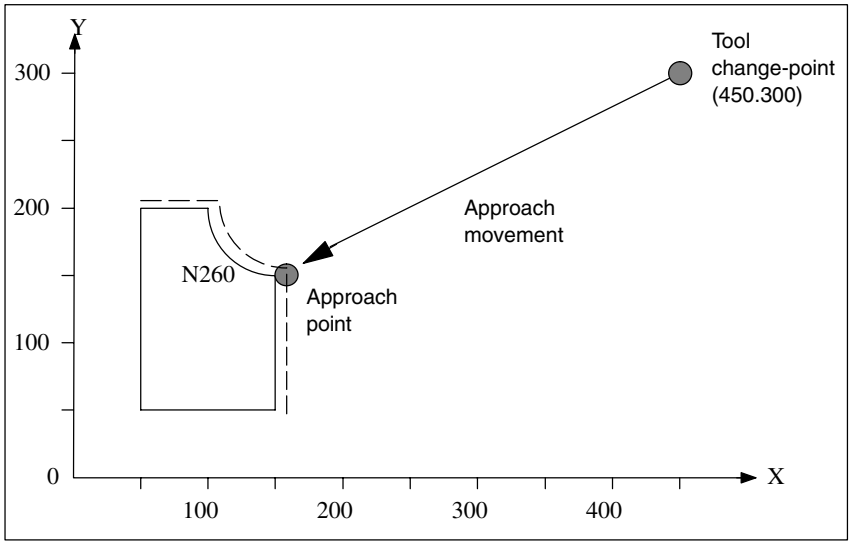


Fig. 2-5 Approach movement for search to contour (target block N260)

Note

"Search to block end point" with target block N260 would result in alarm 14040 (circle end point error).

Parts programs for types 4 and 2

```
PROC WORKPIECE_1
; Main program
...
;Machine contour section 1 with tool "MILL_1"
...
N100 G0 G40 X200 Y200                ; Deselect radius compensation
N110 Z100 D0                         ; Deselect length compensation
; End of contour section 1
;
;Machine contour section 2 with tool "MILL_1"
N200 T="MILL_2"; Preselect tool
N210 WZW                             ; Call up tool change routine
N220 G0 X170 Y30 Z10 S3000 M3 D1     ; Approach block, contour section 2
N230 Z5                              ; Infeed
N240 G1 G64 G42 F500 X150 Y50        ; Contour start point
```

© Siemens AG, 2003. All rights reserved

```
N250 Y150
N260 G2 J50 X100 Y200
N270 G1 X50
N280 Y50
N290 X150
N300 G0 G40 G60 X170 Y30              ; Deselect radius compensation
N310 Z100 D0                          ; Deselect length compensation
; End of contour section 2
...
M30

PROC WZW
; Tool change routine
;
N500 DEF INT TNR_AKTIV                ; Variable for active T number
N510 DEF INT TNR_VORWAHL              ; Variable for preselected
                                      ; T number
;
N520 TNR_AKTIV = $TC_MPP6[9998,1]     ; Read T number of active
                                      ; tool
N530 GETSELT(TNR_VORWAHL)             ; Read T number of preselected
                                      ; tool
;
; Only execute tool change if tool not yet active
N540 IF TNR_AKTIV == TNR_VORWAHL GOTOF ENDE
N550 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0   ; Tool change point
                                             ; approach
N560 M6                               ; Execute tool change
;
ENDE: M17

PROC SUCHLAUF_ENDE SAVE
;ASUP for calling the tool change routine after block search

N1000 DEF INT TNR_AKTIV               ; Variable for active T number
N1010 DEF INT TNR_VORWAHL             ; Variable for preselected
                                      ; T number
N1020 DEF INT TNR_SUCHLAUF            ; Variable for T number determined
                                      ; T number

N1030 TNR_AKTIV = $TC_MPP6[9998,1]    ; Read T number of active
                                      ; tool
N1040 TNR_SUCHLAUF = $P_TOOLNO        ; Read T number
                                      ; determined by search
N1050 GETSELT(TNR_VORWAHL)            ; Read T number of preselected
                                      ; tool
N1060 IF TNR_AKTIV ==TNR_SUCHLAUF GOTOF ASUP_ENDE
N1070 T = $TC_TP2[TNR_SUCHLAUF]       ; T selection via tool name
N1080 WZW                             ; Call tool change routine
N1090 IF TNR_VORWAHL == TNR_SUCHLAUF GOTOF ASUP_ENDE
N1100 T = $TC_TP2[TNR_VORWAHL]        ; Restore T selection via
                                      ; tool name
ASUP_ENDE:
N1110 M90                             ; Checkback to PLC
N1120 REPOSA                          ; End of ASUB
```

© Siemens AG, 2003. All rights reserved

2.5.6 Block search in Program Test mode, SERUPRO

Definition SERUPRO

The “Block search via program test” is in the following referred to as **SERUPRO**. This acronym has been derived from **“Se**arch **Ru**n by **Pro**gram Test”.

Functionality

SERUPRO can be used for a block search across several channels.

This block search permits a block search with calculation through **all** the relevant data from the previous history, so as to acquire all previously valid status data for a particular overall NC status. In this case, the PLC is brought up to the current status.

The NC is operated in program testing mode during this block search so that interactions between one channel and synchronized actions, or between several channels, can take place.

In combination with the MMC/HMI, SERUPRO is provided for the following channels:

- For the current SERUPRO channel only
- For all channels on the NCU
- For all channels with the same workpiece name as the SERUPRO channel
- For all channels with the same mode group as the SERUPRO channel.

All other channels started with SERUPRO are operated in the “Self-Acting SERUPRO” mode. Only the channel in which a target block has been selected can be started with a block search in SERUPRO mode.

Activation

SERUPRO is activated via

PI service “_N_FINDBL” with
PI service parameter searchMode set to 5.

The operator panel interface is supplied with data via this PI service. SERUPRO uses REPOS to approach the target block.

SERUPRO is operated on the MMC/HMI via the “Prog.Test Contour” soft key.

Regarding mode change, reset and POWER ON

- The mode change is permitted during SERUPRO.
- Reset will cancel SERUPRO, i.e. will deselect the internally selected program test. POWER ON will enable SERUPRO again.

Supplementary conditions for Program test

Program test mode can be deactivated only in the RESET channel state in software versions up to SW 5.3. MD 10707: SERUPRO_TEST_MASK allows the mode to be deactivated in the stopped state without the SERUPRO operation being affected. The default setting allows program testing to be deactivated only in the RESET state.

Note

After program testing has been deactivated, a REPOS operation is initiated that is subject to the same restrictions as a SERUPRO approach process.
Any impairment to performance can be avoided by means of an ASUB.

© Siemens AG, 2003. All rights reserved

Supplementary conditions for block search SERUPRO

The SERUPRO function may only be aborted in the “AUTOMATIC” mode in the program status (channel status RESET).

If in normal mode **only** the PLC starts commonly several channels, then this can be simulated by SERUPRO in each channel.

When MD 10708: SERUPRO_MASK bit 1 = 0, alarm 16942 “Channel %1 Start program action %2<ALNX> not possible” aborts the simulation if the parts program “START” is used.

Influence SERUPRO behavior

Machine data MD 10708: SERUPRO_MASK can influence the SERUPRO behavior as follows:

Stop at M0 during the search
Bit 0 = 0 NC will stop at M0 during the search.
Bit 0 = 1 NC will not stop at M0 during the search.

Allow parts program “START” on alarm 16942
Bit 1 = 0 Alarm will abort the search on parts program START.
Bit 1 = 1 Alarm is suppressed. A program uses the parts program “Start(j)” in channel i after the following call:
The preselected program is started in channel j.
Channel j begins to really start with moving axes.
In channel j, Program test can be selected by the user beforehand. Channel j will now **not** select a search target.

SW 6.2 or higher Enable or disable “Group-SERUPRO” function
Bit 2 = 0 enables the function “Group-SERUPRO”.
Bit 2 = 1 enables the function “Group-SERUPRO”.

Before SW 6.4, use SERUPRO simultaneously in all channels,
Bit 3 = 1 and search target found. Exception: A reset has aborted the search run or the channel has reached M30 without finding the search target.

Basic setting for SERUPRO

Machine data MD 20112: START_MODE_MASK defines the basic setting of the control at the start of the parts program with respect to G codes (in particular, the active plane and settable zero offset), tool length compensation, transformation and axis links.
In SW 6.3 and higher it is possible for the SERUPRO operation to use

MD 22620: ENABLE_START_MODE_MASK_PRT

to select a basic setting other than the normal parts program start. The new setting must be stored in MD 22620: START_MODE_MASK_PRT. The meaning of the bits of MD 22620 is identical to those in MD 20112: START_MODE_MASK.

Example: At parts program start, the synchronous spindle coupling is retained at the beginning of the SERUPRO operation.

```
                                                ; Non-configured synchronous spindle coupling
$MC_START_MODE_MASK = 'H400'                    ; is disabled.
$MC_START_MODE_MASK_PRT = 'H00'                 ; remains active
$MC_ENABLE_START_MODE_MASK_PRT = 'H01' ; $MC_START_MODE_MASK_PRT is
                                                evaluated in SERUPRO instead of $MC_START_MODE_MASK.
```

© Siemens AG, 2003. All rights reserved

Chronological sequence of SERUPRO

The sequence of operations from 5. to 9. corresponds to one **SERUPRO operation**.

1. The SERUPRO function is enabled using the channel-specific PI service "_N_FINDBL".
2. All parameters describing the search target (target block) must be stored in the OPI module SPARPF.
3. The searchMode parameter of the PI service has the following value:
   5 for approaching the contour starting point of the target block.
4. The PI service "_N_FINDBL" is positively acknowledged immediately if all parameters are correct. There is no delay while the search target is located.
5. The NC will now start the selected program in Program test mode automatically.
   – The axes will not be traversed.
   – Auxiliary functions $A_OUT and the direct PLC I/O are output.
   – The auxiliary functions of the target block are not output.
6. A large number of operator actions can be performed at this stage, for example:
   – start, stop,
   – axis replacement,
   – deletion of distance-to-go,
   – mode change, ASUBs, etc.
   The program and channel status on the interface (DB21, ... DBB35) or the system variable $AC_PROG are provided analogously to real operation.
7. The parts program command WAITM/WAITE/WAITMC will wait for the partner channels concerned. This will also take place if:
   – the partner channels are in the SERUPRO mode;
   – program test or real run.
8. Selection of program testing and dry run feedrate is rejected with the appropriate alarm 16935.
9. After the target block has been found, the NC stops at the beginning of the target block, deselects "Program test" internally and displays the Stop condition "Search target" found in its block display.
10. As required, the user can start an ASUB that is traversed really.
    Definition: This ASUB is referred to below as the **SERUPRO ASUB**.
11. The user presses Start: The spindles are started;
    then the path axes start a REPOS process leading them to the start-of-block point of the target block. The REPOS process has been implemented using a system ASUB and can be expanded via the function "Editable ASUB".
    Definition: This process is called **SERUPRO approach**.

DB21, ... DBX318.1

The VDI signal provided from the NCK channel (NCK→PLC)
IS "Block search via Program Test is active" (DB21, ... DBX318.1) has the following meaning and effect:

In the internal mode Program test, the NC runs until the target block of the search block is changed in the main run and the program status changes to stopped. During this time, the IS "Block search via Program Test is active" is set to 1.

Cascaded block search

Cascaded SERUPRO:
Following a SERUPRO block search, a further search can be started from the "search target found" state, making it possible to conduct an unlimited number of searches in a program loop.

© Siemens AG, 2003. All rights reserved

For user-defined ASUB after the SERUPRO process

Note

If the machine manufacturer decides to start an ASUB after the SERUPRO operation as described under 10., the following must be observed:

Stopped status acc. to point 9. : The machine data
MD 11602: ASUP_START_MASK and
MD 11604: ASUP_START_PRIO_LEVEL
allow the NCK to start the ASUB from the stopped status automatically via the FC9 module.

Acknowledgment from FC9 only after completion of the REPOS block:
The ASUB can only be signaled completed with “ASUB Done” by the FC9 module if the REPOS block is also completed.

Deselection of the planned REPOS process to point 11. :
Starting of the ASUB will deselect the planned REPOS process!
The ASUB should therefore be completed with REPOSA to keep the REPOS process.

Deletion of an undesired REPOS process:
The undesired REPOS process is deleted if the ASUB is quitted with M17 or RET.

Special handling of ASUB:
Generally, an ASUB that ends with REPOS and which is started from a stopped status is handled in a particular way.
The ASUB will stop automatically prior to each REPOS block and will display this via the IS “ASUB stopped” (DB21, ... DBX318.0).

For further explanations regarding the automatic ASUB in the appropriate operating modes with the appropriate start conditions, please refer to Subsection 2.6.14 Calling the ASUB outside the program operation.

Automatic ASUB start

The ASUB stored in /_N_CMA_DIR/_N_PROG_EVENT_SPF is automatically started in SERUPRO approach with machine data MD 11450: SEARCH_RUN_MODE, Bit1 = 1 in the following sequence:

1. The SERUPRO operation has been performed completely.
2. The user presses Start in order to output the action blocks.
3. The last action block is loaded and automatic ASUB Start is initiated.
4. The NCK stops automatically before the REPOS parts program command Alarm 10208 “Press NC Start to continue the program” appears.
5. The user presses Start again.
6. The NCK executes the REPOS movement and continues the parts program at the target block.

Note

The automatic ASUB start with MD 11450: SEARCH_RUN_MODE needs **2 starts** to continue the program.
The behavior is therefore similar to other types of search.

© Siemens AG, 2003. All rights reserved

StopRe in target block

Modal settings for SERUPRO

The STOPRE block receives all modal settings from the preceding block and can therefore apply conditions in advance in relation to the following actions:

- Synchronize program line currently processing with the main run.
- Derive modal settings for SERUPRO in order, for example, to influence this REPOS motion on approach to SERUPRO.

Example 1: Position a Z axis by specifying an X axis setpoint.

When block “G1 F100 Z=$AA_IM[X]” is interpreted, the preceding STOPRE block ensures synchronization with the main run. The correct setpoint of the X axis is thus read via $AA_IM to move the Z axis to the same position.

Example 2: Read and correctly calculate an external zero offset.

```
N10 G1 X1000 F100                ;
N20 G1 X1000 F500                ;
N30 G1 X1000 F1000               ;
N40 G1 X1000 F5000               ;
N50 SUPA G1 F100 X200            ; Move external zero offset to 200
N60 G0 X1000                     ;
N30                              ;
```

With an implicit STOPRE before N50, the NCK can read and correctly calculate the current zero offset.
With a SERUPRO operation to search target N50, the axis is repositioned at the implicit STOPRE in the SERUPRO approach and the velocity calculated from N40 with F5000.

Implicit preprocessing stop

Situations in which interpreter issues an implicit preprocessing stop:

1. In all blocks in which one of the following variable access operations occurs:
   –Programming of a system variable beginning with $A..
   –Redefined variable with attribute SYNR/SYNRW
2. With the following parts program commands:
   –Parts program command MEACALC, MEASURE
   –Programming of SUPA (suppress frames and online offsets)
   –Programming of CTABDEF (start of curve table definition)
   –Parts program command WRITE/DELETE (write/delete file)
   –A sequence of such commands before first WRITE/DELETE command
   –Parts program command EXTCALL
   –Parts program command GETSELT, GETEXET
   –Tool change and active fine tool offset FTOCON
3. With execution of following commands:
   –Final processing of searches of types 1 and 2
   –(“Block search without calculation” and “Block search at block end point”)

   Block search type 2 (“Block search at contour start point”) functions analogously.

Note: The SERUPRO approach
with supposedly incorrect velocity is absolute when the reapproach velocity in REPOS can be specified explicitly.

SPOS in target block

If a spindle is programmed with M3/M4 and the target block contains an SPOS command, the spindle is switched over to SPOS on completion of the SERUPRO process (search target located). This is also indicated on the VDI interface.

© Siemens AG, 2003. All rights reserved

2.5.7 Stop at any point in the program for SERUPRO (SW 6.4 and higher)

Preselect point in program

After a workpiece has been completed, the operator wants to remachine a small area on the workpiece, e.g. because a tool has broken. The operator would continue the program via a block search at a point before the area to be remachined and repeat the machining operation. At the same time, he must know the parts program block at which the machining operation ends.

The function “Stop at any preselected point in program” allows the operator to take this action. The parts program block at which the NC must stop automatically is selected via the MMC/HMI.

The function is referred to below as the **Stop routine** and the parts program block at which machining must stop is referred to as the **Stop block**.

Stop routine

The following applies for PI service “_N_FINDST” (Abbr.: **find st**op):

1. The PI service activates the stop routine and may be called in

   Automatic mode and the “Reset” program state.

2. The PI service is accepted only

   if the NC has finished the SERUPRO operation,
   i.e. the NC has found a search target.

3. This PI service does not start a program.

The following events abort the stop routine prematurely:

- Reset,
- Block search and block search in SERUPRO program test mode.



Caution

If the operator selects a stop block which is not executed in that particular situation, the NC runs the program to the program end.
This is comparable to the following scenario:
The stop block is skipped with GOTOF in the current processing sequence.

Stop block

The stop block is defined (from line 100) via OPI block “SPARPF”.

Example:
The block can be defined by

- the block number,
- a label,
- a string or by
- the seek pointer.

Parameterization of the OPI block does not yet activate the function.

© Siemens AG, 2003. All rights reserved

Block end

When a parts program is started subsequently

- via PLC (MD 22622: DISABLE_PLC_START Bit 0 == 0) or
- from another channel (MD 22622: DISABLE_PLC_START Bit 0 == 1)

the NC starts a search before the stop block in order to stop automatically at its **block end**. The stop routine is then deactivated again.

2.5.8 Multiple starts with SERUPRO in different channels

Multiple starts with SERUPRO

The SERUPRO function starts a parts program in the Program Test mode, and is comparable to a Start key used to initiate a program start in different channels.

This is illustrated by the following example:
Program Prog1Ch1.mpf of the 1st channel starts program Prog1Ch2.mpf in the 2nd channel 5 times in succession as follows:

- The 1st channel is started in normal mode with the Start key.
- A block search must find two target blocks in the relevant programs.
- The search target in the 2nd channel must not be reached until the 5th program run.

The parts program command START of the 1st channel initiates a SERUPRO operation in the other channel. The search target in the 2nd channel must indicate that the target must not be reached until after the 5th start. To avoid inconsistencies, four complete runs of the parts program in the 2nd channel must be simulated.

This search procedure is referred to as “**Group-Serupro**” and is activated **in SW 6.4 and higher** with MD 10708: SERUPRO_MASK, bit 2 = 1. See Subsection 2.5.9 “Carrying out SERUPRO in a group of channels”.

Multiple starts in the current channel

Required expansions in the OPI blocks SPARPF and SPARPI: The pass counter actual value “invocCount” of the main program level may be higher than 1 and include the number of necessary START( ) commands up to the search target. Variable plcStartReason in both blocks indicates which channel must be started by the PLC to ensure that the current channel actually starts.
The following then applies to the example above

- invocCount = 5 in 2nd channel.
- plcStartReason = 1 in block of 2nd channel.

Multiple approach in all channels

All programs contain an M0 shortly before the program end. The program in channel n starts the channel n+1 until the 10th channel is reached. In normal mode, the PLC only starts the first channel. If M0 is to be approached with SERUPRO in all channels, then search target M0 must be entered in the SPARPF block in all channels. The following must be set

- invocCount = 1 and for the channels 2 to 10
- invocCount = 1 and for the channels 2 to 10

© Siemens AG, 2003. All rights reserved

Example

Multiple starts with SERUPRO for 4 channels

The control is initialized with machine data that take effect on POWER ON.

```
$MN_SERUPRO_MASK = 6
M17
```

The R parameters are 0 and
the NCK GUD variables THE_CHAN_NO and FINISH are preset to zero. All channels have selected the following program:

```
DEF INT MYCHAN
DEF INT NEXTCHAN
DEF INT MAXCHAN
MAXCHAN = 4
THE_CHAN_NO = THE_CHAN_NO +1
MYCHAN = THE_CHAN_NO
IF MYCHAN == MAXCHAN
                    NEXTCHAN= 0

ELSE
                    NEXTCHAN = MYCHAN + 1
                    START( NEXTCHAN )
ENDIF

IF FINISH == 1
                    IF R1 == 0
                     M0
                    ENDIF
                    R1 = R1 + 1
ENDIF

IF MYCHAN < MAXCHAN
                    WAITE( NEXTCHAN )
ENDIF

IF MYCHAN == 1
                    STOPRE
                    FINISH = 1
ENDIF

G0 G91 X10
IF MYCHAN < MAXCHAN
                    START( NEXTCHAN )
ENDIF

IF MYCHAN == 1
                     M0
ENDIF

IF MYCHAN < MAXCHAN
                    WAITE( NEXTCHAN )
ENDIF

STOPRE
THE_CHAN_NO = THE_CHAN_NO –1
M30
```

Remarks:
The program starts the next channel twice. The user starts the program in the 1st channel and, after a brief period, all channels have run to M0. RESET is now issued in all channels, causing the NC to store the **interruption pointer** in block SPARPI on all channels.

© Siemens AG, 2003. All rights reserved

Block SPARPI

Block SPARPI as NC interruption pointer contains the following values:

Channel 1	invocCount=0	plcStartReason=0
Channel 2	invocCount=2	plcStartReason=1
Channel 3	invocCount=3	plcStartReason=1
Channel 4	invocCount=5	plcStartReason=1

This example illustrates that the number of passes of the individual channels is not immediately evident if only the program is analyzed.

2.5.9 Carrying out SERUPRO in a group of channels (SW 6.4 and higher)

Channel definitions

The start conditions of for the relevant channels are decisive for searching in a group of channels. The following definitions are made:

An **independent channel** is a channel whose parts programs are started **only** via the PLC.

A **dependent channel** is a channel whose parts programs are started **only** via the parts program command START from a different channel The PLC start in only permitted for program continuation (e.g. after Stop, M0).

The **original channel** has started the dependent channel finally via the PLC.

Example:
Channel 1 has been started via PLC and starts the 2nd channel via the START parts program command. Channel 2 starts channel 3 via the START parts program command. This means that:

Channel 1 is independent
Channel 2 is dependent on the original channel 1
Channel 3 is dependent on the original channel 1

Restriction for normal operation

To perform a search correctly for a whole group of channels, the NCK must be restricted to normal operation. This is activated with machine data MD 22622: DISABLE_PLC_START, which can block starting via PLC for each channel.

If MD 22622: DISABLE_PLC_START is set in a channel, the machine is “Group-Serupro”-capable and the following restrictions apply to **normal operation**:

- Channels with MD 22622: DISABLE_PLC_START == TRUE are **only** started via the parts program command START from Reset. The channel is therefore a **dependent channel** and is monitoried by Alarm 16947.
- Channels with MD 22622: DISABLE_PLC_START == FALSE may **only** be started via the VDI signal (PLC) from Reset. The channel is therefore an **independent channel** and is monitoried by Alarm 16946.
- The parts program in the independent channel must wait for the end of all channels that were started from this channel. Each dependent channel is checked at program end (M30/M2) to establish whether its original channel is still in operation (channel status not equal to reset, Program status not equal to aborted). Otherwise, the alarm 16948 is output.

© Siemens AG, 2003. All rights reserved

Group-Serupro sequence

The program must be continued at the same point after a program abort.

Complete “GroupSerupro” sequence:

1. Every SPARPF block in the 4 channels is parameterized with the search target. In this instance, SPARPI is simply copied complete to SPARPF.
2. The user starts SERUPRO with channel-specific PI service “_N_FINDBL” in the 1st channel.
   (Analogous to actuation of the Start key in real operation).
3. The NC sets the OPI/VDI variable “seruproActive” in the 1st channel
4. The NC simulates the program in the nth channel and starts SERUPRO in channel m in cases where the START(m) instruction must executed in the parts program of channel n. All channels started in this way form a SERUPRO channel group.
5. When all channels in the channel group have reached the search target, “seruproActive” is canceled in all of them.
6. Each channel can now execute SERUPRO approach separately.

Supplementary conditions

The following supplementary conditions must be observed by the user:

- Only one channel chosen by the user may be started via PLC with PI service “_N_FINDBL”.
- Only channels which will finally be included in the channel group really need a search target.
- The following restriction to normal operation may be applicable:
  The parts program started via the PLC waits for the end of all channels started from this channel, i.e. the parts program waits with WAITE before its own end for the end of the other channels.

Note

The MMC/HMI may use plcStartReason as an orientation as regards the channels to be parameterized, i.e. PI is permissible only for channels with plcStartReason==0.

© Siemens AG, 2003. All rights reserved

2.5.10 SERUPRO sequence without defining a search target beforehand

Self-Acting Serupro, SW 6.1 and higher

The channelspecific function “Selfacting Serupro” permits a SERUPRO sequence **without** prior definition of a search target.

In addition, a special channel, the “serurpoMasterChan”, can be defined for **each** “Self-Acting Serupro”.

The function “SelfActing Serupro” supports the block search SERUPRO acting in several channels.

Functionality

The process “SelfActing Serupro” is not intended to find a search target. If the search target is not reached, no channel is stopped. In certain situations, however, the channel is nevertheless stopped temporarily. In this case, the channel will wait for another channel. Examples are: Wait marks, couplings or axis change.

A WAIT phase occurs:
During such a WAIT phase, the NC checks the channel “seruproMasterChan” whether it has reached a search target. If no search target is reached, the WAIT phase is left.

If the search target is reached,
the SERUPRO process is also ended in the channel. The channel “seruproMasterChan” must naturally have been started in the normal SERUPRO mode.

No WAIT phase occurs:
“SelfActing Serupro” is ended by M30 of the parts program.
The channel status is then in the Reset status again.
A SERUPRO approach does not take place.

Starting a group of channels

If a group of channels is only started with “SelfActing Serupro”, then all channels are ended with “Reset”.

Exceptions:
A channel waits for a partner channel which has not been started at all.

A block search for all channels can be carried out as follows:

- The user or the MMC/HMI chooses the channels which must work together (channel group).
- The user chooses an especially important channel from the channel group for which he wants to select a search target explicitly (target channel).
- The MMC/HMI will then start SERUPRO on the target channel and “SelfActing Serupro” in the remaining channels of the channel group.

The process is completed if **each** channel concerned has deleted “seruproActive”.

NCU-Link SW 6.2 and higher

“Selfacting Serupro” accepts a master channel on another NCU.

© Siemens AG, 2003. All rights reserved

Activation

"SelfActing Serupro" is activated via the

PI service "_N_FINDBL" with
PI service parameter searchMode set to 5.

Prior to the PI service _N_FINDBL, the coding "No search target" must be specified in the OPI block SPARPF as follows:

- The value zero must be entered in the "searchType" variable on **each program level**.

The "searchType" determines how the search target is specified on this level.

The value zero is understood as "No search target" for the appropriate level.

- In addition, the OPI partner (MMC/HMI) can specify the channel "seruproMasterChan" where it assigns the OPI variable "seruproMasterChan" an appropriate value.

Variable "seruproMasterChan" is preset to zero during booting.

2.5.11 System variables updated by the SERUPRO sequence (SW 6.1 and higher)

Detecting the SERUPRO sequence

The SERUPRO sequence can be detected using the following system variables:

$P_ISTEST	is TRUE (this also applies to program test)
$P_SEARCHL	is set to 5 (block search in extended program test)
$AC_ASUP	bit 20 is set in the system ASUB after the search target has been found (SERUPRO sequence, point 11.)

$P_ISTEST AND (5 == $P_SEARCHL) is detected reliably by SERUPRO.

$AC_SEARCH is **not** supplied by the SERUPRO operation.

Note

$P_SEARCHL is set at the beginning of the SERUPRO process and reset on **Reset**. $P_SEARCHL is therefore set in the SERUPRO ASUB and remains set and can be evaluated in the remainder of the parts program.

In contrast, the $P_ISTEST variable is set **only** in the SERUPRO process and is thus suitable for search-specific adaptation of programs.

Synchronized action

In SW 6.2 and higher, SERUPRO can be scanned for is TRUE with system variable

$AC_SERUPRO

In SW 6.4 and higher, REPOS acknowledgments updated by SERUPRO can be scanned using:

"Program-sensitive system variable"	Description
$AC_REPROS_PATH_MODE	Type of repositioning MODE
$AA_REPROS_DELAY	REPOS suppression is currently active for this axis.

© Siemens AG, 2003. All rights reserved

$AC_SERUPRO and $P_ISTEST

If SERUPRO is still active in the main run (SW 6.3 and higher)

Note

A check is carried out to determine when the SERUPRO target block has already been found when system variables $P_ISTEST and $AC_SERUPRO are interpreted. If this is the case, an implicit preprocessing stop is inserted before the two system variables are evaluated.

As a result, interpretation is halted and not continued again until SERUPRO is deactivated in the main run as well. The decision as to whether SERUPRO must be active or inactive is then made correctly.

Classification of system variables

The system variables for SERUPRO are classified according to

- PROGSENSITIVE Program progress in the SERUPRO sequence
- REAL Actual machine status

A parts program reads system variables and saves the values in a buffer memory. The value of a system variable is dependent either on the preceding program block or on the current block. If this block is executed during the SERUPRO sequence, then no axis is really moved. This requires a system variable which includes the program processing status as a result of SERUPRO. This behavior is taken into account only by the socalled “PROGSENSITIVE” system variables which are updated continuously during the SERUPRO sequence.

The actual status of the machine is supplied **only** by the “real system variable”. All classifications **up to SW 6.4** are described below.

“Program-sensitive system variable”	**Description**
$AA_IM	Machine positions: Actual value in MCS
$P_AD	Active tool compensations
$P_ADT	Active tool compensations transformed
$P_TOOLP	Last tool no. programmed TO–T32000
$P_TOOLEXIST	Tool with T no. t exists
$P_D	Current D no. in ISO 2 language mode
$P_H	Current H no. in ISO 2 language mode
“Real system variable”	**Description**
$AA_LOAD and $VA_LOAD	Drive utilization with 611D only
$AA_TORQUE and $VA_TORQUE	Drive torque setpoint with 611D only
$AA_POWER and $VA_POWER	Drive active power with 611D only
$AA_CURR and $VA_CURR	Axis/spindle actual curr. with 611D only
$VA_VALVELIFT	Drive data with 611D hydr. module only
$VA_PRESSURE_A and $VA_PRESSURE_B	Drive data with 611D hydr. module only
$VA_DIST_TORQUE	Disturbing torque/max. motor torque
$AA_FXS	Setpoint status travel to fixed stop
$AN_SETUPTIME a. $AN_POWERON_TIME	Time since last controller power-up
$TC_MAP3, $TC_MPP4 and $TC_MPP6	Tool management magazine data
$TC_MOP2 and $TC_MOP4	Residual service life and residual number of workpieces

© Siemens AG, 2003. All rights reserved

Value assignments for SW 6.4 and higher

System values can be distinguished according to their behavior on SERUPRO To do this, the **SERUPRO** identifier is added to the “description” area of the variable description svar.inp.

The following values can be assigned to this identifier:

- INDEPENDENT Data is independent of SERUPRO
- PROGSENSITIVE Tracks the progress of the program on SERUPRO and may not display the real status of the machine
- REAL Always detects the real machine status
- UNCLASSIFIED Has not yet been classified

Note

The PROGSENSITIVE and REAL identifiers are of particular significance for the SERUPRO block search.

For more detailed information about all known system variables, see
References: /PGA/, Chapter 15 Tables “List of system variables”

Example

The program-sensitive system variable $AA_IM is preprocessed as follows:

System variable:		Explanation:
`name =`	**`"$AA_IM"`**	Variable identifier
`type =`	`DOUBLE`	System variable type
`unit =`	`POSN`	Unit
`readIpo =`	`CO_SYNA_AA_IM_READ_ID`	Read or write
`syncInfo =`	`SYNR`	in synchronized actions
`docu:`		Description

Description in the Programming Guide, Advanced /PGA/:
Use:
$AA_IM[X]
Actual value in machine coordinate system (MCS)
`:end`

```
description:
read = parts program, synchronized action, OPI
preprocessing stop = TRUE
write = FALSE
axis identifier = GEOAX, CHANAX, MACHAX
notAllowed = UNCLASSIFIED
```

SERUPRO = PROGSENSITIVE (conditioning SW 6.4 and higher)

```
range of values = [DBL_MIN, DBL_MAX]
date =
nckVersion =
author = ""
:end
```

© Siemens AG, 2003. All rights reserved

2.5.12 NC functions supported in SERUPRO (SW 6.3 and higher)

Applications

SERUPRO supports the following NC functions:

• Gear step change:	During program test (not fully automatically)
• Linkages:	Setpoint and actual value linkages can be simulated
Master-slave for drives Electronic gear Axial master value coupling	These functions are simulated correctly subject to the relevant restrictions must be considered!
• Coupled motion:	Axis grouping with TRAILON and TRAILOF
• Gantry axes:	Gantry axis grouping traversal
• Tangential control:	Tangential followup of individual axes
• Superimposed movement:	Superimposed motion interpolation
• Travel to fixed stop:	Autom. FXS and FOC in SW 6.3 and higher
• Synchronous spindle:	Synchronous spindle coupling with COUPON
• Autonomous single-axis operations:	Axes controlled by the PLC in SW 6.4 and higher

SERUPRO can be combined with a cascaded block search.

Gear step change

The gear stage change (GSW) function requires the NCK to initiate physical movements in order to engage a new gear. A gear stage change is not necessary during the SERUPRO operation and is performed as follows:

SW 6.1 and lower
Fully automatic gear stage change is not supported. The user must make appropriate adaptations in the PLC and/or in the SERUPRO ASUB. During the SERUPRO process, the VDI interface is operated by the gear stage change. The PLC should be able to acknowledge the requested gear stage change with the new gear stage and engage the last gear stage after the SERUPRO process automatically.

Some gears can only be changed controlled by the NC, since either the axis must oscillate or a certain position must be approached beforehand.

SW 6.3 and higher
The gear stage change can be suppressed selectively in MD 35035: SPIND_FUNCTION_MASK with bits 0 to 2 for DryRun, program testing and SERUPRO.

The gear stage change must then be performed in REPOS; this will work even if the axis involved is to be in “speed control mode” at the target block. In other cases, the automatic gear stage change is denied with an alarm if, for example, the axis was involved in a transformation or coupling between the gear stage change and the target block.

Note

For further information about gear stage changes in DryRun, Program Test and SERUPRO modes, please refer to
References: /FB/, S1, “Spindles” 2.6 Programming

© Siemens AG, 2003. All rights reserved

Couplings

The SERUPRO process is a program simulation in Program Test mode with which setpoint and actual value couplings can still be simulated.

Coupled axes

The following axis links are compatible with the SERUPRO operation:

- Coupled motion “TRAILON” and TRAILOF (SW 6.3 and higher)
- Gantry axes (SW 6.3 and higher)
- Tangential control (SW 6.3 and higher)
- The activatable and deactivatable “master/slave” link as long as activation and deactivation is not selective (SW 6.3 and higher)

Note

The following measure for accelerating the machining velocity is **not** operative for a master axis whose slave axes are configured in another channel
MD 22601: SERUPRO_SPEED_FACTOR = positive

Coupled motion

The motion-synchronous action for coupled motion of an axis grouping with TRAILON and TRAILOF is supported by SERUPRO.

For further information about coupled motion with TRAILON and TRAILOF, please refer to
References: /FB/, M3, “Axis Coupling and ESR”
/PGA/, “Path behavior” and “Motion-synchronous actions”

Gantry axes

Mechanically linked machine axes can be moved without a mechanical offset using the gantry axis function. This operation is simulated correctly with SERUPRO.

For further information about the functionality of gantry axes, please refer to
References: /FB/, G1, “Gantry axes”

Tangential control

The tangential control of individual axes is supported by SERUPRO.

For further information about tangential control, please refer to
References: /FB/, T3, “Tangential control”

Overlaid movement

If “superimposed movements” are used, only the block search via Program Test (SERUPRO) can be used, since the superimposed movements are interpolated accordingly in the main run. This applies in particular to $AA_OFF.

During Program test, a velocity profile must be used which allows “superimposed movements” to be interpolated during the main run. It is thus not possible to interpolate at the maximum axis velocity.

The axis velocity is set in “Dry run feedrate” mode via
SD 42100: DRY_RUN_FEED.

The velocity of the SERUPRO process is selected via
MD 22600: SERUPRO_SPEED_MODE.

© Siemens AG, 2003. All rights reserved

Acceleration measures

The processing speed of the entire SERUPRO process can be substantially accelerated using the machine data below.
MD 22600: SERUPRO_SPEED_MODE and
MD 22601: SERUPRO_SPEED_FACTOR

With MD 22600: SERUPRO_SPEED_MODE == 1, the SERUPRO process will run at the “dry run feedrate”.

With MD 22600: SERUPRO_SPEED_MODE == 0,
MD 22601: SERUPRO_SPEED_FACTOR is evaluated,
and another acceleration is permitted. Dynamic monitoring functions are disabled in this mode.

SPEEDFACTOR for channel axes in the main run
Machine data MD 22600: SERUPRO_SPEED_MODE is effective for the following channel axes in the main run throughout the entire SERUPRO process:

- PLC axes
- Command axes
- Positioning axes
- Reciprocating axes

The functions of MD 22600: SERUPRO_SPEED_MODE and
MD 22601: SERUPRO_SPEED_FACTOR apply only to SERUPRO and not program testing. No axes/spindles are moved.



Caution

The NC as a discrete system generates a sequence of interpolation points. The velocity values configured in MD 22601: SERUPRO_SPEED_FACTOR must be such that all sections of the program can executed, i.e. they must not be so high that the machine is forced to operate at an excessive speed.

Effect during DryRun in SW 6.2 and higher

An active SERUPRO SPEED-FACTOR has the following effect on DryRun:

- DryRun is activated simultaneously.

  This causes a switch from G95/G96/G961/G97/G971 to G94 in order to execute G95/G96/G961/G97/G971 as quickly as possible.

- Tapping and thread cutting are performed at the usual velocity for DryRun.

DryRun and SERUPRO affect the spindle/axis with the following G codes:

- G331/G332 causes the spindle to be interpolated as an axis in a path grouping. In the case of tapping, the drilling depth (e.g. axis X), the pitch and speed (e.g. spindle S) are specified.

  In the case of DryRun, the velocity is specified by X, the speed remains constant and the pitch is adjusted.

  Following the SERUPRO simulation, the position for spindle S deviates from the normal position because the spindle S has rotated less during simulation.

© Siemens AG, 2003. All rights reserved

FXS Travel to fixed stop

In SW 6.2 and higher, the functionality of FXS is repeated automatically with REPOS and is designated as FXS REPOS. Every axis is taken into account and the torque last programmed before the search target is applied.

Furthermore, the meaning of system variable $AA_FXS is redefined for SERUPRO as follows:

- $AA_FXS displays the current status of program simulation.
- $VA_FXS always describes the real machine status.

The two system variables $AA_FXS and $VA_FXS have the same values continuously **outside** the SERUPRO function.

The user can treat FXS and FOC as special commands in a SERUPRO ASUB.

Example of FXS

The following example shows the main program “Travel to fixed stop”:

Main program
fxsSerupro.mpf

```
DO $R0=0                                ; Delete FXS flag R0
N10 G0 G90 X0 Y0 Z0 FXS[X]=0            ;

N15 ID=1 WHEN $AA_IW[X] > 45 DO FXS[X]=15 FXST[X]=22 ;

N20 FXST[AX1] = 10.0 FXSW[AX1] = 25.0
N30 G1 G91 F1000 X30                    ;
WHEN $AA_IW[X] > 35 DO FXST[X] = 5 FXSW[X]=20 ;
DO $R0=1                                ; Enable FXS flag synchronously with N40
N40 FA[X]=100 POS[X]=200 FXS[X]=1       ;

N100 F1000 Y20                          ;
N110 F2000 Z100                         ;

DO $R0=0                                ; Disable FXS flag synchronously with N200
N200 FXS[X]=0 G90 X–10 F1000            ;
                                        :
N250 F2000 Y–10 Z–10                    ;

DO $R0=1                                ; Enable FXS flag synchronously with N300
N300 G91 FXST[X]=5 FXS[X]=1 POS[X]=300 FA[X]=100 ;
                                        :
N350 G90 F1000 Y50 Z50                  ;

DO $R0=0                                ; Disable FXS flag synchronously with N400
N400 FXS[X]=0 X0 F1000                  ;

N500 G0 G90 X0 Y0 Z0                    ;
N9999 M30                               ;
```

ASUB
fxsSeruproAsup.mpf

```
                                        ;
N1000 when (R[0]==1) AND ($AA_FXS[X]==0)
                                        ; Evaluate FXS flag R0 whether during
DO $R0=1 FXS[X]=1 FXST[X]=12            ; the Serupro process FXS has been enabled
N2000 when (R[0]==0) AND ($AA_FXS[X]==1)
                                        ; or was deactivated
DO $R0=0 FXS[X]=0 FXSW[X]=21            ; in order to enable or disable FXS
N1020 REPOSA                            ; for the REPOSA block
```

© Siemens AG, 2003. All rights reserved

Machine data extract
for the program fxsSerupro.mpf

```
                                        ;
                                        ; Allow ASUB start from the
                                        ; stopped state
$MN_ASUP_START_MASK = 'H03'
$MN_ASUP_START_PRIO_LEVEL=100

M17
```

FXS and FOC SW 6.1 and lower

SERUPRO approach does not automatically take account of the FXS instruction and travel with limit torque/force FOC (Force Control).

If FXS “Travel to fixed stop” is programmed between the beginning of the program and the search target, the instruction is not executed by the NC. The actual status of the function does not change and the motion is only simulated up to the end point.

The user can log the enabling/disabling of FXS in the parts program and enable or disable FXS in the SERUPRO ASUB (see point 10.)

FOC SW 6.2 and higher

The meaning of system variable $AA_FOC is redefined for SERUPRO as follows:

- $AA_FOC represents the current status of program simulation.
- $VA_FOC always describes the real machine status.

The FOC REPOS function behaves analogously to the FXS REPOS function. The functionality of FOC is repeated automatically with REPOS and designated FOC REPOS below. Every axis is taken into account and the torque last programmed before the search target is applied.

Condition

A continuously changing torque characteristic can **not** be implemented with FOC REPOS.

Example:
A program moves axis X from 0 to 100 and activates FOC every 20 increments for 10 increments at a time. This torque characteristic is usually generated with nonmodal FOC and cannot therefore be traced by FOC REPOS. FOC REPOS will traverse axis X from 0 to 100, with or without FOC, according to the last programming.

Note

For further information about the SERUPRO block search in relation to FXS or FOC, please see

References: /FB/, F1, “Travel to fixed stop” 2.2 General functionality.

Synchronous spindle SW 6.3 and higher

The synchronous spindle operation with main spindle and any number of following spindles can be simulated in all existing channels with SERUPRO.

For further information about synchronous spindles in SW 4 and higher, please refer to
References: /FB/, S3, “Synchronous Spindle”

© Siemens AG, 2003. All rights reserved

Restrictions up to SW 6.3

SERUPRO supports the following NC functions subject to certain restrictions:

NCK functionality	Restrictions
Master-slave for drives and SERUPRO	Selective enabling and disabling of the master-slave link with MASLON active
Axis enables and SERUPRO	Real servo enable missing during program testing
Axis replacement and SERUPRO	Axes traversing as path axes before RELEASE are ignored on REPOS
Electronic gear (EG)	The electronic gear link is not simulated correctly with EGON, EGONSYN, EGONSYNE during program testing
Axial master value coupling	The link with LAEDON and LEADOF is not supported

Master-slave and SERUPRO

The link status should only be updated without calculating the associated positions during block search.

A system ASUB can be started automatically when the block search is finished. In this subroutine, the user can control the link status and the associated axis positions subsequently. The information needed can be read from additional block search system variables.

System variable for master-slave

The following system variables are needed for the position offset between the axes to be coupled with the desired link status:

NCK variable in SW 6.1 and higher	Description
$P_SEARCH_MASLD[X] Slave*	Position offset between slave and master axis when the link is closed.
$P_SEARCH_MASLC[X] Slave*	Current status of a master-slave link was changed during block search.
$AA_MASL_STAT[X] Slave*	Current status of a master-slave link active.

Slave* for slave axis identifier

Note

This block search for the master-slave link is effective only if the position offset between the axes can be determined.

In order to determine the programmed positions, the axes to be linked must be in the same channel at the time of the block search. If this is not the case, the block search is aborted with alarm 15395.

Variables $P_SEARCH_MASLD, $P_SEARCH_MASLC and $AA_MASL_STAT are cleared on MASLON.

For further information on master-slave link please refer to:
References: /FB/, TE3, “Speed/Torque Coupling, Master-Slave”

© Siemens AG, 2003. All rights reserved

System ASUB

The system ASUB is called progevent.spf and must be available in the /_N_CMA_DIR directory. The contents might be as follows:

```
progevent.spf
; X=master axis, Y=slave axis

N10 IF(($S_SEARCH_MASLC[Y]< >0) AND ($AA_MASL_STAT[Y]< >0))
N20 MASLOF(Y)
N30 SUPA Y=$AA_IM[X]–$P_SEARCH_MASLD[Y]
N40 MASLON(Y)
N50 ENDIF
N60 REPOSA
```

The following machine data must be set to enable the ASUB to be started automatically:
$MN_ASUP_START_MASK = 'H03'
$MN_ASUP_START_PRIO_LEVEL = 100
$MN_SEARCH_RUN_MODE = 'H02'

Axis enables and SERUPRO

The axial IS "Enable traversing, program testing" (DB31, ... DBX3.7) affects axis enable signals when no control enables are to (or can) be issued at the machine and is active only during the program test or when SERUPRO is active.

It is possible to issue this enable via the interface signal PLC→NCK IS "Enable traversing, program testing" (DB31, ... DBX3.7). If the real servo enable is missing during program testing or SERUPRO, the effect on the axes/spindles is as follows:

- As soon as the simulated program run intends to move an axis/spindle, the message "Waiting for axis enable" or "Waiting for spindle enable" is displayed and the simulation is stopped.
- If the VDI signal IS "Enable traversing, program testing" (DB31, ... DBX3.7) is canceled again, alarm 21612: "Channel %1 axis %2 VDI signal 'servo enable' reset during motion" is activated.

Autonomous single-axis operations SW 6.4 and higher

Autonomous single-axis operations are axes controlled by the PLC that can also be simulated on SERUPRO. During SERUPRO operation, as in normal operation, the PLC can take over or give up control of an axis. If required, this axis can also be traversed using FC18. The PLC takes over control of the axis **BEFORE** the approach block and is responsible for positioning this axis. This is valid for all block search types.

For further information about autonomous single-axis operations, please refer to
References: /FB/, P2, "Positioning Axes"

Axis replacement and SERUPRO

After SERUPRO, an axis should be neutral. The user can suppress the REPOS movement for that axis completely in the SERUPRO ASUB. See Subsection 2.5.13.



Caution

Axes which could be moved as path axes before the RELEASE command must also be capable of path axis motion in REPOS.

The axis controlled by the PLC is still **not** repositioned (up to SW 7.1)!

© Siemens AG, 2003. All rights reserved

Axis coupling in SW 6.4 and higher

A programmable interrupt pointer available in SW 6.4 and higher can be used to prevent simulation of program point.

Please refer to 2.5.7 “Preventing program point for SERUPRO”

The functions that did not work properly up to SW 6.3, such as electronic gear and axial leading-value coupling can be used under certains conditions in SW 6.4 and higher with user-supported simulation and approach strategies for SERUPRO.

Specifications for EG simulation

For simulation of EG, the following definitions apply:

1. Simulation always takes place with setpoint coupling.
2. If under SERUPRO there are only some, i.e. not all, leading axes, the simulation is aborted with alarm 16952 “Reset Clear/No Start”. This can occur with multi-channel couplings.
3. Axes that have only one encoder from the point of view of the NCK and are operated with external data, cannot be simulated correctly. These axes mut not be integrated in couplings.



Caution

In oder to simulate a coupling correctly, it must be deactivated berforehand. This can be done with machine data MD 10708: SERUPRO_MASK.

Specifications for active couplings

The SERUPRO procedure simulates axis couplings always assuming that they are setpoint couplings. In this way, the end points are calculated for **all** axes that are used as target points for SERUPRO approach. The coupling is also active with “Search target found”. The path from the current point to the end point is carried out for SERUPRO approach with the active coupling.

LEADON

The following specifications apply for the simulation of leading-value couplings:

1. Simulation always takes place with setpoint coupling.
2. SERUPRO approach takes place with active coupling and a overlaid motion of the following axis in order to reach the simulated target point.

The following axis that is moved solely by the coupling cannot always reach the target point. In SERUPRO approach, a overlaid linear motion is calculated for the following axis to appraoch the simulated point!

Reaching simulated target point for LEAD with JOG

At the time of “Search target found”, the coupling is already active, especially for the JOG movements. If the target point is not reached, SERUPRO approach can be used to traverse the following axis with active coupling and an overlaid motion manually to the target point.

Note

For further information on restarting axis couplings, see
Subsection 2.5.15 “Continue machining after SERUPRO search target found”.

© Siemens AG, 2003. All rights reserved

2.5.14 SERUPRO ASUB (SW 6.3 and higher)

SERUPRO ASUB

The following NC functions are supported by the user in the SERUPRO ASUB:

- Reference point approach: Referencing via parts program G74
- Tool management: Tool change and magazine data
- Spindle ramp-up: On starting a SERUPRO ASUB
- Determine spindle status: Program new values at any point in the program before REPOSA

G74 Reference point approach

If the statement G74 (reference point approach) is programmed between the program start and the search target, this will be ignored by the NC.

SERUPRO approach will not take into account this statement G74!

Tool management

If the tool management is active, the following setting is recommended:

MD 18080: TOOL_MANAGEMENT_MASK, set BIT 20 = 0.

The tool management command generated during the SERUPRO process is thus **not** output to the PLC!
The tool management command has the following effect:

- The NC acknowledges the commands automatically.
- No magazine data are changed.
- The tool data are not changed.
  Exception:
  The tool status of the tool enabled during the test mode can take the status 'active'. In this case, the wrong tool can be mounted on the spindle after the SERUPRO process.
  Remedy:
  The user starts a SERUPRO ASUB that is traversed really. Prior to the start, the user can start an ASUB changing the correct tool.

SERUPRO process: Functionality: The sequence from point 5 to 9.
SERUPRO-ASUB: Functionality: The sequence of point 10.

Bit 11 must also be set to 1 in machine data
MD 18080: TOOL_MANAGEMENT_MASK, because the ASUB may have to repeat a T selection.

Systems with tool management and auxiliary spindle are not supported by SERUPRO!

© Siemens AG, 2003. All rights reserved

Example

Tool change subroutine

```
PROC L6                                ; Tool change routine
                                       ;
N500 DEF INT TNR_AKTUELL               ; Variable for active T number
N510 DEF INT TNR_VORWAHL               ; Variable for the preselected T number

                                       ; Determine current tool
N520 STOPRE                            ; The current tool is read
N530 IF $P_ISTEST                      ; from the program context
N540 TNR_AKTUELL = $P_TOOLNO           ; during the Program Test mode.
N550 ELSE                              : Otherwise, the tool of the spindle
                                       ; is read out.

N560 TNR_AKTUELL = $TC_MPP6[9998,1]
                                       ; Read T number of the tool on the spindle
N570 ENDIF                             ;
                                       ;
N580 GETSELT(TNR_VORWAHL)              ; Read T number of the preselected tool of the
                                       ; master spindle.
                                       ; Carry out tool change only
                                       ; if tool is not yet current
N590 IF TNR_AKTUELL <> TNR_VORWAHL

                                       ; Approach tool change point
N600 G0 G40 G60 G90 SUPA X450 Y300 Z300 D0
N610 M206                              ; Execute tool change
N620 ENDIF                             ;
N630 M17                               ;
```

ASUB for calling the tool change routine after block search type 5

```
PROC ASUPWZV2

N1000 DEF INT TNR_SPINDEL              ; Variable for active T number
N1010 DEF INT TNR_VORWAHL              ; Variable for preselected T number
N1020 DEF INT TNR_SUCHLAUF             ; Variable for the T number determined during the
                                         block search

N1030 TNR_SPINDEL = $TC_MPP6[9998,1]
                                       ; Read T number of the tool on the spindle

N1040 TNR_SUCHLAUF = $P_TOOLNO
                                       ; Read T no. determined by block search, i.e. this
                                       ; tool determines the current tool compensation

N1050 GETSELT(TNR_VORWAHL)             ; Read T number of the preselected tool

N1060 IF TNR_SPINDEL ==TNR_SUCHLAUF GOTOF ASUP_ENDE1
N1070 T = $TC_TP2[TNR_SUCHLAUF]        ; T selection via tool name
N1080 L6                               ; Call tool change routine
N1085 ASUP_ENDE1:                      ;
N1090 IF TNR_VORWAHL == TNR_SUCHLAUF GOTOF ASUP_ENDE
N1100 T = $TC_TP2[TNR_VORWAHL]         ; Restore T preselection via tool name

N1110 ASUP_ENDE:                       ;
N1110 M90                              ; Checkback to PLC
N1120 REPOSA                           ; End of ASUB
```

In both programs PROC L6 and PROC ASUPWZV2, the tool change is programmed with M206 instead of M6.
The ASUB program “ASUPWZV2” uses various system variables that, on the one hand, know the program progress ($P_TOOLNO) and, on the other hand, represent the current status of the machine ($TC_MPP6[9998,1]).

© Siemens AG, 2003. All rights reserved

Spindle ramp-up

When the SERUPRO ASUB is started, the spindle is not accelerated to the speed specified in the program because the SERUPRO ASUB is intended to move the new tool into the correct position at the workpiece after the tool change.

A spindle ramp-up is performed as follows with the SERUPRO ASUB:

- SERUPRO operation has finished completely.
- The user starts the SERUPRO ASUB via function block FC 9 in order to ramp up the spindle.
- The start after M0 in the ASUB does not change the spindle status.
- SERUPRO ASUB automatically stops before the REPOS parts program block.
- The user presses START.
- The spindle accelerates to the status in the target block if the spindle was not programmed differently in the ASUB.

Note

Modifications for spindle REPOS:
The transitions of speed control mode and positioning mode must be considered in the event of modifications for SERUPRO approach and spindle functionality.

For further information about mode changes for spindles, please refer to
References: /FB/, S1, “Spindles” 2.1 Spindle modes

Detect spindle status in the ASUB

The spindle status to apply at the target block is determined in the ASUB using
$P_SEARCH_S
$P_SEARCH_SDIR
$P_SEARCH_SGEAR
$P_SEARCH_SPOS
$P_SEARCH_SPOSMODE

The spindles can subsequently be programmed with the correct values at any point in the program before REPOSA in the SERUPRO ASUB.

Note

Please note that these variables can **also** contain movements as command or PLC axes. $P_SEARCH_SPOS then contains the end point of the last command or PLC axis movement.
$P_SEARCH_SPOSMODE can be used to detect a replaced axis.

© Siemens AG, 2003. All rights reserved

2.5.15 Continue machining after SERUPRO search target found

User information

Regarding the REPOS process

REPOS is generally used to interrupt an ongoing machining operation and to continue machining after the interruption.

With approach in SERUPRO, however,
a program section must be added. This is the case when SERUPRO has terminated the simulation and is to travel to the target block again. SERUPRO refers to the existing REPOS function that can be adapted by the user in **SW 6.3 and higher**.

The user can change the REPOS response of individual axes at selective times to reposition certain axis types either ealier, later or not at all. This affects SERUPRO approach in particular. Restart movements of some axes can also be controlled independently of SERUPRO approach during the REPOS procedure.



Danger

The **REPOS procedure** moves all axes in a block from the current position to the target start-of-block point. During this process, the NC **cannot** detect any collisions can detect possible collisions with the machine or the workpiece!
Protection zones and software limits are monitored.

The path axes are repositioned together in one block. The spindles and positioning axes are repositioned in the next block.
Neutral axes are **not** repositioned in versions up to SW 6.3.

Set REPOS response

By setting the bits of machine data **MD 11470: REPOS_MODE_MASK**, you can control the behavior of the NC during repositioning.

- Bit 0 = 1 The hold time is resumed at the point of interruption in the residual repositioning block.
- Bit 1 = 1 Reserved
- Bit 2 = 1 Prevent repositioning of individual axes via VDI signals.
- Bit 3: Reposition positioning axes in the approach block via program test (SERUPRO).
- Bit 4 = 1 Positioning axes in approach block on every REPOS.
- Bit 5 = 1 Modified feedrates and spindle speeds are already valid in the residual block. Otherwise, in next block.

In SW 6.4 and higher, the behaviour for repositioning is extended as follows:

- Bit 6 = 1 After SERUPRO, neutral axes and positioning spindles in the approach block are repositioned as command axis.
- 7 = 1 The behavior of IS “REPOSDELAY” (DB31, .. DBX10.0) is read when REPOSA is interpreted. Axes that are neither geometry axes or orientation axes are then excluded by REPOS and not moved.

© Siemens AG, 2003. All rights reserved

Reapproach with controlled REPOS

At any point during processing, a parts program can be interrupted and an ASUP with REPOS started.

For path axes, the REPOS mode can be controlled by the PLC via VDI signals to restart at the contour. This mode is programmed in the parts program and defines the approach response. For information see Subsection 2.5.16.

The REPOS response on individual axes can also be controlled via VDI signals and is enabled with MD 11470: REPOS_MODE_MASK BIT 3 == 1. Path axes cannot be influenced individually. For all other axes that are not geometry axes, repositioning of individual axes can be prevent temporarily and also delayed. VDI signals can be used to enable or keep blocking individual channel axes that REPOS would like to traverse at a later time.



Danger

"Delete axial distance-to-go" gives rise to the following risk for "Prevent repositioning individual axis" when MD 11470: REPOS_MODE_MASK (bit 0==1).
As long as an axis is programmed **incrementally** after the interruption, the NC approaches positions different from those approached with no interruption (see example).

Example: Axis is programmed incrementally

Axis A is positioned at 11 degrees before the REPOS operation; the programmed operation in the interruption block (target block for SERUPRO) specifies 27 degrees. Any number of blocks later, this axis is programmed to move incrementally through 5 degrees with N1010 POS[A]=IC(5) FA[A]=1000.

On IS "REPOSDELAY" (DB31, ... DBX10.0), the axis does not traverse in the REPOS operation and is moved to 32 degrees in N1010.
(The user may need to acknowledge the path from 11 to 27 degrees explicitly).

Caution: The axis is programmed incrementally after the interruption. In the example, the NC moves to 16 degrees instead of 32 degrees.

Start axes individually

The REPOS response for SERUPRO approach with several axes is selected with MD 11470: REPOS_MODE_MASK BIT 3 == 1.

The NC starts SERUPRO approach at a block which moves **all**

positioning axes	to the programmed end and
all path axes	to the target block.

The user starts the individual axes by selecting the appropriate feedrate enables. The target block motion is then executed.

Reposition in restart block
Positioning axes are **not** repositioned in the distance-to-go block but in the restart block and not only affect the block search via program test on SERUPRO approach. When MD 11470: REPOS_MODE_MASK:

Bit 3 = 1	for block search via program test (SERUPRO)
Bit 4 = 1	for each REPOS

Note

If neither bit 3 nor bit 4 is set, "non-path axes" are repositioned in the remaining block in this phase.

© Siemens AG, 2003. All rights reserved

Prefer or ignore REPOS

Further REPOS adaptations can be made by setting the bits in MD 11470: REPOS_MODE_MASK:

- Bit 5 = 1 Modified feedrates and spindle speeds are already valid in the remaining block and are therefore preferred. This response refers to each REPOS procedure.
- Bit 6 = 1 Neutral axes and positioned spindles are repositioned after SERUPRO. Neutral axes that are not repositioned further must be considered NST “REPOSDELAY” (DB31, ... DBX10.0). This deletes the REPOS movement.
- Bit 7 = 1 Axes that are neither geometry or orientation axes are not controlled with IS “REPOSDELAY” (DB31, ... DBX10.0). So these axes are acluded from the REPOS procedure and are not moved.

Delayed approach of axis with REPOS offset

The axial pretriggered VDI signal axis/spindle (PLC→NCK) IS “REPOSDELAY” (DB31, ... DBX10.0) is used, with the edge of IS “REPOSMODEEDGE” (DB21, ... DBX31.4) to traverse the REPOS offset for this axis only after the next time it has been programmed.

Whether this axis is currently subject to a REPOS offset, can be queried via synchronized actions **in SW 6.4 and higher** with $AA_REPOS_DELAY.



Caution

IS “REPOSDELAY” (DB31, ... DBX10.0) has no effect on machine axes that form a path. IS “Path axis” (DB31, ... DBX76.4) determines whether an axis is a path axis.

Transfer time for REPOS VDI signals

With the 0/1 edge of channel-specific VDI signal (PLC→NCK) IS “REPOSMODEEDGE” (DB21, ... DBX31.4) the level signals from IS “REPOSMODE0–2” (DB21, ... DBX31.0–31.2) and IS “REPOSDELAY” (DB31, ... DBX10.0) are transferred to the NC. The levels refer to the current block in the main run and vary in the following two cases:

Case A: **One** reapproach block of a currently active REPOS operation is contained in the main run.

The active REPOS operation is aborted, restarted and the REPOS offsets controlled via signals IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) and IS “REPOSDELAY” (DB31, ... DBX10.0).

Case B: **No** reapproach block of a currently active REPOS operation is contained in the main run.

Each future REPOS operation wishing to reapproach the current main program block is controlled by the level of IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) and IS “REPOSDELAY” (DB31, ... DBX10.0).

© Siemens AG, 2003. All rights reserved

Note

In the current ASUB, IS “REPOSMODEEDGE” (DB21, ... DBX31.4) does **not** affect the final REPOS, unless this signal applies to the REPOS blocks. In Case A, the signal is only allowed in stop mode.

With VDI signals to control SERUPRO approach

SERUPRO approach can be used with
IS “REPOSMODEEDGE” (DB21, ... DBX31.4) and the associated signals in the following phases:

- Between “Search target found” and “Start SERUPRO ASUB”
  From: “SERUPO ASUB stops automatically before REPOS”
  to: “Target block is being executed”
- While the SERUPRO ASUB is being executed, e.g. in the program section before the REPOS operation, IS “REPOSMODEEDGE” (DB21, ... DBX31.4) has no effect on SERUPRO approach.

REPOS operations with VDI signals

Control REPOS with VDI interface signals

REPOS offsets can be positively influenced with the following channel-specific VDI interface signals **from the PLC**:

IS “REPOSPATHMODE0 to 2” (DB21, ... DBX31.0–31.2)	channel-specific *
IS “REPOSMODEEDGE” (DB21, ... DBX31.4)	channel-specific
IS “REPOSDELAY” (DB31, ... DBX10.0)	axis/spindle*

Machine axes which form a path are **not** affected by this axial IS.

* These signals are available in the relevant HMI/MMC or PLC DB.

IS “REPOSDELAY” (DB31, ... DBX72.0)	axis/spindle

REPOS acknowledgment signals

The following VDI signals can be used to acknowledge **from the NCK** functions that control the REPOS response via PLC:

IS “REPOSMODEEDGEACKN” (DB21, ... DBX319.0)	channel-specific
IS “Repos Path Mode Ack0–2” (DB21, ... DBX319.1–319.3)	channel-specific
IS “Repos DEFERRAL Chan” (DB21, ... DBX319.5)	channel-specific
NST “Repos offset” (DB31, ... DBX70.0)	axis/spindle
IS “Repos offset valid” (DB31, ... DBX70.1)	axis/spindle
NST “Repos Delay Ack” (DB31, ... DBX70.2)	axis/spindle
IS “Path axes” (DB31, ... DBX76.4)	axis/spindle

REPOS acknowledgment procedures

The channel-specific VDI signal IS “REPOSMODEEDGEACKN” (DB21, ... DBX319.0) is used to establish a handshake. For this purpose, IS “REPOSMODEEDGE” (DB21, ... DBX31.4) is detected by the NC and acknowledged with IS “REPOSMODEEDGEACKN” (DB21, ... DBX319.0).

A REPOSMODE specified by the PLC is acknowledge by the NCK with
IS “Repos Path Mode Ack0 to 2” (DB21, ... DBX319.1–319.3) and
IS “Repos Delay” (DB31, ... DBX10.0) with
IS “Repos Delay Ack” (DB31, ... DBX70.2) in the following way:

A parts program is stopped on N20. Time 2 Stop

© Siemens AG, 2003. All rights reserved

The NCK stops after the braking ramp. After the PLC has specified the REPOSPATHMODE, the NCK accepts in time 3 the REPOSPATHMODE with the 0/1 edge from REPOSMODEEDGE “Repos Path Mode Ack” remains set until
the ASUP is triggered. Time 4
The REPOS command is started in the ASUP. Time 5
The remaining block of the ASUP is used again Time 6

Note

If IS “REPOSMODEEDGE” (DB21, ... DBX31.4), is **not yet** acknowledged with IS “REPOSMODEEDGEACKN” (DB21, ... DBX319.0), a RESET in this situation causes the program to abort and the REPOS that is to be used to control the REPOSPATHMODE can no longer take place.

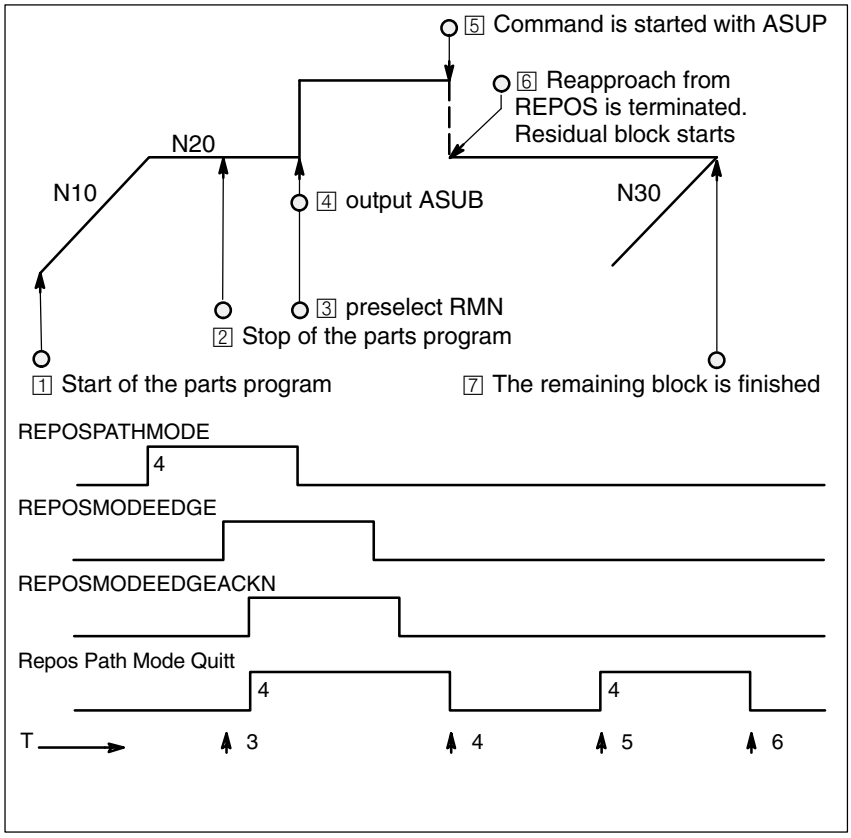


Fig. 2-6 REPOS sequence in parts program with timed acknolwedgement signals from NCK

NCK set the acknowledgement again

Phase with REPOSPATHMODE still active (remaining block of the program stopped at Time 2 is not yet completed).

A soon as the REPOS restart movement of the ASUPs is processed, the NCK sets “Repos Path Mode Ack” again. Time 5
If no REPOSPATHMODE has been selected via VDI signal, the programmed REPOS mode is displayed.

© Siemens AG, 2003. All rights reserved

"REPOS Path Mode Ack" is canceled on activation of the remaining block. Time 6
The parts program block N30 that follows the one in Time 2 is continued.

IS "REPOS Delay Ack" (DB31, ... DBX70.2) is defined as analog.

IS "REPOS offset valid" (DB31, ... DBX70.0)=1, if
IS "REPOS Path Mode Ack 0 to 2" (DB21, ... DBX319.1–319.3)=4 (RMN).

Valid REPOS offsets

When the SERUPRO operation is finished, the user can read out the REPOS offset via the axis/spindle VDI signal (NCK→PLC)
IS "REPOS offset" (DB31, ... DBX70.0) The effects of this signal on the relevant axis are as follows:

Value 0:	The axis does not have to traverse the REPOS offset.
Value 1:	The axis will traverse the REPOS offset.

Scope:
IS "REPOS Delay Ack" (DB31, ... DBX70.0) is supplied at the end of the SERUPRO process. The REPOS offset is invalidated at the start of a SERUPRO ASUB or the automatic ASUB start.

Updating the REPOS offset within the valid scope:
The axis can be moved in JOG mode between the SERUPRO end and Start. The user moves the axis manually through the REPOS offset to set IS "REPOS offset" (DB31, ... DBX70.0) to value 0.
The axis can also be moved via FC18 within the valid scope, during which IS "REPOS offset" (DB31, ... DBX70.0) is updated continuously.

Displaying the scope:
The valid scope of the REPOS offset is displayed with IS "REPOS offset valid" (DB31, ... DBX70.1). It is indicated whether the REPOS offset calculation was valid or invalid:

Value 0:	REPOS offset calculated correctly for this axis.
Value 1:	REPOS offset not calculated correctly for this axis.

REPOS offset after axis exchange
To simplify operation of all axes, the current controlled axis is reassembled after an exchange for each channel with
IS "REPOS DEFERRAL Chan" (DB21, ... DBX319.5).

Value 0	All axes currently controlled by this channel don't have a REPOS offset of their REPOS offsets are not valid.
Value 1	Other.

REPOS offset for synchronized spindle coupling
On restart with SERUPRO, feed starts at the point where interrupted. If a synchronous spindle coupling was already synchronized, there is no REPOS offset of the following spindle and no synchronization path is present. The synchronization signals remain set.

Search target found on block change

The axial VDI signal NCK→PLC NST "Path axis" (DB31, ... DBX76.4) makes the appropriate path axis signals available for the relevant axis type. This signal shows the status of the current block to be executed during block change. Subsequent status changes are ignored. Once the SERUPRO operation is ended with "Search target found", IS "Path axis" (DB31, ... DBX76.4) refers to the target block.

© Siemens AG, 2003. All rights reserved

2.5.16 Reapproach contour with controlled REPOS (SW 6.3 and higher)

Influence path axes individually

During SERUPRO approach, a REPOS operation is initiated in order to reapproach the contour. A large number of axes which the user can control by means of interface signals is frequently moved. The operator panel interface supplies the offsets per channel axis which REPOS intends to traverse.

Restart of individual path axes can be controlled by the PLC and therefore has priority over the actual RMI, RMB and RME commands in the parts program.

Reapproach with RMN

Like RMI, RMB and RME, RMN (REPOS Mode Next) is redefined for SERUPRO approach (SW 6.3 and higher). After an interruption, the reapproach block is not started again completely with RMN, but is merely executed as follows from the next path point:

At the time REPOSA is interpreted, position (B) is referenced in order to find point C at the interruption block with the shortest distance to B. The reapproach block moves from B to C to the end position.

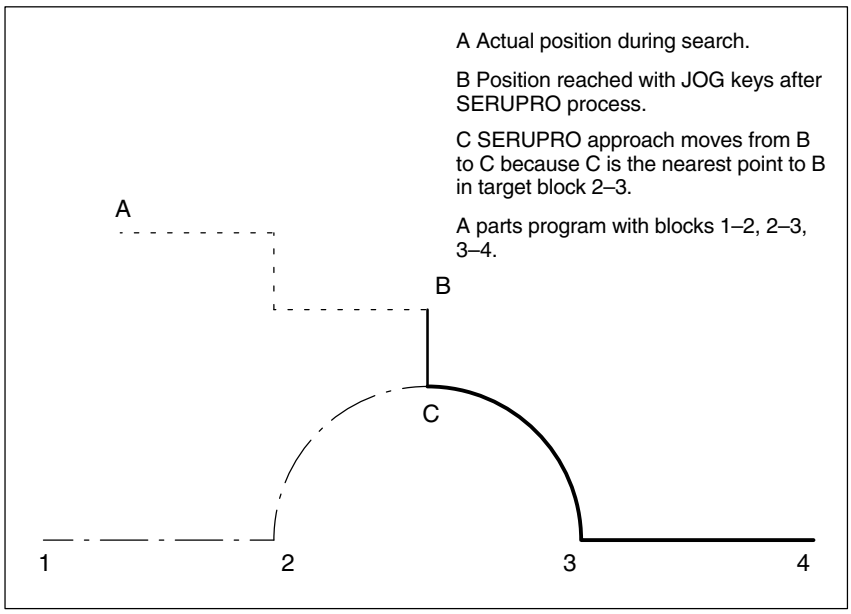


Fig. 2-7 SERUPRO approach with RMN

Application and procedure

SERUPRO approach with RMN offers the opportunity of using the application shown in Fig. 2-7: If a program abort is forced by RESET at any point when the program is advancing from block 2 to 3,

- RMN is used to approach the abort location by the shortest route in order to process just the distance to go from C–3 and 3–4. The user starts a SERUPRO process at the interruption block and uses the JOG keys to move in front of the problem component of the target block.
- B–2, 2–3, 3–4 are always approached with RMI and RMB and the target block thus repeated once completely.

© Siemens AG, 2003. All rights reserved

REPOS-Mode selection

The 3 bits of channel-specific VDI signal PLC→NCK)
IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) can be selected with the 3 bits of the relevant function RMB, RMI, RME or RMN.

Reapproach point

RMNOTDEF	REPOS mode not redefined
RMB	Reapproach start or end of block
RMI	Reapproach interruption point
RME	Reapproach end of block
RMN	Reapproach next point on path

“REPOSPATHMODE” (DB21, ... DBX31.0–31.2)=0 corresponds to **RMNOTDEF**
“REPOSPATHMODE” (DB21, ... DBX31.0–31.2)=1 corresponds to **RMB**
“REPOSPATHMODE” (DB21, ... DBX31.0–31.2)=2 corresponds to **RMI**
“REPOSPATHMODE” (DB21, ... DBX31.0–31.2)=3 corresponds to **RME**
“REPOSPATHMODE” (DB21, ... DBX31.0–31.2)=4 corresponds to **RMN**

With IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) = 0 nothing is overwritten and the current program is valid. The interface signal responds to the level of the corresponding mode.

Note

RMN is a general REPOS extension and is not restricted to SERUPRO. RMI and RMB are identical with respect to SERUPRO.

IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) is used to influence the path as a whole. The path axes cannot be changed individually.

The behavior of the other axis types can be changed individually with IS “REPOSDELAY” (DB31, ... DBX10.0). This REPOS offset is not applied immediately, but only when it is next programmed.

For further information about programming the reapproach point, please refer to
References: /PGA/, “Path traversing behavior” Reapproach contour

---

Read REPOS mode in synchronous actions (SW 6.4 and higher)

The valid REPOS mode of the interrupted block can be read via synchronized actions using the system variables $AC_REPOS_PATH_MODE=

0: not defined	Reapproach not defined
1: RMB	Reapproach at start.
2: RMI	Reapproach to interrupt point.
3: RME	Reapproach to block end point.
4: RMN	Reappraoch to geometrical next path point of interrupted block.

© Siemens AG, 2003. All rights reserved

2.6 Program operation mode

Definition

The term Program operation mode refers to the state where parts programs or parts program blocks are processed in AUTOMATIC or MDA mode.

Channel control

Every channel can be controlled by means of interface signals from the PLC. These can be either mode group-specific or channel-specific interface signals. An overview of these signals is given in Chapter 5 of this Description of Functions.

Status messages

Each channel reports its current program operation status to the PLC with interface signals. These signals are divided up into mode group-specific and channel-specific signals. An overview of these signals is given in Section 7.5 “Interface signals” in this Description of Functions.

2.6.1 Initial settings

Initial settings

Initial settings can be programmed in channel-specific machine data for each channel. These initial settings affect, for example, G groups and auxiliary function output.

Output of auxiliary functions

The timing for output of auxiliary functions can be predetermined via machine data AUXFU_x_SYNC_TYPE (MD 22200, 22210, 22220, 22230, 22240, 22250, 22260), (output timing for M, S, T, H, F, D, E functions). For more detailed information see:
References: /FB/, H2, “Output of Auxiliary Functions to PLC”

G groups

An initial setting can be predefined with
MD 20150: GCODE_RESET_VALUES (initial setting of G groups) for each of the existing G groups. This initial setting is automatically active during program start or in Reset until it is deselected by a G command from the same G group.
The MD 22510: GCODE_GROUPS_TO_PLC (G codes that are output to NCKPLC interface on block change/RESET) activates the output of G codes to the PLC interface.

For a list of the G groups together with their G functions and explanations see:

References: /PG/, “Programming Guide: Fundamentals”

© Siemens AG, 2003. All rights reserved

2.6.2 Part program selection

Channel status

A parts program can only be selected if the channel is concerned in the Reset state.

2.6.3 Start of part program or part program block

START-command, channel status

There are two possible START commands for initiating processing of a parts program or parts program block:

- The channel-specific IS “NC start” (DB21, ... DBX7.1), which is usually controlled from the machine control panel key NC Start, starts program processing in the same channel.
- With the NC instruction START, program processing in the first channel can be started from the second channel, for example (for further information see Chapter Channel synchronization).

The START command can only be executed in AUTOMATIC and MDA modes. The relevant channel must be in the state “Channel status reset” (DB21, ... DBX35.7) or “Channel status interrupted” (DB21, ... DBX35.6).

Necessary signal states

The parts program can be enabled by the START command for processing in the channel only on the condition that certain signal states exist on the machine.

Relevant enable signal settings:

- IS “Mode group ready” must be present (DB11, ... DBX4.4)
- IS “Mode group reset” must not be present (DB11, ... DBX0.7)
- IS “Activate program test” must not be present (DB21, ... DBX1.7)
- IS “NC Start disable” must not be present (DB21, ... DBX7.0)
- IS “NC stop at block limit” must not be present (DB21, ... DBX7.2)
- IS “NC Stop” must not be present (DB21, ... DBX7.3)
- IS “NC Stop axes plus spindle” must not be present (DB21, ... DBX7.4)
- IS “Channel” must not be present (DB21, ... DBX7.7)
- IS “EMERGENCY STOP” must not be applied (DB10, DBX56.1)
- No axis or NCK alarm must be active

For a further explanation of the individual signals see Chapter 5.

© Siemens AG, 2003. All rights reserved

Execution of command

The parts program or parts program block is automatically processed and IS “Channel status active” (DB21, ... and IS “Program status running” (DB21, ... DBX35.0) are set. The program is processed until the end of the program has been reached or the channel is interrupted or aborted by a STOP or RESET command.

Alarms

Under certain conditions the START command will have no effect and on of the following alarms will be triggered:

- 10200 “No NC Start permitted with active alarm”
- 10202 “No NC Start permitted with active command” (see /DA/)
- 10203 “No NC Start permitted for non-referenced axes”.

2.6.4 Parts program interruption

Channel status

The STOP command is executed only if the channel concerned is in the “Channel active” status (DB21, ... D35.5).

STOP commands

There are various commands which stop processing of the program and set the channel status to “interrupted”. These are:

- IS “NC stop at block limit” (DB21, ... DBX7.2)
- IS “NC stop” (DB21, ... DBX7.3)
- IS “NC Stop axes plus spindle” (DB21, ... DBX7.4)
- IS “Single block” (DB21, ... DBX2.0)
- Programming command “M00” or “M01”.

For explanations of the interface signals see Chapter 5. For explanations of the parts program instructions see:
References: /PG/, “Programming Guide: Fundamentals

Execution of command

After execution of the STOP command, IS “Program status interrupted” (DB21, ... DBX35.3) is set. Processing of the interrupted program can continue from the point of interruption with the command START.

© Siemens AG, 2003. All rights reserved

The following actions are executed when the STOP command is triggered:

- Parts program processing is stopped at the next block limit (with NC stop at block limit, M00/M01 or single block), processing is stopped immediately with the other STOP commands.
- Any auxiliary functions of the current block not yet output by this point are no longer output.
- The axes of each channel are brought to a standstill along a braking ramp and parts program processing is then stopped.
- The block indicator stops at the point of interruption.

Possible actions in the interrupt state

The following actions can be executed when the parts program has been interrupted (program status stopped, channel interrupted):

- Overstore
  References: /BA/, “Operator’s Guide”
- power ON, block search
  References: /BA/, “Operator’s Guide”
- Repositioning (machine function REPOS)
  References: /BA/, “Operator’s Guide”
- Orientable tool retraction
  References: /PG/, “Programming Guide: Fundamentals”
- Interrupt routine (see Subsection 2.5.12)
- DRF function, workpiece zero offset
  References: /FB/ H1, “Jogging with the handwheel”
- Starting the interrupted program with IS “NC Start” (DB21, ... DBX7.1) or with the NC instruction START from a different channel.

© Siemens AG, 2003. All rights reserved

2.6.5 RESET command

Channel status

The RESET command can be executed in every channel state. This command is aborted by another command.

Reset commands

The following Reset commands are available:

- IS “Mode group Reset” (DB11, ... DBX0.7)
- IS “Reset” (DB21, ... DBX7.7)

For a further explanation of the individual signals see Chapter 5.

A RESET command can be used to interrupt an active parts program or a parts program block (in MDA).

IS “Channel status reset” (DB21, ... DBX35.7) is set after execution of the Reset command.

The parts program cannot be continued at the point of interruption. All the axes of the channel go into exact stop unless they are in follow-up mode. The same applies to the spindles configured in the channel.

The following actions are executed when the RESET command is triggered:

- Parts program preparation is stopped immediately.
- Axes and, if they exist, spindles in the channel are decelerated along a braking ramp.
- Any auxiliary functions that have not yet been output at this point are not longer output.
- The block indicator is reset to the beginning of the parts program.
- The Reset alarms (channel-specific, axis-specific, spindle-specific) are cleared from the display.

2.6.6 Program status

The status of the selected program is flagged to the interface for each channel. The PLC can then trigger the responses and interlocks configured by the manufacturer depending on the status at the interface.
The program status is only shown in AUTOMATIC and MDA modes. In all other modes the program status is aborted or interrupted.

© Siemens AG, 2003. All rights reserved

Program states

The following program states are available:

- IS “Program status aborted” (DB21, ... DBX35.4)
- IS “Program status interrupted” (DB21, ... DBX35.3)
- IS “Program status stopped” (DB21, ... DBX35.2)
- IS “Program status wait” (DB21, ... DBX35.1)
- IS “Program status running” (DB21, ... DBX35.0)

For a further explanation of the individual signals see Chapter 5.

The effect of commands/signals

The program status can be modified by activating different commands or interface signals. The following table shows the resulting program status when these signals are set (status before the signal is set –> Program status running).

Table 2-2 Effect on program status

Commands	Program processing statuses				
	Aborted	Interrupted	Stopped	Wait	Running
IS “Reset”	X				
IS “NC stop”			X		
IS “NC stop at block limit”			X		
IS “NC stop axes and spindles”			X		
IS “Read-in disable”					X
IS “Feed hold, channel-sp.”					X
IS “Feed hold, axis-sp.”					X
Feedrate override = 0%					X
IS “Spindle stop”					X
M02/M30 in a block	X				
M00/M01 in a block			X		
IS “Single block”			X		
IS “Delete distance-to-go”					X
Auxiliary functions output to PLC but not yet acknowledged			X		
Wait instruction in program				X	

© Siemens AG, 2003. All rights reserved

2.6.7 Channel status

The current channel status is signaled at the interface. The PLC can then trigger certain responses and interlocks configured by the manufacturer depending on the status at the interface.
The channel status is displayed in all operating modes.

Channel statuses

The following channel statuses are available:

- IS “Channel status Reset” (DB21, ... DBX35.7)
- IS “Channel status interrupted” (DB21, ... DBX35.6)
- IS “Channel status active” (DB21, ... DBX35.5)

For a further explanation of the individual signals see Chapter 5.

The effect of commands/signals

The channel status can be modified through the activation of various commands or interface signals. The following table shows the resulting channel status when these signals are set (status before the signal is set –> Channel status active).
The “Channel status active” signal is obtained when a parts program or parts program block is being executed or when the axes are traversed in JOG mode.

Table 2-3 Effect on channel status

Commands	Resulting channel status		
	Reset	Interrupted	Active
IS “Reset”	X		
IS “NC stop”		X	
IS “NC stop at block limit”		X	
IS “NC stop axes and spindles”		X	
IS “Read-in disable”			X
IS “Feed hold, channel-sp.”			X
IS “Feed hold, axis-sp.”			X
Feedrate override = 0%			
IS “Spindle stop”			X
M02/M30 in a block	X		
M00/M01 in a block		X	
IS “Single block”		X	
IS “Delete distancetogo”			X
Auxiliary functions output to PLC but not yet acknowledged			X
Wait instruction in program			X

© Siemens AG, 2003. All rights reserved

2.6.8 Response to operator and program actions

The following table shows the channel and program states that result after certain operator and program actions.

The lefthand side of the table shows the channel and program statuses and the mode groups from which the initial situation can be selected. Various operator/ program actions are listed on the righthand side of the table, the number of the situation after the action has been carried out is shown in brackets after each action.

Table 2-4 Responses to operator or program actions

Situation	Channel status			Program status					Active mode			Operator or program action (Situation after the action)
	R	U	A	N	U	S	W	A	A	M	J	
1		x						x	x			RESET (4)
2		x						x		x		RESET (5)
3		x						x			x	RESET (6)
4	x			x					x			NC Start (13); Mode change (5 or 6)
5	x			x						x		NC Start (14); Mode change (4 or 6)
6	x			x							x	Direction key (15); Mode change (4 or 5)
7		x		x						x		NC Start (14)
8		x		x							x	NC Start (15)
9		x			x				x			NC Start (13); Mode change (10 or 11)
10		x			x					x		NC Start (16); Mode change (9 or 11)
11		x			x						x	Direction key (17); Mode change (9 or 10)
12		x				x			x			NC Start (13); Mode change (10 or 11)
13			x					x	x			NC Stop (12)
14			x	x						x		NC Stop (7); at block end (5)
15			x	x							x	NC Stop (8); at JOG end (6)
16			x		x					x		NC Stop (10); at block end (10)
17			x		x						x	NC Stop (11); at JOG end (11)
18			x				x		x			Reset (4); wait for other channel (18)

Channel status:
R → aborted
U → interrupted
A → running

Program status:
N → aborted
U → interrupted
S → stopped
W → waiting
A → running

Mode:
A → AUTOMATIC
M → MDA
J → JOG

© Siemens AG, 2003. All rights reserved

2.6.9 Start part program

Table 2-5 Typical program sequence

Sequence	Command	Supplementary Conditions (must be satisfied before the command)	Comments
1	Load program (via the operator interface or parts program)		
2	Select AUTOMATIC mode		
3	Program preselection	– Channel preselected – Preselected channel in Reset state – User identifier for program preselection sufficient	
4	NC start for preselected channel	– NC start disable not set – Reference point approached with all axes	
5			Program processing
6	M02/M30/Reset	None	End of program

© Siemens AG, 2003. All rights reserved

2.6.10 Time diagram example for a program run

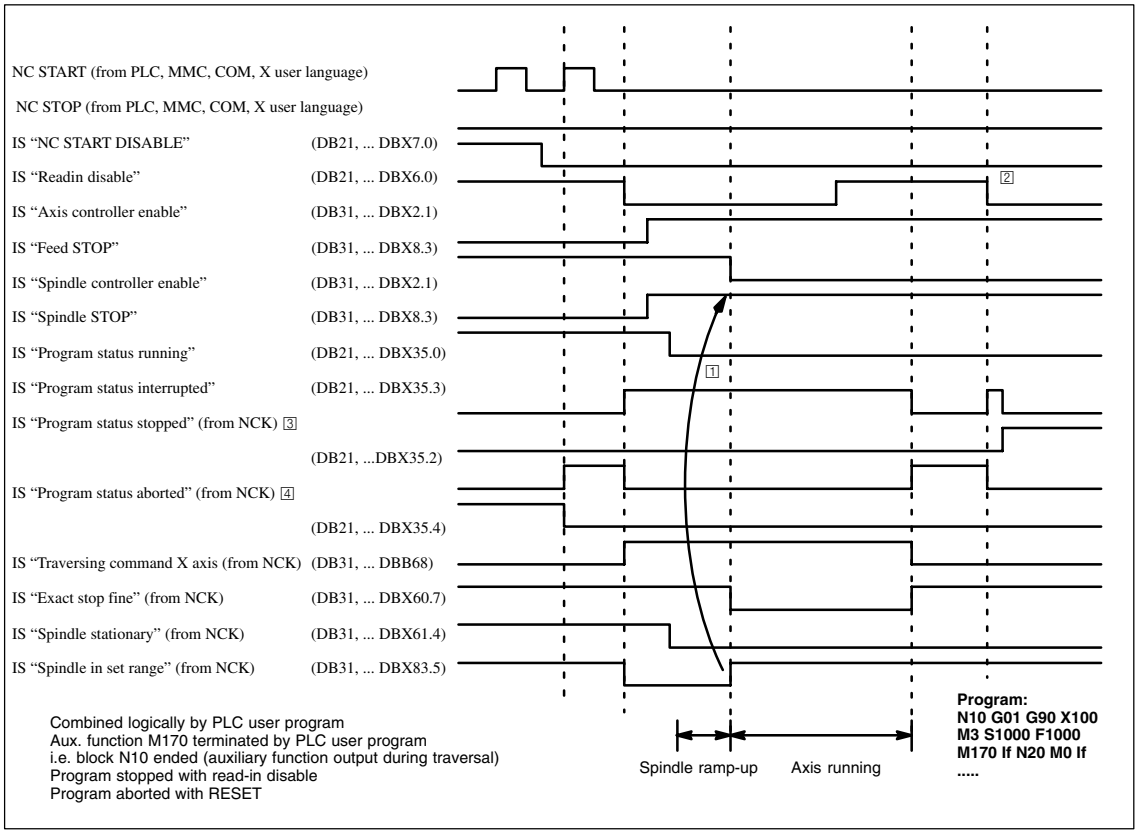


Fig. 2-8 Examples of signals during a program run

© Siemens AG, 2003. All rights reserved

2.6.11 Repeating program sections (SW 4 and higher)

Basic information

Program sections which have already been programmed are frequently needed again.

Instead of using subroutines, you can repeat existing program sections in any order.

The advantage compared with copying blocks is that you save memory and can make modifications easily. Changes to positions only need to be made at one point.

Labels

Labels are used within a program to identify a block or a program section.

References: /PG/, Programming Guide: Fundamentals

Repetition

The options available for repeating part of a program are:

- Repeat a block
- Repeat area starting at a label
- Repeat an area between two labels
- Repeat an area between a label and the end label

Reference: /PGA/, Programming Guide Advanced

Activation

Program section repeats are activated by programming instructions.

Example

Milling: Machine drilling position with different technology sequences

```
N10 CENTER DRILL()          ; Load center drill
N20 POS_1:                  ; Drilling positions 1
N30 X1 Y1
N40 X2
N50 Y2
N60 X3 Y3
N70 ENDLABEL:
N80 POS_2:                  ; Drilling positions 2
N90 X10 Y5
N100 X9 Y5
N110 X3 Y3
N120 ENDLABEL:
N130 BOHRER()               ; Change drill and drilling cycle
N140 GEWINDE(6)             ; Load tap M6 and threading cycle
N150 REPEAT POS_1           ; Repeat program section once from Pos 1 up to
                            ; end label
N160 BOHRER()               ; Change drill and drilling cycle
N170 GEWINDE(8)             ; Load tap M6 and threading cycle
N180 REPEAT POS_2           ; Repeat program section once from Pos 2 up to
                            ; end label
N190 M30
```

© Siemens AG, 2003. All rights reserved

2.6.12 Event-driven program calls (SW 6.1 and higher)

Application

In the case of certain events, an implied user program is to start. The user can thus make initial settings of functions or initializations via parts program commands.

MD 20108 Selection which event

The machine data MD 20108: PROG_EVENT_MASK can be set channel-specifically which of the following events will enable the user program:

- Bit=1: Parts program start
- Bit1 = 1: Parts program end
- Bit2 = 1: Operator panel reset
- Bit3 = 1: Power-up (of the NC control)

The user program must be stored under the fixed path name **/_N_CMA_DIR/_N_PROG_EVENT_SPF**.

The same protection mechanisms can be enabled as can generally be activated for cylinders (protection levels for writing, reading etc.).

MD 20108: PROG_EVENT_MASK is ignored in the simulation run.

Request which start event

In the user program, the system variable $P_PROG_EVENT can be used to request the event which has enabled the parts program.

Event **Parts program start**

Table 2-6 Sequence when starting a parts program

Sequence	Command	Supplementary Conditions (must be satisfied before the command)	Comments
1	Channel selection: Reset status Mode selection: AUTO or AUTO and Overstore or MDA or TEACH IN	Specification of initial state: Channel in reset status Mode: AUTO or AUTO and Overstore or MDA or TEACH IN	Select channel and mode Channel: in reset status and mode: as per selection
2	NC Start	None	Start NCK
3	MD 20112: START_MODE_MASK	Initialization sequence with Evaluation	Initialization sequence with evaluation of MD 20112
4	/_N_CMA_DIR/_N_PROG_ EVENT_SPF	as a subroutine	Implied call of the path name as a subroutine
5		None	Processing of the data part of the main program
6		None	Processing of the program part of the main program

© Siemens AG, 2003. All rights reserved

Event Parts program end

Table 2-7 Sequence at parts program end

Sequence	Command	Supplementary Conditions (must be satisfied before the command)	Comments
1	Channel selection: Reset status Mode selection: AUTO or AUTO and Overstore or MDA or TEACH IN	Specification of initial state: Channel in active status Mode: AUTO or AUTO and Overstore or MDA or TEACH IN	Select channel and mode, channel: in active status and mode: as per selection
2	NC Start	Block with end of parts program	Block is changed
3	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES MD 20152: GCODE_RESET_MODE	Control enabled: Reset sequence with evaluation	Control enabled Reset sequence with evaluation of MD 20110 and MD 20150 and MD 20152
4	/_N_CMA_DIR/_N_PROG_ EVENT_SPF	as an ASUB	Implied call of the path name as an ASUB
5	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES MD 20152: GCODE_RESET_MODE	Control enabled: Reset sequence with evaluation	Control enabled Reset sequence with evaluation of MD 20110 and MD 20150 and MD 20152. Significance: The G code setting is continued to be set via machine data.

Event Operator panel reset

Table 2-8 Sequence with operator panel reset

Sequence	Command	Supplementary Conditions (must be satisfied before the command)	Comments
1	Select channel and mode: Optional	Initial state: Any mode, any channel status	Select mode / channel status from any state
2	Reset		
3	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES MD 20152: GCODE_RESET_MODE	Control enabled: Reset sequence with evaluation	Control enabled Reset sequence with evaluation of MD 20110 and MD 20150 and MD 20152
4	/_N_CMA_DIR/_N_PROG_ EVENT_SPF	as an ASUB	Implied call of the path name as an ASUB
5	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES MD 20152: GCODE_RESET_MODE	Control enabled: Reset sequence with evaluation	Control enabled Reset sequence with evaluation of MD 20110 and MD 20150 and MD 20152. Significance: The G code setting is continued to be set via machine data.

© Siemens AG, 2003. All rights reserved

Event **Power-up**

Table 2-9 Sequence with Power-up

Sequence	Command	Supplementary Conditions (must be satisfied before the command)	Comments
1	Reset	after power-up	
2	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES MD 20152: GCODE_RESET_MODE	Control enabled after Power-up: Reset sequence with evaluation	Control enables the reset sequence of MD 20110 and MD 20150 and MD 20152 after power-up
3	/_N_CMA_DIR/_N_PROG_EVENT_SPF	as an ASUB	Implied call of the path name as an ASUB
4	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES MD 20152: GCODE_RESET_MODE	Control enabled: Reset sequence with evaluation	Control enabled Reset sequence with evaluation of MD 20110 and MD 20150 and MD 20152. Significance: The G code setting is continued to be set via machine data.

Chronological sequences

With parts program start and parts program end (SW 6.3 and higher):

Time sequence of “Program status” and “Channel status” of the VDI signals (DB21–DB30, ... DBB35) on processing a parts program with event-controlled program call for parts program start and parts programend:

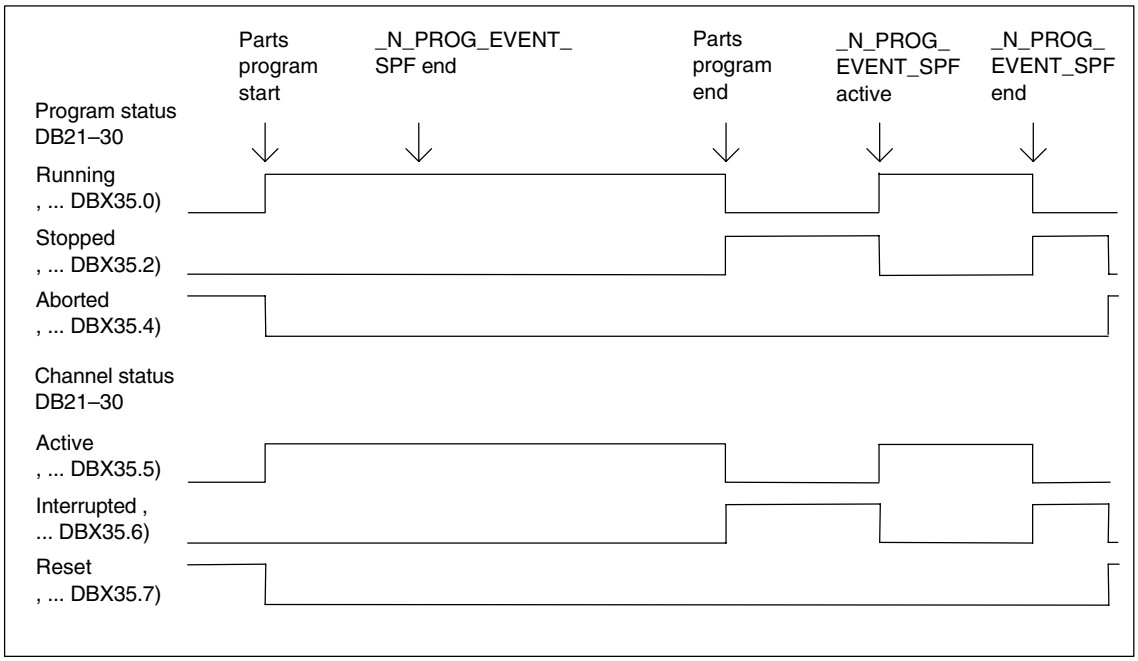


Fig. 2-9 Time sequence of the interface signals for program status and channel status

© Siemens AG, 2003. All rights reserved

With operator panel reset (SW 6.3 and higher):

Time sequence of “Program status” and “Channel status” of the VDI signals (DB21–DB30, ... DBB35) on processing with event-controlled program call:

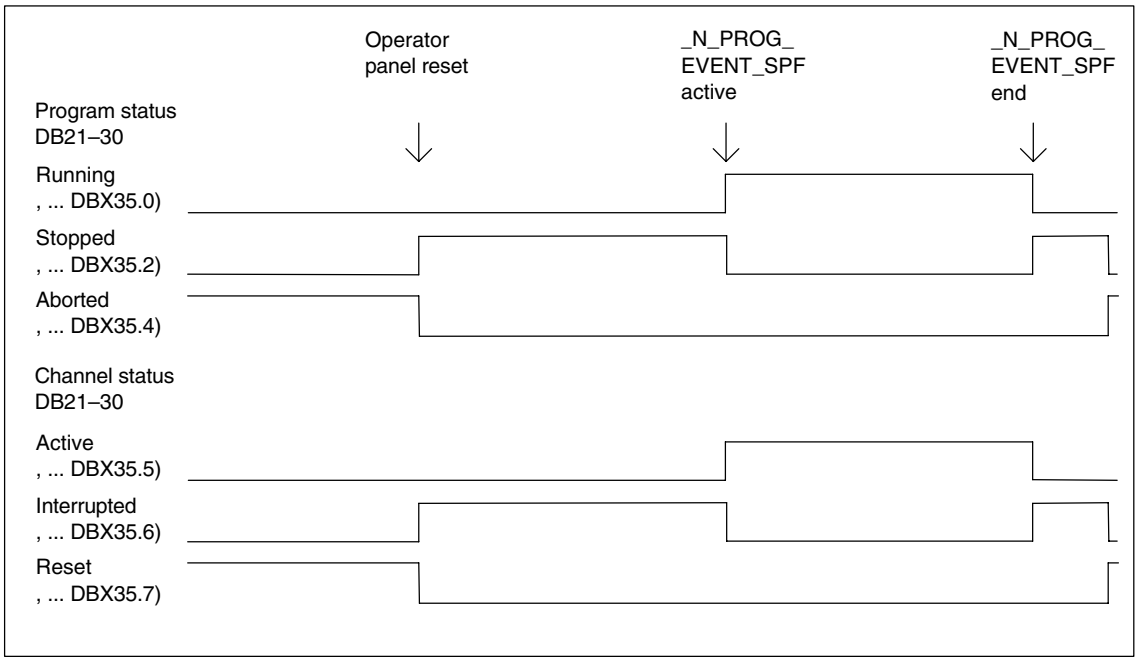


Fig. 2-10 Time sequence of the interface signals for program status and channel status

Software modification

The chronological sequences from SW 6.3 and higher are shown:

Notice

The signal charts for the program state and channel state have been altered between SW 6.2 and SW 6.3 for “Parts program start and parts program end” and for “Operator panel reset”.

SW 6.2 and lower:
IS “Program status aborted” (DB21, ... DBX35.4) is terminated independent of “Program status halted” (DB21, ... DBX35.2) change from “1” to “0”.
“Channel status Reset” (DB21, ... DBX35.7) is termianted independent of “Channel status interrupted” (DB21, ... DBX35.2) change from “1” to “0”.

SW 6.3 and higher (see also chronological charts):
“Program status aborted” (DB21, ... DBX35.4) and “Channel status Reset” (DB21, ... DBX35.7) is not assumed until _N_PROG_EVENT_SPF has ended.
Neither the “Program aborted state” (DB21, ... DBX35.4) nor the “Channel reset state” (DB21, ... is assumed between program end and start of the program event. DBX35.7).
The same applies between operator panel reset and the start of the program event.

© Siemens AG, 2003. All rights reserved

Special points to be noted

The following must be observed with the user program _N_PROG_EVENT_SPF:

- is run with the lowest priority and can therefore be interrupted by the user ASUB.
- The PLC can be advised of the processing status of _N_PROG_EVENT_SPF via user M functions. No special NCKPLC signals are intended for this.
- is generally processed in the channel in which the appropriate event has occurred. By requesting MD 20000: CHAN_NAME, it is possible to determine in the program _N_PROG_EVENT_SPF which channel is currently processed.
  Note: Power-up is an event that takes place in all channels.
- With each new configuration of MD 20108: PROG_EVENT_MASK, /_N_CMA_DIR/_N_PROG_EVENT_SPF must be loaded or enabled. Otherwise, the alarm 14011 “Program _N_PROG_EVENT_SPF does not exist or not enabled for execution” is output.
- The display can be suppressed in the current block display using the DISPLOF attribute in the PROC statement.
- Single block stop can be disabled using the command SBLOF attribute or via MD 10702: IGNORE_SINGLEBLOCK_MASK with bit 0.
  MD 10702: IGNORE_SINGLEBLOCK_MASK with bit 0.

No sequences for **start/end of parts program** are passed:

- If a user ASUB is started from the reset status, the described sequences for the event for start/end of parts program are not passed.
- **Settable ProgEvent properties (SW 6.3 and higher)**

  Additional properties of the “Eventdriven program calls” can be set for specific channels via machine data MD 20109: PROG_EVENT_MASK_PROPERTIES from SW 6.3 and higher:

  – Bit=0: An ASUB started from the RESET channel state is followed by an “Event-driven program call” as in earlier versions

  – Bit=1: An ASUB started from the RESET channel state is not followed by an “Event-driven program call”

When **starting a parts program:**

- /_N_CMA_DIR/_N_PROG_EVENT_SPF is executed as a subroutine. _N_PROG_EVENT_SPF must be ended with M17 or RET.
  No return using the REPOS command is permitted; this will generate the alarm 16020 “Repositioning not possible”.

Errors **in the case of operator panel reset** or **after powerup**:

- If an error is present in the case of operator panel reset or after power-up EMERGENCY STOP or Mode group/NCK Continue, then _N_PROG_EVENT_SPF will only be processed after EMERGENCY STOP has been acknowledged or the error has been acknowledged in all channels.

Generally, the following applies to **powerup**:

- The power-up event occurs in **all** channels at the same time.

© Siemens AG, 2003. All rights reserved

Call exampleing all events

For MD 20108: PROG_EVENT = ‘H0F‘, i.e.
call of _N_PROG_EVENT_SPF with
parts program start,
parts program end, operator panel reset
and power-up:

PROC PROG_EVENT DISPLOF

Sequence for **parts program start**

```
IF ($P_PROG_EVENT == 1)
N 10 MY_GUD_VAR = 0                ; Initialize GUD variable
N 20 M17                           ;
ENDIF
```

Sequence for **parts program end** and **operator panel reset**

```
IF ($P_PROG_EVENT == 2) OR ($P_PROG_EVENT == 3)

N10 DRFOF                          ; Disable DRF offsets
N20 IF $MC_CHAN_NAME == “CHAN1”
N30 CANCEL(2)                      ; Delete modal synchronized action 2
N40 ENDIF                          ;
N50 M17                            ;

ENDIF
```

Sequence for **power-up**

```
IF ($P_PROG_EVENT == 4)

N10 IF $MC_CHAN_NAME == “CHAN1”
N20 IDS=1 EVERY $A_INA[1] > 5.0 DO $A_OUT[1] = 1
N30 ENDIF                          ;
N40 M17                            ;

ENDIF
M17
```

Example for calling operator panel reset

For MD 20108: PROG_EVENT = ‘H04‘

PROC PROG_EVENT DISPLOF

```
N10 DRFOF                          ; Disable DRF offsets
N20 M17
```

Note

For block search:
By setting MD 11450: SEARCH_RUN_MODE, bit 1 = TRUE,
/_N_CMA_DIR/_N_PROG_EVENT_SPF will also implicitly enable “automatic ASUB start” after block search when changing the last action block.

For further explanations, please refer to Section 2.4 Program test “Automatic start of an ASUB after block search”.

For further information regarding the power-up of the control, please see
References: /IAD/, Chapter 5 “Turning on and Power-up”

© Siemens AG, 2003. All rights reserved

2.6.13 Asynchronous subroutines (ASUBs), interrupt routines

Overview

Interrupt inputs allow the NC to interrupt the current processing operation in the NC so that it can react to more urgent events in interrupt routines or ASUBs.

Identical functionality is identified by the terms ASUB and Interrupt routines. For the purposes of simplification, only the term interrupt routine will be used from now on.

- Interrupt routines are normal parts programs which are started by interrupt events (interrupt inputs, process or machine status) related to the machining process or the relevant machine status. Any parts program block currently being executed will be interrupted by the routine if it is not specifically declared to be locked against interruption. It is possible to continue the subroutine at the point of interruption at a later stage.
- Where several interrupt routines exist, the have to be given different levels of priority so that incoming activation signals arriving at the same time can be placed in a certain order.
- Interrupt routines can be activated from the PLC program by means of a “Function call”.

Note

With software versions up to and including version 3, interrupt routines can be called only if the mode group is in program operation mode, i.e. in cases where parts program blocks are being processed in either AUTOMATIC or in MDA mode. With SW 4 and higher, this restriction no longer fully applies. For the conditions under which interrupt routines can be called additionally in SW 4, please refer to 2.6.14.

© Siemens AG, 2003. All rights reserved

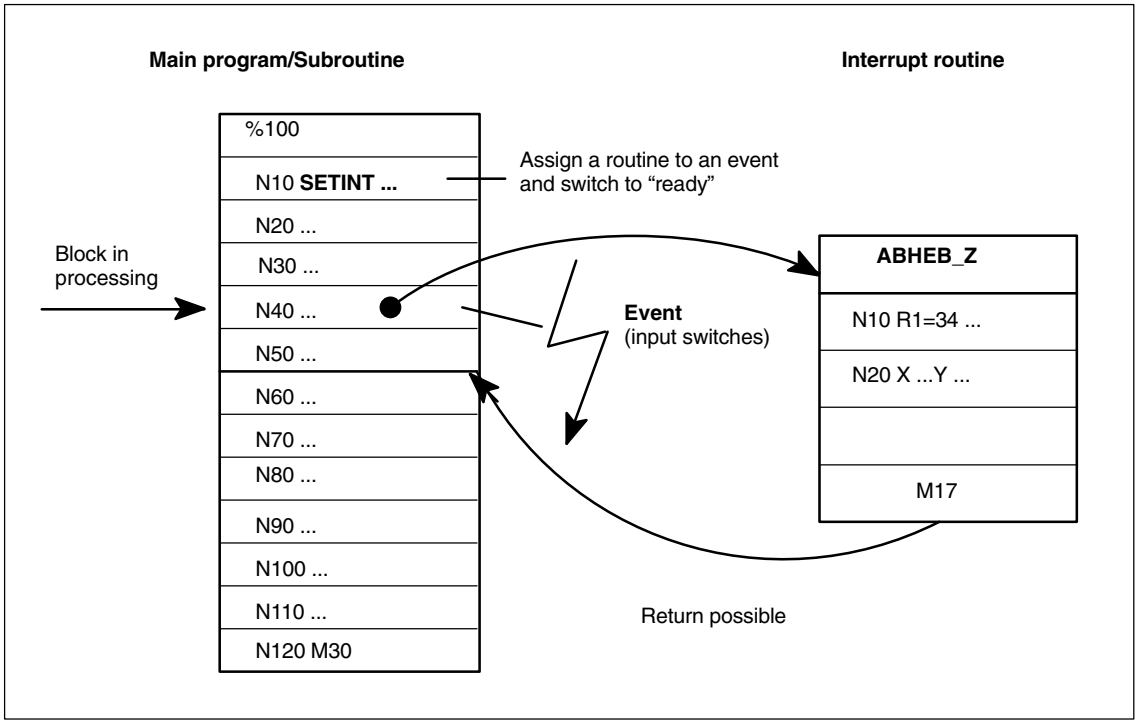


Fig. 2-11 Working with interrupt routines

Parameterization by SETINT

An interrupt signal must be assigned to the parts programs via the NC instruction SETINT. This turns the parts program into an interrupt routine. The following parameters can be used in the SETINT instruction:

- LIFTFAST: When the interrupt signal arrives, a “Fast retraction of the tool from the contour” is executed before the interrupt routine starts. The direction of the retraction is defined by the NC address ALF.
  If mirroring is active for machining with frames,
  MD 21202: LIFTFAST_WITH_MIRROR can be used to control whether the retraction direction is also mirrored with lift fast.
  The traversing path for rapid retraction is stored for the 3 geometry axes in MD 21200: LIFTFAST_DIST (traversing path for rapid retraction from contour).
  When LIFTFAST (rapid retraction) is parameterized, the maximum axis acceleration for positioning operations (MD 32300: MAX_AX_ACCEL) is reduced by the factor set in MD 20610: ADD_MOVE_ACCEL_RESERVE (acceleration margin for overlaid motions).

- BLSYNC: If this parameter is set, the current program block is processed and only then is the interrupt routine started.

References: /PG/, “Programming Guide: Fundamentals”

© Siemens AG, 2003. All rights reserved

Interrupt signals

A total of 8 interrupt signals are available.
All inputs can be controlled via the PLC.
The first four interrupt signals are controlled via the 4 rapid NC inputs on the NCU module (X121).
The signal state of the rapid NC inputs can be read out via the PLC interface (DB10).
Transmission of the rapid NC input signals to the interrupt signals can be disabled via the PLC interface (DB10).

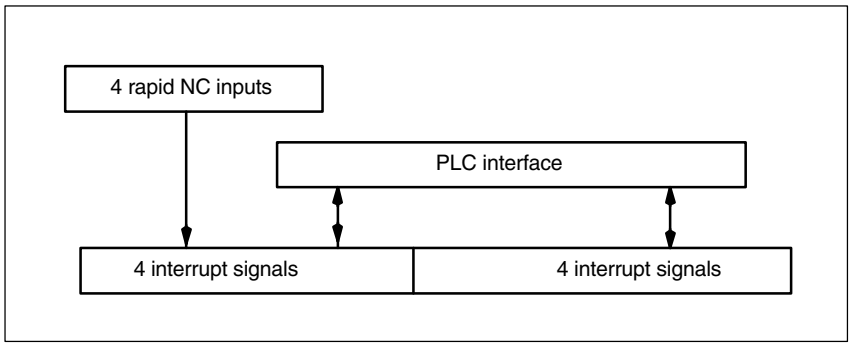


Fig. 2-12 Interrupt signals

For further information about PLC control of the rapid NC inputs (interrupt signals) see:

References: /FB/, A4, “Digital and Analog NCK I/Os”.

Activating the Interrupt routine

Interrupt routines can be activated by two different methods:

- By a 0/1 transition of the interrupt signal, triggered by a 0/1 transition at the rapid NC input
- By the function call ASUPST (/B1/, /P3/)

After activation, all machine axes are braked along the acceleration ramp (MD32300: MAX_AX_ACCEL (axis acceleration)) until they stop and the axis positions are stored.

Reorganization

In addition to decelerating the axes, the previously decoded calculation blocks are calculated back to the interruption block. I.e. all the variables, frames and G codes are assigned the value that they would have at the point of interruption if the parts program had not been previously decoded. These values are put in the buffer so that they can be called up again when the interrupt routine is completed.

Exceptions where reorganization is not possible:

- In thread cutting blocks
- With complex geometries (e.g. spline, radius compensation).

© Siemens AG, 2003. All rights reserved

Processing of interrupt routine

The “Interrupt” program is automatically started on completion of reorganization. It is treated by the system like a normal subroutine (displayed on the operator interface, nesting depth etc.)

End of the interrupt routine

After the end identifier (M02, M30, M17) of the “Interrupt” routine has been processed, **the axis traverses as standard to the end position programmed in the parts program block following the interruption block**.
A REPOS instruction must have been programmed at the end of the “interrupt” routine if return positioning to the point of interruption is required, e.g. REPOSL M17

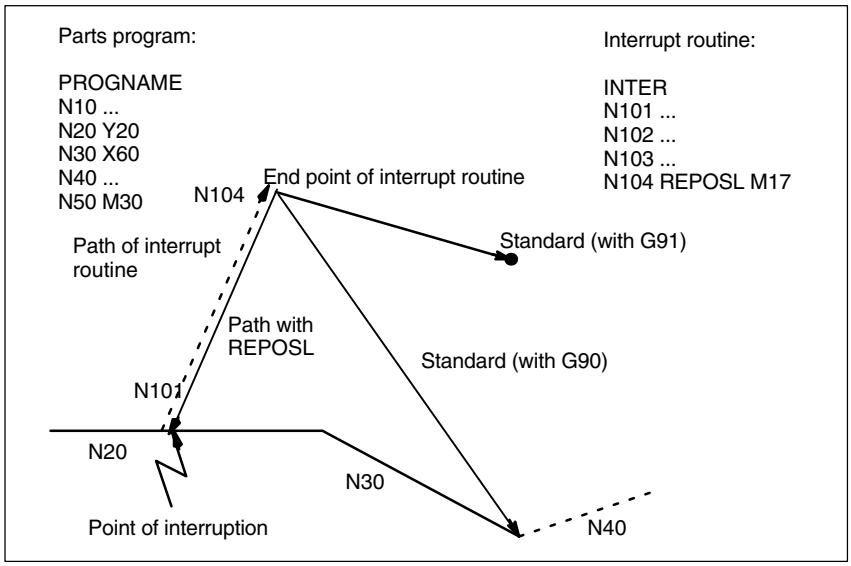


Fig. 2-13 End of interrupt routine

For more information about the REPOS instruction (e.g. syntax) see:
References: /PG/, “Programming Guide: Fundamentals”

SAVE command

If the SAVE command has been used to define the interrupt routine, the G codes, frames and transformations previously active in the interrupted parts program become operative again as soon as the interrupt routine is ended.

Interrupt disable

The DISABLE command can be set to protect parts program sections from being interrupted by an interrupt routine. The assignment interrupt signal <–>parts program is maintained but the interrupt routine does not respond to the 0/1 signal transition.
The DISABLE command can be reset with the ENABLE command. Interrupt routines are not activated until the next 0/1 transition of the interrupt signal.

© Siemens AG, 2003. All rights reserved

Clear assignment

The assignment interrupt signal <–> parts program is cleared when the following happens:

- Channel in Reset state
- CLRINT instruction in parts program.

For more information on the DISABLE, ENABLE and CLRINT instruction (e.g. syntax) please see:

References: /PG/, Programming Guide: Fundamentals

2.6.14 Calling the ASUB outside program operation (SW 4 and higher)

New ASUB activation states

In SW 4 and higher, ASUBs/interrupt routines can also be enabled in the program states or operating modes listed below, in addition to AUTOMATIC mode and MDA:

- JOG, JOG REF
- MDA Teach In, MDA Teach In REF, MDA Teach In JOG, MDA REF, MDA JOG
- AUTOMATIC, stopped, ready
- Not referenced.

If an interrupt routine is activated in JOG or REF mode, it will interrupt any jogging and referencing operations in progress.

Start conditions

Individual basic frames can be deleted with MD 11602: ASUP_START_MASK can be set to enable ASUBs for conditions under which ASUBs are otherwise locked by a default setting.

- Stop by means of Stop key, M0, M01
- Not all axes are referenced yet
- Readin disable is active.

If the default setting remains valid, the response is the same as for SW version 3.

Explicit ASUB start

If MD 11602: ASUP_START_MASK is set such that the ASUB may not be started automatically, the routine can still be activated by the Start key. Any rapid retraction that may be parameterized is always started.

Priorities

The MD 11604: ASUP_START_PRIO_LEVEL can be set to define the priority level up to which the settings in MD 11602: ASUP_START_MASK must be effective (1–128, 1 corresponds to highest priority). The priority specified in the MD covers all priority levels from 1 up to the specified level.

© Siemens AG, 2003. All rights reserved

Effect of VDI signals on mode group channels

MD11600: BAG_MASK can be set to control the effect of mode group signals (mode group reset, mode group stop axes and spindles, mode group switchover disable) on mode group channels in which interrupt routines are currently being processed. See Section 4.2, General machine data.

The machine data also defines whether the internal program processing mode applies only to the one channel in which the interrupt routine has been activated or to all channels in the mode group.

If via MD11600: BAG_MASK, the channel in which the interrupot is running has left the BAG, the BAG signals BAG-RESET, BAG-STOP and BAG-STOPALL are not active in this channel. So the ASUP of BAG signals can proceed unhindered.

NC response

The following table describes the NC response in the individual operational states:

Table 2-10

Status of NC	Start of ASUB	Control system reaction
Program is active	Interrupt, (PLC)	Rapid retraction or stop axes Interruption of program for duration of ASUB Approach to interruption point if ASUB contains REPOS Continuation of parts program. (Corresponds to SW 3)
RESET	Interrupt, (PLC)	The ASUB is executed like a main program. RESET (without M30) is executed at the end of the ASUB. The next control system status depends on machine data MD 20110: RESET_MODE_MASK and MD 20112: START_MODE_MASK. **References:** /FB/, K2, “Workpiece-Related Actual Value System”
Program operation mode AUTOMATIC or MDA and **channel stopped**	Interrupt, (PLC)	ASUB is executed. The channel resumes the stopped state at the end of the routine. REPOS in ASUB: Processing of the ASUB ceases prior to the approach block. The approach movement can be initiated with the Start key.
	Start key	Once the ASUB has been executed, processing of the interrupted program is resumed. (As with SW 3)
Manual mode, **channel stopped**	Interrupt, (PLC)	Control system assumes the status “internal program processing mode” for the addressed channel (not evident externally) and then activates the ASUB. The selected operating mode remains valid. The original status is resumed after execution of the ASUB (M17).
JOG AUTO Teach In, AUTO Teach Reference pnt.	Interrupt, (PLC)	Stop processing, evaluate setting in MD11602: ASUP_START_MASK and MD 11604: ASUP_START_PRIO_LEVEL, internal switchover to “internal program execution mode” if appropriate, activate ASUB, restore status prior to ASUB start. Any LIFTFAST defined with SETINT is not activated in JOG mode.
MDA JOG, MDA Teach In, MDA Teach reference pnt.	Interrupt, (PLC)	

© Siemens AG, 2003. All rights reserved

Table 2-10

Status of NC	Start of ASUB	Control system reaction
Manual mode, **channel running**	Interrupt, (PLC)	The current active motion is stopped. The distance-to-go is deleted. The remaining sequence of operations is the same as for “Manual mode, channel stopped”.
Processing of INITIAL.INI, Block search, alarm which cannot be reset by an NC start, digitizing activated, channel in error state	**ASUB cannot be started**	The signal “Interrupt request not possible” is generated. See 5 Signal Descriptions.

Activation options

With SW 4 and higher, ASUBs can also be activated by using synchronized actions to set outputs that enable the input of the interrupt indirectly by shortcircuit.
Example:

1. Define number of active digital I/Os
   FASTIO_DIG_NUM_INPUTS=3
   FASTIO_DIG_NUM_OUTPUTS=3
2. Generate a shortcircuit with the following MDs
   FASTIO_DIG_SHORT_CIRCUIT[0]=’H0102B102’
   FASTIO_DIG_SHORT_CIRCUIT[1]=’H0202B202’
3. Hardware assignment of external NC input byte for NC program interrupt
   SETINT_ASSIGN_FASTIN=2 ; better to use 1 byte more than needed
4. SETINT (1) PRIO=1 SYNCASUP; Define input as ASUB trigger
5. IDS=1 EVERY $$AC_PATHN>=0.5 DO $A_OUT_[9]=1

References: /FBSY/, Synchronized Actions

REPOS

In the case of ASUBs that could only be called from MDA or AUTOMATIC in SW 3, it was always possible to call REPOS at the end of an ASUB to reposition on the interruption point. With the new options provided in SW 4, ASUB sequences may be generated for which there is no unambiguous return to an interruption point in the block processing sequence. System variable:

$P_REPINF

can be used to scan the ASUB to determine whether a REPOS is possible.

Value	Meaning
0	Repositioning with REPOS is not possible because: – Not called in ASUB – ASUB was executed from RESET state – ASUB was executed from JOG state
1	Repositioning with REPOS possible in ASUB

© Siemens AG, 2003. All rights reserved

SW 5.3 and higher

Acknowledgment signal DB21, ... DBX318.0

Control systems can also be stopped/started from AUTOMATIC mode automatically prior to the end of the ASUB. In the stopped status, the VDI acknowledgment signal IS “ASUP is stopped” (DB21, ... DBX318.0) can be set at the PLC interface.

ASUB with REPOSA

An ASUB with REPOSA can be initiated in the AUTOMATIC status. **Example:**

```
                    ; ASUB program
                    ;
N10 G0 G91          ;
N20 Y10             ;
N30 X20             ;
N40 REPOSA          ;
```

If an ASUB is started after the collecting blocks have been output, the NCK will stop before executing the REPOSA block, and the IS “ASUB is stopped” (DB21, ... DBX318.0) is set.

Note

In this situation, “ASUB Done” is not yet set by the FC9 block.
In the case of ASUBs ending without REPOS, the signals “ASUB Done” and “ASUB is stopped” occur at the same time.

For further explanations regarding a description of the signals at the VDI interface, please refer to:

Chapter 5 Signal Descriptions

Read-in disable

The MD 20116: IGNORE_INHIBIT_ASUP instructs the control that user ASUBs for specified interrupt channels must be executed completely even through a read-in disable is active.

For running ASUB programs, the VDI signal IS “Read-in disable” (DB21, ... DBX6.1) can be ignored depending on MD 11604: ASUP_START_MASK. How the ASUB is further processed, depends on the programmed block end (RET or REPOS).

Single block processing

The MD 20117: IGNORE_SINGLEBLOCK_ASUP instructs the control that user ASUBs for specified interrupt channels must be executed completely even through single block processing mode is selected.

© Siemens AG, 2003. All rights reserved

2.6.15 User-defined system ASUBs (SW 4 and higher)

Preconditions

The SW supplied with the NCK contains preprogrammed processes (internal ASUPs) for implementation of the RET and REPOS functions. With SW 4 and higher, the machine manufacturer can replace these standard routines with ones of his own.

Replacement of system routine

MD 11610: ASUP_EDITABLE is set to determine which system routines are to be replaced by user-defined routines.

Value	Meaning
0	The user-defined routine stored in _N_ASUP_SPF in directory _N_CUS_DIR is not activated for either RET or REPOS. Response as for SW 3.
1	The user-defined routine is activated for RET, the routine provided in the system is activated for REPOS.
2	The user-defined routine is activated for REPOS, the routine provided in the system is activated for RET.

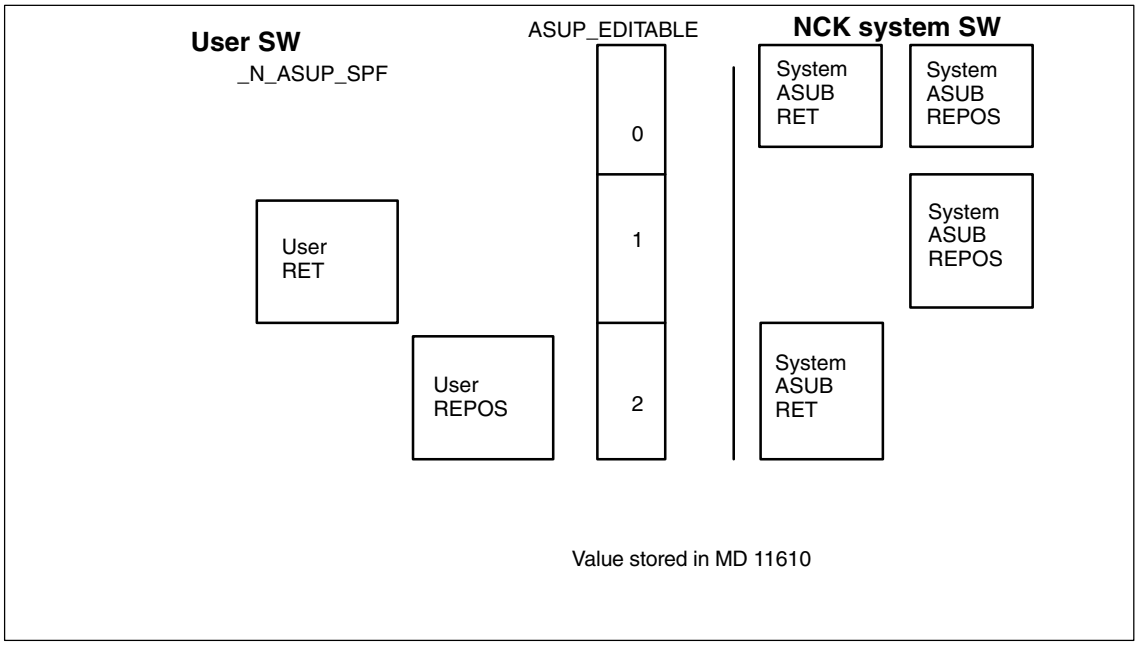


Fig. 2-14 Replacement of system ASUBs by user-defined routines

The following diagram illustrates which routines are used:

© Siemens AG, 2003. All rights reserved

Installation of user system ASUBs

One routine named _N_ASUP_SPF can be loaded to directory _N_CUS_DIR. This routine must implement the actions required by the user for:
RET when MD 11610 is set to 1
REPOS when MD 11610 is set to 2.

$AC_ASUP

Bit	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RET							1			1	*		1							
REPOS	1	1	1	1	1	1		1	1		*	1		1	1					

*) If bit 9 is set, then the branch is determined by MD 20114:
MODESWITCH_MASK:
Bit 0 in MD 20114 is set: →RET
Bit 0 in MD 20114 is not set: →REPOS

Meaning of bits

The meaning of the bits of system variable $AC_ASUP is as follows:

Bit	Meaning
0	User interrupt “ASUB with Blsync” Continuation: Freely selectable REORG or RET
1	User interrupt “ASUB”; The position at which the routine was **stopped** is stored for continuation with REPOS. Continuation: Freely selectable REORG or RET
2	User interrupt “ASUB from channel status Ready” Continuation: Freely selectable REORG or RET
3	User interrupt “ASUB in a manual mode and in channel status not Ready” Continuation: Freely selectable REORG or RET
4	User interrupt “ASUB”; The **current** position as the interrupt occurred is stored for continuation with REPOS. Continuation: Freely selectable REORG or RET
5	Abortion of subroutine repeat Continuation: With system ASUB REPOS
6	Activation of decoder single block Continuation: With system ASUB REPOS
7	Activation of delete distance-to-go Continuation: With system ASUB RET
8	Activation of axis synchronization Continuation: With system ASUB REPOS
9	Mode change Continuation: With system ASUB REPOS or RET depending on MD 11610
10	Program continuation during Teach In or after Teach In deactivation Continuation: With system ASUB RET
11	Overstore selection Continuation: With system ASUB REPOS
12	Alarm with reaction correction block with REPOS Continuation: With system ASUB REPOS

© Siemens AG, 2003. All rights reserved

Bit	Meaning
13	Retraction motion with G33 and stop Continuation: With system ASUB RET
14	Activation of dry run feedrate Continuation: With system ASUB REPOS
15	Deactivation of dry run feedrate Continuation: With system ASUB REPOS
16	Activation of skip block Continuation: With system ASUB REPOS
17	Deactivation of skip block Continuation: With system ASUB REPOS
18	Activate machine data Continuation: With system ASUB REPOS
19	Activate user data Continuation: With system ASUB REPOS

Protection level for user routine

If a user-defined routine is to be used, i.e. if a value other than 0 is programmed in MD 11610: ASUP_EDITABLE, a protection level can be defined for user-specific routine _N_ASUP_SPF. The protection level can be set to values between 0 and 7 and must be entered in MD 11612: ASUP_EDIT_PROTECTION_LEVEL. For further information about protection levels, please read

References: /IAD/, Installation and StartUp Guide, System of Protection

Single block processing

Bit 0 in machine data 10702: IGNORE_SINGLEBLOCK_MASK determines whether internal ASUBs or ASUBs defined by the user must be processed without interruption after every block even when single block processing is selected:

Bit 0 is set:	Internal ASUBs are not interrupted in every block.
Bit 0 is not set:	Internal ASUBs are interrupted in every block.



Danger

The machine manufacturer is responsible for the contents of ASUB routines used to replace ASUP.SYF supplied by Siemens.

© Siemens AG, 2003. All rights reserved

2.7 Block

With the single block function, the user can process a parts program block by block. There are 3 types of setting for the single block function:

- SLB1 := IPO single block
  When the SLB1 function is active, machining stops or pauses after each machine action block (Ipo block).
- SLB2 := Decoding single block.
  When the SLB2 function is active, machining always stops or pauses after each parts program block. If a parts program block is processed in several IPO blocks, machining stops after every Ipo block. Thread cutting is an exception.

SW 4 and higher

1. In many situations and with certain blocks, it is not desirable that machining should stop.
   1st example: Change after jogging if reorganization or repositioning not possible, MD 10702, bits 6 and 7. If machining stops at the end of block in a block which cannot be reorganized or repositioned, JOG mode cannot be selected.
   2nd example: Change after jogging on a STOPRE block, MD 10702, bits 6 and 7
   When changing from AUTO to JOG while a STOPRE block is active, the remaining block and one or (on Decode single block) two STOPRE blocks follow after continuation apart from system ASUB2.
   A logic operation which always triggers a parts program start in single block and then always changes to JOG mode remains indefinitely at the STOPRE block.
   3rd example: DISPOF: Deactivate block display, MD 10702, bits 6 and 7
   When DISPOF was programmed in a subroutine the block display is suppressed. The operator must continuously press start up to the end of the subroutine.
2. When single block is deactivated there is no stop at end of block.
3. When STOPRE blocks are displayed, the main run and preprocessing are synchronized in the decoding single block.

2.7.1 Decoding single block SBL2 with implicit preprocessing stop (SW 6.3 and higher)

Establish reference between variable values and current block display

The advance processing of parts program blocks can result in loss of reference between the current block display referred to the main run status on the NCK and the variable values displayed on the HMI. The operator display then shows implausible variable values.

Using channel-specific setting data SD 42200: SINGLEBLOCK2_STOPRE a preprocessing stop is executed on every block when SBL2 is active. This suppresses preprocessing of parts program blocks and maintains the relationship between the current block display and the variable values display.

© Siemens AG, 2003. All rights reserved

Note

This variant of SBL2 does **not** maintain **an accurate contour**. In other words, as a result of the preprocessing stop, a different contour may be generated than the one created without single block mode or with SBL1.

Application: Debug mode for testing parts programs.

2.7.2 Single block suppression via SBLOF (SW 4.2 and higher)

SBLOF

Programs identified by the language command SBLOF are executed completely in one block as with every type of single block.

SBLOF is also valid in subroutines which are called.

Example for subroutine without stop in single block:

```
PROC BEISPIEL SBLOF
G1 X10
RET
```

The return command determines whether or not the program halts at the end of the subroutine:

Return jump with M17	Stop at the end of the subroutine
Return jump with RET	No stop at the end of the subroutine

In the program

SBLOF must appear on its own in a block. With effect from this block, single block is deactivated until the next programmed SBLON or until the end of the active subroutine level. If SBLOF is active, this also applies in subroutines which are called.

Example: The area between N20 and N60 is executed as one step in single block mode.

```
N10 G1 X100 F1000
N20 SBLOF                    ; Deactivate single block
N30 Y20
N40 M100
N50 R10=90
N60 SBLON                    ; Reactivate single block
N70 M110
N80 ...
```

Asynchronous subroutines

The asynchronous subroutines ASUP1.SYF and ASUP2.SYF, which are started internally in the system on Reorg/REPOS, have to be executed step by step in software versions up to SW 4. In SW 4 and higher, the system ASUB can be executed in one step by programming SBLOF.

Example: ASUP.SPF:

```
N10 SBLOF
N20 IF $AC_ASUP == 'H200'
N30 RET                      ; No REPOS on mode change
N40 ELSE
N50 REPOSA                   ; REPOS in all other cases
N60 ENDIF
N70 RET
```

© Siemens AG, 2003. All rights reserved

Supplementary conditions

- The current block display can be suppressed in cycles using DISPLOF.
- If DISPLOF is programmed together with SBLOF, the cycle call continues to be displayed on single block stops within the cycle.
- The default setting for the response of asynchronous programs with single block, which is defined in MD 20117: IGNORE_SINGLEBLOCK_ASUP, can be overwritten in a specific program with SBLOF.

Examples

Example 1: A cycle is to act like a command for a user.

```
Main program:
N10 G1 X10 G90 F200
N20 X4 Y6
N30 CYCLE1
N40 G1 X0
N50 M30

Program cycle:1
N100 PROC CYCLE1 DISPLOF SBLOF          ; Suppress single block
N110 R10=3*SIN(R20)+5
N120 IF (R11 <= 0)
N130  SETAL(61000)
N140 ENDIF
N150 G1 G91 Z=R10 F=R11
N160 M17
```

CYCLE1 is processed for an active single block, i.e. the Start key must be pressed once for machining with CYCLE1.

Example 2: An ASUB, which is started by the PLC in order to activate a modified zero offset and tool offsets, is to be executed invisibly.

```
N100 PROC NV SBLOF DISPLOF
N110 CASE $P_UIFRNUM OF 0 GOTOF _G500 1 GOTOF _G54 2 GOTOF _G55 3
                    —>GOTOF _G56 4 GOTOF _G57 DEFAULT GOTOF END

N120 _G54: G54 D=$P_TOOL T=$P_TOOLNO
N130 RET
N140 _G54: G55 D=$P_TOOL T=$P_TOOLNO
N150 RET
N160 _G56: G56 D=$P_TOOL T=$P_TOOLNO
N170 RET
N180 _G57: G57 D=$P_TOOL T=$P_TOOLNO
N190 RET
N200 END: D=$P_TOOL T=$P_TOOLNO
N210 RET
```

© Siemens AG, 2003. All rights reserved

2.7.3 Preventing single block suppression (SW 4.2 and higher)

As a function of MD 10702: IGNORE_SINGLEBLOCK_MASK, setting bits 0 to 12 = 1 can suppress stopping at the end of the block during the following machining processes.

Program processing must not stop after single blocks in the case of the following even if SBL processing is selected:

1. An internal Asub
2. A user Asub
3. Subroutines with the attribute DISPLOF
4. Intermediate blocks
5. Block search group blocks
6. Init blocks
7. Blocks which are not reorganizable
8. Blocks which are not repositionable
9. A reapproach block which contains no traversing information.
10. A prerun/main run/synchronization block, due to Reorg
11. At a Tool selection block.
12. At a GET block
13. During a single block type 2 (SW 6.4 and higher)

If an ASUB is activated during the single block, for example, execution of the ASUB is completed. The deceleration movement does not take place until after the end of the ASUB or the first Ipo block in which single block suppression is not activated. If the velocity is too large for the deceleration to be performed in this block (with continuous-path mode G64 active), further block changes must be enabled.

For decoding single block, MD 10702 is only effective with “internal ASUB”, “user ASUB” and “subroutines with the attribute DISPLOF”. In these cases, it is already clear at the time of interpretation that the block belongs to one of the above categories. In these cases, further blocks can be generated.

SBLON in ASUB

The single block stop for an internal ASUB or user ASUB programmed with MD 10702: IGNORE_SINGLEBLOCK_MASK can be activated again by programming SBLON in the ASUB.

This functionality can be suppressed again with MD 20117: IGNORE_SINGLEBLOCK_ASUP, which disables the command SBLON.

Supplementary conditions

The following restriction applies to decoding single block SBL2:

- Block search approach blocks
- Block not in ASUB; DISPLOF, SBLOF
- Non-reorganizable and non-repositionable blocks
- Blocks which are not generated in the interpreter, e.g. intermediate blocks

© Siemens AG, 2003. All rights reserved

2.8 Program control

2.8.1 Control via the operator interface and the PLC

The user can control parts program processing via the operator interface or PLC.

Selection Different functions are available under the soft key Program control. Selection affects an interface signal in the PLC. These signals are to be understood as selection signals from the operator interface, and do not activate the selected function.

Activation These signal states must be transferred to another area of the data block to activate the selected function. With program control by the PLC the signals are to be set directly.

Feedback The activated functions are partly signalled back to the PLC from the NCK.

Table 2-11 Program control

Function	Selection signal	Activation signal	Feedback signal
SKP skip blocks 0 to 7 SKP skip blocks 8 to 9	DB21, ... DBX26.0–26.7 DB21, ... DBX27.0–27.1	DB21, ... DBX2.0–2.7 DB21, ... DBX31.6–31.7	
DRY dry run feedrate	DB21, ... DBX24.6	DB21, ... DBX0.6	DB21, ... DBX318.6
ROV Rapid traverse override	DB21, ... DBX25.3	DB21, ... DBX6.6	
Single block with stop after:	Preselection of SBL1, SBL2 or SBL3 via program control display of HMI		
SBL1 each machine function.	DB21, ... DBX27.4	DB21, ... DBX0.4	DB21, ... DBX32.6
SBL2 each block	HMI operator panel	OPI variable: suppProgFunc bit0 and bit1	
SBL3 stop in cycle	HMI operator panel		
M01 programmed stop 1	DB21, ... DBX24.5	DB21, ... DBX0.5	DB21, ... DBX32.5
MFct programmed Stop 2	DB21, ... DBX24.4	DB21, ... DBX30.5	DB21, ... DBX318.5
DRF selection	DB21, ... DBX24.3	DB21, ... DBX0.3	DB21, ... DBX33.3
PRT program test	DB21, ... DBX25.7	DB21, ... DBX1.7	DB21, ... DBX33.7

References: /LIS/, Lists “Interface signals”
/BA/ Operator’s Guide “Machine Operating Area”

© Siemens AG, 2003. All rights reserved

2.8.2 Multiple skip levels (SW 5 and later)

It is possible to skip blocks which are not to be executed every time the program runs. Blocks to be skipped are indicated in the parts program by an oblique "/" before the block number.
The skip levels in the parts program are specified by"/0" to "/7".
Only one skip level can be specified in a parts program block.

MD 9423: MAX_SKP_LEVEL defines the number of skip levels.

Example:
Blocks which are not to be executed every time the program runs can be skipped as follows (e.g. program running in blocks):

```
/          ; Skip block, (DB21, ... DBX2.0)        1st skip level
/ N005     ; Skip block, (DB21, ... DBX2.0)        1st "
/0 N005    ; Skip block, (DB21, ... DBX2.0)        1st "
/1 N010    ; Skip block, (DB21, ... DBX2.1)        2nd "
/2 N020    ; Skip block, (DB21, ... DBX2.2)        3rd "
/3 N030    ; Skip block, (DB21, ... DBX2.3)        4th "
/4 N040    ; Skip block, (DB21, ... DBX2.4)        5th "
/5 N050    ; Skip block, (DB21, ... DBX2.5)        6th "
/6 N060    ; Skip block, (DB21, ... DBX2.6),       7th "
/7 N070    ; Skip block, (DB21, ... DBX2.7)        8th  skip level
```

The 8 skip levels "/0" to "/7" are activated by the PLC setting the PLC –> NCK interface signals.

Skip levels 8 and 9 in SW 6.3 and higher

2 new skip levels are added to make 10 in total (/0 to /9)

Example:

```
/8 N080    ; Skip block, (DB21, ... DBX31.6),      9th skip level
/9 N090    ; Skip block, (DB21, ... DBX31.7)       10th skip level
```

The function is activated from the HMI via the "Program control" menu in the Machine operating area for skip levels

- "/0" to "/7" via interface MMC –> PLC (DB21 to DB30) in DBB26.
- "/8" to "/9" via interface MMC –> PLC (DB21, ... DBX27.0–27.1).

References: /BA/ Operator's Guide "Machine Operating Area"

Note

The levels to be skipped can only be changed when the control is in the Stop/Reset state.

© Siemens AG, 2003. All rights reserved

2.8.3 Use of a reduced interpolation buffer

Introduction

The channel-specific interpolator executes prepared blocks from the interpolation buffer during the parts program run. The maximum number of blocks requiring space in the interpolation buffer at any given point in time is defined by memory-configuring MD 28060: MM_IPO_BUFFER_SIZE. For some applications it may be meaningful not to use the full buffer capacity in order to minimize the “interval” between preparation and interpolation.

Solution

With the setting data SD 42990: MAX_BLOCKS_IN_IPOBUFFER the number of blocks in the interpolation buffer can be dynamically limited to a value lower than MM_IPO_BUFFER_SIZE (minimum of 2 sets).

Value of the setting data SD 42990: MAX_BLOCKS_IN_IPOBUFFER:

Value	Effect
< 0	No interpolation buffer limit active. The max. possible interpolation buffer as set in MD 28060: MM_IPO_BUFFER_SIZE is activated.
0 or 1	The minimum permissible interpolation buffer with 2 blocks is activated.
< MM_IPO_BUFFER_SIZE	The interpolation buffer is activated with no more than the maximum specified number of blocks.
>= MM_IPO_BUFFER_SIZE	The interpolation buffer is activated with the number of blocks specified in MD 28060: MM_IPO_BUFFER_SIZE.

Note

If SD 42990: MAX_BLOCKS_IN_IPOBUFFER is set in the parts program, the limitation of the interpolation buffer takes effect immediately when the block with the SD is executed by the interpreter at the preprocessing stage, i.e. the limitation of the IPO buffer may take effect a few blocks before the intended activation point (see also MD 28070: MM_NUM_BLOCKS_IN_PREP). To avoid premature activation and make the limitation of the IPO buffer take effect **in synchronism with the block**, a **STOPRE** (preprocessing stop) must be programmed before the SD is set in the parts program.

Validity

SD 42990: MAX_BLOCK_IN_IPOBUFFER has global, channel-specific validity and can also be modified in a parts program. This modified value is maintained at program end. If this setting data is to be reset again on defined events, an event-controlled program must be created to do this. This setting data could als be set to a predefined value on RESET, for example. For information see Subsection 2.6.12.

© Siemens AG, 2003. All rights reserved

Application

The interpolation buffer limitation can be used whenever the number of blocks between block preparation and interpolation must be minimized, e.g. when actual positions in the parts program must be read and processed for other purposes.

Example

```
N10 ...
N20 ...
.....
N100 $SC_MAX_BLOCKS_IN_IPOBUFFER = 5
                ; Limit IPO buffer to 5 NC blocks
N110 ...
N120 ...
.....
N200 $SC_MAX_BLOCKS_IN_IPOBUFFER = –1
                ; Cancel limit on IPO buffer
N210 ...
.....
```

2.8.4 Basic block display (SW 6.4 and higher)

Look Ahead basic block display

A so-called basic block display showing all blocks which will generate an action on the machine can be called in parallel to the existing block display. The actually approached end positions are shown as an absolute position. The positional data refer either to the workpiece coordinate system (WCS) or the settable zero system (SZS).

The number of Look Ahead display blocks stored in the display buffer depends on the number of prepared blocks in the NCK preprocessing buffer in the relevant processing state. If a preprocessing stop is processed, the number of display blocks is reduced to zero and increases again after the stop is acknowledged. In the case of REORG events (e.g. mode change, ASUB start), the display blocks stored for Look Ahead are deleted and preprocessed again afterwards.

Processed values

Values preprocessed in the basic block display coincide with the:

- selected tool
- feed and spindle speed
- position values actually approached.
  Exceptions:
  Deviations may occur when tool radius compensation is active.
  With modulo axes, the basic block display shows the programmed value which might also lie outside the modulo range.

Note

Positions are always shown in either the WCS or the SZS.
The basic block display function can be activated or deactivated via setting data SD 42750: ABSBLOCK_ENABLE.

© Siemens AG, 2003. All rights reserved

Configure basic block display

The basic block display can be configured via the following machine data:

NCK machine data for basic block display	Significance:
MD 28400: MM_ABSBLOCK	activate basic block display
MD 28402: MM_ABSBLOCK_BUFFER_CONF[2]	size of display buffer
Machine data display	position values to be set:
MD 9004: DISPLAY_RESOLUTION	for metric measurements
MD 9011: DISPLAY_RESOLUTION_INCH	for inch measurements
MD 9010: SPIND_DISPLAY_RESOLUTION	for spindle display resolution
	coordinate system to be set
MD 9424: MA_COORDINATE_SYSTEM	for actual value display in WCS or SZS

These display machine data are copied to NCK machine data MD 17200: GMMC_INFO_UNIT[0] to MD 17200: GMMC_INFO_UNIT[3], allowing them to be accessed from the NCK.

Activation

The basic block display is activated by MD 28400: MM_ABSBLOCK by means of a POWER ON. If MD 28400: MM_ABSBLOCK is set to 1, a channel-specific display buffer (FIFO)

Size of the display buffer (FIFO) parameterized in (MD 28060: MM_IPO_BUFFER_SIZE + MD 28070: MM_NUM_BLOCKS_IN_PREP) multiplied by 128 bytes is set up at boot. This corresponds to a size of 6KB in the machine data default setting.

Optimize size of display buffer:
The memory requirement can be optimized by entering a value between 128 and 512. The display blocks preprocessed in the display buffer are transferred to the HMI via a configurable upload buffer.

The maximum size of the upload buffer is calculated by multiplying (MD 28402: MM_ABSBLOCK_BUFFER_CONF[0] + MD 28402: MM_ABSBLOCK_BUFFER_CONF[1] + 1) with the block length configured in MD 28400: MM_ABSBLOCK.

The value configured here is the number of blocks located **before** the current block in MD 28402: MM_ABSBLOCK_BUFFER_CONF[0] and **after** the current block in MD 28402: MM_ABSBLOCK_BUFFER_CONF[1].

Supplementary conditions

When the length of a display block configured in MD 28400: MM_ABSBLOCK is exceeded, this display block is truncated accordingly. This is represented by string "..." at the end of the block.

In the case of precompiled cycles (MD 10700: PREPROCESSING_LEVEL > 1), the display block contains **only** axis positions.

Additional boundary conditions for the basic block display:

- Modal synchronized action blocks with absolute values are not taken into account.
- The basic block display is deactivated during block search with or without computation.
- Polar coordinate programming is not shown in Cartesian system.

© Siemens AG, 2003. All rights reserved

Radius / diameter values

Diameter values shown in the basic block display and position display may be needed as a radius for internal calculation. These values can be controlled by the following methods:

- G code DIAMCYCOF (expansion of G code group 26 in SW 6.4 and higher)
  Like DIAMOF, this G code deactivates the diameter programming function. The previously active G code setting remains valid for the position and basic block displays and enables the cycle to calculate with radius, while a diameter value remains displayed in the basic block display.
- MD 27100: ABSBLOCK_FUNCTION_MASK
  Bit0 = 1 Traverse axis setpoints are always shown as diameter values in the basic block display.

Behavior while the compressor is active

With active compressor and G/Code group 30 not equal to COMPOF, two display blocks are generated. The

- First contains the G/Code of the active compressor.
- The second contains the string "..." as character for missing display blocks.

Example:

```
G0 X10 Y10 Z10 ; Block which is still to be processed for basic block display
COMPCAD        ; Compressor for optimized surface quality (CAD-Prog.) on
...            ; String as character for missing display blocks.
```

To avoid bottlenecks in the NCK performance, the basic block display is deactivated automatically. A display block containing the string "..." is generated as character for missing display blocks.
All display blocks are always created in the single block.

Structure for a DIN block

Basic structure of display block for a DIN block

- Block number/label
- G function of first G group
  (only when altered as compared to the last machine function block).
- Axis positions
  (sequence as specified in MD 20070: AXCONF_MACHAX_USED).
- Other modal G functions
  (only when altered as compared to the last machine function block).
- Other addresses as programmed.

The display block for the basic block display is directly derived from the programmed parts program blocks according to the following rules:

- Macros are expanded.
- Skip identifiers and comments are omitted.
- Block number and labels are transferred from the original block, but omitted if DISPLOF is active.
- The number of decimal places is defined in display machine data MD 9004, MD 9010 and MD 9011 via the HMI.

© Siemens AG, 2003. All rights reserved

HMI display machine data	Access in NCK machine data
MD 9004: DISPLAY_RESOLUTION	MD 17200: GMMC_INFO_NO_UNIT[0]
MD 9011: DISPLAY_RESOLUTION_INCH	MD 17200: GMMC_INFO_NO_UNIT[1]
MD 9010: SPIND_DISPLAY_RESOLUTION	MD 17200: GMMC_INFO_NO_UNIT[2]
MD 9424: MA_COORDINATE_SYSTEM	MD 17200: GMMC_INFO_NO_UNIT[3]

Programmed axis positions are displayed as absolute positions in the coordinate system (WKS / SZS) selected via MD 9424: MA_COORDINATE_SYSTEM.

Note

The modulo offset is omitted for modulo axes which means that positions outside the modulo range can be displayed. It also means that the basic block display differs from the position display in which values are always moduloconverted.

Examples

Comparisons between display block (original block) and basic block display:

1. **Programmed positions** are displayed absolutely.
   Addresses AP/RP are shown with their programmed values.

Original block:		Display block:
N10 G90 X10.123	–>	N10 X10.123
N20 G91 X1	–>	N20 X11.123

2. **Address assignments** (non-DIN addresses) are displayed in the form <address> = <constant>.

Original block:		Display block:
N110 R1 = –67.5 R2 = 7.5		
N130 Z = R1 RND = R2	–>	N130 Z–67.5 RND = 7.5

3. **Address indices** (address extensions) are displayed as constants <address > [<constant>] = <constant>

Original block:		Display block:
N220 DEF AXIS AXIS_VAR		
N240 FA[AXIS_VAR] = R2	–>	N240 FA[X] = 1000

4. **DIN addresses without address extension** are displayed in the form <din_adresse> <constant> .

Original block:		Display block:
N410 DEF REAL FEED = 1.5		
N420 F = FEED	–>	N420 F1.5

   The following applies for **H functions**: The value programmed in each case is displayed regardless of the output method to the PLC (MD 22110: AUXFU_H_TYPE_INT).

© Siemens AG, 2003. All rights reserved

5. For the **tool selection via T command,**
   display information in the form of T<value> or T=<string> is generated. If an address extension has been programmed, this is displayed as well.

   If several spindles have been configured or if the “Tool change function via toolholder” (MD 20124: TOOL_MANAGEMENT_TOOLHOLDER) is active, then the T number is always output with address extension.

   If no address extension has been programmed, the number of the master spindle or the master toolholder is used instead (T<spindel_nummer/tool_holder>=).

6. The following rule applies to **spindle programming** via S, M3, M4, M5, M41 – M45 and M70 (or MD 20094: SPIND_RIGID_TAPPING_M_NR) with respect to address extension:
   If an address extension has been programmed, this is displayed as well.

   If several spindles have been configured, the address extension is always displayed as well.
   If no address extension has been programmed, the number of the master spindle is used (S<spindel_nummer>=).

7. **Indirect G code programming** in form G[<group>] =< > is substituted by the corresponding G code.

Original block:		Display block:
N510 R1=2		
N520 G[8]= R1	–>	N520 G54

8. **Modal G codes** that do not generate an executable block are collected and output with the display block of the next executable block if permitted by the syntax (DIN block). If this is not the case (e.g. predefined subroutine call TRANSMIT), a separate display block containing the modified G codes is placed in front of the next executable block.

Original block:		Display block:
N610 G64		G64
N620 TRANSMIT	–>	N620 TRANSMIT

9. For **parts program lines** in which the addresses **F und FA** occur, a display block is always generated
   (including for MD 22240: AUXFU_F_SYNC_TYPE = 3).

Original block:		Display block:
N630 F1000		F630 F1000
N640 X100	–>	N640 X100

10. The **display blocks generated for the block display** are derived **directly** from the programmed parts program blocks. If intermediate blocks (e.g. tool radius compensation G41/G42, radius/chamfer RNDM, RND, CHF, CHR) are generated in the course of contour preprocessing, these are assigned the display information from the parts program block on which the motion is based.

Original block:		Display block:
N710 Y157.5 G42		N710 Y157.5 G42
N720 Z–67.5 RND=7.5	–>	N720 Z–67.5 RND=7.5

11. With the **EXECTAB command** (processing a table of contour elements), the block generated by EXECTAB is shown in the display block.

Original block:	Display block:
N810 EXECTAB (KTAB[5])	N810 G01 X46.147 Z–25.38

© Siemens AG, 2003. All rights reserved

2.8.5 Execution from external source

Application

Individual machining steps for producing complex workpieces may involve program sequences that require so much memory that they cannot be stored in the NC memory. This type of program can be selected and executed from an external device in “Execution from external source” mode. This “external device” takes the following form with HMI Advanced and, with SW 6.2 and higher, with HMI Embedded:

HMI Advanced:	HMI Embedded: (ab SW 6.2)
Local hard disk or a	ATA card or a
Network drive	Network drive

In principle, any program that is accessible via the directory structure of the data management system of the HMI operator interface can be selected and retroloaded.

- **Execution from external source via RS232 interface:**
  With HMI Embedded it is possible to transfer external programs to the NC via the RS232 interface using soft key “Execute from external”.

Extensions

Extensions to “Execution from external” source function:

- **External subroutine (SW 4 and higher)**
  Process subroutines in “Process from external mode”

  The “external” subroutine is called via parts program command EXTCALL with specification of a call path (optional) and the subroutine identifier.

  For more information please see Subsection 2.8.6.

- **Modal execution from external source (SW 4 and higher):**
  “Process from external” selection remains active after Reset or parts program end

  Deselection by next program in the NC memory:
  If a program has been selected for “Execution from external source” via the HMI, this selection remains valid after Reset / parts program end. The “Process from external” mode is not deselected until another program stored in the NC memory is selected for execution.

- **Settable size of reload memory / FIFO buffer (SW 4 and higher):**

  A FIFO buffer must be set up in the NCK in order to execute a program (main program or subroutine) in “Process from external mode”. The default setting for this buffer size is 30KB.

  The machine data MD 18360: MM_EXT_PROG_BUFFER_SIZE defines the size of the reload buffer. The number of the reload buffer is parameterized with MD 18362: MM_EXT_PROG_BUFFER_NUM. A reload buffer must be made available for each of the programs (main run or subroutine) that are processed simultaneously in “Execute from external source” mode.

  Reload buffers are set up in the NCK DRAM. If there is insufficient DRAM available, alarm 4077 “New value of machine data MD 18360/18362 not set” is generated.

© Siemens AG, 2003. All rights reserved

2.8.6 Execute external subroutine (SW 4 and higher)

EXTCALL

The “external” subroutine is called with parts program command

EXTCALL with specified <path and optional> <program name>

Program command EXTCALL is capable of reloading a program in “Execution from external source” mode via the HMI. The external programs must be accessible via the directory structure on the HMI operator interface. They can be stored on the following data media:

HMI Advanced: Local hard disk or network drive

HMI Embedded: ATA card or network drive (with HMI SW 6.2 and higher)

SD 42700 EXT_PROG_PATH

The call path can be set flexibly in SD 42700: EXT_PROG_PATH. SD 42700: EXT_PROG_PATH contains a path name which, together with the programmed subroutine identifier, represents the absolute path name of the program to be called.

If an external subroutine is called without an absolute path name, HMI Advanced runs through the same search path as applied when a subroutine is called from the NC main memory.

Call

An external subroutine is called by means of parts program command **EXTCALL**.

The subroutine name programmed in EXTCALL and

setting data SD 42700: EXT_PROG_PATH

result in the program path by means of a string of characters comprising

- the content of SD 42700: EXT_PROG_PATH (e.g. /_N_WKS_DIR/_N_WKST1_WPD)
- the character “/” as a separator (if a path has been specified via SD 42700: EXT_PROG_PATH)
- the subroutine path or identifier programmed in EXTCALL.

SD 42700: EXT_PROG_PATH is a blank. If an external subroutine is called without an absolute path name, HMI Advanced runs through the same search path as applied when a subroutine is called from the NCK memory.

1. Current directory / Subroutine identifier
2. Current directory / Subroutine identifier_SPF
3. Current directory / Subroutine identifier_MPF
4. /_N_SPF_DIR / Subroutine identifier_SPF
5. /_N_CUS_DIR / Subroutine identifier_SPF
6. /_N_CMA_DIR / Subroutine identifier_SPF
7. /_N_CST_DIR / Subroutine identifier_SPF

“Current directory”: stands for the directory in which the main program has been selected.
“Subroutine identifier”: stands for the subroutine identifier programmed in EXTCALL.

© Siemens AG, 2003. All rights reserved

Examples

Program to be reloaded is stored on the local hard disk of HMI Advanced:

1. SD 42700: EXT_PROG_PATH = "_N_WKS_DIR/_N_WST1"

   Main program _N_MAIN_MPF
   (stored in the NC memory and selected for processing)
   :N010 PROC MAIN
   N020 ....
   N030

   EXTCALL "ROUGHING"
   N040 .....
   N050 M30

   Subroutine _N_ROUGHING_SPF
   (stored in the HMI memory under workpieces–>WST1)
   N010 PROC ROUGHING
   N020 G1 F1000
   N030 X= ... Y= ... Z= ...
   N040 .....
   ....
   ....
   N999999 M17

Program to be reloaded is stored on network drive or ATA card

2. EXTCALL Windows path name

   Call for network drive (HMI Embedded or HMI Advanced) e.g.
   EXTCALL \\R4711\Workpieces\Contour.1.spf

   Call for ATA card (HMI Embedded) e.g.
   EXTCALL C:\Workpieces\Contour.2.spf

Note

An absolute path must always be specified in HMI Embedded.

For further information about using HMI Embedded /Advanced, please see:
References: /BEM/, HMI Embedded, Section 6.9 "Processing from network", Sections 6.10 "Processing from ATA card" and 6.11 "EXTCALL"
/BAD/, HMI Advanced, Access to external network drive/ computer

---

Internal sequence of operations:

The HMI receives the program path of the subroutine to be called in the channel-specific OPI variable "mmcCmdPrep". As a means of identifying that a new "mmcCmdPrep" signal has been transferred, NCK-specific OPI variable "mmcCmdPrepCounter" is incremented.

HMI then starts downloading the requested subroutine and acknowledges this operation in channel-specific OPI variable mmcCmdQuitPrep.

The FIFO buffer for processing the program is set up in NCK directory /_N_SYF_DIR. The buffer name is generated by the NCK and sent to the HMI together with the subroutine path in "mmcCmdPrep".

If a subroutine level is operating in "Process from external" mode, then variable "extProgFlag" in the OPI block SPARPP for the current program pointer is set accordingly.

© Siemens AG, 2003. All rights reserved

POWER ON, RESET

External subroutine calls are aborted and the respective FIFO (reload) buffer erased by a RESET or POWER ON.

A program selected for “Process from external” remains selected for “Process from external” after a Reset. A POWER ON deletes the selection. Default program MPF0 is selected instead.

Note

In “Execution from external source” mode, programs must not contain any branch instructions such as GOTOF, GOTOB, CASE, FOR, LOOP, WHILE, or REPEAT.

Subroutine calls, including nested EXTCALL calls, may be used.

In SW 6.3 and higher, IF ELSE ENDIF CONSTRUCTS are supported.

© Siemens AG, 2003. All rights reserved

2.9 Reset response

2.9.1 System settings for power-up, RESET/end of parts program, and parts program start

Concept

The control system response can be altered for G codes such as tool length compensation, transformation, coupled axis groupings, tangential follow-up and programmable synchronous spindle functions after

- Power up (POWER ON),
- Reset/parts program end and
- Parts program start

by means of machine data
MD 20110: RESET_MODE_MASK
MD 20150: GCODE_RESET_VALUES and, in SW 4.4 and higher, its extension
MD 20152: GCODE_RESET_MODE (definition of control initial setting with Reset) and
MD 20112: START_MODE_MASK (definition of control initial setting after parts program start).

For certain applications (e.g. grinding) it is advisable not to capture the reset position of individual functions until parts program start. This is possible by setting MD 20110, 20150, 20152 and MD 20112.

Table 2-12 Change system settings by means of MD

Status	Can be changed in MD
Power up (POWER ON)	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES
RESET/parts program end	MD 20110: RESET_MODE_MASK MD 20150: GCODE_RESET_VALUES MD 20152: GCODE_RESET_MODE
Parts program start	MD 20110: START_MODE_MASK and MD 20112: RESET_MODE_MASK

© Siemens AG, 2003. All rights reserved

Procedure Select the desired system response.

- After power up (POWER ON)
  MD 20110: RESET_MODE_MASK, bit 0 = 0 or 1

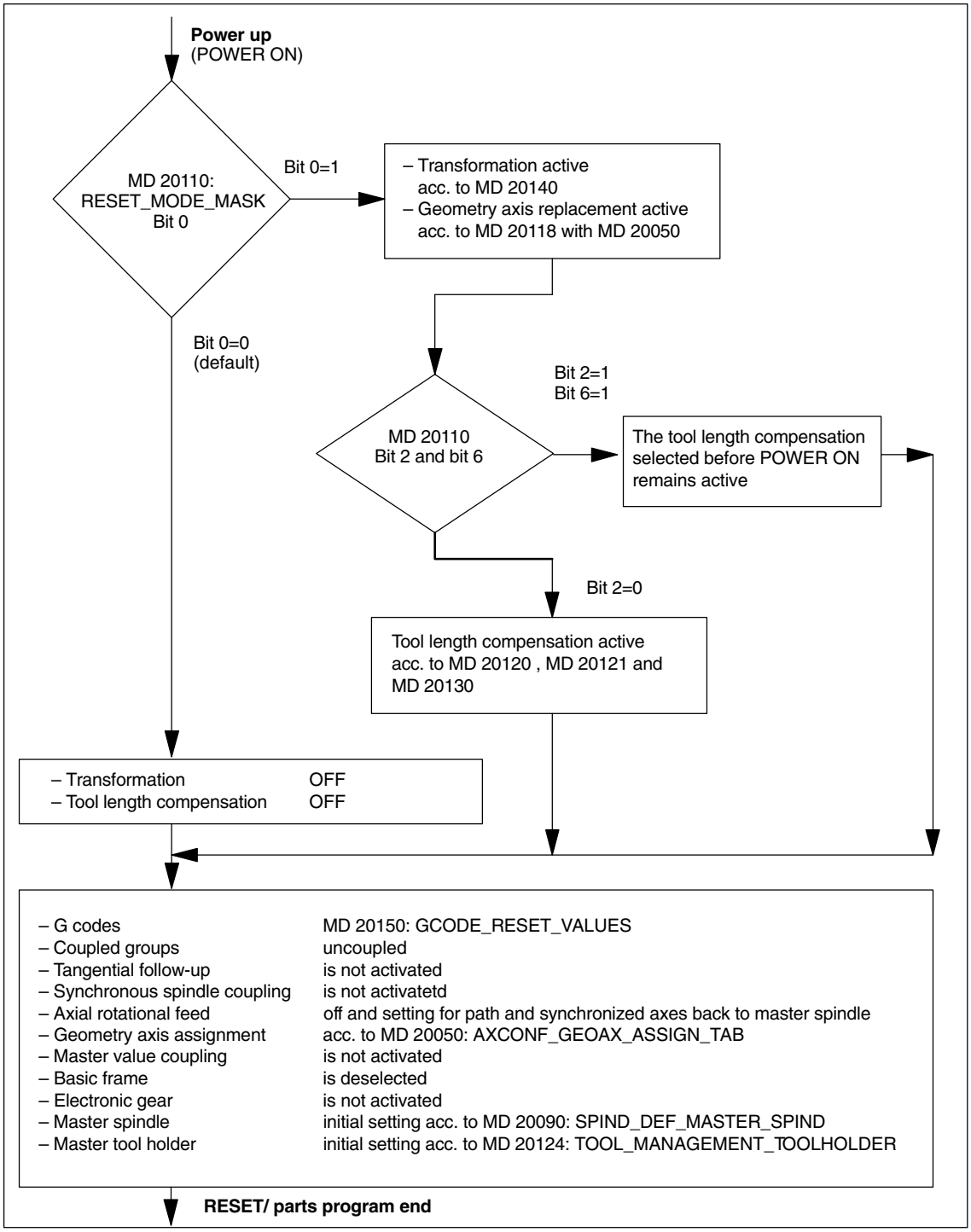


Fig. 2-15 System settings after power up

© Siemens AG, 2003. All rights reserved

- After power up (POWER ON)
  MD 20110: RESET_MODE_MASK, bit 0 = 0 or 1

Bits 4 – 13 can be combined optionally.

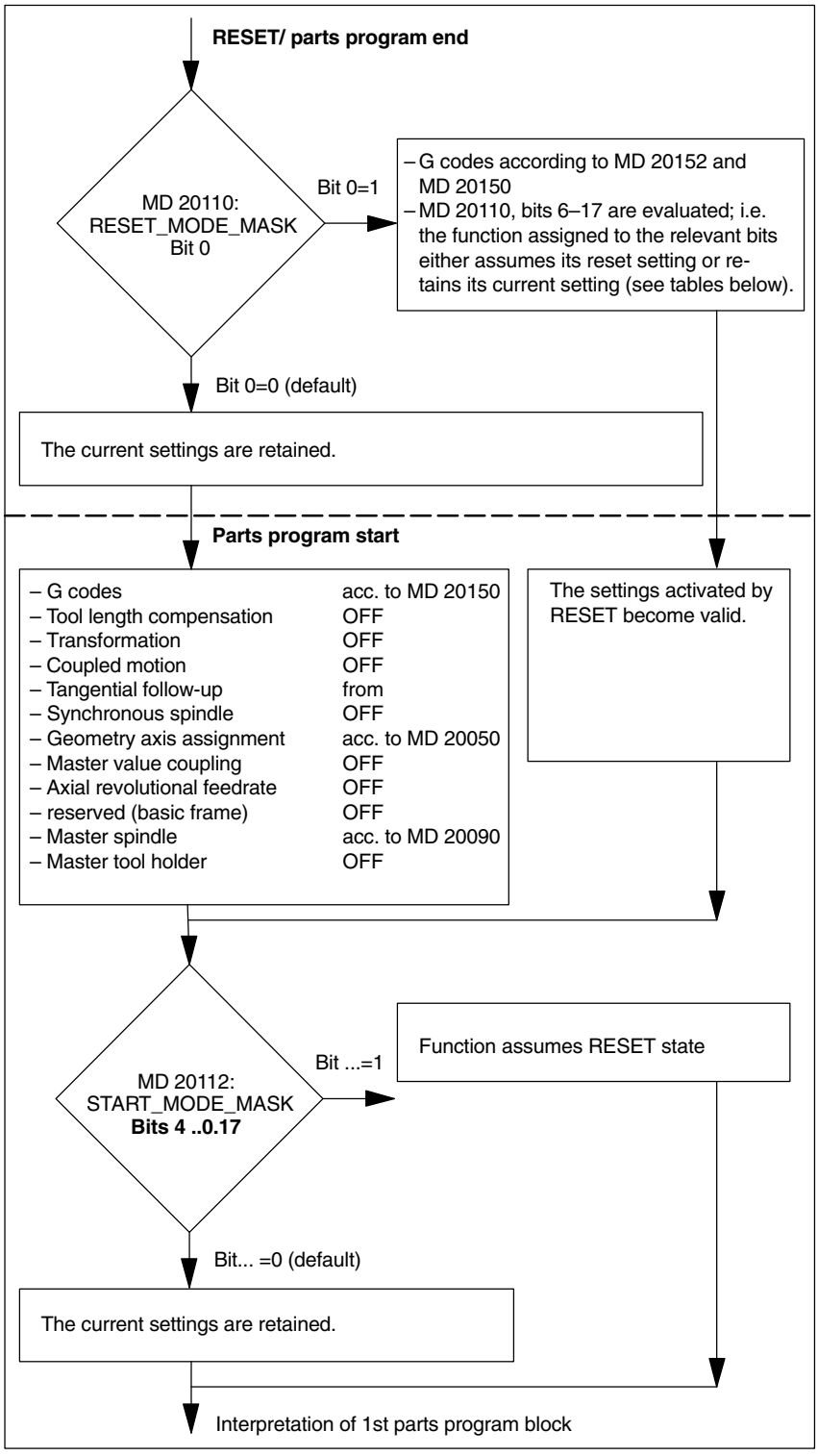


Fig. 2-16 System settings after RESET/parts program end and parts program start

© Siemens AG, 2003. All rights reserved

Table 2-13 Selection of RESET and powerup response

20110 MD no.	**RESET_MODE_MASK** Definition of control initial setting after power up and reset/parts program end	
Behavior after	**Bit 0 = 0**	**Bit 0 = 1**
POWER ON (booting)	– Transformation not active – Tool length compensation not active – G codes according to MD 20150: GCODE_RESET_VALUES – No coupled-axis groupings active – No tangential follow-up active – Non-configured Synchronous spindle coupling is switched off – Geometry axis assignm. acc. to MD 20050 – No master value coupling active – Axial revolutional feedrate off – Basic frame is deselected	– Transformation active acc. to MD 20140: TRAFO_RESET_VALUE – Tool length compensation active acc. to MD 20120: TOOL_RESET_VALUE, MD 20121: TOOL_PRESEL_RESET_VALUE and MD 20130: CUTTING_EDGE_RESET_VALUE, if bits 2 and 6 in MD 20110 = 1, tool length compensations selected at POWER ON, otherwise TLC acc. to MD 20120, 20121, 20130 – G codes according to MD 20150: GCODE_ RESET_VALUES – No coupled-axis groupings active – No tangential follow-up active – Non-configured Synchronous spindle deactivated – Geometry axis assignment acc. to MD 20118, MD 20050 – No master value coupling active – Axial revolutional feedrate off – Basic frame is deselected
RESET/ progr. end	The current settings are retained. With the next parts program start, the following initial setting is activated:	When a synchronous spindle coupling is configured, the coupling is set as a function of MD 21330: COUPLE_RESET_MODE_1.
	– G codes acc. to MD 20150: GCODE_RESET_VALUES – Tool length compensation not active – Transformation not active – No coupled-axis groupings active – No tangential follow-up active – Basic frame is deselected – Geometry axis assignm. acc. to MD 20050 – Axial revolutional feedrate off	GCODES acc. to MD 20150, 20152

Table 2-14 Effect of MD 20110: **RESET_MODE_MASK** Bits 0...6

Bit 0 = 1	**Bit 1 = 1**	**Bit 2 = 1**	**Bit 3 = 1**	**Bit 4 = 1**	**Bit 5 = 1**	**Bit 6 = 1**
Initial setting after power up and RESET/end of parts program see above	No D, T, M output on tool selection; with active tool management irrelevant	If bit 0=1 and bit 6 =1, the tool offset of the last active tool is active after POWER ON (irrelevant with active tool management)	Use active tool or tool offset from last test program terminated in test mode	Current plane is retained, as of SW 5 replaced by MD 20152 index 5 see below	Current settable frame is retained, as of SW 5 replaced by MD 20152 index 7 see below	active tool offset is retained
Bit 0 = 0	**Bit 1 = 0**	**Bit 2 = 0**	**Bit 3 = 0**	**Bit 4 = 0**	**Bit 5 = 0**	**Bit 6 = 0**
Initial setting after power up and RESET/end of parts program see above	D, T, M output on tool selection; with active tool management irrelevant	No tool offset active after POWER ON (no effect when tool management active)	Use active tool or tool offset from last program terminated before program testing activated	Plane acc. to MD 20150: GCODE_ RESET_ VALUES; as of SW 5 replaced by MD 20152 index 5 see below	Frame acc. to MD 20150: GCODE_ RESET_ VALUES; as of SW 5 replaced by MD 20152 index 7 see below	TO acc. to MD 20120: TOOL_RESET_VALUE, MD 20121: TOOL_PRESEL_RESET_VALUE, MD 20130: CUTTING_ EDGE_RESET_VALUE; Output of D, T, M to PLC, depending on bit 1

© Siemens AG, 2003. All rights reserved

Table 2-14 Effect MD 20110: **RESET_MODE_MASK** Bits 7 ...12

Bit 7 = 1	**Bit 8 = 1**	**Bit 9 = 1**	**Bit 10 = 1**	**Bit 11 = 1**	**Bit 12 = 1**
Active transformation is retained	Coupled-motion groupings are retained	Tangential correction is retained	Non-configured synchronous spindle coupling remains active	Current setting for revolutional feedrate is retained	Modified geometry axis assignment is retained
Bit 7 = 0	**Bit 8 = 0**	**Bit 9 = 0**	**Bit 10 = 0**	**Bit 11=0**	**Bit 12 = 0**
Transformation acc. to MD 20140: TRAFO_RESET_VALUE	Coupled-axis groupings are separated	Tangential follow-up is disabled	Non-configured synchronous spindle coupling is switched off	Revolutional feedrate is no longer valid	Modified geometry axis assignment is deleted according to MD 20050, depending on MD 20118 (because of compatibility)

Table 2-15 Effect of MD 20110: **RESET_MODE_MASK** Bits 13...17 (SW 6.4 and higher bits 16 to 17)

Bit 13 = 1	**Bit 14 = 1**	**Bit 15 = 1**	**Bit 16 = 1**	**Bit 17 = 1**	
Active guide value coupling is retained	The current setting of the basic frame is retained.	The active electronic gears are deactivated	The current setting of the master spindle is retained.	The current setting of the master tool holder is retained.	
Bit 13 = 0	**Bit 14 = 0**	**Bit 15 = 0**	**Bit 16 = 0**	**Bit 17 = 0**	
Guide value coupling is separated	Basic frame is deselected	The active electronic gears are retained	Initial setting for the master spindle acc. to MD 20090	Initial setting for the master tool holder acc. to MD 20124	

RESET behavior of the master spindle up to SW 6.3

Up to SW 6.3 the master spindle setting was reset to the configured value on M30/RESET. Two cases can apply depending on the setting of bit 0 of MD 20110: RESET_MODE_MASK:

Bit 0 = 0: No init blocks are generated. (Default: The current settings are retained)
Bit 0 = 1: Init blocks are generated. The settings activated by RESET become valid.

MD 20152: GCODE_ RESET_MODE (SW 5 and higher)

In SW 5 and higher, MD 20152: GCODE_RESET_MODE replaces bits 4 and 5 from MD 20110: RESET_MODE_MASK. There is also an extension of the setting options:

In SW 4 and lower,
the following applied for bits 4 and 5 of MD 20110: RESET_MODE_MASK:
Bit 4: Control of level
Bit 5: Control of configurable frames

In SW 5 and higher, every G code group in MD 20150: GCODE_RESET_VALUES[i] can be controlled directly with the additional MD 20152: GCODE_RESET_MODE[i].

MD 20152: GCODE_RESET_MODE[i] (i = G code group –1)	**0**	**1**
MD 20150: GCODE_RESET_VALUES[i]	The value stored in MD 20150 is active	The last active/current value is active

© Siemens AG, 2003. All rights reserved

Note

The previous setting in machine data MD20110: RESET_MODE_MASK no longer applies!
The corresponding bits of this MD are tagged as reserved. Write operations involving these bits are automatically redirected to the corresponding array elements of MD20150: GCODE_RESET_MODE and an alarm (4502) is output.

When reading MD20110: RESET_MODE_MASK, the information is read from the corresponding array elements of MD20150: GCODE_RESET_MASK and written to MD20110: RESET_MODE_MASK.

The effect of MD 20112: START_MODE_MASK is unchanged in SW 5.

Parts program start

The basic setting of the control system at parts program start, e.g. G codes (especially active plane and active settable zero offset), active tool length compensation, transformation and axiscoupling, is determined according to the following table.

Application

By setting a bit in MD 20112: START_MODE_MASK, the Reset action of the relevant function can be delayed until the start of the parts program.

Table 2-16 Effect of MD 20112: **START_MODE_MASK** Bits 1 ..0.7

Bit 1 = 1	**Bit 2 = 1**	**Bit 3 = 1**	**Bit 4 = 1**	**Bit 5 = 1**	**Bit 6 = 1**	**Bit 7 = 1**
No D, T, M output on tool selection; with active tool management irrelevant	reserved	reserved	Plane acc. to MD 20150: GCODE_RESET_VALUES	Frame acc. to MD 20150: GCODE_RESET_VALUES	TO acc. to MD 20120: TOOL_RESET_VALUE, MD 20121: TOOL_PRESEL_RESET_VALUE, and MD 20130: CUTTING_EDGE_RESET_VALUE; Output of D, T, M to PLC, depending on bit 1	Transformation acc. to MD 20140: TRAFO_RESET_VALUE
Bit 1 = 0	**Bit 2 = 0**	**Bit 3 = 0**	**Bit 4 = 0**	**Bit 5 = 0**	**Bit 6 = 0**	**Bit 7 = 0**
D, T, M output on tool selection; with active tool management irrelevant	reserved	reserved	Current plane is retained	Current settable frame is retained	Active tool length offset is retained	Active transformation is retained

© Siemens AG, 2003. All rights reserved

Table 2-17 Effect of MD 20112: **START_MODE_MASK** Bits 8 ..0.12

Bit 8 = 1	**Bit 9 = 1**	**Bit 10 = 1**	**Bit 11 = 1**	**Bit 12 = 1**
Coupled-motion groupings are deactivated	Tangential follow-up from deactivated	Non-configured synchronous spindle coupling is switched off	reserved	Geometry axis assignment is deleted according to MD 20050, depending on MD 20118 (because of compatibility)
Bit 8 = 0	**Bit 9 = 0**	**Bit 10 = 0**	**Bit 11 = 0**	**Bit 12 = 0**
Coupled-motion groupings are retained	Tangential correction is retained	Non-configured synchronous spindle coupling remains active	reserved	Modified geometry axis assignment is retained

Table 2-17 Effect MD 20112: **START_MODE_MASK** Bits 13...17 (SW 6.4 and higher bits 16 to 17)

Bit 13 = 1	**Bit 14 = 1**	**Bit 15 = 1**	**Bit 16 = 1**	**Bit 17 = 1**
Guide value coupling is separated	reserved for Basic frame	reserved for electronic gears	Initial setting for master spindle acc. to MD 20090: SPIND_DEF_MASTER_SPIND	Only if MD 20124: TOOL_MANAGEMENT_TOOLHOLDER > 0: Initial setting for the master tool holder acc. to MD 20124. Otherwise setting for master spindle.
Bit 13 = 0	**Bit 14 = 0**	**Bit 15 = 0**	**Bit 16 = 0**	**Bit 17 = 0**
Active guide value coupling is retained	reserved for Basic frame	reserved for electronic gears	current setting of the master spindle (SETMS) is retained	current setting of the Master tool holder (SETMS) is retained

Note

- Bits set to 1 in MD 20110: RESET_MODE_MASK cause settings to be retained,
- in MD 20112: START_MODE_MASK cause settings to be retained.

Meaning of the machine data

The channel-specific machine data in the table have the following meanings. Details are specified in Subsection 4.4.3.

MD 20120: TOOL_RESET_VALUE
Definition of tool (T number),
whose assigned tool length compensation values must be taken into account on RESET and power up according to **MD 20110: RESET_MODE_MASK** are to be taken into account.

MD 20121: TOOL_PRESEL_RESET_VALUE
Definition of tool (T number) as preselected tool whose assigned tool length compensation values must be taken into account on RESET and power up according to MD **20110: RESET_MODE_MASK** are to be taken into account.

© Siemens AG, 2003. All rights reserved

MD 20130: CUTTING_EDGE_RESET_VALUE
Definition of edge number (D number) of tool in $MC_TOOL_RESET_VALUE

MD 20140: TRAFO_RESET_VALUE
Definition of transformation data block (TRAORI, TRAANG, SW 4 and higher: TRANSMIT)

MD 20150: GCODE_RESET_VALUES
Initial settings of G groups

MD 20152: GCODE_RESET_MODE
GCODE initial setting on RESET

MD 20152 defines for each entry in MD 20150: GCODE_RESET_VALUES whether the setting in MD 20150: GCODE_RESET_VALUES is applied again on a reset/end of parts program (entry in MD 20152=0), or whether the current setting is retained (entry in MD 20152=1).

MD 21330: COUPLE_RESET_MODE_1
Cancellation of an axis coupling

MD 20050 AXCONF_GEOAX_ASSIGN_TAB
Assignment of geometry axis to channel axis
active with MD 20110/20112 on bit 12
MD 20118 GEOAX_CHANGE_RESET
active with MD 20110/20112 on bit 12

Example:

1. Activate RESET setting on RESET:
   MD 20110 = 'H01' (Bit 0)
   MD 20112 = '0'
2. Transformation is retained on reset/parts program start:
   MD 20110 = 'H81' (Bit 0 + Bit 7)
   MD 20112 = '0'
3. Tool length compensation is retained beyond reset/parts program start:
   MD 20110 = 'H41' (Bit 0 + Bit 6)
   MD 20112 = '0'
4. Active plane (bit 4) and settable frame (bit 5) are retained beyond reset and are reset on parts program start:
   MD 20110 = 'H31' (Bit 4 + Bit 5)
   MD 20112 = '30'

Note

For bit 5 and bit 6:

If the tool length compensation or a frame is active on parts program start/MDA start, due to the configuration of MD 20110/MD 20112, the first programming instruction for axes must use absolute dimensions (because of the offset response).
Exception: MD 42442/MD 42440 has been used to suppress the offset response on G91.

© Siemens AG, 2003. All rights reserved

2.10 Subroutine call with M and T functions

General

For some applications it can be advantageous to replace M and/or T functions by a subroutine call. This can be used, for example, to call the tool change routine. The machine data below can be used to define and control subroutines accordingly with M and/or T functions:

Machine data for M function	**Significance:**
MD 10715: M_NO_FCT_CYCLE	M function to be replaced by SR
MD 10716: M_NO_FCT_CYCLE_NAME	SR name for M function replacement
MD 10718: M_NO_FCT_CYCLE_PAR	M function replacement with parameter transfer (SW 6.3 and higher)
Machine data for T function	
MD 10717: T_NO_FCT_CYCLE_NAME	SR name for T function replacement
MD 10719: T_NO_FCT_CYCLE_MODE	Parameterization of T function replacement

During block searches, subroutine calls with M and T functions behave in the same way as standard subroutine calls.

Mapping M and T programming onto cycle calls works in both ISO-dialect-mode and in Siemens-mode.

M functions with predetermined meaning

A subroutine call **must not** be superimposed on M functions with predetermined meaning. In the case of a conflict, alarm 4150 is generated.

M functions with predetermined meaning:

- M0 to M5,
  M17, M30,
  M19,
  M40 to M45,
- for switching between spindle/axis mode as set in
  MD 20094: SPIND_RIGID_TAPPING_M_NR (default: M70),
  MD 10714: M_NO_FCT_EOP (M function for spindle active after RESET)
  MD 22254: AUXFU_ASSOC_M0_VALUE (additional M fct. for program stop)
  MD 22256: AUXFU_ASSOC_M1_VALUE (additional M fct. for conditional stop)
- for nibbling/punching as configured via
  MD 26008: NIBBLE_PUNCH_CODE if activated via
  MD 26012: PUNCHNIB_ACTIVATION.
- for applied external language (MD 18800: MM_EXTERN_LANGUAGE) plus M98 and M99.
- MD 10804: EXTERN_CHAN_M_NO_SET_INT (default setting M96)
  MD 10806: EXTERN_CHAN_M_NO_DISABLE_INT (default setting M97)
- MD 10814: EXTERN_M_NO_MAC_CYCLE (macro call M fct.)
- MD 20095: EXTERN_RIGID_TAPPING_M_NR (default setting M29)

Exception:

Subroutines configured in MD 22560: TOOL_CHANGE_M_CODE for tool changes.

© Siemens AG, 2003. All rights reserved

2.10.1 M function replacement

Assign subroutine

Using machine data MD 10715: M_NO_FCT_CYCLE and
MD 10716: M_NO_FCT_CYCLE_NAME, it is possible to assign a subroutine with name to an M function. Subroutine calls using an M function are referred to below as M function replacement.
The M functions with predetermined meaning shown in Fig. 2.10 must not have a subroutine call superimposed on them.

Start subroutine

If the M function defined in MD 10715: M_NO_FCT_CYCLE[n] is included in a parts program block, the subroutine defined in
MD 10716: M_NO_FCT_CYCLE_NAME[n] is started at the end of the block.

When the M function is occurs again in the subroutine, it is not replaced by a subroutine call.
Exception:
The replacement is made in the ASUB even if it has been started in a replaced subroutine.

Using system variable $C_ME it is possible to interrogate the address extension of the substituted M function in the replaced subroutine.

Supplementary conditions

The following conditions apply to subroutine calls with M function:

- A subroutine call must not be superimposed on M functions with predetermined meaning.

  Exception:
  The M function parameterized in MD 22560: TOOL_CHANGE_M_CODE for tool changes.

- The following applies to blocks containing an M function replacement:
  – No modal subroutine call may be active or
  – No subroutine return jump may be programmed or
  – No parts program end may be programmed.

- Subroutines configured in MD 10716: M_NO_FCT_CYCLE_NAME[n] and MD 10717: T_NO_FCT_CYCLE_NAME must not be active simultaneously in the same block (parts program line), i.e. no more than one M or T function replacement (or generally just one subroutine call) may be executed per block. Conflicts with other subroutine calls are signaled by alarm 14016.

- A maximum of one M function may be configured in SW 6.2 and lower.

Extensions SW 6.3 and higher

In SW 6.3 and higher, the following features are added to M function replacement:

- Configuring of a maximum of 10 M function replacements via machine data MD 10715: M_NO_FCT_CYCLE and MD 10716: M_NO_FCT_CYCLE_NAME. No further M function replacement is made in the substituted subroutine.

- Parameter transfer per system variable as for T function replacement for an M function replacement selected in machine data MD 10718: M_NO_FCT_CYCLE_PAR. This feature enables the tool change cycle to be called via M function.

© Siemens AG, 2003. All rights reserved

MD 10718 **M functions with parameter transfer per system variable**

If an M function replacement has been configured with MD 10715: M_NO_FCT_CYCLE[n] and MD 10716: M_NO_FCT_CYCLE_NAME[n], it is possible to specify a parameter transfer per system variable (analogous to T function replacement) using MD 10718: M_NO_FCT_CYCLE_PAR. The following system variables are available:

$C_ME	: Address extension of substituted M function
$C_T_PROG	: TRUE if address T has been programmed
$C_T	: Value of address T (integer)
$C_TE	: Address extension of address T
$C_TS_PROG	: TRUE if address TS has been programmed
$C_TS	: Value of address TS (string, with tool management only)
$C_D_PROG	: TRUE if address D has been programmed
$C_D	: Value of address D
$C_DL_PROG	: TRUE if address DL has been programmed
$C_DL	: Value of address DL

Note

Values passed to the cycle have not yet been executed and must therefore be programmed again in the cycle.
Configuring errors in MD 10718: M_NO_FCT_CYCLE_PAR are signaled at boot with alarm 4150.

Boundary condition:

Programming of the M function replacement with parameter transfer is restricted in the respect that both address extension as well as M function value must be programmed as constants. Indirect programming is rejected with alarm 14017.

Permissible programming variants:

– M<m function value>
– M=<m function value>
– M[<address extension>]=<m function value>
<m function value> and <address extension> are constants.

Impermissible programming variants:

– M=<variable_1>
– M[<variable_2>]=<variable_1>

Examples

Configuring examples:

- Call of subroutine SUB_M101 by M function M101
  MD 10715: M_NO_FCT_CYCLE[0] = 101
  MD 10716: M_NO_FCT_CYCLE_NAME[0] = “SUB_M101”
- Call of subroutine SUB_M102 by M function M102
  MD 10715: M_NO_FCT_CYCLE[1] = 102
  MD 10716: M_NO_FCT_CYCLE_NAME[1] = “SUB_M102”
- Call of subroutine SUB_M6 by M function M6 with parameter transfer
  MD 10715: M_NO_FCT_CYCLE[2] = 6
  MD 10716: M_NO_FCT_CYCLE_NAME[2] = “SUB_M6”
  MD 10718: M_NO_FCT_CYCLE_PAR = 6

© Siemens AG, 2003. All rights reserved

2.10.2 T function replacement

Assign subroutine

The MD 10717: T_NO_FCT_CYCLE_NAME is used to assign a subroutine to a T command. Each block containing a T command is executed first before the subroutine is called.

The T value is not output, the T word must be programmed again in the cycle.

Call subroutine

If a T function is programmed in a parts program block, the subroutine defined in MD 10717: T_NO_FCT_CYCLE_NAME[n] is called at the end of the block. Subroutine calls using an T function are referred to below as T function replacement.

The subroutine is supplied with the following parameters via system variables in the cycle:

$C_ME	: Address extension of substituted M function
$C_T_PROG	: TRUE if address T has been programmed
$C_T	: Value of address T (integer)
$C_TE	: Address extension of address T
$C_TS_PROG	: TRUE if address TS has been programmed
$C_TS	: Value of address TS (string, with tool management only)
$C_D_PROG	: TRUE if address D has been programmed
$C_D	: Value of address D
$C_DL_PROG	: TRUE if address DL has been programmed
$C_DL	: Value of address DL

Values passed to the cycle have not yet been executed and must therefore be programmed again in the cycle.

Supplementary conditions

The following conditions apply to subroutine calls with T function:

- A maximum of one T function may be configured.
- The following applies to blocks containing a T function replacement:
  – No modal subroutine call may be active or
  – No subroutine return jump may be programmed or
  – No parts program end may be programmed.
- Subroutines configured in MD 10716: M_NO_FCT_CYCLE_NAME[n] and MD 10717: T_NO_FCT_CYCLE_NAME must not be active simultaneously in the same block (parts program line), i.e. no more than one T function replacement (or generally just one subroutine call) may be executed per block. Conflicts with other subroutine calls are signaled by alarm 14016.
- **M and T functions for tool change in a block**
  If T function replacement is programmed in addition to M function replacement with parameter transfer, then the system response is as follows if these settings conflict, i.e. T and M functions for tool change in one block:
  – The T function replacement is not executed.
    Instead, the T value is made available to the M function replacement via the appropriate system variable $C_T... .
  – Programming the address T in the substituted M function subroutine does not result in another substitution.

© Siemens AG, 2003. All rights reserved

Extensions SW 6.4 and higher

The T function replacement has been extended such that machine data MD 10719: T_NO_FCT_CYCLE_MODE can be set do define when D or DL and T are programmed in the same block whether

- D or DL must be passed as the parameter to the T replacement cycle or
- whether they must be executed before the T replacement cycle is called.

MD 10719

Parameterization of T function replacement with machine data MD 10719: T_NO_FCT_CYCLE_MODE as follows:

Value 0: As earlier, the D or DL number is passed directly to the cycle (default setting)

Value 1: The D or DL number is calculated directly in the block

This function is active only if the tool change has been configured with M function (MD 22550: TOOL_CHANGE_MODE = 1); the D or DL values are otherwise always transferred.

2.10.3 Examples for M/T function replacement for tool change

M6 not active

Tool change with M6 not active (behavior as in earlier version):

```
MD 22550: TOOL_CHANGE_MODE = 0
MD 10719: T_NO_FCT_CYCLE_MODE = 0
MD 10717: T_NO_FCT_CYCLE_NAME = “MY_T_CYCLE”   ; T replacement cycle

N110 D1                          ;
N120 G90 G0 X100 Y100 Z50        ; D1 is active
N130 D2 X110 Z0 T5               ; D1 remains active, programmed D2 is made
...                              ; available to T replacement cycle as variable
```

M6 active and MD 10719 = 0

Tool change with M6 active and MD 10719 T_NO_FCT_CYCLE_MODE = 0 behavior as in earlier versions.

```
MD 10719: T_NO_FCT_CYCLE_MODE = 0
MD 10717: T_NO_FCT_CYCLE_NAME = “MY_T_CYCLE”   ; T replacement cycle

N210 D1                          ;
N220 G90 G0 X100 Y100 Z50        ; D1 is active
N230 D2 X110 Z0 T5               ; D1 remains active, programmed D2 is made
                                 ; available to T replacement cycle as variable
N240 M6                          ; New tool is selected
```

© Siemens AG, 2003. All rights reserved

M6 active and MD 10719 = 1

Tool change with M6 active and MD 10719 T_NO_FCT_CYCLE_MODE = 1
Behavior **in SW 6.4 and higher**

MD 22550: TOOL_CHANGE_MODE = 1
MD 10719: T_NO_FCT_CYCLE_MODE = 1
MD 10717: T_NO_FCT_CYCLE_NAME = “MY_T_CYCLE” ; **T replacement cycle**

```
N310 D1                          ;
N320 G90 G0 X100 Y100 Z50        ; D1 is active
N330 D2 X110 Z0 T5               ; D2 is activated, D2 is not passed
                                 ; to T replacement cycle as variable
N340 M6                          ; T5 is activated.
```

Plus parameter transfer

Tool change with M6 active and MD 10719 T_NO_FCT_CYCLE_MODE = 1
Behavior **in SW 6.4 and higher**

A replacement cycle for T and M6 has been configured. In addition, parameter transfer to the M6 replacement cycle has been configured with MD 10718: M_NO_FCT_CYCLE_PAR.

If M6 is now programmed with D or DL in the block, the D or DL numbers are also passed as parameters to the M6 replacement cycle when machine data MD 10719 T_NO_FCT_CYCLE_MODE = 1.

MD 22550: TOOL_CHANGE_MODE = 1
MD 10719: T_NO_FCT_CYCLE_MODE = 1
MD 10717: T_NO_FCT_CYCLE_NAME = “MY_T_CYCLE” ; **T replacement cycle**

```
N410 D1                          ;
N420 G90 G0 X100 Y100 Z50        ; D1 is active
N4130 D2 X110 Z0 T5 M6           ; D1 remains active, D2 and T5 are passed to M6
                                 ; replacement cycle as variables
```

Tool change with M function

Programming example:

```
PROC MAIN
...
N10 T1 D1 M6
...
N90 M30

PROC SUB_M6
...
N110 IF $C_T_PROG == TRUE        ; Scan whether address T has been programmed
N120   T[ $C_TE ] = $C_T         ; Make T selection
N130 ENDIF
N140 M[$C_ME] = 6                ; Execute tool change
N150 IF $C_D_PROG == TRUE        ; Scan whether address D has been programmed
N160   D = $C_D                  ; Make D selection
N170 ENDIF
N190 M17
```

© Siemens AG, 2003. All rights reserved

2.11 Program runtime/workpiece counter (SW 5.2 and higher)

Introduction

To assist the operator on the machine tool, information on the program runtime and for part counting are provided.

The functions defined for this purpose are **not identical to the functions of tool management** and are intended primarily for systems without tool management

The desired information must be specified in defined machine data and can be manipulated via system variables in the NC and/or PLC program. The information is available to the MMC via the operator panel ↔ PLC interface.

A dedicated display screen for the SINUMERIK 802D control is provided for program runtime and workpiece counter.

2.11.1 Program runtime

Functionality

The “Program runtime” function presents timers in system variables which can be used in particular for monitoring technological processes. Access to these timers is read-only.

The following timers are defined as NCKspecific system variables and are always active:

Name	Meaning	Description
$AN_SETUP_TIME	Time since the last control power-up with default values (“Cold start” in min.)	Counts the time since the last control power-up with default values. It is automatically reset to the default values each time the control is powered up.
$AN_POWERON_TIME	Time since the last normal control powerup (“Warm start” in min.)	Counts the time since the last normal control power-up. It is automatically reset to the default values on each normal control power-up.

Several timers are available as channel-specific system variables and can be activated via a channel machine data. Each active runtime measurement is interrupted automatically by a program status ≠ “Program running” and an active override = 0.

The behavior of the activated timers for active dry run feedrate and program testing can be specified using machine data.

© Siemens AG, 2003. All rights reserved

The following channel-specific timers are defined:

Name	Meaning	Description
$AC_OPERATING_TIME	Total runtime of NC programs in Automatic mode (in s)	Totals the accumulated runtimes of all programs between NC Start and end of program/NC reset. Is automatically reset each time the control is powered up.
$AC_CYCLE_TIME	Runtime of the selected NC program (in s; always only one active per channel)	The runtime between NC Start and end of program / NC reset is measured in the selected NC program. Is automatically reset each time a new NC program is started. See also MD 27860.
$AC_CUTTING_TIME	Tool operating time (in s)	The measurement produces the runtime of the path axes (at least one is active) without active rapid traverse in all NC programs between NC Start and program end / NC reset with active tool. The measurement is also interrupted when a dwell time is active. It is automatically reset to the default values each time the control is powered up.

Note

The MMC has continuous read-only access to the timers.

Condition

All timers are reset to the default values each time the control is powered up and can be read independent of their activation.

Activation

The timers for normal control power-up and control power-up with default values are always active. All other timers must be activated channel-specifically in machine data 27860: $MC_PROCESSTIMER_MODE. These timers are automatically activated with the standard machine data for the SINUMERIK 802D control.

Examples

1. Activating the runtime measurement for the active NC program (no measurement with active dry run feedrate and program testing):
   $MC_PROCESSTIMER_MODE = ‘H2‘
2. Activating the measurement for the tool action time (measurement also with dry run feedrate and program testing):
   $MC_PROCESSTIMER_MODE = ‘H34‘
3. Activating the measurement for the total runtime and the tool action time (measurement also with program test):
   $MC_PROCESSTIMER_MODE = ‘H25‘

© Siemens AG, 2003. All rights reserved

2.11.2 Workpiece counter

Functionality

The “workpiece counter” function prepares counters which can be used in particular for internal counting of workpieces on the control. These counters exist in the form of channel-specific system variables with read-only and read/write access and have a value range from 0 to 999 999 999.

The channel machine data can be used to influence the timer activation, the reset time and the counting algorithm.

The following counters are available:

Name	Meaning	Description
$AC_REQUIRED_PARTS	Number of workpieces required (workpiece setpoint)	Determination of the number of workpieces at which the counter for the current workpieces is reset ($AC_ACTUAL_PARTS). The generation of the display alarm workpiece setpoint reached and the channel VDI signal “workpiece setpoint reached” can be activated via MD.
$AC_TOTAL_PARTS	Total number of workpieces produced (total actual)	Specifies the total number of all workpieces produced since the start time. Is automatically reset to the default values only when the control is powered up.
$AC_ACTUAL_PARTS	Number of actual workpieces (actual)	Records the number of all workpieces produced since the start time. Automatically reset when the workpiece setpoint ($AC_REQUIRED_PARTS) is reached (whereby it is assumed: $AC_REQUIRED_PARTS $\neq$ 0).
$AC_SPECIAL_PARTS	Number of workpieces specified by the user	Allows the user to define how the workpieces are counted. An alarm output can be defined on matching $AC_REQUIRED_PARTS. Must be reset manually by the user.

Note

The “workpiece counter” function is independent of the tool management functions. All counters can be read and written from the MMC.

Supplementary condition

All counters are reset to the default values each time the control is powered up and can be read and written independent of their activation.

If the counter $AC_REQUIRED_PARTS = 0, no ID check is performed with the counter pulse for $AC_ACTUAL_PARTS or $AC_SPECIAL_PARTS even if the MD bit is set.

Activation

Each counter is activated via a channel machine data. All workpiece counters are declared as inactive in the default machine data set.

The counters are activated by default for the SINUMERIK 802D.

© Siemens AG, 2003. All rights reserved

Examples

1. Activating the workpiece counter $AC_REQUIRED_PARTS:
   $MC_PART_COUNTER = ‘H3‘
   $AC_REQUIRED_PARTS is active
   display alarm with
   $AC_REQUIRED_PARTS ≡ $AC_SPECIAL_PARTS
2. Activating the workpiece counter $AC_TOTAL_PARTS:
   $MC_PART_COUNTER = ‘H10‘
   $MC_PART_COUNTER_MCODE[0] = 80
   $AC_TOTAL_PARTS is active;
   counter is increased by 1 with each M02.
   $MC_PART_COUNTER_MCODE[0] has no meaning.
3. Activating the workpiece counter $AC_ACTUAL_PARTS:
   $MC_PART_COUNTER = ‘H300‘
   $MC_PART_COUNTER_MCODE[1] = 17
   $AC_TOTAL_PARTS is active;
   the counter is increased by 1 with each M17.
4. Activating the workpiece counter $AC_SPECIAL_PARTS:
   $MC_PART_COUNTER = ‘H3000‘
   $MC_PART_COUNTER_MCODE[2] = 77
   $AC_SPECIAL_PARTS is active;
   the counter is increased by 1 with each M77.
5. Deactivating the workpiece counter $AC_ACTUAL_PARTS:
   $MC_PART_COUNTER = ‘H200‘
   $MC_PART_COUNTER_MCODE[1] = 50
   $AC_TOTAL_PARTS is not active; remainder without significance.
6. Activating all counters in examples 1–4:
   $MC_PART_COUNTER = ‘H3313‘
   $MC_PART_COUNTER_MCODE[0] = 80
   $MC_PART_COUNTER_MCODE[1] = 17
   $MC_PART_COUNTER_MCODE[2] = 77
   $AC_REQUIRED_PARTS is active
   display alarm with
   $AC_REQUIRED_PARTS ≡ $AC_SPECIAL_PARTS

   $AC_TOTAL_PARTS is active;
   counter is increased by 1 with each M02.
   $MC_PART_COUNTER_MCODE[0] has no meaning.

   $AC_ACTUAL_PARTS is active;
   the counter is increased by 1 with each M17.

   $AC_SPECIAL_PARTS is active;
   the counter is increased by 1 with each M77.

Response at POWER ON

The machine data provided determine the response at POWER ON. A mode change or an NC reset does not affect the values of the counters.

Machine/ option data

The function “Workpiece counter” is a standard function (not an option).
The new channel machine data MD 27880: PART_COUNTER and MD 27882: PART_COUNTER_MCODE are described in detail in Section 4.4.

■

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

Number of channels and mode groups

For the number of channels and mode groups, please refer to the following documentation:

References: /BU/, “Order Document, Catalog NC 60”

Asynchronous subprograms (ASUB), interrupt routines

The function Interrupt routines/ASUBs is an option and is available in SW 4 and higher.

Rapid lift

The function Rapid lift with ASUBs is also available for SINUMERIK 810D CCU1 in SW 3.2 (08.99) and higher.

Hide several program levels

With SW 5 and higher, it is possible to skip up to 8 program levels.

4 Data Descriptions (MD, SD)

4.1 Machine data for operator panel

9421 MD number	**MA_AXES_SHOW_GEO_FIRST** Display geo axes of channel first		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 4/4	Unit: –	
Data type: BYTE	Applies from SW 2		
Significance:	If the value of machine data MD 9421: MA_AXES_SHOW_GEO_AXES = 1, the geo axes of the channel are displayed first. **Note:** (SW 6.1 and higher) This functionality is available with HMI Embedded and HMI Advanced.		
Application example(s)	–		
Related to	–		

© Siemens AG, 2003. All rights reserved

9422 MD number	**MA_PRESET_MODE** PRESET / basic offset in JOG. 0 no SK, 1 PRESET, 2 Preset actual value memory		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 2	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5, System frame in SW 6.1 and higher		
Significance:	This machine data defines the RESET / basic offset function in JOG mode. Basic configuration for the RESET function: 0: No soft key, 1: The old PRESET in the Machine operating area (default) 2: Preset actual value memory NCK **without** system frame: Can only be set when G500 is active in basic offset 1, otherwise error message. NCK **with** system frame: Can always be set in the system frame. The basic frame is not used with G500 in the system frame. 3: NCK with/without system frame: Can be set in the currently active frame. **Note:** (SW 6.1 and higher) The PRESET function in JOG mode is available with HMI Embedded and HMI Advanced.		
Related to	–		

9423 MD number	**MA_MAX_SKP_LEVEL** Define number of skippable program levels		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 8	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.1		
Significance:	This data defines the number of program levels which can be skipped in the parts program with “/” . The additional skip levels in the parts program are specified by“/0” to “/7”. **Note:** (SW 6.1 and higher) This functionality is available with HMI Embedded and HMI Advanced.		
Application example(s)	/ N005 : Block is skipped when MD9423=1 or higher. /3 N030 : Block is skipped when MD9423=4 or higher. /7 N070 : Block is skipped when MD9423=8 or higher.		
Special cases, errors, ...	This machine data MD 9423: MA_MAX_SKP_LEVEL must be entered even if only one mode group is configured.		

Note

All CTM operator panel machine data are described in the Description of Functions for turning in ManualTurn, and all CMM operator panel machine data in the Description of Functions for milling in ShopMill, see

References: /FBMA/, Description of Functions ManualTurn
/FBSP/, Description of Functions ShopMill

© Siemens AG, 2003. All rights reserved

4.2 General machine data

10010 MD number	**ASSIGN_CHAN_TO_MODE_GROUP[n]** Channel valid in mode group		
Default setting: 1/0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	The channel is assigned to a mode group with this MD. Input value 1 => Assigned to 1st mode group Input value 2 => Assigned to 2nd mode group etc. In SW 4 and higher, it is permissible to assign no mode group number to individual channels (channel gaps), in order to allow uniform configuration of similar machine designs. In this case, the number 0 is assigned to the channel instead of a mode group number greater than or equal to 1. The channel is not activated, but is treated as an active channel in counts. e.g. ASSIGN_CHAN_TO_MODE_GROUP[0] = 1 ASSIGN_CHAN_TO_MODE_GROUP[1] = 1 ASSIGN_CHAN_TO_MODE_GROUP[2] = 0 ; gap ASSIGN_CHAN_TO_MODE_GROUP[3] = 1 Note: By definition it is not permissible to set Channel 1 inactive. The 1st channel must always be active.		
Application example(s)	Select desired channel via MMC and set ASSIGN_CHAN_TO_MODE_GROUP = 1.		
Special cases, errors, ...	This MD also has to be entered if only one mode group exists.		

10280 MD number	**PROG_FUNCTION_MASK** Compare commands “>” and “<” compatible to SW 6.3		
Default setting: 0	Minimum input limit: 0x1	Maximum input limit: 0x1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Bit mask for parameterizing different parts program commands Bit0: 0 Processing of compare commands “>” and “<” as in SW 6.3 and lower Parts program data of REAL type are represented internally in IEEE format with 64 bits. An intrinsic characteristic of this type of representation is that decimal numbers are mapped imprecisely if the 52-bit mantissa is not sufficient to represent the number in powers of two. **Remedy:** All compare commands (==, < >, >=, <=, , > and <) are checked for a relative equality of 12^{-12}. Bit0: 1 **SW 6.4 and higher:** Setting bit 0 deactivates this operation for comparisons for Greater Than (>) and Smaller Than (<), thus making it compatible with SW 6.3.		
Related to	–		

© Siemens AG, 2003. All rights reserved

10617 MD number	**FRAME_SAVE_MASK** Response of frames to subroutine return jump with SAVE		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0x3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.1		
Significance:	This machine data defines which frames must be restored on return from a subroutine with a SAVE attribute. Bit 0: Settable frames G54 to G599 Bit 0 =0: If the same G code is active on return from the subroutine as in the subroutine call, then the active settable frame is retained. If this is not the case, the settable frame at the instant the subroutine was called is reactivated. Bit 0 =1: Compatibility with systems up to SW 5.3 On return from a subroutine the basic frame at the instant the subroutine was called is reactivated. Bit 1: Basic frames $P_CHBFR[ ] and $P_NCBFR[ ] Bit 1 = 0: The active basic frame is retained on return from a subroutine. Bit 1 = 1: Compatibility with systems up to SW 5.3 On a return from a subroutine, the basic frame at the instant the subroutine was called is reactivated.		
Related to	–		

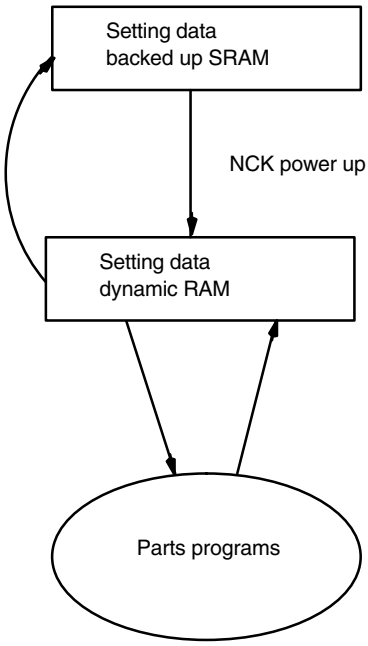
10702 MD number	**IGNORE_SINGLEBLOCK_MASK** Do not stop after single block		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.2		
Significance:	Program processing must not stop after single blocks in the case of the following even if SBL processing is selected: Bit 0 In no block of an internal ASUB: RETURN, REPOS Bit 1 In no block of a user ASUB (applies only in connection with MD 20117: IGNORE_SINGLEBLOCK_ASUP) Bit 2 In no intermediate blocks Bit 3 In block search group block Bit 4 In Init blocks Bit 5 In no block of a subroutine with DISPLOF Bit 6 In no block that cannot be reorganized Bit 7 In no block that cannot be repositioned Bit 8 In residual block without traversing information Bit 9 In preprocessing/main run synchronization block repeated due to an interruption Bit 10 At a “Tool selection block”. Bit 11 At an explicit U_N_Dimplicit GET block Bit 12 In single block type 2 (SBL2) with SBLON block (SW 6.4 and higher).		
Related to	MD 20117: IGNORE_SINGLEBLOCK_ASUP		

© Siemens AG, 2003. All rights reserved

10707 MD number	**PROG_TEST_MASK** Program test modes		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.1		
Significance:	Bit-coded mask for program test operation **Bit 0 Effect:** 0 Program test can be deselected only in the RESET state 1 Program test cannot be deselected in program state Bit 1 to bit 31 are not yet used. **Caution:** After program testing has been deactivated, a REPOS operation is initiated that is subject to the same restrictions as a SERUPRO approach process. These restrictions can be avoided by means of an ASUB.		

10708 MD number	**SERUPRO_MASK** Block change modes		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 15	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.1, in SW 6.4 and higher Bit 3		
Significance:	Bit-coded screen form for block search via Program Test (acronym SERUPRO). SERUPRO means **SE**arch **R**un by **PRO**gram test, i.e. the program test is run from the start of the program up to the search target. **Note:** Program Test will not move any axes. The block search SERUPRO is enabled via the PI service _N_FINDBL mode parameter == 5. Bit 0 = 0 A stop is carried out in the case of M0 during the search phase. Bit 0 = 1 No stop is carried out in the case of M0 during the search phase. Bit 1 = 0 Alarm 16942 will abort the search phase in the case of the parts program command START. Bit 1 = 1 Alarm 16942 is suppressed. **Caution:** A Starts program command in the search process will under certain circumstances start the other channel really! This channel can be switched to program test mode beforehand. **Extension in SW 6.2 and higher** Group SERUPRO activated Bit 2=1 Search for multiple starts with SERUPRO (“GroupSerupro”). Bit 2=1 Search for multiple starts with SERUPRO (“GroupSerupro”). **Extension in SW 6.4 and higher** SERUPRO end not simultaneous. Bit 3 = 0 forces all channels that have started SERUPRO to terminate SERUPRO at the same time unless they were aborted per reset, or the channel has reached M30 without finding the search target. This means: All channels that find the search target (alos self-acting SERUPRO) terminate SERUPRO at the same time. Bit 3 = 1 All channels that have started SERUPRO terminat SERUPRO at the same time. Exception: A reset has aborted the search run or the channel has reached M30 without finding the search target. Behavior **up to SW 6.4**. **Caution:** Replacement of axes between these channel is not permiited! In thsi state, axis replacement can produce unexpected target positions if a channel has already been shut down while another channel is still in simulation mode. Bit 4 to bit 31 are not yet used.		

© Siemens AG, 2003. All rights reserved

10710 MD number	**PROG_SD_RESET_SAVE_TAB[n] (setting data to be updated).** Setting data to be updated Index: 0...9 The values of the SD listed in this table are stored in nonvolatile memory, i.e. remain valid after POWER ON. The setting data whose MMC-numbers were entered in the backup list are written after the parts program into the (buffered) active file system on Reset.
	Programmable setting data are: 42000 $SC_THREAD_START_ANGLE SF 42400 $ SC_PUNCH_DWELLTIME (PDELAYON) 42800 $SC_SPIND_ASSIGN_TAB SETMS 43210 $SA_SPIND_MIN_VELO_G25 G25 43220 $SA_SPIND_MAX_VELO_G26 G26 43230 $SA_SPIND_MAX_VELO_LIMS LIMS 43300 $SA_ASSIGN_FEED_PER_REV_SOURCE (FPRAON) 43420 $SA_WORKAREA_LIMIT_PLUS G26 43430 $SA_WORKAREA_LIMIT_MINUS G25 43510 $SA_FIXED_STOP_TORQUE FXST 43520 $SA_FIXED_STOP_WINDOW FXSW 43700 $SA_OSCILL_REVERSE_POS1 OSP1 43710 $SA_OSCILL_REVERSE_POS2 OSP2 43720 $SA_OSCILL_DWELL_TIME1 OST1 43730 $SA_OSCILL_DWELL_TIME2 OST2 43740 $SA_OSCILL_VELO FA 43750 $SA_OSCILL_NUM_SPARK_CYCLES OSNSC 43760 $SA_OSCILL_END_POS OSE 43770 $SA_OSCILL_CTRL_MASK OSCTRL 43780 $SA_OSCILL_IS_ACTIVE OS
Default setting: 0	Minimum input limit: 0 / Maximum input limit: ***
Changes effective after POWER ON	Protection level: 2 / Unit: –
Data type: DWORD	Applies from SW 1.1
Fig.	On RESET, M02, M30 or M17 if PROG_SD_RESET_SAVE_TAB is supplied accordingly 
Application example(s)	The value of SD 43420: WORKAREA_LIMIT_PLUS (working area limitation plus) and SD43430: WORKAREA_LIMIT_MINUS (working areas limitation minus) is to be stored in the battery-backed RAM after every RESET, M02, M30 or M17. ⇒ PROG_SD_RESET_SAVE_TAB[0] = 43420 ⇒ PROG_SD_RESET_SAVE_TAB[1] = 43430

© Siemens AG, 2003. All rights reserved

10715 MD number	**M_NO_FCT_CYCLE** M function to be replaced by subroutine		
Default setting: –1	Minimum input limit: 0	Maximum input limit: 10	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	M number via which a subroutine is called. A maximum of 10 M functions can be configured. Only one M function replacement can be operative per block. The subroutine name is stored in MD 10716: M_NO_FCT_CYCLE_NAME[n]. If the M function defined in MD 10715: M_NO_FCT_CYCLE[n] is included in a parts program block, the subroutine defined in MD 10716: M_NO_FCT_CYCLE_NAME[n] is started at the end of the block. If the M function is programmed again in the subroutine, it is not replaced again by a subroutine call. MD 10715: M_NO_FCT_CYCLE[ ] works in both Siemens mode G290 and in external language mode G291. A subroutine call must not be superimposed on M functions with predetermined meaning. In the case of a conflict, alarm 4150 is generated. – M0 to M5, – M17, M30, – M19, – M40 to M45, – M function fro changing spindle/axis mode acc. to MD 20094: SPIND_RIGID_TAPPING_M_NR (default: M70), – M function for nibbling/punching acc. to configuration via MD 26008: NIBBLE_PUNCH_CODE if activated via MD 26012: PUNCHNIB_ACTIVATION. – For applied external language (MD 18800: MM_EXTERN_LANGUAGE) M96 and M99 additionally. Exception: The M function parameterized in MD 22560: TOOL_CHANGE_M_CODE for tool change. Subroutines configured in MD 10716: M_NO_FCT_CYCLE_NAME[n] and MD 10717: T_NO_FCT_CYCLE_NAME must not be active simultaneously in the same block (parts program line), i.e. no more than one M/T function replacement may be operative per block. Neither an M98 nor a modal subroutine call may be programmed in the block with the M function replacement. Neither is a subroutine return jump nor a parts program end allowed. In the case of a conflict, alarm 14016 is generated.		
Related to	MD 10716: M_NO_FCT_CYCLE_NAME, MD 10718: M_NO_FCT_CYCLE_PAR, MD 10717: T_NO_FCT_CYCLE_NAME		

10716 MD number	**M_NO_FCT_CYCLE_NAME** Subroutine name for M function replacement		
Default setting: " "	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: STRING	Applies from SW 5.2		
Significance:	Cycle name for call via M function defined with MD 10716: M_NO_FCT_CYCLE_NAME[n].		
Related to	MD 10715: M_NO_FCT_CYCLE, MD 10718: M_NO_FCT_CYCLE_PAR, MD 10717: T_NO_FCT_CYCLE_NAME		

© Siemens AG, 2003. All rights reserved

10717 MD number	**T_NO_FCT_CYCLE_NAME** Subroutine name for T function replacement		
Default setting: " "	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: STRING	Applies from SW 5.2		
Significance:	Cycle name for tool change routine for calls via T function. If a T function is programmed in a parts program block, the subroutine defined in MD 10717: T_NO_FCT_CYCLE_NAME[n] is called at the end of the block. The programmed T number can be scanned in the cycle via system variable $C_T or $C_T_PROG as a decimal value and via $C_TS or $C_TS_PROG as a string (with tool management TMMO only). MD 10717: T_NO_FCT_CYCLE_NAME is active in both Siemens mode G290 and in external language mode G292. MD 10716: M_NO_FCT_CYCLE_NAME and MD 10717: T_NO_FCT_CYCLE_NAME must not be active simultaneously in the same block (parts program line), i.e. no more than one M/T function replacement may be operative per block. Neither an M98 nor a modal subroutine call may be programmed in the block with the T function replacement. Neither is a subroutine return jump nor a parts program end allowed. In the case of a conflict, alarm 14016 is generated.		
Related to	MD 10715: M_NO_FCT_CYCLE MD 10716: M_NO_FCT_CYCLE_NAME		

10718 MD number	**M_NO_FCT_CYCLE_PAR** M function replacement with parameters		
Default setting: –1	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	If an M function replacement has been configured with MD 10715: M_NO_FCT_CYCLE[n] / MD 10716: M_NO_FCT_CYCLE_NAME[n], it is possible to specify a parameter transfer per system variable (analogous to T function replacement) using MD 10718: M_NO_FCT_CYCLE_PAR. The parameters stored in the system variables always refer to the parts program line in which the M function to be replaced is programmed. The following system variables are available: $C_ME : Address extension of substituted M function $C_T_PROG : TRUE if address T has been programmed $C_T : Value of address T (integer) $C_TE : Address extension of address T $C_TS_PROG : TRUE if address TS has been programmed $C_TS : Value of address TS (string, with tool management only) $C_D_PROG : TRUE if address D has been programmed $C_D : Value of address D $C_DL_PROG : TRUE if address DL has been programmed $C_DL : Value of address DL		
Related to	MD 10715: M_NO_FCT_CYCLE MD 10716: M_NO_FCT_CYCLE_NAME		

© Siemens AG, 2003. All rights reserved

10719 MD number	**T_NO_FCT_CYCLE_MODE** Parameterization of T function replacement		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	This machine data is set to define whether D or DL is passed as a parameter to the T replacement cycle (default), or must be executed before the T replacement cycle is called, when D or DL and T are programmed in the same block. Value 0: As in earlier versions, the D or DL number is passed to the cycle (default setting) Value 1: The D or DL number is calculated directly in the block This function is active only if the tool change has been configured with M function (MD 22550: TOOL_CHANGE_MODE = 1); the D or DL values are otherwise always transferred.		
Related to	MD 10717: T_NO_FCT_CYCLE_NAME		

© Siemens AG, 2003. All rights reserved

11450 MD number	**SEARCH_RUN_MODE** Control response and output of spindle auxiliary functions after block search		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0x1F	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.3, bit 3, in SW 6.1 and higher extended		
Significance:	SW 4.3 and higher: The machine data defines the control behavior after completion of the block search. SW 5.3 and higher: The extension with regard to the data type from BYTE to DWORD The extension with bit 2 defines whether the spindle auxiliary functions are to be output after a block search. SW 6.1 and higher: Automatic start of ASUB after block search with **bit 1 = 1** and cascaded block search with **bit 3 = 0**. The following bits can be used to influence the behavior after block search during the action blocks: Bit 0 = 0 Standard response (SW 4.3 and higher) NC Stop after output of the last action block; VDI signal “Last action block active” is set; Alarm 10208 is output. Bit 0 = 1 NC Stop after the last action block has been output; VDI signal “Last action block active” is set; the alarm 10208 is only output if the PLC requests this by setting the VDI signal “PLC action completed”. **Application:** PLC starts an ASUB after block search. **Note:** To continue the program, NC Start is required which should only be displayed after the end of the ASUB. Bit 1 = 1 Automatic start of /_N_CMA_DIR/_N_PROG_EVENT_SPF as an ASUB. Bit 2=0 The spindle auxiliary functions (M3, M4, M5, M19, M70 and S) are output as previously in the action blocks. Bit 2=1 The output of the spindle auxiliary functions in the action blocks is suppressed. The values programmed for the spindle and accumulated during block search can be output later (e.g. in an ASUB). To this aim, the program data are stored in the system variables $P_SEARCH_S, $P_SEARCH_SDIR, $P_SEARCH_SGEAR, $P_SEARCH_SPOS, $P_SEARCH_SPOSMODE. Expansions of functionality to bit 3 in SW 6.1 and higher Bit 3=0 Cascaded block search enabled. (multiple search target specification possible). Bit 3=1 Cascaded block search is disabled Bits 4 to 31 are not yet in use.		
Application	PLC starts an ASUB after block search. The note for the operator, that an NC Start is necessary to continue the program, should not be displayed until after the ASUB end.		
References	/FB/, S1, “Spindle auxiliary functions”, no output of spindle auxiliary functions after block search		

© Siemens AG, 2003. All rights reserved

11470 MD number	**REPOS_MODE_MASK** Repositioning properties		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3, bits 6 and 7 in SW 6.4 and higher		
Significance:	The bits in this machine data can be used to control the response of the NC during repositioning. **Bit** 1 I 0 **Effect:** 0 I 0 Normal reaction to mode group signals in all group channels (as for SW 3) All channels switched over to a program operation mode in response to interrupt. The dwell time is repeated completely in the Reset block. 0 I 1 The hold time is resumed at the point of interruption in the repositioning reset block. 1 I 0 Reserved 1 I 1 Reserved 2 I 0 The VDI signals have no effect on repositioning (same response as before) 2 I 1 Prevent repositioning of individual axes via VDI signals. 3 I 0 Repositioning of positioning axes not in the approach block during SERUPRO. 3 I 1 Positioning axes are repositioned on every REPOS in the approach block during block search by program test (SERUPRO). 4 I 0 Repositioning of positioning axes not on every REPOS in the approach block 4 I 1 Positioning axes are repositioned on every REPOS in the approach block during block search by program test (SERUPRO). 5 I 0 Modified feedrates and spindle speeds are not valid until the next block after residual block. 5 I 1 Modified feedrates and spindle speeds are valid immediately in the residual block. 6 I 0 Neutral axes and positioned spindles are not repositioned after SERUPRO in the approach block. 6 I 1 Neutral axes and positioned spindles are not repositioned after SERUPRO in the approach block as command axes. 7 I 0 The behavior of IS “REPOSDELAY” (DB31, .. DBX10.) is not changed. 7 I 1 The behavior of IS “REPOSDELAY” (DB31, .. DBX10.) is read if REPOSA is interpreted. Axes that are neither geometry or orientation axes are then excluded by REPOS and **not** moved.		

11600 MD number	**BAG_MASK** Defines mode response in relation to ASUBs		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0x3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	The machine data described the effect of the VDI signals on the channels of a mode group with reference to the ASUBs/interrupt routines. **Bit** 1 I 0 **Effect:** 0 I 0 Normal reaction to mode group signals in all group channels (as for SW 3) All channels switched over to a program operation mode in response to interrupt. 0 I 1 No reaction of other mode group VDI signals on a channel in which an interrupt routine is being executed. (mode group RESET, mode group STOP single type A and B, mode selection). (available soon) 1 I 0 Reserved 1 I 1 An internal operating mode switchover takes place only in those channels which have received an interrupt request. 3 I 0 No reaction of mode group Vdi signals to a channel is which an ASUB is running (mode group Reset, mode group Stop, single type A and B, mode selection). 3 I 1 An internal operating mode switchover takes place only in the channel which has received an interrupt request. (only if bit 0 is set).		

© Siemens AG, 2003. All rights reserved

11602 MD number	**ASUP_START_MASK** Ignore stop conditions for ASUB		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0x7	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	If an ASUB is started and its priority is higher than specified in ASUP_START_PRIO_LEVEL, subsequent stop reasons are ignored. This means that the ASUB starts independently although the NCK is stopped. Stop reasons: Bit 0 : The active program has been halted with the stop key, M01 or single-block. If NCK is in Reset state a ASUB is started immediately. (No ASUB can be started in RESET/JOG without this bit.) Bit 1 : ASUB may start even if all axes have not been referenced Bit 2: The start is allowed even if the read-in disable is active, i.e. the blocks of the ASUB program are loaded and executed immediately. The MD IGNORE_INHIBIT_ASUP is therefore ineffective. The MD response is equivalent to the MD assignment IGNORE_INHIBIT_ASUP = FFFFFFFF. If not bit is set the ASUB is selected internally but not processed until the read-in disable is canceled. Further bits: Reserved.		
Related to	ASUP_START_PRIO_LEVEL, IGNORE_INHIBIT_ASUP		
References			

11604 MD number	**ASUP_START_PRIO_LEVEL** Priorities for 'ASUP_START_MASK effective'		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 128	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	This machine data defines the ASUB priority from which machine data ASUP_START_MASK is to be applied. MD ASUP_START_MASK is applied from the level specified here up to maximum ASUB priority level 1.		
Related to	ASUP_START_MASK		
References			

11610 MD number	**ASUP_EDITABLE** Activation of a userspecific program (ASUB) for RET and REPOS		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.2		
Significance:	This MD determines whether user-specific routine: _N_ASUP_SPF stored in directory _N_CUS_DIR must be used to process RET and REPOS instead of routines provided by the system. Value: Significance: 0 Routine _N_ASUP_SPF is not activated for either RET or REPOS. 1 User-specific routine _N_ASUP_SPF is executed for RET, REPOS activates the Routine supplied by the system 2 User-specific routine _N_ASUP_SPF is executed for REPOS, REPOS activates the Routine supplied by the system 3 User-specific routine _N_ASUP_SPF is executed for both RET and REPOS Routine _N_ASUP_SPF		
Related to	MD 11612: ASUP_EDIT_PROTECTION_LEVEL		
References	/IAD/ Installation and StartUp Guide 840D		

© Siemens AG, 2003. All rights reserved

11612 MD number	**ASUP_EDIT_PROTECTION_LEVEL** Protection level of user-specific ASUB for RET and/or REPOS		
Default setting: 2	Minimum input limit: 0	Maximum input limit: 7	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.2		
Significance:	Protection level of user-specific ASUP for RET and/or REPOS The data is effective only if MD 11610: ASUP_EDITABLE is set to a value other than 0. This machine data defines the protection level of the program _N_ASU_CUS.		
MD irrelevant for ...	ASUP_EDITABLE set to 0		
Related to	ASUP_EDITABLE		

17200 MD number	**GMMC_INFO_NO_UNIT** Global HMI info (without physical unit)		
Default setting: Value MD no.	Minimum input limit: Value MD no.	Maximum input limit: Value MD no.	
Changes effective after POWER ON	Protection level: 0/7	Unit: –	
Data type: DOUBLE	Applies from SW 6.4		
Significance:	The global display machine data: MD 9004: DISPLAY_RESOLUTION MD 9011: DISPLAY_RESOLUTION_INCH MD 9010: SPIND_DISPLAY_RESOLUTION MD 9424: MA_COORDINATE_SYSTEM are stored by the HMI in NCK machine data MD 17200: GMMC_INFO_NO_UNIT[0] to MD 17200: GMMC_INFO_NO_UNIT[3], allowing them to be accessed from the NCK. These display machine data are updated by the HMI and evaluated by the NCK on parts program start.		

17201 MD number	**GMMC_INFO_NO_UNIT_STATUS** Global HMI status info (without physical unit)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 0/7	Unit: –	
Data type: BYTE	Applies from SW 6.4		
Significance:	The meaning of the machine data is as follows: Value 0: Entry unassigned. Value 1: Entry assigned.		

4.3 Machine data for processing programs from external (SW 4 and later)

18360 MD number	**MM_EXT_PROG_BUFFER_SIZE** Reload buffer (DRAM) size for a program level with “Execution from external source”		
Default setting: 30	Minimum input limit: 30	Maximum input limit: 1000000	
Changes effective after POWER ON	Protection level: 2/7	Unit: Function: Processing HD	
Data type: DWORD	Applies from SW 4.2 (HMI Advanced)		
Significance:	Size of FIFO buffer for one program level (main program or subroutine) processed in the “Execute from external source” mode (reload mode) (DRAM memory in KB). Memory is reserved for MD 18360: MM_EXT_PROG_BUFFER_SIZE program levels. If there is insufficient DRAM available, alarm 4077 is generated.		
Related to	MD 18362: MM_EXT_PROG_NUM Number of external program levels (DRAM)		
References	/PGA/ Programming Guide, Advanced, Chapter 2		

© Siemens AG, 2003. All rights reserved

18362 MD number	**MM_NUM_EXT_PROG** Number of external program levels (DRAM)		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 13	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 4.2		
Significance:	Number of program levels in the NCK that can operate simultaneously in “Process from external”. DRAM is reserved in MD 18360: MM_EXT_PROG_BUFFER_SIZE for each program level. If there is insufficient DRAM available, alarm 4077 is generated.		
Related to	MD 18360: MM_EXT_PROG_BUFFER_SIZE FIFO buffer size for one program level		

© Siemens AG, 2003. All rights reserved

4.4 Channel-specific machine data

20000 MD number	**CHAN_NAME** Channel name		
Default setting: CHAN1/ CHAN2	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: STRING	Applies from SW 1.1		
Significance:	The channel name can be defined in this MD. The channel name is only used for the display on the MMC.		

20108 MD number	**PROG_EVENT_MASK** Eventdriven program calls		
Default setting: 0	Minimum input limit: 0x0	Maximum input limit: 0xF	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.1		
Significance:	Parameterization of events in response to which user program /_N_CMA_DIR/_N_PROG_EVENT_SPF is called implicitly: Bit = 0: ProgEvent deactivated Bit = 1: ProgEvent activated Bit 0: Parts program start (NC Start initiates ProgEvent) Bit 1: Parts program end (Program end initiates ProgEvent) Bit 2: Operator panel reset (RESET initiates ProgEvent) Bit 3 : Power-up (NC boot initiates ProgEvent)		

20109 MD number	**PROG_EVENT_MASK_PROPERTIES** ProgEvents properties		
Default setting: 0	Minimum input limit: 0x0	Maximum input limit: 0x1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	Parameterization of other properties of eventdriven program calls (abbreviated to ProgEvent), i.e. MD 2018: PROG_EVENT_MASK is parameterized further. Bit 0 = 0: An ASUB started from the RESET channel state is followed by a ProgEvent. Bit 0 = 1: An ASUB started from the RESET channel state is **not followed** by a ProgEvent.		

© Siemens AG, 2003. All rights reserved

20114 MD number	**MODESWITCH_MASK** Setting for REPOS		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 3.2		
Significance:	After program interruption in MDA mode (e.g. in order to carry out a measurement on the workpiece and to correct the tool wear values or after tool break) the tool can be manually withdrawn from the contour by changing into “JOG” mode. In this case, the control stores the coordinates of the position of the interruption and indicates the path differences traversed by the axes in JOG mode as REPOS offset. When MDA mode is selected again, the axis is repositioned on the contour. This response can be canceled by means of this machine data. Bit 0 (LSB) = 0: When MDA is deselected (JOG, JOGREF, JOGREPOS, MDAREF and MDAREPOS) in the stopped state, the REPOS system ASUB is selected. Bit 0 (LSB) = 1: When MDA is deselected (JOG, JOGREF, JOGREPOS, MDAREF and MDAREPOS) in the stopped state, the REPOS system ASUB is not selected. Bit 1 (LSB) = 0: If the NCK stops program execution at a parts program block in which repositioning is not possible, an attempt to switch to a manual mode generates alarm 16916. Bit 1 (LSB) = 1: If the NCK stops program execution at a parts program block in which repositioning is not possible, an attempt to switch to a manual mode does not generate an alarm.		
Related to			

20160 MD number	**CUBIC_SPLINE_BLOCKS** Number of blocks for C spline		
Default setting: 8	Minimum input limit: 4	Maximum input limit: 9	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Number of motion blocks across which a spline section is calculated with the cubic spline (C spline) function. The larger the value, the closer the generated contour is to the ideal mathematical cubic spline. However, the higher the value, the longer the block search calculation time.		
References	/PA/, “Programming Guide Fundamentals”		

20170 MD number	**COMPRESS_BLOCK_PATH_LIMIT** Maximum traversing length of NC block for compression		
Default setting: 1.0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The machine data defines the maximum traversing length of a block that can be compressed. Longer blocks interrupt the compression and are traversed in the normal way.		
Related to	MD 33100: COMPRESS_POS_TOL (maximum tolerance with compression)		
References	/PA/, “Programming Guide Fundamentals”		

© Siemens AG, 2003. All rights reserved

20610 MD number	**ADD_MOVE_ACCEL_RESERVE** Acceleration reserve for overlaid movements		
Default setting: 0.2	Minimum input limit: 0	Maximum input limit: 0.9	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The value of this machine data is multiplied by the setting in MD 32300: MAX_AX_ACCEL (max. axis acceleration). The result is an acceleration value. The max. axis acceleration is reduced by this acceleration value when the function “Fast retraction from the contour” is active. This is done so that there is sufficient reserve for the velocity control for the overlaid movement during fast retraction.		
References	/PA/, “Programming Guide Fundamentals”		

21000 MD number	**CIRCLE_ERROR_CONST** Circle end point monitoring constant		
Default setting: 0.01	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	This machine data defines the permissible absolute circle difference. In circle programming, the radii from the programmed center point to the starting point and to the end point are usually not equal (the circle is “overdefined”). The maximum permissible difference between these two radii which is accepted without an alarm being triggered is defined by the larger value in the following data: – CIRCLE_ERROR_CONST – Start radius multiplied by MD 21010: CIRCLE_ERROR_FACTOR i.e. the tolerance is a fixed value (CIRCLE_ERROR_CONST) for small circles and proportional to the start radius for large circles.		
Application example(s)	CIRCLE_ERROR_CONST = 0.01 $\hat{=}$ 10 µm CIRCLE_ERROR_FACTOR = 0.001 With these MDs and a radius of $\leq$ 10mm, the constant has an effect, with > 10mm, the proportional factor has an effect.		
Related to	MD 21010: CIRCLE_ERROR_FACTOR (circle end point monitoring factor)		

21010 MD number	**CIRCLE_ERROR_FACTOR** Circle end point monitoring factor		
Default setting: 0.001	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Factor for permissible radius difference. Defines the factor for large circles by which the starting and end radius can deviate from each other. (See also MD 21000: CIRCLE_ERROR_CONST circle end point monitoring constant)		

21200 MD number	**LIFTFAST_DIST** Traversing path for fast retraction from the contour		
Default setting: 0.1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The machine data determines the absolute length of the traverse movement for fast retraction. The direction of the traverse movement is defined in the parts program by the command ALF.		
References	/PA/, “Programming Guide Fundamentals”		

© Siemens AG, 2003. All rights reserved

21202 MD number	**LIFTFAST_WITH_MIRROR** Lift fast with mirror		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 5		
Significance:	1: The direction of retraction is also mirrored if an active mirror is applied to the contour. The mirror is only related to the direction component vertical to the tool direction. 0: The mirror is not applied to the retraction direction.		
References	V1, Velocities and Accelerations for Lift Fast		

21210 MD number	**SETINT_ASSIGN_FASTIN** NCK input bytes for interrupts		
Default setting: 1	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: HEX	
Data type: DWORD	Applies from SW 2 ext. in 5		
Significance:	Hardware assignment of high-speed input byte for NC program interrupt Bits 0 to 7: Number of input used Bits 16 to 23 Mask of signals which the channel is not to evaluate Bits 24 to 31 Mask of signals which are to be evaluated inverted. Bit set: Interrupt triggered by falling edge. Possible inputs: 1: Onboard inputs of the 840D (4 high-speed + 4 bits via VDI command) 2–5: External digital inputs (highspeed NCK I/O or VDI command) 1228–129: Comparator byte (resulting from high-speed analog inputs or VDI command		

22500 MD number	**GCODE_OUTPUT_TO_PLC** G functions to PLC, **(up to SW 3.1)**		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	0: No output of G codes to the PLC 1: Programmed G codes are output to the PLC. The G codes most recently programmed in the block are output. Output is limited to 8 G codes.		

22510 MD number	**GCODE_GROUPS_TO_PLC** G codes that are output to the NCK/PLC interface on block change/Reset, (SW 3.2 and higher)		
Default setting: 0000 0000, ...	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 3.2		
Significance:	Specification of G code that are output to the NCK/PLC interface on block change/Reset. The interface is updated after every block change and Reset.		
Special cases, errors, ...	It cannot be guaranteed that a PLC user program will have a block-synchronous connection between the active NC block and the applied G codes at any given time. Example: Continuous-path mode with very short blocks.		

© Siemens AG, 2003. All rights reserved

27800 MD number	TECHNOLOGY_MODE Technology in channel		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 255	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 4.2		
Significance:	This machine data allows a machining technology to be specified on a channel-dependent basis. MD=0 Milling MD=1 Turning MD=2 Grinding MD=3 Nibbling MD=4 ...(Add other technologies if required)		
Application example(s)	Among other functions, this information is provided for evaluation by the HMI, MMC, PLC and standard cycles.		

27860 MD number	PROCESSTIMER_MODE Activation of program runtime measurement		
Default setting: 0x0	Minimum input limit: 0x0	Maximum input limit: 0x07F	
Changes effective after Reset	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 5.2		
Significance:	The following runtime measurements are performed automatically: Bit 0 = 1 Measurement of total NC program runtime active ($AC_OPERATING_TIME) Bit 1 = 1 Measurement of current NC program runtime active ($AC_CYCLE_TIME) Bit 2 = 1 Measurement of tool operating time ($AC_CUTTING_TIME) Bit 3 Reserved Bit 4/5 only when bit 0 or 1 or 2 = 1 Bit 4=0 No measurement at active dry run feedrate Bit 4 = 1 Measurement at active dry run feedrate Bit 5 = 0 No measurement in program test mode Bit 5 = 1 Measurement in program test mode. Bit 6 only when bit 1 = 1 Bit 6=0 Delete $AC_CYCLE_TIME even with start via ASUB and PROG_EVENTs Bit 6 = 1 $AC_CYCLE_TIME is not deleted with start via ASUB and PROG_EVENTs parts program end. Bit 7 Reserved		

27880 MD number	PART_COUNTER Activation of the workpiece counters		
Default setting: 0x0	Minimum input limit: 0x0	Maximum input limit: 0x0FFF	
Changes effective after Reset	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	This MD can be used to configure the workpiece counters: Bit 0 = 1 $AC_REQUIRED_PARTS counter is active Bit 1 = 0 Alarm/VDI output on $AC_REQUIRED_PARTS = $AC_ACTUAL_PARTS Bit 1 = 1 Alarm/VDI output on $AC_REQUIRED_PARTS = $AC_SPECIAL_PARTS Bit 4 = 1 $AC_TOTAL_PARTS counter is active Bit 5 = 1 $AC_TOTAL_PARTS counter is incremented by 1 on M2/M30. Bit 6=1 $AC_TOTAL_PARTS is incremented by 1 on the M command from MD MC_PART_COUNTER_MCODE[0] incremented by value of 1. Bit 8 = 1 $AC_ACTUAL_PARTS counter is active Bit 9 = 0 $AC_ACTUAL_PARTS counter is incremented by 1 on M2/M30. Bit 9=1 $AC_ACTUAL_PARTS counter is incremented by 1 on the M command from MD MC_PART_COUNTER_MCODE[1] incremented by value of 1. Bit 12 = 1 Activate counter $AC_SPECIAL_PARTS Bit 13 = 0 $AC_SPECIAL_PARTS counter is incremented by 1 on M2/M30 Bit 13 = 1 $AC_SPECIAL_PARTS counter is incremented by 1 on the M command from MD MC_PART_COUNTER_MCODE[2] incremented by value of 1.		

© Siemens AG, 2003. All rights reserved

27882 MD number	**PART_COUNTER_MCODE**[ ] Workpiece counting via M command		
Default setting: 2	Minimum input limit: 0	Maximum input limit: 99	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE[3]	Applies from SW 3.2		
Significance:	The workpiece counters are incremented by 1 on output of the following M command at the VDI interface: $MC_PART_COUNTER_MCODE[0] for $AC_TOTAL_PARTS $MC_PART_COUNTER_MCODE[1] for $AC_ACTUAL_PARTS $MC_PART_COUNTER_MCODE[2] for $AC_SPECIAL_PARTS Note: Only relevant if counting via M command has been activated.		

4.4.1 Block search

20128 MD number	**COLLECT_TOOL_CHANGE** Collect tool changes during block search		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 1/1	Unit: –	
Data type: BOOLEAN	Applies from SW 4.3		
Significance:	This MD is only relevant when tool management is active. It determines whether the tool change M code defined by MD 22560: TOOL_CHANGE_M_CODE is accumulated during block search with calculation. 1: Tool change M code is accumulated 0: Tool change M code is not accumulated The tool determined in the block search is displayed and interpreted as the current tool; the T number output is unaffected. The tool offset data determined by the NCK are effective. There is no change in the magazine data, etc. The tool change M code is not accumulated without tool management if it is not assigned to an auxiliary function group.		
Related to	MD 22560: TOOL_CHANGE_M_CODE		

© Siemens AG, 2003. All rights reserved

22600 MD number	**SERUPRO_SPEED_MODE** Velocity with block search type 5		
Default setting: 1111 1111, ...	Minimum input limit: –	Maximum input limit: –	
Changes effective immediately	Protection level: 2/2	Unit: –	
Data type: DWORD	Applies from SW 6.1		
Significance:	This machine data specifies the block search mode: SERUPRO in detail. SERUPRO means **SE**arch **R**un by **PRO**gram Test, i.e. the axes are traversed in Program Test mode from the program start up to the search target. Note: Program Test will not move any axes. The block search SERUPRO is enabled via the PI service _N_FINDBL mode parameter == 5. **Bit** 1 I 0 **Effect:** 0 I 0 The axis traverses at the dry run feedrate set in MD 22601: SERUPRO_SPEED_FACTOR in program test mode. Dynamic limitations for axes are effective. 0 I 1 The axis traverses at the velocity corresponding to dry run feedrate in program test mode. Dynamic limitations for axes are effective. 1 I 0 The axis traverses at the programmed velocity in program test mode. Dynamic limitations for axes are effective. 1 I 1 Not assigned Bit 2 to bit 31 are not yet used. **Notice:** A Starts program command in the search process will under certain circumstances start the other channel really!		
Related to	MD 22601: SERUPRO_SPEED_FACTOR (velocity factor of search type 5) SD 42100: DRY_RUN_FEED (dry run feed)		

22601 MD number	**SERUPRO_SPEED_FACTOR** Velocity factor for block search type 5		
Default setting: 10.0	Minimum input limit: –	Maximum input limit: –	
Changes effective immediately	Protection level: 2/2	Unit: –	
Data type: DOUBLE	Applies from SW 6.1		
Significance:	SERUPRO means **SE**arch **RU**n by **PRO**gram Test, i.e. the axes are traversed in Program Test mode from the program start up to the search target. Note: Program Test will not move any axes. This machine data is only relevant if the first two bits of MD 22600: SERUPRO_SPEED_MODE are set to zero. The sign of the machine data has the following meaning: Positive: MD 22601: SERUPRO_SPEED_FACTOR specifies the factor with which the dry run feedrate is multiplied. Negative: Axis motions are not executed at all. Zero: Programmed end points are approached in an interpolation cycle. Dynamic limitations of axes are always ignored.		
Related to	MD 22600: SERUPRO_SPEED_MODE (velocity with block search type 5) SD 42100: DRY_RUN_FEED (dry run feed)		

© Siemens AG, 2003. All rights reserved

22621 MD number	**ENABLE_START_MODE_MASK_PRT** Enables MD 22620: START_MODE_MASK_PRT for SERUPRO search run		
Default setting: 0x0, 0x0, ...	Minimum input limit: 0	Maximum input limit: 0x1	
Changes effective after Reset	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	The machine data MD 22620: START_MODE_MASK_PRT is enabled via this machine data. In the initial setting of MD 22621: ENABLE_START_MODE_MASK_PRT if Bit 0 = =0 MD 22620: START_MODE_MASK_PRT is inactive. Bit 0–1 If “search run by program test” (SERUPRO) is started from RESET (PI service _N_FINDBL with mode parameter == 5), Set machine data MD 22620: START_MODE_MASK_PRT replaces Set machine data MD 20112: START_MODE_MASK. This allows you to set a different response to PLC start for search run start.		
Related to	MD 20112: START_MODE_MASK Initial function setting. MD 22620: START_MODE_MASK_PRT Initial setting on special NC Start after booting and on RESET.		

22622 MD number	**DISABLE_PLC_START** Allow parts program start via PLC		
Default setting: 0x0, 0x0, ...	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/2	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	This machine data is only activated is machine data MD 10708: SERUPRO_MASK, bit 2 = 1. Bit 0 = =0 A parts program can only be started via the PLC in this channel. Starting per parts program command START is interlocked. Bit 0 = =1 In this channel, a parts program can only be started per parts program command “START” from a differnet channel. Starting per PLC is interlocked.		
Related to	MD 10708: SERUPRO_MODE_MASK program test modes.		

© Siemens AG, 2003. All rights reserved

4.4.2 ASUB

20116 MD number	**IGNORE_INHIBIT_ASUP** Process ASUB in spite of readin disable		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.2		
Significance:	An interrupt channel is assigned to each bit of this mask. Bit 0 is equivalent to interrupt channel 1, bit 1 to interrupt channel 2, etc. Bit x is set: A user ASUB triggered via the interrupt channel (x+1) is processed completely, in spite of the active read-in disable. User ASUBs are linked to an interrupt channel with the parts program command SETINT or PI service _N_ASUP__. The interrupt channel is then activated via the PLC or the highspeed inputs and the ASUB is executed. If ASUP_START_MASK bit 2 = 1, IGNORE_INHIBIT_ASUP is ineffective, i.e. the NCK responds as if IGNORE_INHIBIT_ASUP = FFFFFFFF. Bit x is not set: The assigned ASUB is selected, but is not processed until the read-in disable is canceled.		
Related to	IGNORE_SINGLEBLOCK_ASUP, ASUP_START_MASK		

20117 MD number	**IGNORE_SINGLEBLOCK_ASUP** ASUB in spite of SBL processing		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.2		
Significance:	In spite of selected SBL processing mode, a user ASUB is processed completely for the interrupt channel with the set bit. Bit 0 is assigned to interrupt channel 1. Bit 1 is assigned to interrupt channel 2. If single block stop is suppressed in the user ASUB, then it cannot be activated with SBLON. SBLON can be activated again if the single block stop is suppressed in the system ASUB or user ASUB with MD 10702: IGNORE_SINGLEBLOCK_MASK bit 0 = 0 or bit 1 = 0. The MD is **not** effective for single block type 2		
Related to	MD20116 IGNORE_INHIBIT_ASUP MD10702 IGORE_SINGLEBLOCK_MASK		

© Siemens AG, 2003. All rights reserved

4.4.3 Reset and starting response

20110 MD number	**RESET_MODE_MASK** Definition of control system initial setting on Reset		
Default setting: 0x0, 0x0, ...	Minimum input limit: 0	Maximum input limit: 0x3FFFF	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 2.0 up to bit 17 in SW 6.4 and higher		
Significance:	Definition of control basic setting after powerup and RESET parts program end with respect to G-Codes (especially active plane and settable zero offset), tool length correction and transformation by setting the following bits: Bit 0: (LSB) Reset mode Bit 1: Suppress aux. funct. output on tool selection Bit 2: Select reset response after POWER ON (e.g. tool offset) Bit 3: Select reset response after end of test mode with reference to active tool-offsets. This bit is relevant only if bits 0 and 6 (0x41) are set. It defines the reference for current setting for the active tool length-offset‘; – the program active on end of Program Test – the program active before executing Program Test Bit 4: Up to SW 4.4 reset response G code “current plane”; in SW 5 and higher reserved, setting via MD 20152: GCODE_RESET_MODE Bit 5: Up to SW 4.4 reset response G code “settable zero offset”; in SW 5 and higher reserved, setting via MD 20152: GCODE_RESET_MODE Bit 6: Reset response “active tool length compensation” Bit 7: Reset response “active kinematic transformation” Bit 8: Reset response “coupled-motion axes” Bit 9: Reset response “tangential correction” Bit 10: Reset response “synchronous spindle” Bit 11: Reset response “revolutional feedrate” Bit 12: Reset response “Geoaxis replacement” Bit 13: Reset response “Guide value coupling” Bit 14: Reset response “Basic frame” Bit 15: Reset response “Electronic gear” Bit 16: Reset response “Master spindle” (SW 6.4 and higher) Bit 17: Reset response “Master tool holder” (SW 6.4 and higher) Bits 4 to 11, 16 and 17 are evaluated only when bit 0 = 1. The remaining bits are not assigned. **Meaning of each bit:** Bit **0** = 0: Corresponds to response of SW 1 Initial setting **after power-up:** – G codes acc. to MD 20150: GCODE_RESET_VALUES; GCODE_RESET_VALUES; – Tool length compensation not active – Transformation not active – No coupled motion groupings active – No tangential followup active – Axial revolutional feedrate off – Axial revolutional feedrate off – Geometry axis assignment according to MD 20050 – Guide value coupling not active – Path revolutional feedrate with master spindle (default) Continued on next page.		

© Siemens AG, 2003. All rights reserved

20110 MD number	**RESET_MODE_MASK** Definition of control system initial setting on Reset
Significance: ff	Initial setting **on reset or end of program:** the current settings are retained; with the next parts program start, the following reset state is in effect: – G codes acc. to MD 20150: GCODE_RESET_VALUES; GCODE_RESET_VALUES; – Tool length compensation not active – Transformation not active – No coupled motion groupings active – No tangential follow-up active – Axial revolutional feedrate off – Axial revolutional feedrate off – Geometry axis assignment according to MD 20050 – Guide value coupling not active Bit **0** = 1: Initial setting **after power-up:** – G codes acc. to MD 20150: GCODE_RESET_VALUES; – Tool length compensation active acc. to MD 20120: TOOL_RESET_VALUE and MD 20130: CUTTING_EDGE_RESET_VALUE If bit 2 =1 and bit 6 = 1, then TLC is active at POWER ON. – Transformation active acc. to MD 20140: TRAFO_RESET_VALUE – Transformation not active – No coupled-motion groupings active – No tangential follow-up active – Axial revolutional feedrate off – Axial revolutional feedrate off – Geometry axis assignment according to MD 20050 – Guide value coupling not active Bit **0** = 1: Initial setting **on reset or end of program:** G codes: Setting via MD 20152 Depending on RESET_MODE_MASK, bit 4 to 14 the current settings are retained or the basic settings stored in the MDs are set for the following: – Active plane – Settable zero offset – Tool length compensation – Transformation – etc. Depending on bits 8 and 9 the current settings of coupled-motion axes and tangentially axes are either deactivated or retained. Synchronous spindle link configured: The link is deselected depending on the setting in MD 21330: COUPLE_RESET_MODE_1. Synchronous spindle link not configured: Depending on the setting of bit 10 the link is deactivated or retained Depending on the setting of bit 14 the basic frame is retained or deactivated. COUPLE_RESET_MODE_1. Bit 1 = 0: Aux. funct. output (D, T, M) to PLC on tool selection according to MD 20120: TOOL_RESET_VALUE, MD 20130: CUTTING_EDGE_RESET_VALUE, MD 20121: TOOL_PRESEL_RESET_VALUE and MD 22550: TOOL_CHANGE_MODE. When the tool management function is active, bit 1 has no meaning. Bit 1 = 1: Suppress aux. funct. output to PLC on tool selection. When the tool management function is active, bit 1 has no meaning. Bit 2 = 0: No tool offset after POWER ON active (no effect when tool management active) Bit 2 = 1: If bit 0 and bit 6 = 1, the tool offset of the last active tool is active (no effect when tool management active) Bit 3 = 0: End of test mode: 'Current setting for active tool length compensation is retained' (bits 0 and 6 enabled) refers to the program which was active before activation (!) of test mode. Bit 3 = 1: End of test mode: 'Current setting for active tool length compensation is retained' (bits 0 and 6 enabled) refers to the program which was active at the end (!) of test mode. Continued on next page.

© Siemens AG, 2003. All rights reserved

20110 MD number	**RESET_MODE_MASK** Definition of control system initial setting on Reset
Significance ff.	Bit 4 = 0: SW 4.4 and lower: Initial setting for G code “current plane” after Reset/part program end MD 20150: GCODE_RESET_VALUES; in SW 5 and higher reserved, setting via MD 20152: GCODE_RESET_MODE Bit 4 = 1: SW 4.4 and lower: The current setting for G code “current plane” is retained beyond reset/end of parts program; in SW 5 and higher reserved, setting via MD 20152: GCODE_RESET_MODE Bit 5 = 0: SW 4.4 and lower: Initial setting for G code “settable zero offset” after reset/end of parts program according to MD 20150: GCODE_RESET_VALUES; in SW 5 and higher reserved, setting via MD 20152: GCODE_RESET_MODE Bit 5 = 1: SW 4.4 and lower: The current setting for G code “current plane” is retained beyond reset/end of parts program; in SW 5 and higher reserved, setting via MD 20152: GCODE_RESET_MODE Bit 6 = 0: Initial setting for active tool length compensation according to parts program end acc. to MD 20120: TOOL_RESET_VALUE and MD 20130: CUTTING_EDGE_RESET_VALUE; If MD 22550: TOOL_CHANGE_MODE=1, the tool specified in MD 20121: TOOL_PRESEL_RESET_VALUE is preselected. When the tool management function is active, MD 20120 is not used but MD 20122: TOOL_RESET_NAME; Bit 6 = 1: Current setting for active tool length compensation is retained beyond reset/end of parts program. Bit 7 = 0: Initial setting for active transformation after reset/parts program end according to MD 20140: TRAFO_RESET_VALUE; Bit 7 = 1: Current setting for active transformation is retained beyond reset/parts program end and parts program start; Bit 8 = 0: Coupled-motion groupings are deactivated after reset/end of parts program Bit 8 = 1: Coupled-motion groupings remain active after reset/end of parts program. Bit 9 = 0: Tangential correction is deactivated on reset/end of parts program. Bit 9 = 1: Tangential correction is retained after reset/end of parts program. Bit 10 = 0: Non-configured synchronous spindle link is deactivated on reset/parts program end. Bit 10 = 1: Non-configured synchronous spindle link remains active on reset/parts program end. Bit 11 = 0: With reset/end of parts program, the setting data MD 43300: ASSIGN_FEED_PER_REV_SOURCE is set to 0 for all inactive axes/spindles, i.e. traversing is no longer at revolutional feedrate, and the setting for path and synchronous axes is reset to the master spindle (default setting). Bit 11 = 1: Current setting for revolutional feedrate is retained after reset/end of part - program. With start of parts program, the setting data MD 43300: ASSIGN_FEED_PER_REV_SOURCE is set to 0, i.e. traversing is no longer at revolutional feedrate, and the setting for path and synchronous axes is reset to the master spindle (default setting). Bit 12 = 0: When machine data $MC_GEOAX_CHANGE_RESET is enabled, a modified geometry axis assignment is cleared on Reset or parts program end. The default setting in the machine data for geometry axis assignment becomes active. Bit 12 = 1: A modified geometry axis assignment remains active after RESET/end of parts program. Bit 13 = 0: Master/slave couplings are separated on RESET/end of parts program. Bit 13 = 1: Master/slave couplings remain active after RESET/end of parts program. Bit 14 = 0: The basic frame is deselected. Bit 14 = 1: The current setting of the basic frame is retained. Bit 15 = 0: Active electronic gears are retained on RESET/end of parts program Bit 15 = 1: Active electronic gears are deactivated on RESET/end of parts program Bit 16 = 0: **SW 6.4 and higher** Initial setting for the master spindle acc. to MD 20090: SPIND_DEF_MASTER_SPIND Bit 16 = 1: The current setting of the master spindle (SETMS) is retained. If MD 20124: TOOL_MANAGEMENT_TOOLHOLDER=0, this bit also has an effect on the response of bit 6. Continued on next page.

© Siemens AG, 2003. All rights reserved

20110 MD number	**RESET_MODE_MASK** Definition of control system initial setting on Reset
	Bit 17 = 0: Initial setting for the master tool holder acc. to MD MD 20124: TOOL_MANAGEMENT_TOOLHOLDER Bit 17 = 1: The current setting of the master tool holder (SETMTH) is retained. (Bit 17 is only significant if tool management is active and MD 20124: TOOL_MANAGEMENT_TOOLHOLDER > 0). Otherwise the mas- ter spindle takes its setting from bit 16 (tool management active). This bit also has an effect on the response of bit 6.
Related to	MD 20120: TOOL_RESET_VALUE (tool whose assigned length compensation is selected during power-up (Reset/parts program end)) MD 20130: CUTTING_EDGE_RESET_VALUE (cutting edge whose assigned length is selected during power-up (Reset/end of parts program) MD 20150: GCODE_RESET_VALUES (initial setting of G groups [G group no.]). MD 20152: in SW 5 and higher, extension for MD 20110 with reference to G codes MD 20140: TRAFO_RESET_VALUE (transformation data set that is selected during powerup (Reset/end of parts program)). MD 20112: START_MODE_MASK (definition of control system initial setting after POWER ON and reset. MD 20118: GEOAX_CHANGE_RESET (allow automatic geometry change) MD 43300: ASSIGN_FEED_PER_REV_SOURCE (revolutional feedrate for positioning axes/spindles) MD 20050: AXCONF_GEOAX_ASSIGN_TAB (assignment between geometry axis and channel axis)

!

Important

If the tool management function is used, then bit 14 in MD 20310 TOOL_MANAGEMENT_MASK must be set.

20112 MD number	**START_MODE_MASK** Determination of the basic control setting at NC start after boot and at RESET		
Default setting: 0x400,0x400, ...	Minimum input limit: 0	Maximum input limit: 0x7FFFF	
Changes effective after RESET	Protection level: 1 / 1	Unit:	
Data type: DWORD	Applies from SW 4.3 up to bit 17 in SW 6.4 and higher		

© Siemens AG, 2003. All rights reserved

20112 MD number	**START_MODE_MASK** Determination of the basic control setting at NC start after boot and at RESET
Significance:	Definition (by setting the following bits) of control basic setting at the start of the parts program with respect to G codes (in particular, active plane and active settable zero offset), active tool length compensation, transformation and axis couplings: Bit 0: Not assigned; START_MODE_MASK is evaluated every time a part program is started. Bit 1 = 0: Aux. funct. output (D, T, M) to PLC on tool selection according to MD 20120: TOOL_RESET_VALUE, MD 20130: CUTTING_EDGE_RESET_VALUE, MD 20121: TOOL_PRESEL_RESET_VALUE and MD 22550: TOOL_CHANGE_MODE. No effect when tool management active. Bit 1 = 1: Suppress aux. funct. output to PLC on tool selection. No effect when tool management active. Bit 2 Not assigned (reserved). Bit 3 Not assigned (reserved). Bit 4 = 0: The current setting for G code “current plane” is retained. Bit 4 = 1: Basic setting for G code “current plane” according to MD 20150: GCODE_RESET_VALUES. Bit 5 = 0: Current setting for G code “settable zero offset” is retained. Bit 5 = 1: Initial setting for G code “settable zero offset” acc. to MD 20150: GCODE_RESET_VALUES. Bit 6 = 0: The current setting for active tool length compensation is retained. When the tool management function is active, the tool selected is always the tool currently located in the active toolholder (spindle). If the tool located in the spindle is disabled, it is automatically replaced by a suitable spare tool. If no such tool exists, an alarm is output. Bit 6 = 1: Initial setting for active tool length compensation according to MD 20120: TOOL_RESET_VALUE. MD 20130: CUTTING_EDGE_RESET_VALUE, MD 20123: USEKT_RESET_VALUE and MD 20132: SUMCORR_RESET_VALUE. If MD 22550: TOOL_CHANGE_MODE=1, the tool specified in MD 20121: TOOL_PRESEL_RESET_VALUE is preselected. When the tool management function is active, MD 20120 is not used but MD 20122: TOOL_RESET_NAME instead. Bit 7 = 0: The current setting for the active transformation is retained. Bit 7 = 1: Initial setting for active transformation acc. to MD 20140: TRAFO_RESET_VALUE. Bit 8 = 0: Coupled-axis groupings are retained. Bit 8 = 1: Coupled-axis groupings are separated. Bit 9 = 0: Tangential followup remains active. Bit 9 = 1: Tangential correction is deactivated. Bit 10 = 0: Non-configured synchronous spindle link remains active. Bit 10 = 1: Non-configured synchronous spindle link is deactivated. Bit 11 Reserved (starting response: “Revolutional feedrate”) Bit 12 = 0: A modified geometry axis assignment remains active on parts program start. Bit 12 = 1: A modified geometry axis assignment is deleted on parts program start depending on MD 20118 GEOAX_CHANGE_RESET. Bit 13 = 0: Master/slave couplings remain active. Bit 13 = 1: Master/slave couplings are separated. Bit 14: Reserved (starting response “Basic frame”) Bit 15: Reserved (starting response “Electronic gear”) **SW 6.4 and higher** Bit 16 = 0: Initial setting for the master spindle acc. to MD 20090: SPIND_DEF_MASTER_SPIND Bit 16 = 1: The current setting of the master spindle (SETMS) is retained. Bit 17 = 0: The current setting of the master tool holder (SETMTH) is retained. (It is only significant if tool management is active.) Bit 17 = 1: Only if MD 20124: TOOL_MANAGEMENT_TOOLHOLDER > 0: Basic setting for the master tool holder acc. to MD 20124: TOOL_MANAGEMENT_TOOLHOLDER Otherwise, the setting for the master spindle applies. Continued on next page.

© Siemens AG, 2003. All rights reserved

20112 MD number	**START_MODE_MASK** Determination of the basic control setting at NC start after boot and at RESET
Related to	MD 20050: AXCONF_GEOAX_ASSIGN_TAB (assignment between geometry axis and channel axis) MD 20110: RESET_MODE_MASK (definition of control system initial setting after reset/parts program end) MD 20118: GEOAX_CHANGE_RESET (configuration of geometry axes on Reset/parts program end/parts program start) MD 20120: TOOL_RESET_VALUE (tool whose assigned length compensation is selected during the boot procedure (Reset/parts program end/parts program start)) MD 20121: TOOL_PRESEL_RESET_VALUE (preselected tool whose length compensation is selected during power-up (Reset/parts program end/start) MD 20122: TOOL_RESET_NAME (definition of tool with which length compensation is selected during power-up (Reset/parts program start) MD 20123: USEKT_RESET_VALUE (the value of MC 20123: USEKT_RESET_VALUE depending on MD 20110: according to MD 20110: RESET_MODE_MASK after booting/RESET/parts program start and according to MD 20112: START_MODE_MASK on parts program start) the value of MD 20123: USEKT_RESET_VALUE is preselected) MD 20124: TOOL_MANAGEMENT_TOOLHOLDER (toolholder number) MD 20130: CUTTING_EDGE_RESET_VALUE (cutting edge whose assigned length com pensation is selected on booting (RESET/parts program end) MD 20132: SUMCORR_RESET_VALUE (effective total offset with which the tool length compensation is selected on booting (Reset/parts program end) according to MD 20110: RESET_MODE_MASK and on parts program start according to MD 20112: START_MODE_MASKSTART_MODE_MASK. MD 20140: TRAFO_RESET_VALUE (transformation data set that is selected during booting (Reset/parts program end/parts program start)). MD 20150: GCODE_RESET_VALUES (initial setting of G groups [G group no.]). booting (Reset/parts program end/parts program start)). MD 22620: START_MODE_MASK_PRT Initial setting on special NC Start after booting and on RESET

20118 MD number	**GEOAX_CHANGE_RESET** Allow automatic geometry axis change		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after RESET	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 2.0		
Significance:	0: The current configuration of the geometry axes remains unchanged on Reset and on program start. With this setting, the response is identical to older software versions without geometry axis replacement. 1: The current configuration of the geometry axes remains unchanged on Reset or parts program end depending on MD 20110: RESET_MODE_MASK and, on program start, depending on MD 20112: START_MODE_MASK or is switched to the initial state defined by MD 20050: AXCONF_GEOAX_ASSIGN_TAB.		
Related to	MD 20050: AXCONF_GEOAX_ASSIGN_TAB MD 20110: RESET_MODE_MASK MD 20112: START_MODE_MASK		

20120 MD number	**TOOL_RESET_VALUE** Tool whose tool length compensation is selected during power-up (Reset/parts program end)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 32000	
Changes effective after RESET	Protection level: 2	Unit: 1	
Data type: DWORT	Applies from SW 2.0		
Significance:	Definition of tool for which tool length compensation is selected during boot or on reset or parts program end as a function of MD 20110: RESET_MODE_MASK and, on program start, depending on MD 20112: START_MODE_MASK.		
MD irrelevant for ...	MD 20110: RESET_MODE_MASK, bit 0 = 0		

© Siemens AG, 2003. All rights reserved

20121 MD number	**TOOL_PRESEL_RESET_VALUE** Preselected tool whose length compensation is selected during power-up (Reset/parts program end).		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 32000	
Changes effective after RESET	Protection level: 2	Unit: 1	
Data type: DWORT	Applies from SW 3.3		
Significance:	Definition of preselected tool for which tool length compensation is selected during Power-Up or on reset or parts program end as a function of MD 20110: RESET_MODE_MASK and, on program start depending on MD 20112: START_MODE_MASK. This MD is valid only when the tool management function is not active.		
MD irrelevant for ...	MD 20110: RESET_MODE_MASK, bit 0 = 0		

20130 MD number	**CUTTING_EDGE_RESET_VALUE** Cutting edge whose tool length compensation is selected during power-up (Reset/parts program end)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 9	
Changes effective after RESET	Protection level: 2	Unit: 1	
Data type: DWORT	Applies from SW 2.0		
Significance:	Definition of cutting edge for which tool length compensation is selected during PowerUp or on reset or parts program end as a function of MD 20110: RESET_MODE_MASK and, on program start, depending on MD 20112: START_MODE_MASK.		
MD irrelevant for ...	MD 20110: RESET_MODE_MASK, bit 0 = 0		

20140 MD number	**TRAFO_RESET_VALUE** Active transformation on RESET		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 8	
Changes effective after RESET	Protection level: 2	Unit: 1	
Data type: BYTE	Applies from SW 2.0		
Significance:	Definition of transformation data block which is selected during PowerUp and or RESET or parts program end as a function of MD 20110: RESET_MODE_MASK and, on program start, depending on MD 20112: START_MODE_MASK. Number of transformation data block (1 ...8) corresponding to MD TRAFO_TYPE_1 to TRAFO_TYPE_8. Permissible transformations: TRACYL, TRAANG, for SW 4 and higher, also TRANSMIT		
MD irrelevant for ...	MD 20110: RESET_MODE_MASK, bit 0 = 0		

© Siemens AG, 2003. All rights reserved

20150 MD number	**GCODE_RESET_VALUES[n]** Initial reset for the G groups G group no.: 0...59		
Default setting: see below	Minimum input limit: 0.0	Maximum input limit: plus	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		

Significance:

Specification of a G code which becomes effective on run-up and reset or at parts program end depending on MD 20110: RESET_MODE_MASK (up to SW 4) and
MD: GCODE_RESET_MODE (SW 5 and higher) and at parts program start depending on the setting in MD 20112 START_MODE_MASK.
The index of the G codes in the respective groups must be programmed as the default value.
For a list of the G groups with the G functions they contain, please refer to
References: /PA/, Programming Guide, Advanced.

Title	**Group**	**Default setting** on 840D/810D/FM-NC
GCODE_RESET_VALUES[0]	1	1 (G00)
GCODE_RESET_VALUES[1]	2	0 (inactive)
GCODE_RESET_VALUES[2]	3	0 (inactive)
GCODE_RESET_VALUES[3]	4	2 (START FIFO)
GCODE_RESET_VALUES[4]	5	0 (inactive)
GCODE_RESET_VALUES[5]	6	1 (G17)
GCODE_RESET_VALUES[6]	7	1 (G40)
GCODE_RESET_VALUES[7]	8	1 (G500)
GCODE_RESET_VALUES[8]	9	0 (inactive)
GCODE_RESET_VALUES[9]	10	1 (G60)
GCODE_RESET_VALUES[10]	11	0 (inactive)
GCODE_RESET_VALUES[11]	12	1 (G601)
GCODE_RESET_VALUES[12]	13	2 (G71)
GCODE_RESET_VALUES[13]	14	1 (G90)
GCODE_RESET_VALUES[14]	15	2 (G94)
GCODE_RESET_VALUES[15]	16	1 (CFC)
GCODE_RESET_VALUES[16]	17	1 (NORM)
GCODE_RESET_VALUES[17]	18	1 (G450)
GCODE_RESET_VALUES[18]	19	1 (BNAT)
GCODE_RESET_VALUES[19]	10	1 (ENAT)
GCODE_RESET_VALUES[20]	21	1 (BRISK)
GCODE_RESET_VALUES[21]	22	1 (RTCPOF)
GCODE_RESET_VALUES[22]	23	1 (CDOF)
GCODE_RESET_VALUES[23]	24	1 (FFWOF)
GCODE_RESET_VALUES[24]	25	1 (ORIWKS)
GCODE_RESET_VALUES[25]	26	2 (RMI)
GCODE_RESET_VALUES[26]	27	1 (ORIC)
GCODE_RESET_VALUES[27]	28	1 (WALIMON)
GCODE_RESET_VALUES[28]	29	1 (DIAMOF)
GCODE_RESET_VALUES[29]	30	1 (COMPOF)
GCODE_RESET_VALUES[30]	31	1 (inactive)
GCODE_RESET_VALUES[31]	32	1 (inactive)
GCODE_RESET_VALUES[32]	33	1 (FTCOF)
GCODE_RESET_VALUES[33]	34	1 (OSOF)
GCODE_RESET_VALUES[34]	35	1 (SPOF)
GCODE_RESET_VALUES[35]	36	1 (PDLAYON)
GCODE_RESET_VALUES[36]	37	1 (FNOORM)
GCODE_RESET_VALUES[37]	38	1 (SPF1)
GCODE_RESET_VALUES[38]	39	1 (CPRECOF)
GCODE_RESET_VALUES[39]	40	1 (CUTCONOF)
GCODE_RESET_VALUES[40]	41	1 (LFOF)
:	:	
GCODE_RESET_VALUES[46]	47	1 (LFTXT)
:	:	
GCODE_RESET_VALUES[48]	49	1 (CP) 2 (PTP)
:	:	
GCODE_RESET_VALUES[59]	60	1 (not defined)

Application example(s):

G91 (instead of G90) must be active as the default setting in G group 14.
GCODE_RESET_VALUES[13] = 2

© Siemens AG, 2003. All rights reserved

20152 MD number	**GCODE_RESET_MODE** G code basic setting at RESET		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 5		
Significance:	MD 20152 is operative only if MD 20110, bit 0 = 1. Definition of control initial setting on reset/end of parts program with reference to G codes. 0: The initial setting of the G group is set according to the entry in MD 20150: GCODE_RESET_VALUES 1: The current G group setting is retained on reset/end of parts program. Example: Read initial setting for the 6th G group (current plane) from MD 20150: GCODE_RESET_VALUES on every reset/end of parts program MD 20150: GCODE_RESET_VALUES[5]=1 ;Reset value of 6th G group is G17 MD 20152: GCODE_RESET_MODE[5]=0 ;Initial setting for 6th G group after ;reset/end of parts program corresponds ;to GCODE_RESET_VALUES[5] If the current setting for the 6th G group is to be retained beyond reset/end of parts program then: MD 20150: GCODE_RESET_VALUES[5]=1 ;Reset value of 6th G group is G17 MD 20152: GCODE_RESET_MODE[5]=1 ;Current setting for 6th G group ;is retained after reset/end of part ;program		
Related to	MD 20150: GCODE_RESET_VALUES MD 20110: RESET_MODE_MASK MD 20112: START_MODE_MASK		

22620 MD number	**START_MODE_MASK_PRT** Initial setting at special NC Start after booting and at RESET		
Default setting: 0x400,0x400, ...	Minimum input limit: 0	Maximum input limit: 0x7FFF	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	This machine data is enabled by MD 22621: ENABLE_START_MODE_MASK_PRT. In the initial setting of the control, MD 22621: ENABLE_START_MODE_MASK_PRT is inactive. If MD 22620: START_MODE_MASK_PRT is enabled for “search run by program test” (SERUPRO) then, when the “search run by program test” is started, this machine data replaces machine data MD 20112: START_MODE_MASK. This allows you to set a different response to PLC start for search run start. The meaning of the bit assignments in MD 22620: START_MODE_MASK_PRT is identical to MD 20112: START_MODE_MASK and is as follows: Bits: Bit 0: Not assigned; START_MODE_MASK is evaluated every time a part program is started. Bit 1 = 0: Aux. funct. output (D, T, M) to PLC on tool selection according to MD 20120: TOOL_RESET_VALUE, MD 20130: CUTTING_EDGE_RESET_VALUE, MD 20121: TOOL_PRESEL_RESET_VALUE and MD 22550: TOOL_CHANGE_MODE. No effect when tool management active. Bit 1 = 1: Suppress aux. funct. output to PLC on tool selection. No effect when tool management active. Bit 2 Not assigned (reserved). Bit 3 Not assigned (reserved). Bit 4 = 0: The current setting for G code “current plane” is retained. Bit 4 = 1: Basic setting for G code “current plane” according to MD 20150: GCODE_RESET_VALUES. Bit 5 = 0: Current setting for G code “settable zero offset” is retained. Bit 5 = 1: Initial setting for G code “settable zero offset” acc. to MD 20150: GCODE_RESET_VALUES.		

© Siemens AG, 2003. All rights reserved

22620 MD number	**START_MODE_MASK_PRT** Initial setting at special NC Start after booting and at RESET
Significance:	Bit 6 = 0: The current setting for active tool length compensation is retained. When the tool management function is active, the tool selected is always the tool currently located in the active toolholder (spindle). If the tool located in the spindle is disabled, it is automatically replaced by a suitable spare tool. Bit 6 = 1: Initial setting for active tool length compensation according to MD 20120: TOOL_RESET_VALUE, MD 20130: CUTTING_EDGE_RESET_VALUE, MD 20123: USEKT_RESET_VALUE and MD 20132: SUMCORR_RESET_ VALUE. If MD 22550: TOOL_CHANGE_MODE=1, the tool specified in MD 20121: TOOL_PRESEL_RESET_VALUE is also preselected. When the tool management function is active, MD 20120 is not used but MD 20122: TOOL_RESET_NAME instead. Bit 7 = 0: The current setting for the active transformation is retained. Bit 7 = 1: Initial setting for active transformation after RESET/parts program end according to MD 20140: TRAFO_RESET_VALUE. Bit 8 = 0: Coupled-axis groupings are retained. Bit 8 = 1: Coupled-axis groupings are separated. Bit 9 = 0: Tangential correction remains active. Bit 9 = 1: Tangential correction is deactivated. Bit 10 = 0: Non-configured synchronous spindle link remains active. Bit 10 = 1: Non-configured synchronous spindle link is deactivated. Bit 11 Not assigned (reserved). Bit 12 = 0: A modified geometry axis assignment remains active on parts program start. Bit 12 = 1: A modified geometry axis assignment is deleted on parts program start depending on MD 20118 GEOAX_CHANGE_RESET. Bit 13 = 0: Master/slave couplings remain active. Bit 13 = 1: Master/slave couplings are separated. Bit 14: Reserved (starting response “Basic frame”)
Related to	MD 20050: AXCONF_GEOAX_ASSIGN_TAB (assignment between geometry axis and channel axis) MD 20110: RESET_MODE_MASK (definition of control system initial setting after RESET/parts program end) MD 20112: START_MODE_MASK Initial function setting MD 20118: GEOAX_CHANGE_RESET (configuration of geometry axes on RESET/parts program end/parts program start) MD 20120: TOOL_RESET_VALUE (tool whose assigned length compensation is selected during the boot procedure (RESET/parts program end/parts program start)) MD 20121: TOOL_PRESEL_RESET_VALUE (preselected tool whose length compensation is selected during powerup (RESET/parts program end/parts program start) MD 20122: TOOL_RESET_NAME (definition of tool with which length compensation is selected during power-up (Reset/parts program start) MD 20123: USEKT_RESET_VALUE (the value of MC 20123: USEKT_RESET_VALUE is preselected depending on MD 20110: RESET_MODE_MASK after booting/ RESET/parts program start and depending on MD 20112: START_MODE_MASK on parts program start) MD 20124: TOOL_MANAGEMENT_TOOLHOLDER (toolholder number) MD 20130: CUTTING_EDGE_RESET_VALUE (cutting edge whose assigned length is selected during power-up (Reset/end of parts program) MD 20132: SUMCORR_RESET_VALUE (effective total offset with which the tool length compensation is selected on booting (Reset/parts program end) according to MD 20110: RESET_MODE_MASK on parts program start according to MD 20112: START_MODE_MASK) MD 20110: RESET_MODE_MASK on parts program start according to MD 20112: START_MODE_MASKSTART_MODE_MASK. MD 20140: TRAFO_RESET_VALUE (transformation data set that is selected during booting (RESET/parts program end/parts program start)). MD 20150: GCODE_RESET_VALUES (initial setting of G groups [G group no.]). booting (RESET/parts program end/parts program start)). MD 22621: ENABLE_MODE_MASK_PRT enables MD 22620: START_MODE_MASK_PRT for SERUPRO search run.

© Siemens AG, 2003. All rights reserved

4.4.4 Basic block display

27100 MD number	**ABSBLOCK_FUNCTION_MASK** Parameterization of basic block display		
Default setting: 0x0	Minimum input limit: 0	Maximum input limit: 0x1	
Changes effective after NEW_CONF	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Parameterization of basic block display function Bit 0 = 1: Positions values of traverse axis are always displayed as diameter value.		
Application example(s)	Among other functions, this information is provided for evaluation by the HMI, MMC, PLC and standard cycles.		

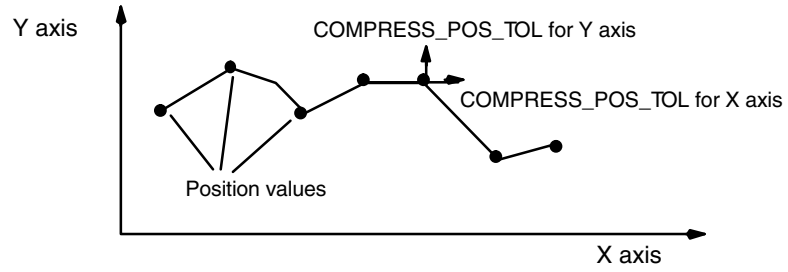
28400 MD number	**MM_ABSBLOCK** Activate basic block display		
Default setting: 2	Minimum input limit: 0	Maximum input limit: 512	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	The meaning of the machine data is as follows: Value 0: Basic block display deactivated. 1: Basic block display activated. A display buffer of the size defined in (MD 28060: MM_IPO_BUFFER_SIZE plus MD 28070: MM_NUM_BLOCKS_IN_PREP) * 128 bytes is set up. ≥ 128: Basic block display activated. A display buffer of the size defined in MD 28060: MM_IPO_BUFFER_SIZE plus MD 28070: MM_NUM_BLOCKS_IN_PREP * < value > is set up.		
Related to	SD 42750: ABSBLOCK_ENABLE (enable basic block display) MD MM_ABSBLOCK_BUFFER_CONF[2] (dimension size of upload buffer)		

28402 MD number	**MM_ABSBLOCK_BUFFER_CONF[2]** Dimension size of upload buffer		
Default setting: [1, 5]	Minimum input limit: –	Maximum input limit: 8 MD 28060 plus MD 28070	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	Dimension size of upload buffer MD 28402: MM_ABSBLOCK_BUFFER_CONF[0]: Number of blocks after the current block MD 28402: MM_ABSBLOCK_BUFFER_CONF[1]: Number of blocks before the current block Recommended default setting: MD 28402: MM_ABSBLOCK_BUFFER_CONF[1] MD 28402: MM_ABSBLOCK_BUFFER_CONF[5] The machine data is checked for the following upper and lower limits at boot: ≤ MD 28402: MM_ABSBLOCK_BUFFER_CONF[0]: ≤ 8 ≤ MD 28402: MM_ABSBLOCK_BUFFER_CONF[1]: ≤ MD 28060 plus MD 28070 MD 28060: MM_IPO_BUFFER_SIZE + MD 28070: MM_NUM_BLOCKS_IN_PREP Alarm 4152 is output if these limits are violated.		
Related to	MD MM_ABSBLOCK_BUFFER (activate basic block display)		

© Siemens AG, 2003. All rights reserved

4.5 Axis/spindle-specific machine data

30600 MD number	**FIX_POINT_POS** Fixed value positions of axes with G75		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The fixed point positions (max. 2) for each axis which can be approached when G75 is programmed are entered in these machine data.		
Application example(s)	Travel to 2nd fixed point: G75 X1=0 FP=2 (a dummy value, here 0, must be defined for the axis).		
References	/PA/, “Programming Guide Fundamentals”		

33100 MD number	**COMPRESS_POS_TOL** Maximum deviation with compensation		
Default setting: 0.1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after	POWER ON, New Conf (SW4.2 and higher)	Protection level: 7/7	Unit: mm, degrees
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The value specifies the maximum permitted path deviation for each axis with compression. The larger the value, the more short blocks can be compressed into a long block. The maximum number of compressible blocks is limited by the size of the spline buffer (currently 10 blocks).		
Fig.			
References	/PA/, “Programming Guide Fundamentals”		

© Siemens AG, 2003. All rights reserved

4.6 Channel-specific setting data

42000 SD number	**THREAD_START_ANGLE** Start angle for thread		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective immediately	Protection level: MMCMD: 9220	Unit: degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The offset between the individual thread turns in multiple-turn thread cutting can be set in this machine data. This setting data can be altered with the command SF in the parts program. The default setting is activated on NC RESET.		
References	/PA/, “Programming Guide Fundamentals”		
Related to	MD 10710 $MN_PROG_SD_RESET_SAVE_TAB		

42200 SD number	**SINGLEBLOCK2_STOPRE** Activate debug mode for SBL2		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after: immediate	Protection level: 7/7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.3		
Significance:	Value of SD 42200: SINGLEBLOCK2_STOPRE = TRUE: A preprocessing stop is executed on every block when SBL2 (single block with stop after every block) is active. This suppresses preprocessing of parts program blocks. This variant of SBL2 does not maintain an accurate contour. In other words, as a result of the preprocessing stop, a different contour may be generated than the one created without single block mode or with SBL1. **Application:** Debug mode for testing parts programs.		
References	/PGA/, “Programming Guide Advanced”		

42444 SD number	**TARGET_BLOCK_INCR_PROG** Continuation mode after block search with calculation		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after: immediate	Protection level: 7/7	Unit: –	
Data type: BOOLEAN	Applies from SW 4.3		
Significance:	If initial programming of an axis is carried out incrementally after “Search with calculation at end of block”, the incremental value is added to the value totalled up to the search target according to setting data SD 42444: TARGET_BLOCK_INCR_PROG SD 42444: = TRUE: incremental value is added to the totalled position SD 42444: = FALSE: incremental value is added to the current actual value. The setting data is evaluated in conjunction with the output of the action blocks on NC start.		

42990 SD number	**MAX_BLOCKS_IN_IPOBUFFER** Control of max. number of blocks in interpolation buffer		
Default setting: –1	Minimum input limit: –	Maximum input limit: –	
Changes effective after: immediate		Unit: –	
Data type: INTEGER	Applies from SW 6.3		
Significance:	This setting data can limit the maximum number of blocks in the interpolation buffer. In this case, the maximum number is defined by MD 28060: MM_IPO_BUFFER_SIZE. A negative setting means that the number of blocks in the IPO buffer is not effectively limited and the number if specified solely by MD 28060: MM_IPO_BUFFER_SIZE (default setting).		
Related to	MD 28060: MM_IPO_BUFFER_SIZE		

© Siemens AG, 2003. All rights reserved

4.7 Setting data for processing programs from external (SW 4 and later)

42700 SD number	**EXT_PROG_PATH** Name of an external program path for subroutine call EXTCALL		
Default setting: –	Minimum input limit: –	Maximum input limit: –	
Changes effective immediately	Protection level: 7/7	Unit: –	
Data type: STRING	Applies from SW 4.2 (HMI Advanced)		
Significance:	The name of the external program path can be specified here for subroutine call EXTCALL. The total path comprises the string of SD 42700: EXT_PROG_PATH + programmed subroutine name **Example:** /_N_WKS_DIR/_N_WKST1_DIR /ROUGHING_SPF		
References	/PGA/ “Programming Guide, Advanced”, Chapter 2		

4.8 Setting data: Enable basic block display (SW 6.4 and later)

42750 SD number	**ABSBLOCK_ENABLE** Enable basic block display		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 7/7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.4		
Significance:	The meaning of this setting data is as follows: Value 0: Disable basic block display Value 1: Enable basic block display		
Related to	MD 28400: MM_ABSBLOCK		

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Signal Descriptions

5

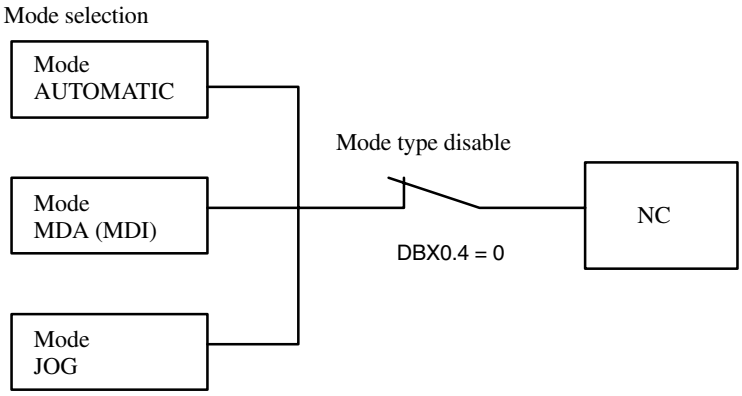
5.1 Mode group-specific signals

DB11, ... **DBX0.0** Data block	**AUTOMATIC mode** Signal(s) to mode group (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	AUTOMATIC mode is selected by the PLC program.	
Signal state 0 or signal transition 1 ——> 0	AUTOMATIC mode is not selected by the PLC program.	
Signal irrelevant for ...	if Operating mode “switchover inhibit” (DB11, ... DBX0.4) signal = 1	
Related to	IS “Active AUTOMATIC” mode (DB11, ... DBX6.0)	

DB11, ... **DBX0.1** Data block	**MDA mode** Signal(s) to mode group (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	MDA mode is selected by the PLC program.	
Signal state 0 or signal transition 1 ——> 0	MDA mode is not selected by the PLC program.	
Signal irrelevant for ...	if IS “Operating mode switchover inhibit” (DB11, ... D0.4) signal = 1	
Related to	IS “Active MDA mode” (DB11, ... DBX6.1)	

DB11, ... **DBX0.2** Data block	**JOG mode** Signal(s) to mode group (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	JOG mode is selected by the PLC program.	
Signal state 0 or signal transition 1 ——> 0	JOG mode is not selected by the PLC program.	
Signal irrelevant for ...	if Operating mode “switchover inhibit” (DB11, ... DBX0.4) signal = 1	
Related to	IS “Active JOG mode” (DB11, ... DBX6.2)	

© Siemens AG, 2003. All rights reserved

DB11, ... **DBX0.4** Data block	**Mode switchover disable** Signal(s) to mode group (PLC ——> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The current active mode (JOG, MDA or AUTOMATIC) of the mode group cannot be changed. The machine functions selected within a mode group can be changed.		
Signal state 0 or signal transition 1 ——> 0	The mode of the mode group can be changed.		
Fig.			

DB11, ... **DBX0.5** Data block	**Mode group stop** Signal(s) to mode group (PLC ——> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	An NC stop is activated for all the channels of the mode group. The channel status of all the active channels changes to interrupted. All the channels in channel status RESET remain in that state. Programs that are running at this point are immediately interrupted (at the earliest possible point, even within a block) and the program status changes to stopped. All the moving axes of the mode group are decelerated according to their acceleleration characteristics without contour violation. The program can be restarted with NC start. None of the spindles of that mode group are affected.		
Signal state 0 or signal transition 1 ——> 0	Channel status and program run are not affected.		
Special cases, errors, ...	All the axes of a mode group that are not triggered by a program or a program block (e.g. axes traversing because traverse keys on machine control panel are being operated) decelerate to rest with mode group stop.		

DB11, ... **DBX0.6** Data block	**Mode group stop axes plus spindles** Signal(s) to mode group (PLC ——> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	An NC stop is activated for all the channels of the mode group. The channel status of all the active channels changes to interrupted. All the channels in channel status RESET remain in that state. Programs that are running at this point are immediately interrupted (at the earliest possible point, even within a block) and the program status changes to stopped. All moving axes and spindles are decelerated to zero speed according to their acceleration ramp without contour violation. The program can be restarted with NC start.		

© Siemens AG, 2003. All rights reserved

DB11, ... **DBX0.6** Data block	**Mode group stop axes plus spindles** Signal(s) to mode group (PLC ——> NCK)
Signal state 0 or signal transition 1 ——> 0	Channel status and program run are not affected.
Special cases, errors, ...	All the axes and spindles of a mode group that are not triggered by a program or a program block (e.g. axes traversing because traverse keys on machine control panel are being operated, spindles are being controlled by the PLC) decelerate to rest with “Mode group stop plus spindles”.

DB11, ... **DBX0.7** Data block	**Mode group reset** Signal(s) to mode group (PLC ——> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	A Reset is activated for all the channels of the mode group. All channels are then in channel status Reset. All the current programs go into program status aborted. All moving axes and spindles are decelerated to zero speed according to their acceleration ramp without contour violation.The initial setting are set (e.g. for G functions). The alarms for the mode group are cleared if they are not POWER ON alarms.	
Signal state 0 or signal transition 1 ——> 0	Channel status and program run are not affected by this signal.	
Related to	IS “Channel reset” (DB21, ... DBX7.7) IS “All channels in reset state” (DB11, ... DBX6.7)	
Special cases, errors, ...	An alarm which cancels IS “Mode group ready” ensures that all the channels in the mode group are no longer in the Reset state. In order to switch to another operating mode, a mode group reset (DB11, ... DBX0.7) must then be initiated.	

DB11, ... **DBX1.0** Data block	**Machine function TEACH IN** Signal(s) to mode group (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function TEACH IN is activated in JOG mode for the mode group.	
Signal state 0 or signal transition 1 ——> 0	Machine function TEACH IN is not activated.	
Signal irrelevant for ...	If JOG mode is not active.	
References	/BA/, “Operator’s Guide”	

DB11, ... **DBX1.1** Data block	**Machine function REPOS** Signal(s) to mode group (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function REPOS is activated in JOG mode for the mode group.	
Signal state 0 or signal transition 1 ——> 0	Machine function REPOS is not activated.	
Signal irrelevant for ...	If JOG mode is not active.	
Application example(s)	Because of an error during parts program operation (e.g. tool breakage) the axis is traversed from the place of error in JOG mode. The axis can then be returned manually to the exact previous position with machine function REPOS so that the program can be continued in AUTOMATIC mode.	
References	/BA/, “Operator’s Guide”	

© Siemens AG, 2003. All rights reserved

DB11, ... **DBX1.2** Data block	**Machine function REF** Signal(s) to mode group (PLC ——> NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function REF is activated in JOG mode for the mode group.		
Signal state 0 or signal transition 1 ——> 0	Machine function REF is not activated.		
Signal irrelevant for ...	If JOG mode is not active.		
References	/FB/, R1, “Reference Point Approach”		

DB11, ... **DBX4.0** Data block	**Selected mode AUTOMATIC** Signal(s) from mode group (MMC ——> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	AUTOMATIC mode is selected by the MMC.		
Signal state 0 or signal transition 1 ——> 0	AUTOMATIC mode is not selected by the MMC.		

DB11, ... **DBX4.1** Data block	**Selected mode MDA** Signal(s) from mode group (MMC ——> PLC)		
Edge evaluation:		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	MDA mode is selected by the MMC.		
Signal state 0 or signal transition 1 ——> 0	MDA mode is not selected by the MMC.		

DB11, ... **DBX4.2** Data block	**Selected mode JOG** Signal(s) from mode group (MMC ——> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	JOG mode is selected by the MMC.		
Signal state 0 or signal transition 1 ——> 0	JOG mode is not selected by the MMC.		

DB11, ... **DBX5.0** Data block	**Selected machine function TEACH IN** Signal(s) from mode group (MMC ——> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function TEACH IN is selected in the mode group by the MMC.		
Signal state 0 or signal transition 1 ——> 0	Machine function TEACH IN is not selected by the MMC.		
References	/BA/, “Operator’s Guide”		

© Siemens AG, 2003. All rights reserved

DB11, ... **DBX5.1** Data block	**Selected machine REPOS function TEACH IN** Signal(s) from mode group (MMC ——> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function REPOS is selected in the mode group by the MMC.		
Signal state 0 or signal transition 1 ——> 0	Machine function REPOS is not selected by the MMC.		
Application example(s)	If an error occurs during parts program operation (e.g. tool breakage) the axis is traversed from the place of error in JOG mode in order to be able to replace the tool. The axis can then be returned manually to the exact previous position with machine function REPOS so that the program can be continued in AUTOMATIC mode.		
References	/BA/, “Operator’s Guide”		

DB11, ... **DBX5.2** Data block	**Selected machine function REF** Signal(s) from mode group (MMC ——> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function REF is selected in the mode group by the MMC.		
Signal state 0 or signal transition 1 ——> 0	Machine function REF is not selected by the MMC.		
References	/FB/, R1, “Reference Point Approach”		

DB11, ... **DBX6.0** Data block	**Active mode AUTOMATIC** Signal(s) from mode group (NCK ——> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	AUTOMATIC mode is active.		
Signal state 0 or signal transition 1 ——> 0	AUTOMATIC mode is not active.		

DB11, ... **DBX6.1** Data block	**Active mode MDA** Signal(s) from mode group (NCK ——> PLC)		
Edge evaluation:		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	MDA mode is active.		
Signal state 0 or signal transition 1 ——> 0	MDA mode is not active.		

DB11, ... **DBX6.2** Data block	**Active mode JOG** Signal(s) from mode group (NCK ——> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	JOG mode is active.		
Signal state 0 or signal transition 1 ——> 0	JOG mode is not active.		

© Siemens AG, 2003. All rights reserved

DB11, ... **DBX6.3** Data block	**Mode group ready** Signal(s) from mode group (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	This signal is set after POWER ON when all voltages are present. The mode group is now ready and parts programs can be processed and axes traversed in the individual channels.		
Signal state 0 or signal transition 1 ——> 0	The mode group is not ready. Possible causes: – A serious axis or spindle alarm is active – Hardware fault – Mode group incorrectly configured (machine data) If the mode group ready signal changes to the “0” state – the axis and spindle drives are decelerated with max. braking current to zero speed – the signals from the PLC to the NCK are set to the inactive state (initial setting).		
Special cases, errors, ...	An alarm that cancels the IS “Mode group ready” ensures that all channels of the mode group are no longer in the reset state. In order to switch to another operating mode, a mode group reset (DB11, ... DBX0.7) must then be initiated.		

DB11, ... **DBX6.7** Data block	**All channels in Reset state** Signal(s) from mode group (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	All the channels that belong to this mode group are in the “Channel status reset state” (DB21, ... DBX7.7).	
Signal state 0 or signal transition 1 ——> 0	At least one of the channels in the mode group is not in “Channel status reset” (DB21, ... DBX7.7).	
Related to	IS “Channel status Reset” (DB21, ... DBX7.7)	

DB11, ... **DBX7.0** Data block	**Active machine function TEACH IN** Signal(s) from mode group (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function TEACH IN is active in the mode group.	
Signal state 0 or signal transition 1 ——> 0	Machine function TEACH IN is not active.	
References	/BA/, “Operator’s Guide”	

DB11, ... **DBX7.1** Data block	**Active machine function REPOS** Signal(s) from mode group (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Machine function REPOS is active in the mode group.	
Signal state 0 or signal transition 1 ——> 0	Machine function REPOS is not active.	
Application example(s)	If an error occurs during parts program operation (e.g. tool breakage) the axis is traversed from the place of error in JOG mode in order to be able to replace the tool. The axis can then be returned manually to the exact previous position with machine function REPOS so that the program can be continued in AUTOMATIC mode.	
References	/BA/, “Operator’s Guide”	

© Siemens AG, 2003. All rights reserved

DB11, ... **DBX7.2** Data block	**Active machine function REF** Signal(s) from mode group (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	Machine function REF is active in the mode group.		
Signal state 0 or signal transition 1 ——> 0	Machine function REF is not active.		
References	/FB/, R1, “Reference Point Approach”		

5.2 Channel-specific signals

DB21, ... **DBX0.4** Data block	**Activate single block** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The program is processed in single-block mode in AUTOMATIC and MDA modes.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Application example(s)	A new program can first be tested in single-block mode in order to monitor the individual program steps more exactly.		
Special cases, errors, ...	– Intermediate blocks can be inserted if cutter radius path compensation is selected. – Single-block mode can only be applied to a series of G33 blocks if “dry run feedrate” is selected. – Calculation blocks are not processed in single step mode if single decoding block mode is active.		
Related to	IS “Program status interrupted” (DB21, ... DBX35.3)		
References	Section 2.4		

DB21, ... **DBX0.5** Data block	**Activate M01** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	M01 programmed in the parts program leads to a programmed stop when processed in AUTOMATIC or MDA mode.		
Signal state 0 or signal transition 1 ——> 0	M01 programmed in the parts program does not lead to a programmed stop.		
Related to	IS “M01 selected” (DB21, ... DBX24.5) IS “M0/M01 active” (DB21, ... DBX32.5)		

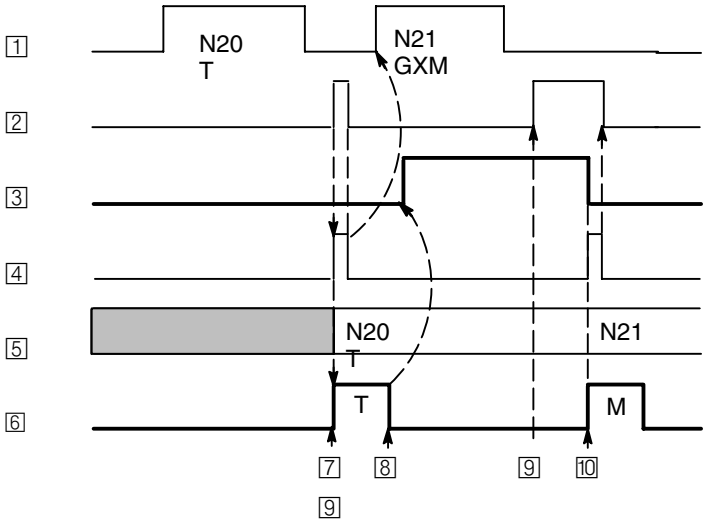
© Siemens AG, 2003. All rights reserved

DB21, ... **DBX1.6** Data block	**PLC action ended** Signal(s) to channel (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 4.3
Signal state 1 or signal transition 0 ——> 1	Axis disable is set internally for all axes (not spindles). Therefore the machine axes do not move when a parts program block or a parts program is being processed. The axis movements are simulated on the operator interface with changing axis position values. The axis position values for the display are generated from the calculated setpoints. The parts program is processed in the normal way. **SW 6.1 and higher** it is possible to manipulate the control response by setting bit 1 = 1 in machine data MD 11450: SEARCH_RUN_MODE such that alarm 10208 is generated after the last action block has been output. Tool management: Because of the axis disable, the assignment of a tool magazine is not changed during program testing. A PLC application must ensure that the integrity of the data in the tool management system and the magazine is not corrupted (see the example on the toolbox diskette).	
Signal state 0 or signal transition 1 ——> 0	Parts program processing is not affected by the function program test	
Related to	IS “Program testing active” (DB 21–22, DBX33.7)	

DB21, ... **DBX1.7** Data block	**Activate program test** Signal(s) to channel (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Axis disable is set internally for all axes (not spindles). Therefore the machine axes do not move when a parts program block or a parts program is being processed. The axis movements are simulated on the operator interface with changing axis position values. The axis position values for the display are generated from the calculated setpoints. The parts program is processed in the normal way. Tool management: Because of the axis disable, the assignment of a tool magazine is not changed during program testing. A PLC application must ensure that the integrity of the data in the tool management system and the magazine is not corrupted (see the example on the toolbox diskette).	
Signal state 0 or signal transition 1 ——> 0	Parts program processing is not affected by the function program test	
Related to	IS “Program test selected” (DB21, ... DBX25.7) IS “Program testing active” (DB 21–22, DBX33.7)	

DB21, ... **DBX2.0** Data block	**Skip block** Signal(s) to channel (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Blocks marked in the parts program with an oblique (/) are skipped. If there is a series of skip blocks, this signal is only active if it is pending before decoding of the first block of the series, ideally before NC start.	
Signal state 0 or signal transition 1 ——> 0	The marked parts program blocks are not skipped. A series of blocks will only be processed if the signal is in the 0 state before the first block is decoded.	
Related to	IS “Skip block selected” (DB21, ... DBX26.0) IS “Program status stopped” (DB21, ... DBX35.2)	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX6.1** Data block	**Read-in disable** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	Data transfer of the next block into the interpolator is disabled. This signal is only active in modes AUTOMATIC and MDA.		
Signal state 0 or signal transition 1 ——> 0	Data transfer of the next block into the interpolator is enabled. This signal is only active in modes AUTOMATIC and MDA.		
Application example(s)	In a case where an auxiliary function has to have been executed before the next block can be processed (e.g. for a tool change), automatic block change must be inhibited with read-in disable.  1 Read block into buffer memory 2 Block processed 3 Read-in disable signal 4 Data transfer 5 Contents of interpolator 6 Output of auxiliary function 7 Data transfer to interpolator 8 Read-in disable for tool change 9 Scan location for readin enable 10 Cancel read-in disable		
Related to	IS “Program status running” (DB21, ... DBX35.0)		

© Siemens AG, 2003. All rights reserved

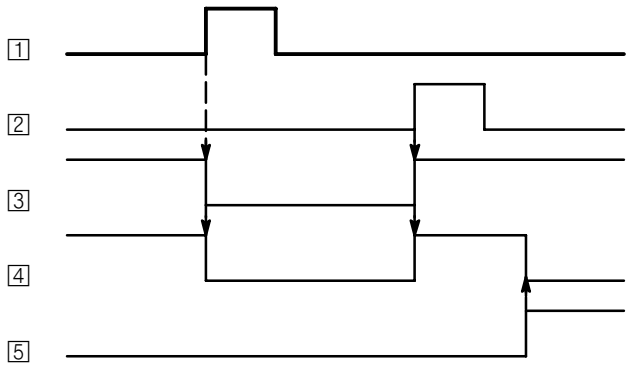
DB21, ... **DBX6.4** Data block	**Program level abort** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: yes		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	On each signal transition 0 –> 1 the current program level being processed (subroutine level, asynchronous subroutine level, save routine) is immediately aborted. Processing of the parts program continues one level higher after the exit point.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Special cases, errors, ...	The main program level cannot be aborted with this IS, only with IS “Reset”.		

DB21, ... **DBX7.0** Data block	**NC start disable** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	IS “NC start” and the START command from the parts program have no effect.		
Signal state 0 or signal transition 1 ——> 0	IS “NC start” and the START command from the parts program have an effect.		
Application example(s)	This signal is used to suppress renewed program processing because, for example, there is no lubricant.		
Related to	IS “NC Start” DB21, ... DBX7.1)		

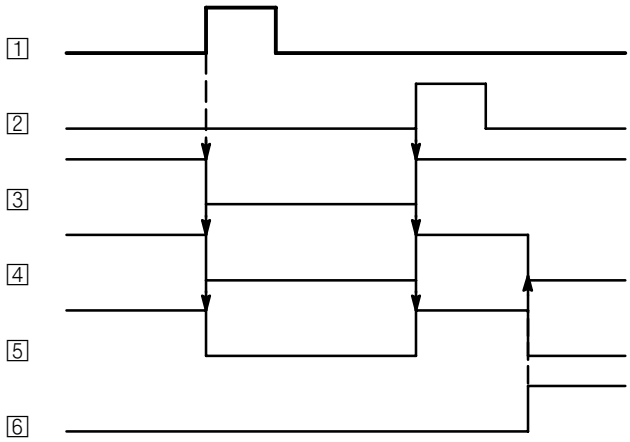
DB21, ... **DBX7.1** Data block	**NC Start** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: yes		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	AUTOMATIC mode: The selected NC program is started or continued or the auxiliary functions placed in the memory during program interruption are output. If data is transferred from the PLC to the NC during program status “program interrupted”, these are calculated immediately on NC start. MDA mode: The entered block information or parts program blocks are released for execution.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Related to	IS “NC Start disable” (DB21, ... DBX7.0)		

DB21, ... **DBX7.2** Data block	**NC Stop at block limit** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The current NC program is stopped after the current parts program block has been processed. Otherwise as for IS “NC Stop” DB21, ... DBX7.3.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Related to	IS “NC Stop” (DB21, ... DBX7.3) IS “NC Stop axes plus spindle” (DB21, ... DBX7.4) IS “Program status stopped” (DB21, ... DBX35.2) IS “Channel status interrupted” (DB21, ... DBX35.6)		

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX7.3** Data block	**NC Stop** Signal(s) to channel (PLC ——> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The current NC program is stopped immediately, the current block is not completed. Only the axes are stopped. These are brought to a controlled stop without contour violation. Distances-to-go are not completed until a start signal is given. The program status switches to stopped, the channel status switches to interrupted. If incremental distances are being traversed in JOG, this movement is aborted. The remaining increments are completed following a renewed NC Start.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Application example(s)	If the NC program has been stopped, auxiliary functions, for example, can be put into the memory. They are then executed on the next NC start. On NC start the program is continued at the point of interruption. If auxiliary functions were put into the memory when the program was interrupted, only these become active on the first NC start. On the second NC start the program is then continued.  1 IS “NC Stop” 2 IS “NC Start” 3 Program running 4 Axis running 5 Block processed		
Special cases, errors, ...	The signal NC stop must be active for at least one PLC cycle. If data was transferred to the NC on NC stop (e.g. tool offset), these are calculated completely on NC start. Spindles are not stopped on MD 35040: SPIND_ACTIVE_AFTER_RESET = 1		
Related to	IS “NC Stop at block limit” (DB21, ... DBX7.2) IS “NC Stop axes plus spindle” (DB21, ... DBX7.4) IS “Program status stopped” (DB21, ... DBX35.2) IS “Channel status interrupted” (DB21, ... DBX35.6)		

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX7.4** Data block	**NC Stop axes plus spindles** Signal(s) to channel (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The current NC program is stopped immediately, the current block is not completed. Distances-to-go are not completed until a start signal is given. The axes and the spindles are stopped. They are brought to a controlled stop. The program status switches to stopped, the channel status switches to interrupted.	
Signal state 0 or signal transition 1 ——> 0	No effect	
Signal irrelevant for ...	Channel status reset Program status aborted	
Special cases, errors, ...	All axes and spindles that were not caused to move by a program or program block (e.g. axes are moved by traversing keys on machine control panel) are not decelerated to zero speed with “NC stop axes plus spindles”. If the NC program has been stopped, auxiliary functions, for example, can be put into the memory. They are then executed on the next NC start. On NC start the program is continued at the point of interruption. If auxiliary functions were put into the memory when the program was interrupted, only these become active on the first NC start. On the second NC start the program is then continued. The signal “NC stop axes plus spindles” must be pending for at least on PLC cycle. If data were transferred to the NC after “NC stop axes plus spindles” (e.g. tool offset), these are calculated immediately on NC start. Spindles are not stopped on MD 35040: SPIND_ACTIVE_AFTER_RESET=1.  1 Signal NC Stop axes 2 Signal NC Start 3 Program running 4 Axis running 5 Spindle running 6 Block processed	
Related to	IS “NC Stop at block limit” (DB21, ... DBX7.2) IS “NC Stop” (DB21, ... DBX7.3) IS “Program status stopped” (DB21, ... DBX35.2) IS “Channel status interrupted” (DB21, ... DBX35.6)	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX7.7** Data block	**Reset** Signal(s) to channel (PLC ——> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The channel is reset. The initial settings are set (e.g. G functions). The alarms for the channel are cleared if they are not POWER ON alarms. The Reset signal must be issued by the PLC (e.g. through gating with the Reset key on the MCP). The signal is only evaluated by the selected channel. The program status changes to aborted, the channel status changes to “Channel status Reset”.	
Signal state 0 or signal transition 1 ——> 0	No effect	
Related to	IS “Mode group Reset” (DB11, ... DBX0.7) IS “Channel status Reset” (DB21, ... DBX35.7)	

DB21, ... **DBX24.5** Data block	**M01 selected** Signal(s) from channel (MMC ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Program control Activate M01 has been selected on the operator interface. This does not activate the function.	
Signal state 0 or signal transition 1 ——> 0	Program control Activate M01 has not been selected via the operator interface.	
Related to	IS “Activate M01” (DB11, ... DBX0.5) IS “M00/M01 active” (DB21, ... DBX33.5)	

DB21, ... **DBX25.7** Data block	**Program test selected** Signal(s) from channel (MMC ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: yes
Signal state 1 or signal transition 0 ——> 1	Program control Program test has been selected via the operator interface. This does not activate the function.	
Signal state 0 or signal transition 1 ——> 0	Program control Program test has not been selected via the operator interface.	
Related to	IS “Activate program test” (DB21, ... DBX1.7) IS “Program test active” (DB21, ... DBX33.7)	

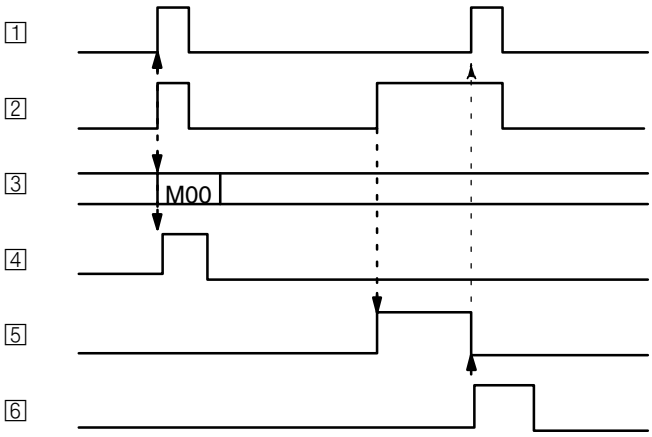
© Siemens AG, 2003. All rights reserved

DB21, ... **DBX31.0–31.2**	**RMNOTDEF**	
DB21, ... **DBX31.0–31.2**	**RMB**	
DB21, ... **DBX31.0–31.2**	**RMI**	
DB21, ... **DBX31.0–31.2**	**RME**	
DB21, ... **DBX31.0–31.2** Data block	**RMN** Signal(s) to channel (PLC ——> NCK) as well as (MMC ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.3
Signal state 1	IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) can be used to select one of the functions for the reapproach point RMB, RMI, RME or RMN based on the following codes: DB21, ... DBX31.0–31.2 = 1 RMB Reapproach at start of block DB21, ... DBX31.0–31.2 = 2 RMI Reapproach at interruption point. DB21, ... DBX31.0–31.2 = 3 RME Reapproach at end of block DB21, ... DBX31.0–31.2 = 4 RMN Reapproach at next point on path	
Signal state 0	DB21, ... DBX31.0–31.2 = 0 RMNOTDEF The REPOSMode is not redefined. No settings are overwritten and the current program is valid.	
Special cases, errors, ...	IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) affects the path as a whole. The path axes cannot be changed individually. The behavior of the other axis types can be changed individually with IS “REPOSDELAY” (DB31, ... DBX10.0).	
Related to	IS “REPOSDELAY” (DB31, ... DBX10.0)	

DB21, ... **DBX31.4** Data block	**REPOSMODEEDGE** Signal(s) to channel (PLC ——> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.3
Signal state 1 or signal transition 0 ——> 1	The level signals of IS “REPOSPASTHMODE0–2 (DB21, ... DBX31.0–31.2) and IS “REPOSDELAY” (DB31, ... DBX10.0) are adopted at the NC. The levels relate to the current block in the main run.	
Signal state 0 or signal transition 1 ——> 0	The level signals of IS “REPOSPASTHMODE0–2 (DB21, ... DBX31.0–31.2) and IS “REPOSDELAY (DB31, ... DBX10.0) are aborted at the NC.	
Application example(s)	To manipulate the SERUPRO approach, JOG can be temporarily shifted to the following areas while SERUPRO is in progress: Between “Search target found” and “Start SERUPRO ASUB” From SERUPOASUP stops automatically before REPOS” to Target block deactivated.	
Related to	IS “REPOSMODEEDGEACKN” (DB21, ... DBX319.0)	

DB21, ... **DBX32.3** Data block	**Action block active** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The action block is being executed.	
Signal state 0 or signal transition 1 ——> 0	No action block active.	
Fig.	The chronological sequence of interface signals is shown in Subsection 2.4.5.	
Application example(s)	Corresponding applications examples are described in Section 6.1.	
References	/BA/, “Operator’s Guide”	

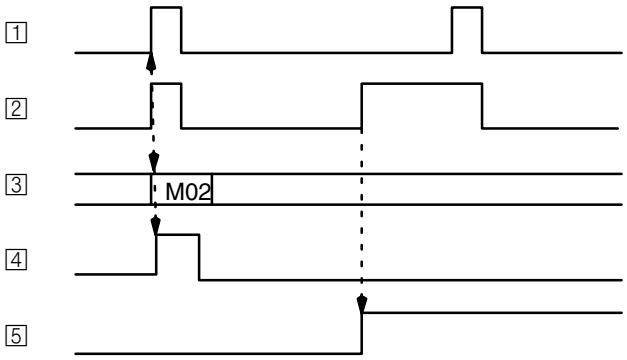
© Siemens AG, 2003. All rights reserved

DB21, ... **DBX32.5** Data block	**M00/M01 active** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The parts program block has been processed, the auxiliary functions have been output and – M00 is stored in the working memory – M01 is stored in the working memory and IS “Activate M01” is active The program status changes to “stopped”.		
Signal state 0 or signal transition 1 ——> 0	– With IS “NC Start” DB21, ... DBX7.1 – On program abort as a result of Reset		
Fig.	 1 Data transfer to working memory 2 Block processed 3 NC block with M00 4 M change signal (1 PLC cycle time) 5 IS “M00/M01 active” 6 IS “Channel state active” (even when axes are moved in JOG mode)		
Related to	IS “Activate M01” (DB21, ... DBX0.5) IS “M01 selected” (DB21, ... DBX24.5)		

DB21, ... **DBX32.6** Data block	**Last action block active** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The last action block is being executed.		
Signal state 0 or signal transition 1 ——> 0	The last action block is not being executed.		
Fig.	The chronological sequence of interface signals is shown in Subsection 2.4.5.		
Application example(s)	Corresponding applications examples are described in Section 6.1.		
References	/BA/, “Operator’s Guide”		

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX33.4** Data block	**Block search active** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The Block search function is active. It has been selected via the operator interface and started by means of IS “NC Start”.		
Signal state 0 or signal transition 1 ——> 0	Search target found.		
Fig.	The chronological sequence of interface signals is shown in Subsection 2.4.5.		
Application example(s)	The block search function makes it possible to jump to a certain block within a parts program and to starts processing of the parts program at this block. Appropriate application examples of block search at end point and block search on contour are described in Section 6.1.		
References	/BA/, “Operator’s Guide”		

DB21, ... **DBX33.5** Data block	**M02/M30 active** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	– NC block with M02 or M30 (or M17 if a subroutine has been started) has been fully executed; if traversing motions are also programmed in this block, the signal is not output until the target position is reached. – The program has been aborted by a Reset, the program status changes to “aborted”. – When the MDA mode or the machine function REF or PRESET is selected – After “Acknowledge EMERGENCY STOP” (DB 10 DBX56.2)	
Signal state 0 or signal transition 1 ——> 0	– No program end or abort – Status after power-up of control system – Start of an NC program	
Fig.	 1 Data transfer to working memory 2 Block processed 3 NC block with M02 4 M change signal (1 PLC cycle time) 5 IS “M02/M30 active”	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX33.5** Data block	**M02/M30 active** Signal(s) from channel (NCK ——> PLC)
Application example(s)	The PLC can detect the end of program processing with this signal and react appropriately.
Special cases, errors, ...	– Functions M02 and M30 have equal priority. – IS “M02/M30 active” is applied statically at the end of a program. – Not suitable for automatic follow-on functions such as workpiece counting, bar advance, etc. M02/M30 must be programmed in a separate block and the word M02/M30 or the decoded M signal used for these functions. – The last block of a program must not contain any auxiliary functions that cause read-in stop or any S values that must be operative with functions other than M02/M30.

DB21, ... **DBX33.6** Data block	**Transformation active** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The NC command TRAORI (activate transformation) is programmed in the NC parts program. This block has been processed by the NC and the transformation is now activated.	
Signal state 0 or signal transition 1 ——> 0	No transformation active.	
References	/PA/, “Programming Guide Fundamentals”	

DB21, ... **DBX33.7** Data block	**Program test active** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Program control Program test is active. Axis disable is set internally for all axes (not spindles). Therefore the machine axes do not move when a parts program block or a parts program is being processed. The axis movements are simulated on the operator interface with changing axis position values. The axis position values for the display are generated from the calculated setpoints. The parts program is processed in the normal way.	
Signal state 0 or signal transition 1 ——> 0	Program control Program test is not active.	
Related to	IS “Activate program test” (DB21, ... DBX1.7) IS “Program test selected” (DB21, ... DBX25.7)	

DB21, ... **DBX35.0** Data block	**Program status running** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The parts program has been started with IS “NC start” and is running. The running program has been halted with IS “Read-in disable” (DB21, ... DBX6.1).	
Signal state 0 or signal transition 1 ——> 0	– Program stopped with M00/M01 or NC stop or mode group change. – The block is processed if single block mode is set. – Program end reached (M02/M30) – Program abort with Reset – No block currently in memory (e.g. with MDA) – Current block cannot be processed	
Signal irrelevant for ...	The parts program has been started with IS “NC start” and is running.	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX35.0** Data block	**Program status running** Signal(s) from channel (NCK ——> PLC)
Special cases, errors, ...	IS “Program status running” does not change to 0 if workpiece machining is stopped by the following events: – Output of feed disable or spindle disable – IS “Input disable” – Feedrate override to 0% – Spindle and axis monitoring respond – Position setpoints set in NC program for axes in “follow-up mode”, for axes without “servo enable” or for “parking axes”.
Related to	IS “Readin disable” (DB21, ... DBX6.1)

DB21, ... **DBX35.1** Data block	**Program status wait** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The current program has encountered the program command WAIT_M or WAIT_E in an NC block. The Wait condition specified in the WAIT command for the channel or channels has not yet been fulfilled.	
Signal state 0 or signal transition 1 ——> 0	Program status wait is not active.	
References	/PA/, “Programming Guide Fundamentals”	

DB21, ... **DBX35.2** Data block	**Program status stopped** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The NC parts program has been stopped by “NC stop”, “NC stop axes plus spindles”, “NC stop at block limit”, programmed M00 or M01 or single block mode.	
Signal state 0 or signal transition 1 ——> 0	Program status stopped is not active	
Related to	IS “NC Stop” (DB21, ... DBX7.3) IS “NC Stop axes plus spindle” (DB21, ... DBX7.4) IS “NC Stop at block limit” (DB21, ... DBX7.2)	

DB21, ... **DBX35.3** Data block	**Program status interrupted** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	When changing from AUTOMATIC or MDA mode (when program status stopped is active) after JOG the program status switches to interrupted. The program can be continued at the point of interruption in AUTOMATIC or MDA mode when NC start is operated.	
Signal state 0 or signal transition 1 ——> 0	Program status interrupted is not active.	
Special cases, errors, ...	IS “Program status interrupted” signifies that the parts program can be continued after an NC start.	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX35.4** Data block	**Program status aborted** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The program has been selected but not started or the current program was aborted with Reset.		
Signal state 0 or signal transition 1 ——> 0	Program status interrupted is not active.		
Related to	IS “Reset” (DB21, ... DBX7.7)		

DB21, ... **DBX35.5** Data block	**Channel status active** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	In this channel – a parts program is currently being processed in AUTOMATIC or MDA or – at least one axis is being traversed in JOG mode.		
Signal state 0 or signal transition 1 ——> 0	“Channel status interrupted” or “Channel status Reset” is active.		

DB21, ... **DBX35.6** Data block	**Channel status interrupted** Signal(s) from channel (NCK –> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The NC program in AUTOMATIC or MDA or a traversing movement in JOG has been interrupted by “NC stop”, “NC stop axes plus spindles”, “NC stop at block limit”, programmed M00 or M01 or single-block mode. After an NC start the parts program or the interrupted traversing movement can be continued.		
Signal state 0 or signal transition 1 ——> 0	“Channel status active” or “Channel status Reset” is active.		

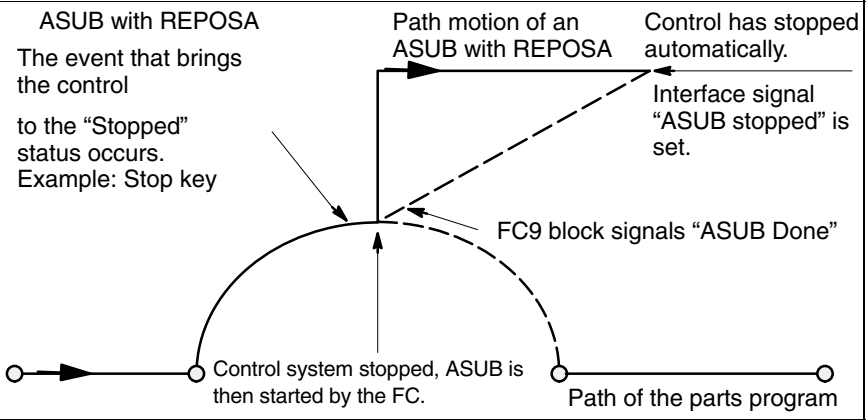
DB21, ... **DBX35.7** Data block	**Channel status reset** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The signal changes to 1 as soon as the channel goes into Reset, i.e. no processing taking place.		
Signal state 0 or signal transition 1 ——> 0	The signal changes to 0 if anything if processing is taking place in the channel, e.g.: – processing of a parts program – block search – TEACH IN active – overstore active etc.		

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX36.4** Data block	**Interrupt processing active** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation:	Signal(s) updated:	Signal(s) valid from SW: 4.1
Signal state 1 or signal transition 0 ——> 1	One or several channels in the mode group are not in the desired operating state as the result of an active interrupt routine. The signal is not set if an interrupt routine is running in a program operation mode.	
Signal state 0 or signal transition 1 ——> 0	All channels are operating in the desired mode.	
Related to	MD BAG_MASK	

DB21, ... **DBB208–267** Data block	**Active G function of groups 1 to 60** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state <>0	A G function or mnemonic identifier of the G group is active. The active G function can be determined from the value (starting with 1). 1 = 1st G function of the G group 2 = 2nd G function of the G group 3 = 3rd G function of the G group : Please refer to the Programming Guide for a list of possible G groups and their functions.	
Signal state = 0	No G function or G group mnemonic identifier is active.	
Application example(s)	The active G group is stored in binary code in the relevant byte. The following place value applies: Bit 7 6 5 4 3 2 1 0 place value 128 64 32 16 8 4 2 1 Example: G90 0 1 0 1 1 0 1 0	
Special cases, errors, ...	In contrast to auxiliary functions, G functions are not have to be acknowledged when output to the PLC, i.e. processing of the parts program is continued immediately after the G function has been output.	
References	/PA/, “Programming Guide Fundamentals”	

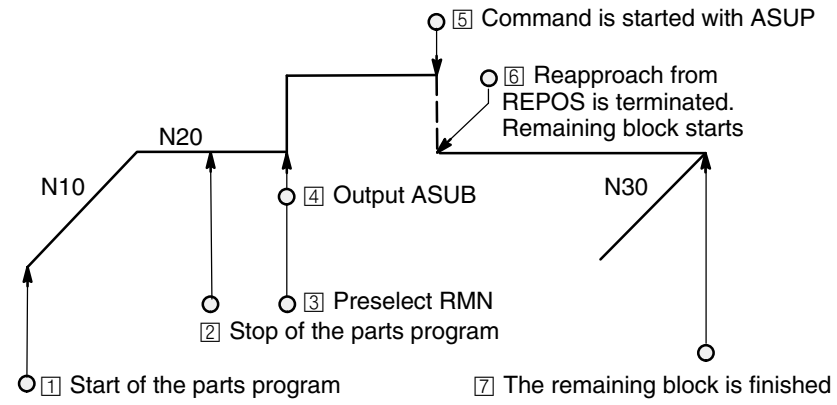
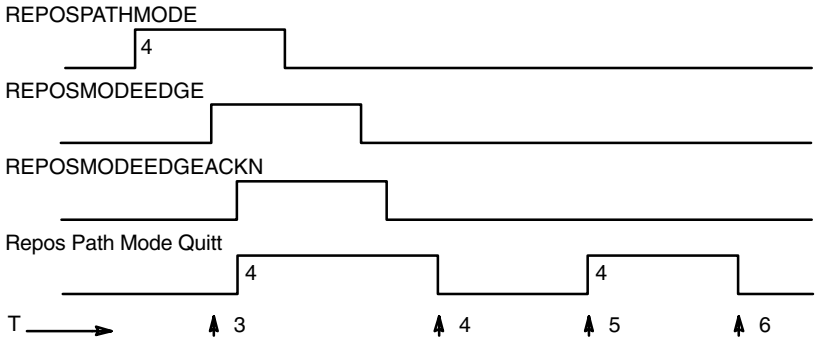
© Siemens AG, 2003. All rights reserved

DB21, ... **DBX318.0** Data block	**ASUB is stopped** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 5.3	
Signal state 1 or signal transition 0 ——> 1	The signal is set to 1 if the control stops automatically prior to the end of the ASUB. The IS “ASUB is stopped” is only supplied in the case “Interrupt in a program mode and channel status stopped”.		
Signal state 0 or signal transition 1 ——> 0	Start and Reset will set the IS “ASUB is stopped” to 0.		
Fig.: Typical sequence of an ASUB with REPOSA			
ASUB with REPOSA is triggered in the status AUTOMATIC mode stopped	If the interrupt program is completed with REPOSA, the following sequence is typical: – Pressing the Stop key, StopAll key or an alarm will stop the parts program. – The control system will change to the program status “Stopped”. – The PLC triggers an ASUB via the block FC9. – Before reapproaching the contour, the control system stops and changes to the program status “Stopped”. The IS “ASUB is stopped” is set. – The user presses Start. The IS “Asup is stopped” is reset, and the reapproach movement is started. – With the end of the reapproach movement, the FC9 signal “ASUB Done” is set, and the path of the interrupted parts program is continued.		

DB21, ... **DBX318.1** Data block	**Block search via Program Test is active** Signal(s) from channel (NCK ——> PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.1	
Signal state 1 or signal transition 0 ——> 1	The PI service for _N_FINDBL, parameter 5, will enable SERUPRO. Time for the active signal status: In the internal mode Program test, the NC runs until the target block of the search block is changed in the main run and the program status changes to “stopped”.		
Signal state 0 or signal transition 1 ——> 0	The NCK will stop at the beginning of the target block and will internally deselect Program test again. NCK will now display the Stop condition “Search target” found.		
Special cases, errors, ...	The SERUPRO function may only be enabled in the “AUTOMATIC” mode in the program status “Aborted”.		

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX319.0** Data block	**REPOSMODEEDGEACKN** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.3
Signal state 1 or signal transition 0 ——> 1	The interface signal detected by the NCK, IS “REPOSMODEEDGE” (DB21, ... DBX31.4), is acknowledged with IS “REPOSMODEEDGEACKN” (DB21, ... DBX319.0) if level signals from IS “REPOSMODE0–2” (DB21, ... DBX31.0–31.2) and IS “REPOSDELAY” (DB31, ... DBX10.0) were transferred to the NC. The levels relate to the current block in the main run.	
Signal state 0 or signal transition 1 ——> 0	SERUPRO–ASUP stops automatically before REPOS and IS “REPOSMODEEDGE” (DB21, ... DBX31.4) is not active on approach SERUPRO.	
Related to	IS “REPOSMODEEDGE” (DB21, ... DBX31.4)	

DB21, ... **DBX319.1–319.3** Data block	**Repos Path Mode Quitt0–2** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.4
Signal state 1	IS “Repos Path Mode Ack0–2” (DB21, ... DBX31.0–319.3) can be used to select one of the functions for the reapproach point RMB, RMI, RME or RMN based on the following codes: DB21, ... DBX319.1–319.3 = 1 RMB Reapproach at start of block DB21, ... DBX319.1–319.3 = 2 RMI Reapproach at interruption point. DB21, ... DBX319.1–319.3 = 3 RME Reapproach at end of block DB21, ... DBX319.1–319.3 = 4 RMN Reapproach at next point on path	
Signal state 0	DB21, ... DBX319.1–319.3 = 0 RMNOTDEF The non-defined ReposMode is acknowledged to the PLC.	
Fig.: Sample sequence for REPOS acknowledgements In the parts program		
Signal timing for acknowledgement by the NCK		
Related to	IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) IS “REPOSMODEEGGE” (DB21, ... DBX31.4) IS“REPOSMODEEGGEACKN” (DB21, ... DBX319.0) IS “Repos Delay Ack” (DB31, ... DBX70.2	
References	Section for block search, “Reapproach after SERUPRO target found”	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX319.5** Data block	**Repos DEFERAL Chan** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.4
Signal state 1 or signal transition 0 ——> 1	All axes that are controlled by this channel are either not a REPOS shift or their REPOS shifts are invalid.	
Signal state 0 or signal transition 1 ——> 0	Miscellaneous	
Related to	IS “Repos offset” (DB31, ... DBX70.0)	

© Siemens AG, 2003. All rights reserved

5.3 Axis-specific signals

DB 31, ... **DBX10.0** Data block	**REPOSDELAY** Signal(s) to channel (PLC ——> NCK) as well as (MMC ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.3
Signal state 1 or signal transition 0 ——> 1	The REPOS offset for this axis is not applied until it is next programmed.	
Signal state 0 or signal transition 1 ——> 0	The REPOS offset for this axis is not active.	
Special cases, errors, ...	Please note the following special cases: – If axes are always programmed absolutely, then IS “REPOSDELAY” has the same significance for operations with or without offset. – Machine axes which form a path are not affected by IS “REPOSDELAY”.	
Related to	IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) IS “Repos Delay Ack” (DB31, ... DBX70.2) IS “REPOSDELAY” (DB31, ... DBX72.0) Signal (MMC ——> PLC)	

DB 31, ... **DBX70.0** Data block	**REPOS offset** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.3
Signal state 1 or signal transition 0 ——> 1	A REPOS offset must be applied for the appropriate axis.	
Signal state 0 or signal transition 1 ——> 0	No REPOS offset need be applied for the appropriate axis.	
Related to	IS “REPOS offset valid” (DB31, ... DBX70.1)	

DB 31, ... **DBX70.1** Data block	**REPOS offset valid** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.3
Signal state 1 or signal transition 0 ——> 1	The range of validity of the REPOS offset is indicated by value 1. The REPOS offset calculation is valid.	
Signal state 0 or signal transition 1 ——> 0	A value of zero indicates that the REPOS offset calculation is invalid.	
Application example(s)	Update REPOS offset in range of validity: The axis can be moved in JOG mode between the SERUPRO end and Start. The operator moves the REPOS offset to a value of zero.	
Related to	IS “Repos offset” (DB31, ... DBX70.0)	

DB 31, ... **DBX70.2** Data block	**REPOS Delay Ack** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.4
Signal state 1 or signal transition 0 ——> 1	The REPOS shift of the axis to be programmed next is acknowledged. This signal behaves like IS “Repos Path Mode Quitt0–2” (DB21, ... DBX319.1–319.3).	
Signal state 0 or signal transition 1 ——> 0	The value zero acknowledges that the REPOS shift is not active for this axis. This signal is cancelled on activation of the remaining block.	
Application example(s)	See sequence of REPOS acknowledgments in the parts program and signal sequence of IS “Repos Path Mode Ack0–2” (DB21, ... DBX319.1–319.3).	
Related to	IS “REPOSDELAY” (DB31, ... DBX10.0)	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX72.0** Data block	**REPOSDELAY** Signal(s) from channel (MMC ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.4
Signal state 1 or signal transition 0 ——> 1	The REPOS offset for this axis is not applied until it is next programmed.	
Signal state 0 or signal transition 1 ——> 0	The REPOS offset for this axis is not active.	
Special cases, errors, ...	Please note the following special cases: – If axes are always programmed absolutely, then IS “REPOSDELAY” has the same significance for operations with or without offset. – Machine axes which form a path are not affected by IS “REPOSDELAY”.	
Related to	IS “REPOSPATHMODE0–2” (DB21, ... DBX31.0–31.2) IS “REPOSDELAY” (DB31, ... DBX10.0) Signal (PLC ——> NCK) IS “Repos Delay Ack” (DB31, ... DBX70.2)	

DB 31, ... **DBX76.4** Data block	**Path axis** Signal(s) from channel (NCK ——> PLC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 6.4
Signal state 1 or signal transition 0 ——> 1	The current state of a block to be executed for a path axis is displayed with value 1.	
Signal state 0 or signal transition 1 ——> 0	Once the SERUPRO operation is ended with search target found, IS “Path axis” with value zero refers to the target block.	

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

6 Example

A number of different, context-specific examples of block searches can be found in the the relevant sections of this document.

■

7 Data Fields, Lists

Reference For machine data not explained in this Description of Functions, the corresponding Description of Functions is indicated in the Reference column (e.g. /K1/ means that this machine data is described in Part K1 of the Description of Functions)

7.1 General machine data

7.1.1 General machine data for operator panel

Number		Identifier	Name	Doc. Refer to
MMC machine data ($MM_ ...)				
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBEDDED (SW 6.1 and higher)	
9421	9421	MA_AXES_SHOW_GEO_FIRST	Display geo axes of channel first	IM2/IM4
9422	9422	MA_PRESET_MODE	PRESET / basic offset in JOG. 0: No SK, 1: PRESET, 2: Preset actual value memory 3: Set actual value in currently active frame	IM2/IM4
9423	9423	MA_MAX_SKP_LEVEL	This data defines the number of program levels which can be skipped in the parts program with “/” .	IM2/IM4

© Siemens AG, 2003. All rights reserved

7.1.2 General machine data mode group/program operation

Number	Identifier	Name	Refer-ence
General ($MN_ ...)			
10010	ASSIGN_CHAN_TO_MODE_GROUP[n]	Channel valid in mode group [channel no.]: 0, 1	
10280	PROG_FUNCTION_MASK	Compare commands “>” and “<” compatible to SW 6.3	
10617	FRAME_SAVE_MASK	Behavior of frames on subroutine return jump with SAVE (SW 6.1 and higher)	
10700	PREPROCESSING_LEVEL	Program preprocessing level	V2
10702	IGNORE_SINGLEBLOCK_MASK	Prevent single block stop	
10707	PROG_TEST_MASK	Program test modes (SW 6.1 and higher)	
10708	SERUPRO_MASK	Block search modes (SW 6.1 and higher)	
10710	PROG_SD_RESET_SAVE_TAB[n]	Setting data to be updated [Index]: 0...9	
10713	M_NO_FCT_STOPRE	M function with preprocessing stop (SW 6.3 and higher)	H2
10715	M_NO_FCT_CYCLE	M function to be replaced by SR	FBFA
10716	M_NO_FCT_CYCLE_NAME	SR name for M function replacement	FBFA
10717	T_NO_FCT_CYCLE_NAME	Name of tool change cycle for T function (SW 5.2 and higher)	FBFA
10718	M_NO_FCT_CYCLE_PAR	M function replacement with parameter transfer (SW 6.3 and higher)	FBFA
10719	T_NO_FCT_CYCLE_MODE	Parameterization of T function replacement (SW 6.4 and higher)	FBFA
11450	SEARCH_RUN_MODE	Control system response and output of the spindle auxiliary functions after block search (SW 5.3 and higher)	
11470	REPOS_MODE_MASK	Repositioning properties (SW 6.3 and higher)	
11600	BAG_MASK	Mode group response to ASUB	
11602	ASUP_START_MASK	Ignore stop conditions for ASUB	
11604	ASUP_START_PRIO_LEVEL	Priorities for “ASUP_START_MASK effective”	
11610	ASUP_EDITABLE	Activation of a user ASUB for RET/REPOS	
11612	ASUP_EDIT_PROTECTION_LEVEL	Protection level of user-specific ASUB for RET and/or REPOS	
17200	GMMC_INFO_NO_UNIT	Global HMI info (without physical unit)	
17201	GMMC_INFO_NO_UNIT_STATUS	Global HMI status info (without physical unit)	
18360	MM_EXT_PROG_BUFFER_SIZE	FIFO buffer size for one program level	
18362	MM_NUM_EXT_PROG	Number of external program levels (DRAM)	

© Siemens AG, 2003. All rights reserved

7.2 Channel-specific machine data

7.2.1 Basic machine data of channel

Number	Identifier	Name	Reference
Channel-specific ($MC_ ...)			
20000	CHAN_NAME	Channel name	
20050	AXCONF_GEOAX_ASSIGN_TAB[n]	Assignment between geometry axis and channel axis [GEOaxis no.]: 0...2	K2
20060	AXCONF_GEOAX_NAME_TAB[n]	Geometry axis name in channel [GEOaxis no.]: 0...2	K2
20070	AXCONF_MACHAX_USED[n]	Machine axis number valid in channel [channel axis no.]: 0...7	K2
20080	AXCONF_CHANAX_NAME_TAB[n]	Channel axis name in channel [channel axis no.]: 0...7	K2
20090	SPIND_DEF_MASTER_SPIND	Initial setting of master spindle in channel	S1
20100	DIAMETER_AX_DEF	Geometry axis with transverse axis function	P1
20108	PROG_EVENT_MASK	Event-driven program calls (SW 6.1)	
20109	PROG_EVENT_MASK_PROPERTIES	Properties of ProgEvents (SW 6.3 and higher)	
20114	MODESWITCH_MASK	Setting for REPOS	
20116	IGNORE_INHIBIT_ASUP	Process user ASUBs completely in spite of read-in disable	
20117	IGNORE_SINGLEBLOCK_ASUP	Process user ASUBs completely in spite of single-block processing	
20160	CUBIC_SPLINE_BLOCKS	Number of blocks for C spline	
20170	COMPRESS_BLOCK_PATH_LIMIT	Maximum traversing length of NC block for compression	
20210	CUTCOM_CORNER_LIMIT	Max. angle for intersection calculation with tool radius compensation	W1
20220	CUTCOM_MAX_DISC	Maximum value with DISC	W1
20230	CUTCOM_CURVE_INSERT_LIMIT	Maximum angle for intersection calculation with tool radius compensation	W1
20240	CUTCOM_MAXNUM_CHECK_BLOCKS	Blocks for predictive contour calculation with tool radius compensation	W1
20250	CUTCOM_MAXNUM_DUMMY_BLOCKS	Max. no. of dummy blocks with no traversing movements	W1
20270	CUTTING_EDGE_DEFAULT	Basic setting of tool cutting edge without programming	W1
20400	LOOKAH_USE_VELO_NEXT_BLOCK	Look Ahead to programmed following block velocity	B1
20430	LOOKAH_NUM_OVR_POINTS	No. of override switch points for Look Ahead	B1
20440	LOOKAH_OVR_POINTS[n]	Compensation switch caracteristics for Look Ahead [characteristic no.]: 0...1	B1
20500	CONST_VELO_MIN_TIME	Minimum time with constant velocity	B2
20600	MAX_PATH_JERK	Path-related maximum jerk	B2
20610	ADD_MOVE_ACCEL_RESERVE	Acceleration reserve for overlaid movements	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
20700	REFP_NC_START_LOCK	NC start disable without reference point	R1
20750	ALLOW_GO_IN_G96	G0 logic in G96	V1
20800	SPF_END_TO_VDI	Subprogram end to PLC	H2
21000	CIRCLE_ERROR_CONST	Circle end point monitoring constant	
21010	CIRCLE_ERROR_FACTOR	Circle end point monitoring factor	
21100	ORIENTATION_IS_EULER	Angle definition for orientation programming	F2
21110	X_AXIS_IN_OLD_X_Z_PLANE	Coordinate system for automatic Frame definition	K2
21200	LIFTFAST_DIST	Traversing path for fast retraction from the contour	
21210	SETINT_ASSIGN_FASTIN	NCK input bytes for interrupts	
21202	LIFTFAST_WITH_MIRROR	Lift fast with mirror	
21250	START_INDEX_R_PARAM	Number of first channel-specific R parameter	S7

7.2.2 Block search

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
20128	COLLECT_TOOL_CHANGE	Collect tool changes during block search	
22600	SERUPRO_SPEED_MODE	Velocity at block search type 5 (SW 6.1 and higher)	
22601	SERUPRO_SPEED_FACTOR	Velocity at block search type 5 (SW 6.1 and higher)	
22621	ENABLE_START_MODE_MASK_PRT	Enables MD 22620: START_MODE_MASK_PRT for SERUPRO search run (SW 6.3 and higher)	
22622	DISABLE_PLC_START	Permit part program start via PLC (SW 6.4 and higher)	

7.2.3 Reset response

Number	Identifier	Name	Refer-ence
Channelspecific ($MC_ ...)			
20110	RESET_MODE_MASK	Definition of control system initial setting on RESET	
20112	START_MODE_MASK	Determination of the basic control setting at NC start after boot and at RESET	
20118	GEOAX_CHANGE_RESET	Allow automatic geometry axis change	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
20120	TOOL_RESET_VALUE	Tool whose tool length compensation is selected during power-up (Reset/parts program end)	
20121	TOOL_PRESEL_RESET_VALUE	Preselected tool whose length compensation is selected in ramp-up (Reset/TP End)	
20130	CUTTING_EDGE_RESET_VALUE	Determination of the tool cutting edge in ramp-up (Reset/TP End)	
20140	TRAFO_RESET_VALUE	Active transformation on RESET:	
20150	GCODE_RESET_VALUES[n]	Initial setting of G groups [G group no.]: 0...59	
20152	GCODE_RESET_MODE	G code basic setting at RESET	
20156	MAXNUM_GCODES_EXT	Reset response of the external G groups (SW 6.3 and higher)	FBFA
22620	START_MODE_MASK_PRT	Initial setting on special NC Start after booting and on RESET (SW 6.3 later)	

7.2.4 Auxiliary function settings of the channel

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
22000	AUXFU_ASSIGN_GROUP[n]	Auxiliary function group [aux. func. no. in channel]: 0...49	H2
22010	AUXFU_ASSIGN_TYPE[n]	Auxiliary function type [aux. func. no. in channel]: 0...49	H2
22020	AUXFU_ASSIGN_EXTENSION[n]	Auxiliary function extension [aux. func. no. in channel]: 0...49	H2
22030	AUXFU_ASSIGN_VALUE[n]	Auxiliary function value [aux. func. no. in channel]: 0...49	H2
22200	AUXFU_M_SYNC_TYPE	Output timing for M functions	H2
22210	AUXFU_S_SYNC_TYPE	Output timing of S functions	H2
22220	AUXFU_T_SYNC_TYPE	Output timing of T functions	H2
22230	AUXFU_H_SYNC_TYPE	Output timing for H functions	H2
22240	AUXFU_F_SYNC_TYPE	Output timing of F functions	H2
22250	AUXFU_D_SYNC_TYPE	Output timing of D functions	H2
22260	AUXFU_E_SYNC_TYPE (available soon)	Output timing of E functions.	–
22300	AUXFU_AT_BLOCK_SEARCH_END	Output of auxiliary functions after block search	H2
22400	S_VALUES_ACTIVE_AFTER_RESET	S function active after RESET	S1
22410	F_VALUES_ACTIVE_AFTER_RESET	F function active after RESET	V1
22500	GCODE_OUTPUT_TO_PLC	G functions to PLC	
22510	GCODE_GROUPS_TO_PLC	G codes that are output to the NCK/PLC interface on block change/RESET	
22550	TOOL_CHANGE_MODE	New tool offset for M function	W1
22560	TOOL_CHANGE_M_CODE	M function for tool change	W1

© Siemens AG, 2003. All rights reserved

7.2.5 Transformation definitions in channel

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
24100	TRAFO_TYPE_1	Definition of transformation 1 in channel	F2
24110	TRAFO_AXES_IN_1[n]	Axis assignment for transformation [axis index]: 0...7	F2
24120	TRAFO_GEOAX_ASSIGN_TAB_1[n]	Assignment between GEO axis and channel axis for transformation 1 [GEO axis no.]: 0...2	F2, M1
24200	TRAFO_TYPE_2	Definition of transformation 2 in channel	F2
24210	TRAFO_AXES_IN_2[n]	Axis assignment for transformation 2 [axis index]: 0...7	F2
24220	TRAFO_GEOAX_ASSIGN_TAB_2[n]	Assignment between GEO axis and channel axis for transformation 2 [GEO axis no.]: 0...2	F2, M1
24300	TRAFO_TYPE_3	Definition of transformation 3 in channel	F2
24310	TRAFO_AXES_IN_3[n]	Axis assignment for transformation 3 [axis index]: 0...7	F2
24320	TRAFO_GEOAX_ASSIGN_TAB_3[n]	Assignment between GEO axis and channel axis for transformation 3 [GEO axis no.]: 0...2	F2, M1
24400	TRAFO_TYPE_4	Definition of transformation 4 in channel	F2
24410	TRAFO_AXES_IN_4[n]	Axis assignment for transformation 4 [axis index]: 0...7	F2
24420	TRAFO_GEOAX_ASSIGN_TAB_4[n]	Assignment between GEO axis and channel axis for transformation 4 [GEO axis no.]: 0...2	F2, M1
24430	TRAFO_TYPE_5	Definition of transformation 5 in channel	F2
24432	TRAFO_AXES_IN_5[n]	Axis assignment for transformation [axis index]: 0...7	F2
24434	TRAFO_GEOAX_ASSIGN_TAB_5[n]	Assignment between GEO axis and channel axis for transformation 5 [GEO axis no.]: 0...2	F2, M1
24440	TRAFO_TYPE_6	Definition of transformation 6 in channel	F2
24442	TRAFO_AXES_IN_6[n]	Axis assignment for transformation 6 [axis index]: 0...7	F2
24444	TRAFO_GEOAX_ASSIGN_TAB_6[n]	Assignment between GEO axis and channel axis for transformation 6 [GEO axis no.]: 0...2	F2, M1
24450	TRAFO_TYPE_7	Definition of transformation 7 in channel	F2
24452	TRAFO_AXES_IN_7[n]	Axis assignment for transformation 7 [axis index]: 0...7	F2
24454	TRAFO_GEOAX_ASSIGN_TAB_7[n]	Assignment between GEO axis and channel axis for transformation 7 [GEO axis no.]: 0...2	F2, M1
24460	TRAFO_TYPE_8	Definition of transformation 8 in channel	F2
24462	TRAFO_AXES_IN_8[n]	Axis assignment for transformation 8 [axis index]: 0...7	F2
24464	TRAFO_GEOAX_ASSIGN_TAB_8[n]	Assignment between GEO axis and channel axis for transformation 8 [GEO axis no.]: 0...2	F2, M1
24500	TRAFO5_PART_OFFSET_1[n]	Offset vector of 5axis transformation 1 [axis no.]: 0...2	F2
24510	TRAFO5_ROT_AX_OFFSET_1[n]	Position offset of rotary axes 1/2 for 5-axis transformation 1 [axis no.]: 0...1	F2

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
24520	TRAFO5_ROT_SIGN_IS_PLUS_1[n]	Sign of rotary axis 1/2 for 5axis transformation 1 [axis no.]: 0...1	F2
24530	TRAFO5_NON_POLE_LIMIT_1	Definition of pole limit for 5axis transformation 1	F2
24540	TRAFO5_POLE_LIMIT_1	Pole end angle tolerance for interpolation for 5-axis transformation 1	F2
24550	TRAFO5_BASE_TOOL_1[n]	Vector of base tool with activation of 5-axis transformation 1 [axis no.]: 0...2	F2
24560	TRAFO5_JOINT_OFFSET_1[n]	Vector of kinematic offset for 5-axis transformation 1 [axis no.]: 0...2	F2
24600	TRAFO5_PART_OFFSET_2[n]	Offset vector of 5-axis transformation 2 [axis no.]: 0...2	F2
24610	TRAFO5_ROT_AX_OFFSET_2[n]	Position offset of rotary axes 1/2 for 5-axis transformation 2 [axis no.]: 0...1	F2
24620	TRAFO5_ROT_SIGN_IS_PLUS_2[n]	Sign of rotary axis 1/2 for 5-axis transformation 2 [axis no.]: 0...1	F2
24630	TRAFO5_NON_POLE_LIMIT_2	Definition of pole limit for 5-axis transformation 2	F2
24640	TRAFO5_POLE_LIMIT_2	Pole end angle tolerance for interpolation for 5-axis interpolation 2	F2
24650	TRAFO5_BASE_TOOL_2[n]	Vector of base tool with activation of 5-axis transformation 2 [axis no.]: 0...2	F2
24660	TRAFO5_JOINT_OFFSET_2[n]	Vector of kinematic offset for 5-axis transformation 2 [axis no.]: 0...2	F2

7.2.6 Channel-specific memory settings

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
25000	REORG_LOG_LIMIT	Percentage of IPO buffer for log file enable	S7
28000	MM_REORG_LOG_FILE_MEM	Memory size for REORG (DRAM)	S7
28010	MM_NUM_REORG_LUD_MODULES	Number of blocks for local user variables for REORG (DRAM)	S7
28020	MM_NUM_LUD_NAMES_TOTAL	Number of local user variables (DRAM)	S7
28030	MM_NUM_LUD_NAMES_PER_PROG	Number of local user variables per program (DRAM)	S7
28040	MM_LUD_VALUES_MEM (available soon)	Memory size for local user variables (DRAM)	–

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Reference
Channel-specific ($MC_ ...)			
28050	MM_NUM_R_PARAM	Number of channel-specific R parameters (SRAM)	S7
28060	MM_IPO_BUFFER_SIZE	Number of NC blocks in IPO buffer (DRAM)	S7
28070	MM_NUM_BLOCKS_IN_PREP (available soon)	Number of blocks for block preparation (DRAM)	–
28080	MM_NUM_USER_FRAMES	Number of settable Frames (SRAM)	S7
28090	MM_NUM_CC_BLOCK_ELEMENTS	Number of block elements for compile cycles (DRAM)	S7
28100	MM_NUM_CC_BLOCK_USER_MEM	Size of block memory for compile cycles (DRAM)	S7
28400	MM_ABSBLOCK	Dimension basic block display	
28402	MM_ABSBLOCK_BUFFER[2]	Dimension size of upload buffer	
28500	MM_PREP_TASK_STACK_SIZE	Stack size of preparation task (DRAM)	S7
28510	MM_IPO_TASK_STACK_SIZE	Stack size of IPO task (DRAM)	S7

7.2.7 Program runtime and workpiece counter (SW 5.2 and higher)

Number	Identifier	Name	Reference
Channel-specific ($MC_ ...)			
27860	PROCESSTIMER_MODE	Activate the runtime measurement	
27880	PART_COUNTER	Activate the workpiece counter	
27882	PART_COUNTER_MCODE[ ]	Workpiece counting via M command	

7.3 Axis/spindle-specific machine data

Number	Identifier	Name	Reference
Axis/spindle-specific($MA_ ...)			
30550	AXCONF_ASSIGN_MASTER_CHAN	Reset position of channel for axis change	K5
30600	FIX_POINT_POS	Fixed value positions of axes with G75	
33100	COMPRESS_POS_TOL	Maximum deviation with compensation	

© Siemens AG, 2003. All rights reserved

7.4 Channel-specific setting data

Number	Identifier	Name	Refer-ence
Channel-specific ($SC_ ...)			
42000	THREAD_START_ANGLE	Start angle for thread	V1
42010	THREAD_RAMP_DISP	Starting and deceleration distance of feed axis in thread cutting	V1
42100	DRY_RUN_FEED	Dry run feed	V1
42200	SINGLEBLOCK2_STOPRE	Activate debug mode for SBL2 (SW 6.3 and higher)	
42444	TARGET_BLOCK_INCR_PROG	Set up mode after block search with calculation (SW 4.3 and higher)	
42700	EXT_PROG_PATH	Name of an external program path for subroutine call EXTCALL	
42750	ABSBLOCK_ENABLE	Enable basic block display (SW 6.4 and higher)	
42990	MAX_BLOCKS_IN_IPOBUFFER	Control of max. number of blocks in interpolation buffer	

© Siemens AG, 2003. All rights reserved

7.5 Interface signals

In the tables below a crossreference always indicates where the interface signal or group of interface signals is described (e.g. /R1/ for Description of Functions R1). Signals which are described in detail in Chapter 5 and intuitive signals do not have a reference.

You will find a complete list of all interface signals in
References: /LIS/, Lists

7.5.1 Analog inputs and outputs

DB number	Byte.Bit	Name	Reference
General signals to NCK			
10	DBX56.1	EMERGENCY STOP	N2

7.5.2 DB11, ... Mode group signals

Description of interface signals

The mode group DB signals from PLC → NCK are described in Subsection 3.5.1 of this Description of Functions.
The mode group DB signals from NCK → PLC are described in Subsection 3.5.2 of this Description of Functions.

DB number	Byte.Bit	Name	Reference
PLC to NCK			
11	DBX0.0	AUTOMATIC mode	
	DBX0.1	MDA mode	
	DBX0.2	JOG mode	
	DBX0.4	Mode group switchover disable	
	DBX0.5	Mode group stop	
	DBX0.6	Mode group stop axes plus spindles	
	DBX0.7	Mode group RESET	
11	DBX1.0	TEACH IN machine function	
	DBX1.1	Machine function REPOS	
	DBX1.2	Machine function REF	

© Siemens AG, 2003. All rights reserved

DB number	Byte.Bit	Name	Refer-ence
NCK to PLC			
11	DBX4.0	Selected mode AUTOMATIC	
	DBX4.1	Selected mode MDA	
	DBX4.2	Selected mode JOG	
11	DBX5.0	TEACH IN machine function selected	
	DBX5.1	Selected machine REPOS function TEACH IN	
	DBX5.2	Selected machine function REF	
11	DBX6.0	Active mode AUTOMATIC	
	DBX6.1	Active mode MDA	
	DBX6.2	Active mode JOG	
	DBX6.3	Mode group ready	
	DBX6.7	All channels in Reset state	
11	DBX7.0	TEACH IN machine function active	
	DBX7.1	Active machine function REPOS	
	DBX7.2	Active machine function REF	

7.5.3 DB21, ... Channel signals

Signals at mode group from PLC to NCK

DB number	Byte.Bit	Name	Refer-ence
To channel			
21	DBX0.3	Activate DRF	H1
	DBX0.4	Activate single block	
	DBX0.5	Activate M01	
	DBX0.6	Activate dry run feed	V1
21	DBX1.0	Activate referencing	R1
	DBX1.6	PLC action ended	
	DBX1.7	Activate program test	
21	DBB2	Skip blocks /0 to /7	
21	DBX7.0	NC Start disable	
	DBX7.1	NC Start	
	DBX7.2	NC Stop at block limit	
	DBX7.3	NC Stop	
	DBX7.4	NC Stop axes plus spindles	
	DBX7.7	Reset	
21	DBX31.0–DBX31.2*	REPOSPATHMODE0–2 (SW 6.3 and higher)	
21	DBX31.4	REPOSMODEEDGE (SW 6.3 and higher)	

* These signals to the mode group are also available from HMI/MMC to NCK.

© Siemens AG, 2003. All rights reserved

Signals from mode group to PLC

DB number	Byte.Bit	Name	Ref.
From channel			
21	DBX24.3	DRF selected	H1
	DBX24.4	Single block selected	
	DBX24.5	M01 selected	
	DBX24.6	Dry run feedrate selected	V1
21	DBX25.3	Feed override for rapid traverse selected	V1
	DBX25.7	Program test selected	
21	DBB26	Skip block selected	
21	DBX27.0	Skip block selected	
21	DBX27.1	Skip block selected	
21	DBX31.6	Block skip active /8	
21	DBX31.7	Block skip active /9	
21	DBX32.3	Action block active	
	DBX32.4	Approach block active	
	DBX32.5	M00/M01 active	
	DBX32.6	Last action block active	
21	DBX33.0	Referencing active	R1
	DBX33.4	Block search active	
	DBX33.5	M02/M30 active	
	DBX33.6	Transformation active	
	DBX33.7	Program test active	
21	DBX35.0	Program status: Running	
	DBX35.1	Program status: Wait	
	DBX35.2	Program status: Stopped	
21	DBX35.3	Program status: Interrupted	
	DBX35.4	Program status: Aborted	
	DBX35.5	Channel status: disable active	
	DBX35.6	Channel status: Interrupted	
	DBX35.7	Channel status: Reset	
21	DBB 208	Number of active G function of G function group 1 (binary)	
21	DBB 209	Number of active G function of G function group 2 (binary)	
21	DBB 210	Number of active G function of G function group 3 (binary)	
21	DBB ...	Number of active G function of G function group ... (binary)	
21	DBB266	Number of active G function of G function group n1 (binary)	
21	DBB 208	Number of active G function of G function group n (binary)	
21	DBX318.0	ASUB is stopped (SW 5.3 and higher)	
21	DBX318.1	Block search via program test is active (SW 6.1 and higher)	
21	DBX319.0	REPOSMODEEDGEACKN (SW 6.3 and higher)	
21	DBX319.1–DBX319.3	Repos Path Mode Quitt0–2 (SW 6.4 and higher)	
21	DBX319.5	Repos DEFERAL Chan (SW 6.4 and higher)	

© Siemens AG, 2003. All rights reserved

7.5.4 DB 31, ... Axis-specific signals

DB number	Byte.Bit	Name	Refer-ence
To channel			
31	DBX10.0	REPOSDALAY (SW 6.3 and higher)	
From channel			
31	DBX70.0	REPOS offset (SW 6.3 and higher)	
31	DBX70.1	REPOS offset valid (SW 6.3 and higher)	
31	DBX70.2	REPOS Dalay Quitt (SW 6.4 and higher)	
31	DBX72.0	REPOSDALAY (SW 6.4 and higher)	
31	DBX76.2	Path axis (SW 6.4 and higher)	

7.6 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Axes, Coordinate Systems, Frames (K2)

1 Brief Description 1/K2/1-5

1.1 Axes 1/K2/1-5

1.2 Coordinate systems 1/K2/1-7

1.3 Frames 1/K2/1-9

2 Detailed Description 1/K2/2-13

2.1 Axes 1/K2/2-13
2.1.1 Machine axes 1/K2/2-14
2.1.2 Channel axes 1/K2/2-15
2.1.3 Geometry axes 1/K2/2-15
2.1.4 Replaceable geometry axes 1/K2/2-15
2.1.5 Special axes 1/K2/2-19
2.1.6 Path axes 1/K2/2-19
2.1.7 Positioning axes 1/K2/2-20
2.1.8 Main axes 1/K2/2-21
2.1.9 Synchronous axes 1/K2/2-22
2.1.10 Axis configuration 1/K2/2-23
2.1.11 Link axes (SW 5 and higher) 1/K2/2-26

2.2 Zeros and reference points 1/K2/2-29
2.2.1 Reference points in working space 1/K2/2-29
2.2.2 Position of coordinate systems and reference points 1/K2/2-30

2.3 Coordinate systems 1/K2/2-32
2.3.1 Machine coordinate system (MCS) 1/K2/2-34
2.3.2 Basic coordinate system (BCS) 1/K2/2-35
2.3.3 Additive offsets 1/K2/2-36
2.3.4 Basic zero system (BZS) 1/K2/2-38
2.3.5 Settable zero system (SZS) 1/K2/2-40
2.3.6 Configurable SZS (SW 6.4 and higher) 1/K2/2-41
2.3.7 Workpiece coordinate system (WCS) 1/K2/2-42

2.4 Frames 1/K2/2-43
2.4.1 Overview 1/K2/2-43
2.4.2 Rough and fine offsets (translations) 1/K2/2-44
2.4.3 Frame rotations 1/K2/2-46
2.4.4 SCALE 1/K2/2-54
2.4.5 MIRROR 1/K2/2-54
2.4.6 Frames and switchover of geometry axes (SW 5 and higher) 1/K2/2-57
2.4.7 Channel-specific system frames (SW 6.1 and later) 1/K2/2-58
2.4.8 Configuring the current system frames (SW 6.1 and higher) 1/K2/2-59

© Siemens AG, 2003. All rights reserved

2.4.9 NCU global basic frames (SW 5 and higher) 1/K2/2-62
2.4.10 Channel-specific basic frames 1/K2/2-63
2.4.11 Frames active in the channel 1/K2/2-63
2.4.12 FRAME chaining 1/K2/2-67
2.4.13 Frames with G91 (SW 4 and higher) 1/K2/2-71
2.4.14 Suppression of frames with G53, G153 and SUPA (SW 6.1 and later) 1/K2/2-72
2.4.15 Operating states and frames 1/K2/2-74

2.5 Workpiece-related actual-value system 1/K2/2-77
2.5.1 Use of workpiece-related actual-value system 1/K2/2-77
2.5.2 Special reactions 1/K2/2-79

3 Supplementary Conditions 1/K2/3-81

3.1 Axes 1/K2/3-81

3.2 Coordinate systems 1/K2/3-82

3.3 Frames 1/K2/3-82

3.4 Workpiece-related actual value system/reset response 1/K2/3-82

4 Data Descriptions (MD, SD) 1/K2/4-83

4.1 Operator panel machine data 1/K2/4-83

4.2 General machine data 1/K2/4-88

4.3 Channel-specific machine data 1/K2/4-94
4.3.1 Axes/coordinate system 1/K2/4-94

4.4 Axis-specific machine data 1/K2/4-99

4.5 Channel-specific setting data 1/K2/4-100

5 Signal Description 1/K2/5-101

5.1 Axis/spindle-specific signals 1/K2/5-101
5.1.1 Axes 1/K2/5-101
5.1.2 Coordinate systems 1/K2/5-101
5.1.3 Workpiece-related actual-value system 1/K2/5-101

5.2 Channel-specific signals 1/K2/5-102
5.2.1 Axes 1/K2/5-102
5.2.2 Coordinate systems 1/K2/5-102
5.2.3 Workpiece-related actual-value system 1/K2/5-102

6 Examples 1/K2/6-105

6.1 Axes 1/K2/6-105

6.2 Coordinate systems 1/K2/6-107

6.3 Frames 1/K2/6-108

7 Data Fields, Lists 1/K2/7-111

7.1 Interface signals 1/K2/7-111
7.1.1 Axes 1/K2/7-111
7.1.2 Coordinate systems 1/K2/7-111
7.1.3 Workpiece-related actual-value system 1/K2/7-111

© Siemens AG, 2003. All rights reserved

7.2	Machine data	1/K2/7-112
7.2.1	General machine data for operator panel	1/K2/7-112
7.2.2	Axes/coordinate systems	1/K2/7-113
7.3	Setting data	1/K2/7-114
7.4	Alarms	1/K2/7-114
7.5	System variable	1/K2/7-114

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

1 Brief Description

1.1 Axes

Machine axes Machine axes are the axes that actually exist on a machine tool.

Channel axes Every geometry axis and every special axis is assigned to a channel and therefore a channel axis. Geometry axes and additional axes are always traversed in “their” channel.

Geometry axes The three geometry axes always make up a fictitious rectangular coordinate system, the basic coordinate system (BCS). By using FRAMES (offset, rotation, scaling, mirroring), it is possible to image geometry axes of the workpiece coordinate system (WCS) on the BCS.

Special axes In contrast to geometry axes, no geometrical relationship is defined between the special axes.

Path axes Path axes are interpolated together (all the path axes of a channel have a common path interpolator). All the path axes of one channel have the same acceleration phase, constant travel phase and delay phase.

Positioning axes Positioning axes are interpolated separately (each positioning axis has its own axis interpolator). Each positioning axis has its own feedrate and acceleration characteristic.

Synchronous axes Synchronous axes are interpolated together with path axes (all path axes and synchronous axes of one channel have a common path interpolator). All path axes and all synchronous axes of a channel have the same acceleration phase, constant travel phase and deceleration phase.

© Siemens AG, 2003. All rights reserved

Axis configuration

The figure below shows the assignment between the geometry axes, special axes, channel axes and machine axes as well as the names of the individual axis types.
MD 20050 AXCONF_GEOAX_ASSIGN_TAB,
MD 20060 AXCONF_GEOAX_NAME_TAB,
MD 20070 AXCONF_MACHAX_USED,
MD 20080 AXCONF_CHANAX_NAME_TAB,
MD 10000 AXCONF_MACHAX_NAME_TAB,
MD 35000 SPIND_ASSIGN_TO_MACHAX.

Replaceable geometry axes

The “Replaceable geometry axes” function allows the geometry axes in a grouping to be replaced by other channel axes.
Axes that are initially configured as synchronous special axes in a channel can replace any selected geometry axis in response to a program command.

Link axis (SW 5 and higher)

Link axes are axes which are physically connected to another NCU and which are controlled by its position control system. Link axes can be assigned dynamically to channels of **another** NCU. Link axes are not → local axes from the perspective of a particular NCU.
The **axis container** concept is used for the dynamic modification of the assignment to an NCU. Axis replacement with GET and RELEASE from the parts program is not available for link axes across NCU boundaries.
The link axes are described in
References: /FB/ B3 Several Operator Panels and NCUs

Axis container (SW 5 and higher)

An axis container is a circular buffer data structure in which local axes and/or link axes are assigned to channels. The entries in the circular buffer can be **shifted cyclically**.

In addition to the direct reference to local axes or link axes, the link axis configuration in the logical machine axis image also allows references to axis containers. Such a reference consists of:

- An axis container number and
- a slot (circular buffer location within the corresponding container)

The entry in a circular buffer location contains:

- A local axis **or**
- a link axis

The axis container function is described in
References: /FB/ B3 Several Operator Panels and NCUs

© Siemens AG, 2003. All rights reserved

1.2 Coordinate systems

MKS

The machine coordinate system (MCS) has the following properties:

- It is defined by the machine axes.
- The machine axes can be perpendicular to each other to form Cartesian system or arranged in any other way.
- The names of the machine axes can be defined.
- The machine axes can be linear or rotary axes.

BCS

The basic coordinate system (BCS) has the following properties:

- The geometry axes form a perpendicular Cartesian coordinate system.
- The BCS is derived from a kinematic transformation of the MCS.

BZS

The basic zero system (BZS) is the basic coordinate system with a basic offset.

SZS

The settable zero system (SZS) is the workpiece coordinate system with a programmable frame from the viewpoint of the WCS. The workpiece zero is defined by the settable frames G54 ... G599.

WCS

The workpiece coordinate system (WCSMachine) has the following properties:

- In the workpiece coordinate system all the axes coordinates are programmed (parts program).
- It is made up of geometry axes and special axes.
- Geometry axes always form a perpendicular Cartesian coordinate system
- Special axes form a coordinate system without any geometrical relation between the special axes.
- The names of the geometry axes and special axes can be defined.
- The workpiece coordinate system can be translated, rotated, scaled or mirrored with FRAMES (TRANS, ROT, SCALE, MIRROR). Multiple translations, rotational movements etc. are also possible.

© Siemens AG, 2003. All rights reserved

WOE

The work offset external has the following properties:

- At a time defined in the PLC, a predefined additional zero offset between the basic and the workpiece coordinate systems is activated.
- The magnitudes of the offsets can be set by the following for each of the axes involved:
  - PLC
  - Operator panel
  - Parts programs
- Activated offsets take effect at the instant the first motion block of the relevant axes is processed after offset activation. The offsets are superimposed on the programmed path (no interpolation). The velocity at which the work offset external is shifted is as follows: Programmed F value plus 1/2 JOG velocity. The work offset external is traversed at the end of G0 blocks.
- The activated offsets are retained after RESET and end of program.
- After POWER ON, the last active offset is still stored in the control but must be reactivated by the PLC.

© Siemens AG, 2003. All rights reserved

1.3 Frames

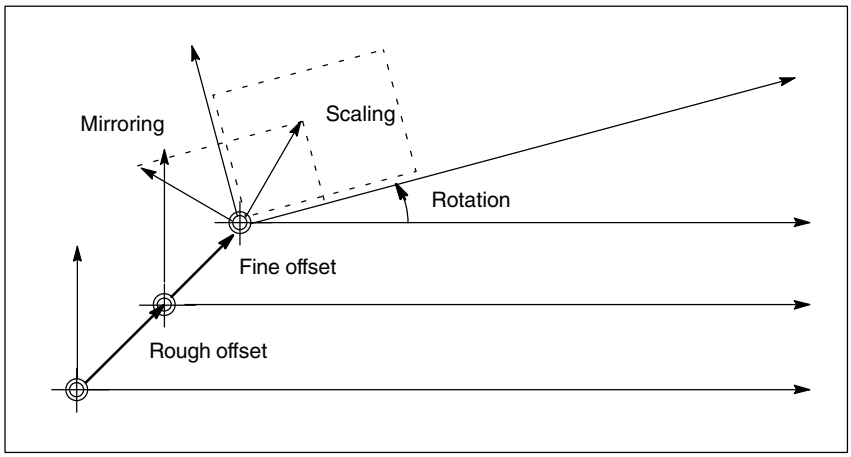


Fig. 1-1 FRAME components

A FRAME is a closed calculation rule that translates one Cartesian coordinate system into another.

A FRAME consists of the following components:

- Offset
  Rough offset:
  –> Programmable with TRANS,
  –> ATRANS (additive translation component) and
  –> CTRANS (zero offset for several axes)
  –> G58 (axial zero offset).

  Fine offset:
  –> Programmable with CFINE and
  –> G59 (axial zero offset).
- Rotation –> Programmable with ROT, AROT and
  –> with ROTS, AROTS and CROTS in SW 5.3 and higher
- Scaling –> Programmable with SCALE and ASCALE
- Mirroring –> Programmable with MIRROR and AMIRROR

Features in relation to axes

The rough and fine offsets, scaling and mirroring can be programmed for geometry and special axes. A rotation can also be programmed for geometry axes.

Rough and fine offsets

The translation component of FRAMES comprises:

Rough offset with TRANS, ATRANS and CTRANS:
Defined by the tool setter. The access rights can be restricted via MMC input.

The programmable offsets for all geometry axes and special axes are programmed with TRANS.

Fine offset with CFINE:
Can be defined by the operator via the MMC within certain input limits.

© Siemens AG, 2003. All rights reserved

G58, G59 (SW 5 and higher)

G58 and G59 can be programmed to replace the rough and fine offsets of the programmable frame on an axial basis. These functions can only be used if the fine offset is configured.

Rough offset with G58:
G58 changes only the absolute translation component (rough offset) for the specified axis; the total of additively programmed translations (fine offset) is retained.

Fine offset with G59:
G59 is used for axial overwriting of the additively programmed translations for the specified axes which were programmed with ATRANS.

Frame rotations

Orientations in space are defined via frame rotations as follows:

- Rotation with ROT defines the individual rotations for all geometry axes.
- Solid angles with ROTS, AROTS, CROTS define the orientation of a plane in space.
- Frame rotation with TOFRAME defines a frame with a Z axis pointing in the tool direction.

Scaling

SCALE is used to program the programmable scale factors for all geometry axes and special axes.

ASCALE must be programmed if a new scaling is to be based on a previous scaling, rotation, translation or mirroring.

Mirroring (SW 5 and higher)

MD10610: MIRROR_REF_AX can be used to define the axis in relation to which mirroring will be performed.

MD10610 = 0:
Mirroring is performed around the programmed axis.

MD10610 = 1 or 2 or 3:
Depending on the input value, mirroring is mapped onto the mirroring of a specific reference axis and rotation of two other geometry axes.

Frame chaining

The current Frame consists of the total basic frame, the settable FRAME, SW 6.1 and higher the 4 system frames and the programmable FRAME. The current total frame is calculated according to the following formula:

$P_ACTFRAME =$P_SETFRAME : $P_EXTFRAME : $P_PARTFRAME : $P_ACTBFRAME : $P_IFRAME : $P_TOOLFRAME : $P_PFRAME

© Siemens AG, 2003. All rights reserved

Frames with G91 (SW 4 and higher)

Incremental programming with G91 is defined such that the compensation value is traversed additively to the incrementally programmed value when a zero offset is selected.

With the setting data SD 42440: FRAME_OFFSET_INCR_PROG is used to set the following:

- Value = 1
  Zero offset is retracted on FRAME and incremental programming of an axis (= default setting)
- Value = 0
  Only the programmed path is traversed.

Suppression of frames

Current frames can be suppressed with the following instructions:

- G53 Current zero offset (ZO)
- G153 Current frame, incl. basic frame
- SUPA Current ZO, incl. programmed offsets

NCU global basic frames

For the rotary indexing machine technology, for example, it is necessary to initialize frames for various channels from one channel.
These cross-channel frames are referred to below as “NCU global basic frames”.

The NCU global basic frames:

- Can be read and written from all channels.
- Can only be activated in the channels.
- Up to 16 NCU global base frames are available.

Global frames can be used to apply offsets, scale factors and mirroring operations to channel and machine axes.
All base frames (up to 16 global and 16 channel-specific) are concatenated to produce the complete base frame. The standard configuration is designed for at least one base frame per channel.

Settable frames can be defined as either NCU global or channel-specific.

Channel coordination

With NCU global frames, the user must ensure channel coordination and activation of the frames (e.g. using the WAITMC command), to allow the frames to be calculated at the desired point in the program.

Cross-channel activation of frames is not supported.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

2 Detailed Description

2.1 Axes

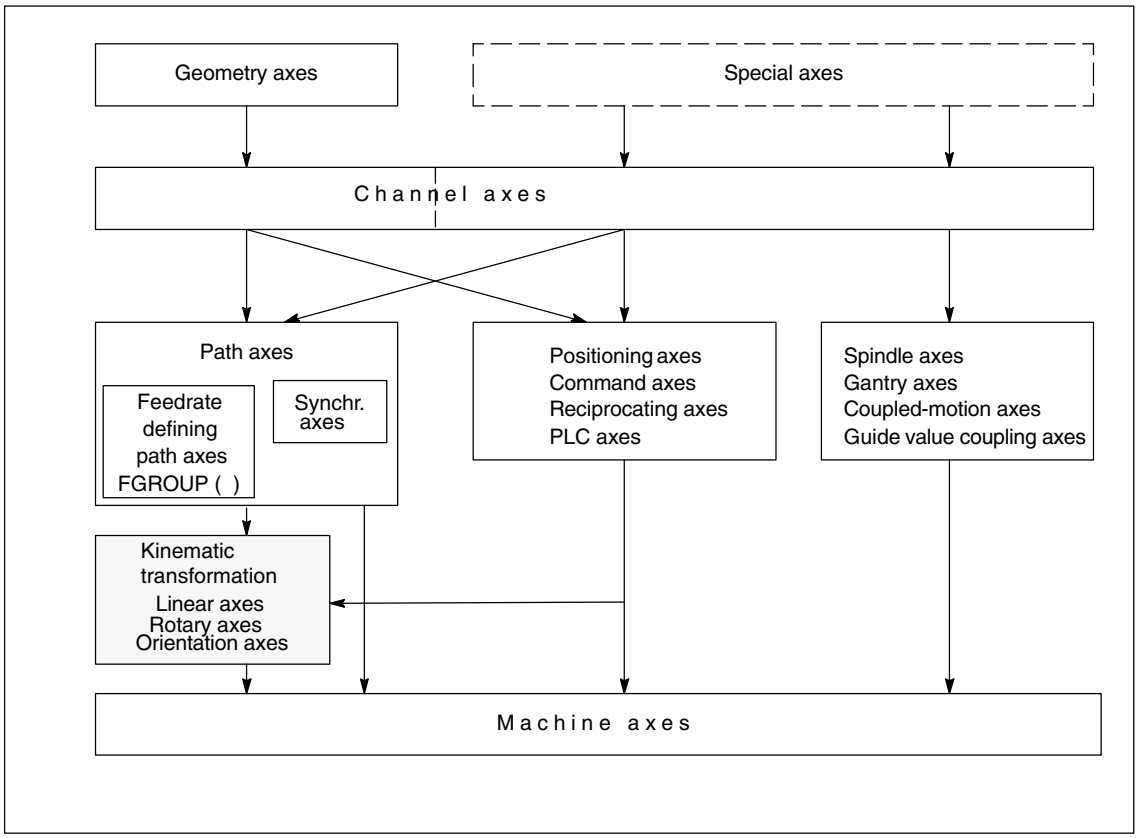


Fig. 2-1 Relationship between the axis types

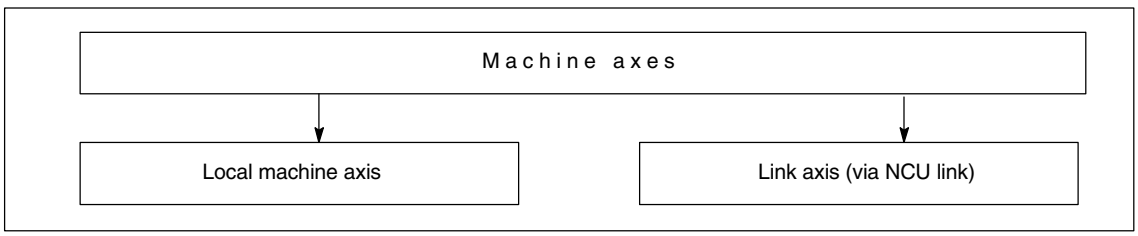


Fig. 2-2 Local axes and link axis

© Siemens AG, 2003. All rights reserved

2.1.1 Machine axes

Meaning

Machine axes are the axes that actually exist on a machine tool.

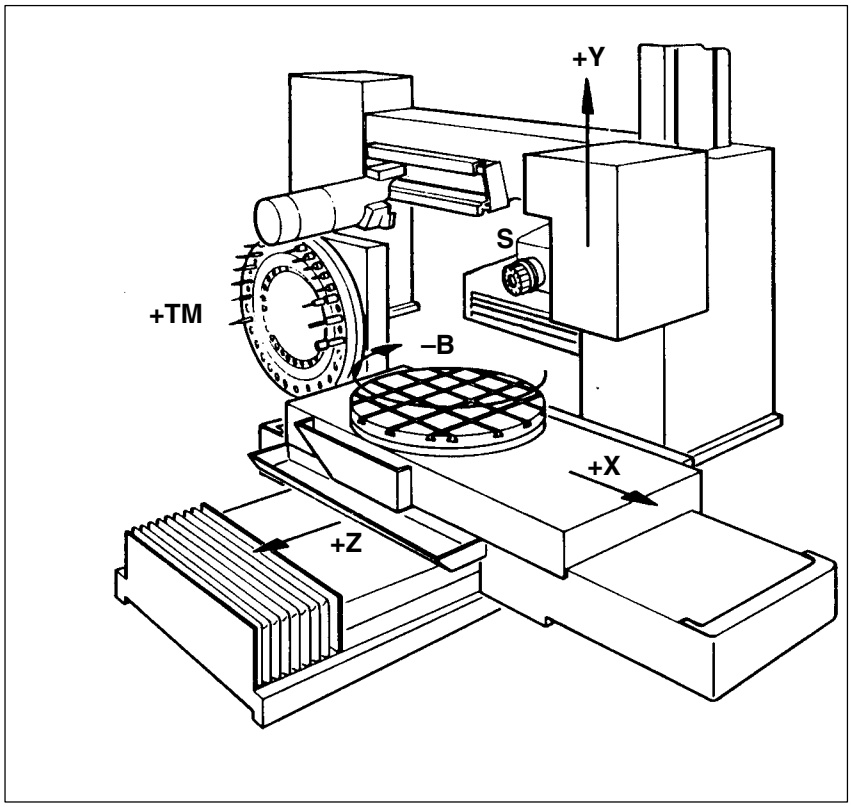


Fig. 2-3 Machine axes X, Y, Z, B, S on a Cartesian machine

Application

The following can be machine axes:

- Geometry axes X, Y, Z
- Orientation axes A, B, C
- Loader axes
- Tool turrets
- Axes for tool magazine
- Axes for automatic tool changer
- Spindle sleeves
- Axes for pallet changers
- etc.

© Siemens AG, 2003. All rights reserved

2.1.2 Channel axes

Meaning Each geometry axis and each special axis is assigned to a channel. Geometry axes and additional axes are always traversed in “their” channel.

2.1.3 Geometry axes

Meaning The three geometry axes always make up a fictitious rectangular coordinate system. By using FRAMES (offset, rotation, scaling, mirroring), it is possible to image geometry axes of the workpiece coordinate system (WCS) on the BCS.

Application Geometry axes are used to program the workpiece geometry (the contour).

Plane selection G17, G18 and G19 (DIN 66217) always refers to the three geometry axes. That is why it is advantageous to name the three geometry axes X, Y and Z.

2.1.4 Replaceable geometry axes

Meaning The “Replaceable geometry axes” function allows the geometry axes in a grouping to be replaced by other channel axes.

Axes that are initially configured as synchronous special axes in a channel can replace any selected geometry axis in response to a program command.

Example On a machine with two Z axes, Z1 and Z2, either of the Z axes can be programmed as the geometry axis in response to an instruction in the parts program.

Activation Axis replacement is activated by the program command

GEOAX([n, channel axis name]...)

n=0:	In order to remove an axis from the geometry axis-grouping
n=1, 2, 3:	Index of geometry axis
GEOAX( ):	Establishes the basic setting defined via MD for the assignment of channel axes to geometry axes
Channel axis name:	Name of channel axis which is to operate as a geometry axis.

A channel axis which has been designated a geometry axis can only be addressed under its geometry axis name. The geometry axes names themselves remain unchanged.

Geometry axes can be replaced either individually or as a group in one command.

© Siemens AG, 2003. All rights reserved

Supplementary conditions

As a basic rule, any channel axis designated as a geometry axis can be replaced by another channel axis. In this case, the following restrictions apply:

- Rotary axes may not be programmed as geometry axes.
- A geometry axis which has the same name as a channel axis cannot be replaced by another channel axis (alarm message). Nor can an axis of this type be removed from the geometry axis grouping. It cannot change its position within the geometry axis grouping.
- Both axes in each of the axis pairs involved in the replacement operation must be block-synchronized.
- The following functions may not be active when geometry axes are replaced:
  - Transformation
  - Spline interpolation
  - Tool radius compensation
  - Tool fine compensation
- Any active DRF offset or work offset external will remain operative. They both act on channel axes. The channel axis assignment is not affected by the replacement of geometry axes.

Replacement of geometry axes

All frames, protection zones and working area limitations are deleted.

They may need to be reprogrammed after the replacement operation.

The system response to replacement of geometry axes is therefore identical to its response to a change (switch on/off, switchover) in a kinematic transformation.

Tool length compensation

Any active tool length compensation remains operative and is applied to the new geometry axes after replacement.

The system treats tool length compensations as not yet applied for all geometry axes

- which have been newly added to the geometry axis grouping
- which have changed their positioning within the geometry axis grouping

Geometry axes which retain their position within the geometry axis grouping after a replacement operation also retain their status with respect to tool length compensation.

RESET

MD 20110: RESET_MODE_MASK

Bit 12: Reset behavior of the modified geometry axis assignment

Bit 12 = 0 When machine data $MC_GEOAX_CHANGE_RESET is enabled, a modified geometry axis assignment is cleared on Reset or on parts program end. The basic setting defined in the MD for the geometry axis assignment becomes active.

Bit 12 = 1 A modified geometry axis assignment remains active after a Reset/parts program end.

© Siemens AG, 2003. All rights reserved

MD 20118: GEOAX_CHANGE_RESET

FALSE: The current configuration of the geometry axes remains unchanged on reset and program start. With this setting, the response is identical to older software versions without geometry axis replacement.

TRUE: The configuration of the geometry axes remains unchanged or is switched to the basic state defined in machine data AXCONF_GEOAX_ASSIGN_TAB on Reset and parts program end (according to machine data $MC_RESET_MODE_MASK) and on parts program start (according to machine data $MC_START_MODE_MASK).

Program start

Analogously to the Reset response, the behavior on Start is determined by:

MD 20112: START_MODE_MASK

Bit 12: Behavior of the modified geometry axis assignment

Bit 12 = 0 A modified geometry axis assignment remains active on parts program start.

Bit 12 = 1 A modified geometry axis assignment is deleted on parts program start.

Reference point approach

When the “Reference point approach” mode is selected, the geometry axis configuration defined by the machine data is automatically set.

M code

The PLC is informed of a replacement of geometry axes by means of GEOAX( ) through the optional output of an M code which is set in MD 22532: GEOAX_CHANGE_M_CODE.

Note

If this machine data is set to one of the values 0 to 6, 17, 30, then no M code is output.

Transformation changeover

The following interrelationships must be noted with respect to kinematic transformation and geometry axis replacement:

- Geometry axis assignments cannot be modified when the transformation is active.
- Activating a transformation deletes the current geometry axis configuration and replaces it by the geometry axis assignment stored in the machine data of the activated transformation.
- When the transformation is deactivated, the MD-defined basic setting for the geometry axis configuration becomes valid again.

© Siemens AG, 2003. All rights reserved

Should it be necessary to modify the geometry axis assignment in connection with transformations, then another new transformation must be configured.

For this purpose, there are a total of 8 simultaneously available transformations in the channel. A maximum of two transformations are available simultaneously per channel from the following transformation groups:

- Orientation transformations
  (3-axis, 4-axis, 5-axis and nutation transformation)
- TRAANG (inclined axis)
- TRANSMIT
- TRACYL

References: /FB/, F2, “5-axis transformation”
/FB/, M1, “Kinematic transformation”

Example

In the following example, it is assumed that there are 6 channel axes with channel axis names XX, YY, ZZ, U, V, W and three geometry axes with names X, Y, Z. The basic setting is defined in machine data such that the geometry axes are imaged on the first three channel axes, i.e. on XX, YY and ZZ.

```
GEOAX( )                ;The geometry axis assignment defined via
                        ;MD AXCONF_GEOAX_ASSIGN_TAB becomes
                        ;effective, i.e. XX, YY and ZZ become
                        ;geometry axes.
G0 X0 Y0 Z0 U0 V0 W0    ;Traverse all axes in rapid traverse to position 0.
GEOAX(1, U, 2, V, 3, W) ;Channel axis U becomes the 1st geometry axis,
                        ;V the second and W the third.
GEOAX(1, XX, 3, ZZ)     ;Channel axis XX becomes the first geometry axis,
                        ;ZZ becomes the third geometry axis. The second
                        geometry axis remains unchanged.
G17 G2 X20 I10 F1000    ;Semicircle in the XY plane. Channel axes
                        ;XX and V traverse.
GEOAX(2, W)             ;Channel axis W becomes the 2nd geometry-
                        ;axis. The first and third geometry axes
                        ;remain unchanged.
G17 G2 X20 I10 F1000    ;Full circle in the XY plane. Channel axes XX and
                        ;W traverse.
GEOAX( )                ;The geometry axis assignment defined via
                        ;MD AXCONF_GEOAX_ASSIGN_TAB becomes
                        ;effective, i.e. XX, YY and ZZ become
                        ;geometry axes.
GEOAX(1, U, 2, V, 3, W) ;U, V and W become the first, second and
                        ;third geometry axes.
G1 X10 Y10 Z10 XX=25    ;Channel axes U, V, W each traverse to
                        ;position 10, XX traverses to position 25.
GEOAX(0, V)             ;V is removed from the geometry axis grouping
                        ;again. U and W remain geometry-axes. The
                        second geometry axis is no longer assigned.
```

© Siemens AG, 2003. All rights reserved

```
GEOAX(1, U, 2, V, 3, W)    ;U, V and W become the first, second and
                           ;third geometry axes, i.e. U and W remain
                           ;unchanged.
GEOAX(3, V)                ;V becomes the third geometry axis. As a result,
                           ;W, which was previously the 3rd geometry axis, is
                           ;removed from the geometry axis grouping.
                           ;The second geometry axis is no longer assigned.
```

2.1.5 Special axes

Meaning

In contrast to geometry axes, no geometrical relationship is defined between the special axes.

Note: Geometry axes have an exactly defined relationship in the form of a right-angled coordinate system.

Special axes are part of the basic coordinate system (BCS). With FRAMES (translation, scaling, mirroring), special axes of the workpiece coordinate system can be mapped on the basic coordinate system.

Application

Typical special axes are:

- Rotary axes
- Machine tool axes
- Tool revolver axes
- Loader axes

2.1.6 Path axes

Meaning

Path axes are interpolated together (all the path axes of a channel have a common path interpolator). All the path axes of one channel have the same acceleration phase, constant travel phase and delay phase. The feedrate programmed under address F (path feedrate) applies to all the path axes programmed in a block, with the following exceptions:

- An axis has been programmed that has been defined as having no control over the path velocity with instruction FGROUP – see Chapter “Synchronous Axes”
- Axes programmed with instructions POS or POSA have an individual feedrate setting (axis interpolator) – see Chapter “Positioning Axes”

Application

Path axes are used to machine the workpiece with the programmed contour.

© Siemens AG, 2003. All rights reserved

2.1.7 Positioning axes

Meaning

Positioning axes are interpolated separately (each positioning axis has its own axis interpolator). Each positioning axis has its own feedrate and acceleration characteristic. Positioning axes can be programmed in addition to path axes (even in the same block). Path axis interpolation (path interpolator) is not affected by the positioning axes. Path axes and the individual positioning axes do not necessarily reach their block end points at the same time.

Instructions POS and POSA are used to program positioning axes and define block change criteria:

- POS: Block change takes place when the path axes and positioning axes have reached their block end points.
- POSA: Block change takes place when the path axes have reached their block end point. Positioning axes continue to traverse beyond block limits to their block end point.

Concurrent positioning axes differ from positioning axes in that they:

- Only receive their block end points from the PLC
- Can be started at any time (not at block limits)
- Do not affect the processing of current parts programs

Application

The following are typical applications of positioning axes:

- Loaders for transporting workpieces
- Tool magazine/turret

Reference

References: /FB/, P2, “Positioning Axes”
/FB/, S1, “Spindles”
/FB/, G1, “Gantry axes”
/FB/, M3, “Coupled motion and master/slave coupling”
/FB/, P3, “PLC basic program” (PLC axes)
/FB/, P5, “Oscillation”
/FB/, FBSY/, “Synchronized actions” (command axes)

© Siemens AG, 2003. All rights reserved

2.1.8 Main axes

Meaning

A main axis is an axis that is interpolated by the main run. This interpolation can be started:

- From synchronized actions
  as command axes due to an event via block-related, modal or static synchronous actions.
- From the PLC via special function blocks in the PLC basic program
  as competing positioning axis or also called PLC axis.
- Per setting data or from the parts program
  as asynchronous or block-synchronous oscillating axis.

Channel control

An axis interpolated by the main axis reacts in terms of:

- NC stop
- Alam handling
- Program control
- End of program
- RESET

Note

The behavior at the end of the program varies. The axis movement need not always be completed and therefore may carry on beyond the end of the program.

Application

In SW 6.3 and higher, certain axes in the main run can be be decoupled at the the channel response triggered by the NC program sequence and controlled from the PLC. These axes are also interpolated in the main run and respond independently for the channel and program sequence.

A PLC-controlled axis can then be controlled indeendently by the NC. This affects

- the sequence for cancelling the axis (equivalent to delete distance-to-go)
- stopping or interrupting the axis
- continuing the axis (continue motion)
- resetting the axis to its basic status

© Siemens AG, 2003. All rights reserved

2.1.9 Synchronous axes

Meaning

Synchronous axes are components of the path axes which are not referenced in order to calculate the tool path velocity. They are interpolated together with path axes (all path axes and synchronous axes of one channel have a common path interpolator). All path axes and all synchronous axes of a channel have the same acceleration phase, constant travel phase and deceleration phase. The feedrate (path feedrate) programmed under address F applies to all the path axes programmed in a block but not to the synchronous axes. Synchronous axes take the same time to cover the programmed path as the path axes.

FGROUP command

The instruction FGROUP determines whether the axis is a feed-determining **path axis** (used in calculation of path velocity) or a **synchronous axis** (**not** used in calculation of path velocity).

Example

N05 G00 G94 G90 M3 S1000 X0 Y0 Z0

N10 FGROUP(X,Y)	Axes X and Y are path axes Axis Z is a synchronous axis
N20 G01 X100 Y100 F1000	Progr. feedrate 1000 mm/min Feedrate of axis X = 707 mm/min Feedrate of axis Y = 707 mm/min
N30 FGROUP (X)	Axis X is a path axis Axis Y is a synchronous axis
N20 X200 Y150	Progr. feedrate 1000 mm/min Feedrate of axis X = 1000 mm/min Feedrate of axis Y is automatically set to 500 mm/min because only half the path has to be traversed.

Note

The channel axis name must be used for the FGROUP command (defined by MD: AXCONF_CHANAX_NAME_TAB).

Application

In the case of helical interpolation FGROUP can be programmed to determine whether

- the programmed feedrate is to apply on the path (all three programmed axes are path axes) or
- the programmed feedrate is to apply on the circle (two axes are path axes and the infeed axis is a synchronous axis).

© Siemens AG, 2003. All rights reserved

2.1.10 Axis configuration

The figure below shows the assignment between the geometry axes, special axes, channel axes and machine axes as well as the names of the individual axis types. The assignment is made in the MD (see Chapter 4).

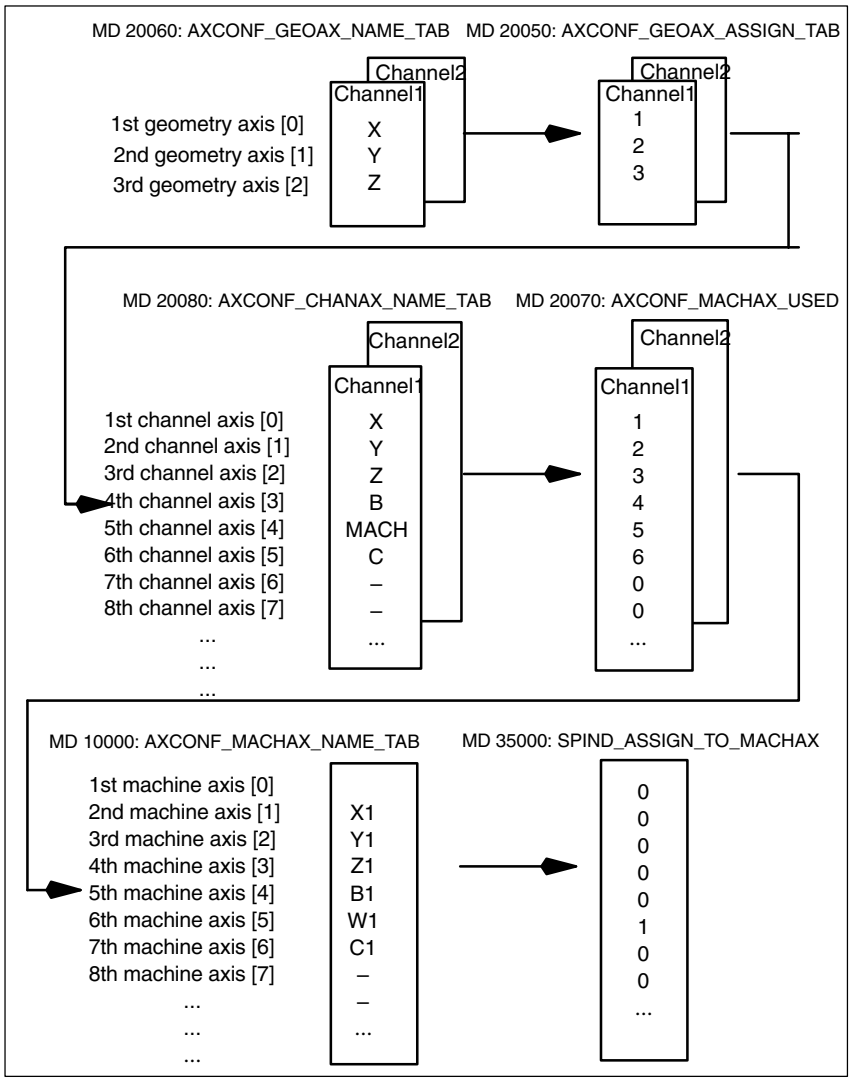


Fig. 2-4 Axis configuration

Note

The geometry axes must be assigned to the channel axes in ascending order leaving no gaps.

© Siemens AG, 2003. All rights reserved

Special points to be noted

Three geometry axes are assigned to the channel axes in the MD.

- All channel axes that are not assigned to the three geometry axes are special axes.
- The channel axes are assigned to machine axes.
- The spindles are also assigned to machine axes.

Channel axis gaps SW 5.1

In SW 5 and higher, not every channel axis per
MD 20080: AXCONF_CHANAX_NAME_TAB
has to be assigned to a machine axis (local or link axis) via
MD 20070: AXCONF_MACHAX_USED.
If the actual machine in the application has a specific machine axis which is assigned to the channel axis, values entered in MD 20070: AXCONF_MACHAX_USED refer to the logical machine axis map
MD 10002: AXCONF_LOGIC_MACHAX_TAB,
otherwise 0 is entered.

Application:
Consistent, semi-defined channel axis names for various machine versions of a manufacturer's machine series.
Advantages:
Standard basic configurations for different machines, easy reconfiguration when expanding a machine, portability of programs.

Reliability of channel axis gaps

Channel axis gaps must be declared explicitly in MD 11640: ENABLE_CHAN_AX_GAP. If you omit to do this, a 0 entry in MD 20070: AXCONF_MACHAX_USED terminates the assignment of further machine axes to channel axes.

References: /FB/, B3 "Several Operator Panels and NCUs" (Distributed Systems) in SW 5 and higher

© Siemens AG, 2003. All rights reserved

Example

In the following example, a machine tool channel axis is specified without a real machine axis. The shaded structures are available in SW 5 and higher.

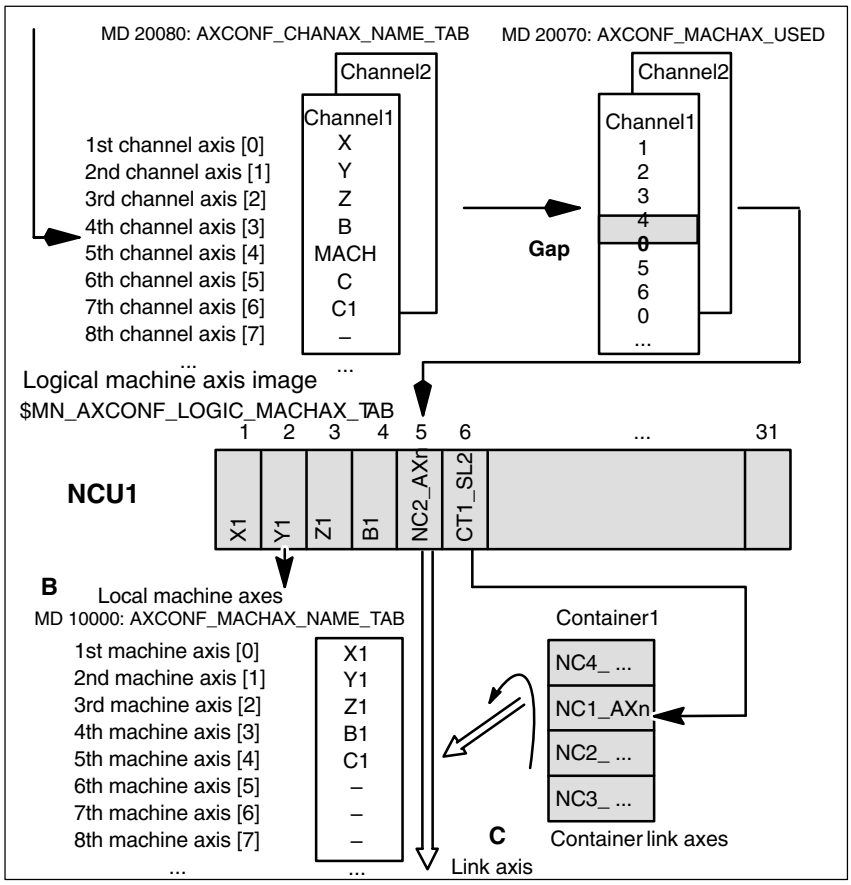


Fig. 2-5 Axis configuration with channel axis gap

Note

The gaps count as axes with reference to the number of channel axes and their indices.

If you attempt to define a channel axis gap as a geometry axis with MD 20050: AXCONF_GEOAX_ASSIGN_TAB, the attempt is denied without an alarm.

The use of channel axes in
MD 24120 ff.: TRAFO_GEOAX_ASSIGN_TAB1...8 and
MD 24110 ff.: TRAFO_AXES_IN1...8,
to which no machine axes are assigned via MD 20070:
AXCONF_MACHAX_USED (gap) generates alarm 4346 or 4347.

© Siemens AG, 2003. All rights reserved

2.1.11 Link axes (SW 5 and higher)

Link axes are axes which are physically connected to another NCU and which are controlled by its position control system. Link axes can be assigned dynamically to channels of **another** NCU. Link axes are not local axes from the perspective of a particular NCU.
The **axis container** concept is used for the dynamic modification of the assignment to an NCU. Axis replacement with GET and RELEASE from the parts program is not available for link axes across NCU boundaries.

Requirements:

- The participating NCUs, NCU1 and NCU2 must be connected by means of high-speed communication via the link module.

  References: /PHD/, Configuring Manual NCU 571–573.2, Link Module

- The axis must be configured appropriately by machine data.
- The link axis option must be installed.

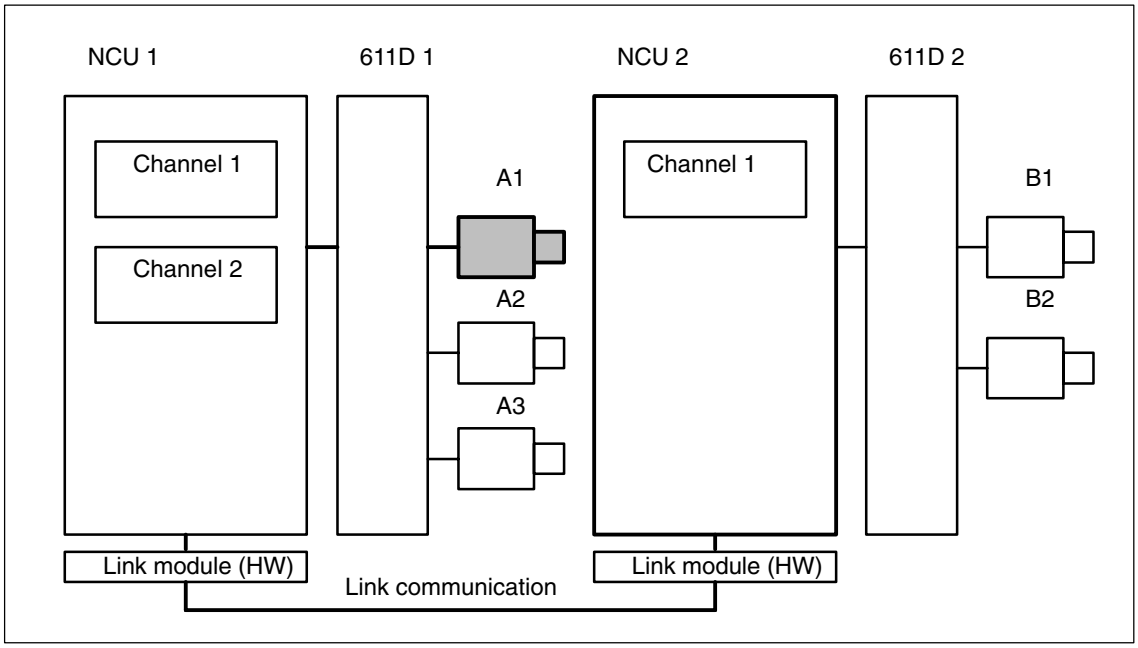


Fig. 2-6 Overview of link axes

The link axes are described in
References: /FB/ B3, “Several Operator Panels and NCUs” in SW 5 and higher

Note

The link axis functionality is currently not available with the SINUMERIK 840Di.

© Siemens AG, 2003. All rights reserved

Axis container (SW 5 and higher)

An axis container is a circular buffer data structure in which local axes and/or link axes are assigned to channels. The entries in the circular buffer can be **shifted cyclically**.

In addition to the direct reference to local axes or link axes, the link axis configuration in the logical machine axis image also allows references to axis containers.

Such a reference consists of:

- A container number and
- a slot (circular buffer location within the corresponding container)

The entry in a circular buffer location contains:

- A local axis **or**
- a link axis

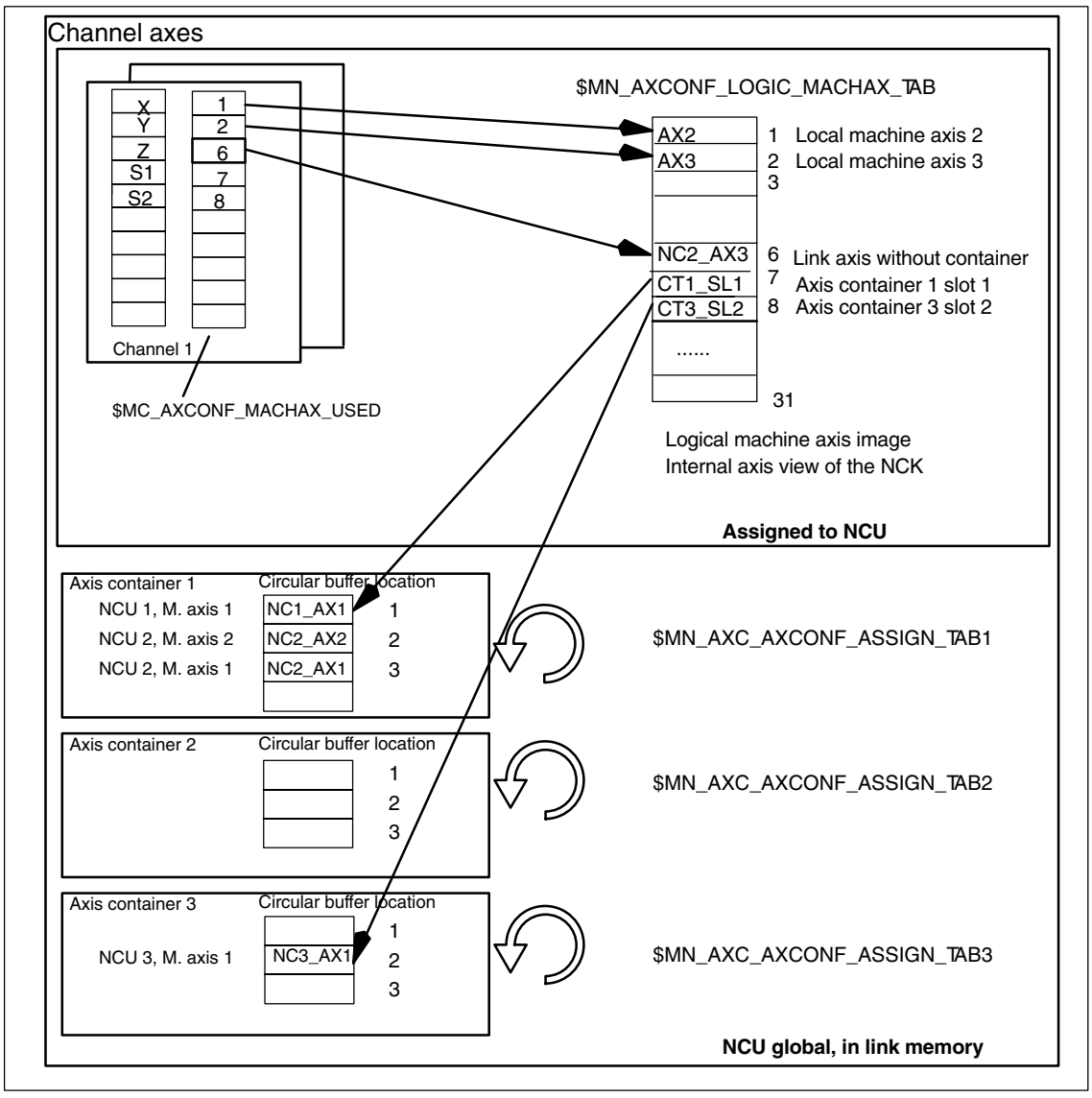


Fig. 2-7 Mapping of channel axes onto axis containers via logical machine axis image

© Siemens AG, 2003. All rights reserved

Axis container entries contain local machine axes or link axes from the perspective of an individual NCU. The entries in the logical machine axis image $MN_AXCONF_LOGIC_MACHAX_TAB of an individual NCU are fixed.

Note

The axis container functionality is currently not available with the SINUMERIK 840Di.

The axis container function is described in
References: /FB/ B3, “Several Operator Panels and NCUs” in SW 5 and higher

© Siemens AG, 2003. All rights reserved

2.2 Zeros and reference points

2.2.1 Reference points in working space

The following zeros and reference points are defined in the working space in which tools move when machining workpieces:

Table 2-1 Zeros and reference points

Zeros		**Reference points**	
	**M** = Machine zero		**R** = Reference point
	**W** = Workpiece zero		**T** = Toolholder reference point

- **M**achine zero **M**
  The machine zero M defines the machine coordinate system MCS. All other reference points refer to the machine zero.
- **W**orkpiece zero **W**
  The workpiece zero W defines the workpiece coordinate system in relation to the machine zero M. The programmed parts program blocks are executed in the workpiece coordinate system WCS.
- **R**eference point **R**
  The position of reference point R is defined by cam switches. Reference point R calibrates the position measuring system.

  The reference point must be approached every time the control system is switched on. The control can only then work with the measuring system and transfer all position values to the coordinate systems.
- **T**oolholder reference point **T**
  The toolholder reference point T is on the toolholder. By entering the tool lengths, the control calculates the distance between the tool tip (TCP Tool Center Position) and the toolholder reference point.

© Siemens AG, 2003. All rights reserved

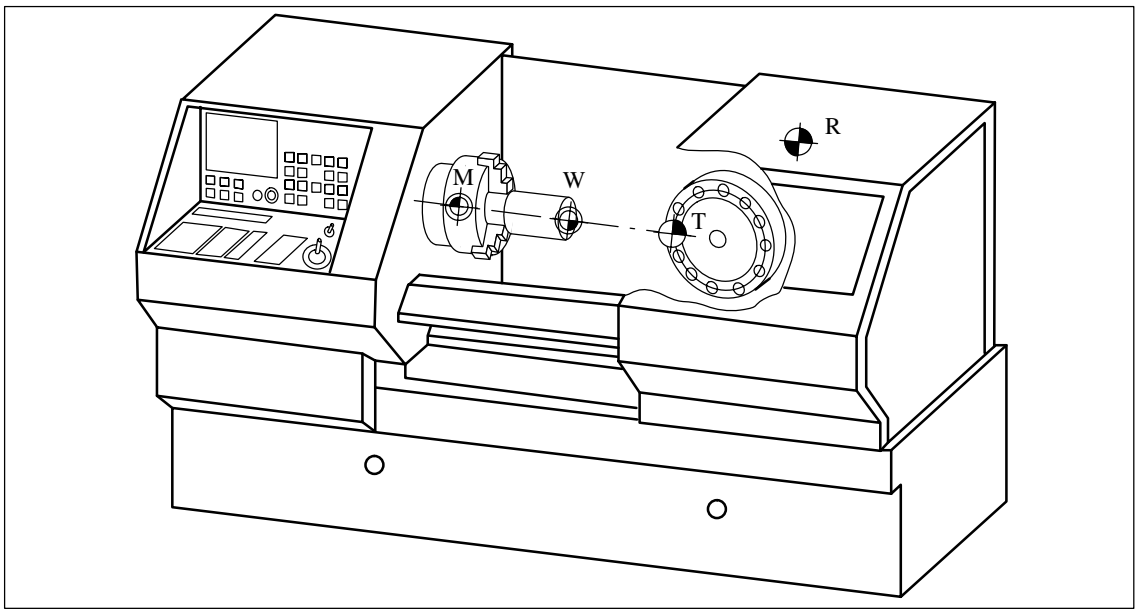


Fig. 2-8 Zeros and reference points on a turning machine

The zero of the coordinate system MCS corresponds to M and the zero of the WCS corresponds to W. In the working space of the machine the reference point is defined as R and the toolholder reference point with T.

2.2.2 Position of coordinate systems and reference points

Control POWER ON

For incremental measuring probes, the reference point must be approached each time the control is activated so that the control can transfer all position values to the coordinate system.

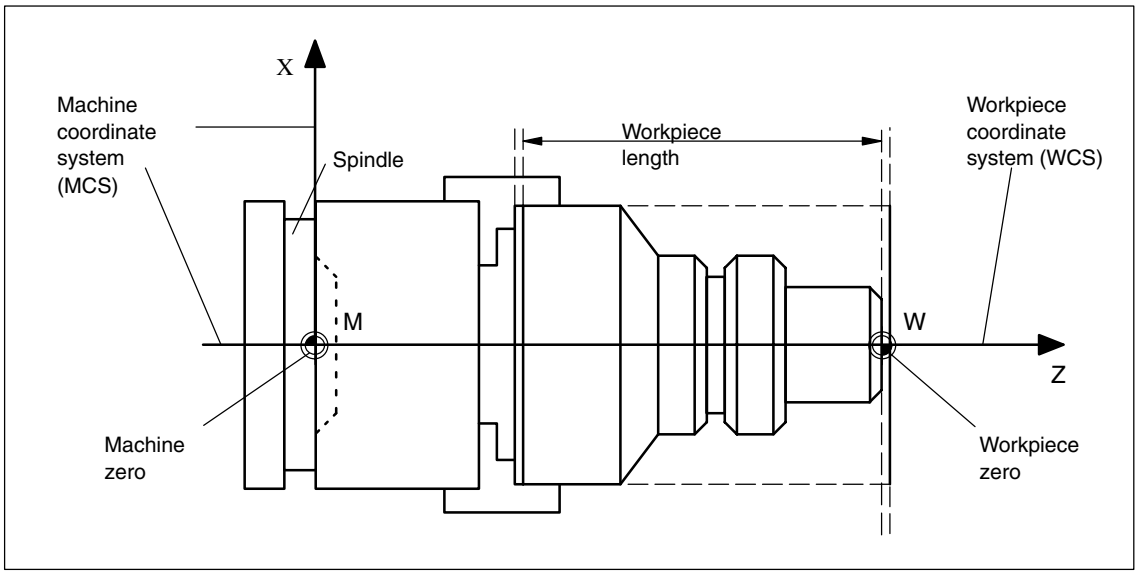


Fig. 2-9 Position of coordinate systems by machine zero M and workpiece zero W

© Siemens AG, 2003. All rights reserved

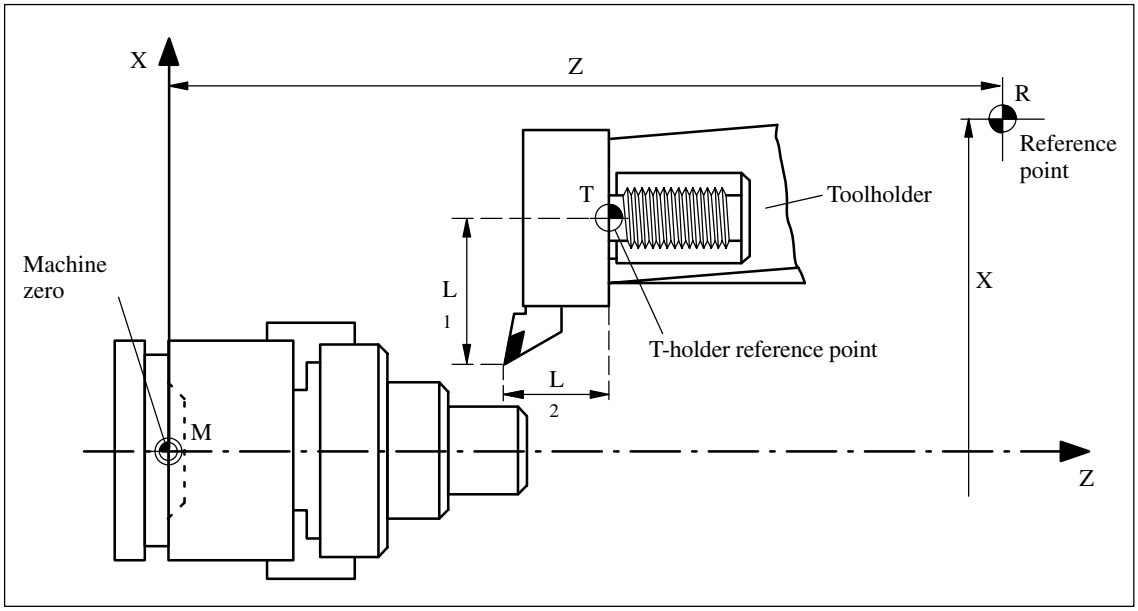


Fig. 2-10 Position of reference point in relation to machine zero

© Siemens AG, 2003. All rights reserved

2.3 Coordinate systems

Meaning

DIN 66217 specifies the use of clockwise, rectangular (Cartesian) coordinate systems for machine tools.

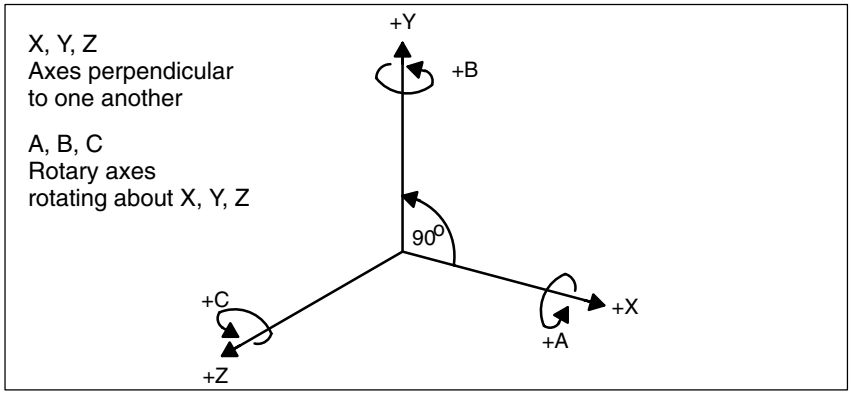


Fig. 2-11 Clockwise, rectangular Cartesian coordinate system

The following coordinate systems are defined:

- MCS **M**achine **C**oordinate **S**ystem
- BCS **B**asic **C**oordinate **S**ystem
- BZS **B**asic **Z**ero **S**ystem
- SZS **S**ettable **Z**ero **S**ystem
- WCS **W**orkpiece **C**oordinate **S**ystem

The coordinate systems are determined by the kinematic transformation and the FRAMES. Fig. 2-12 shows this relationship.

A kinematic transformation is used to derive the BCS from the MCS. If no kinematic transformation is active, the BCS is the same as the MCS.

The basic frame maps the BCS onto the BZS.

An activated settable FRAME G54...G599 converts the BZS to the SZS.

The WCS, which is the basis for programming, is defined by the programmable FRAME.

© Siemens AG, 2003. All rights reserved

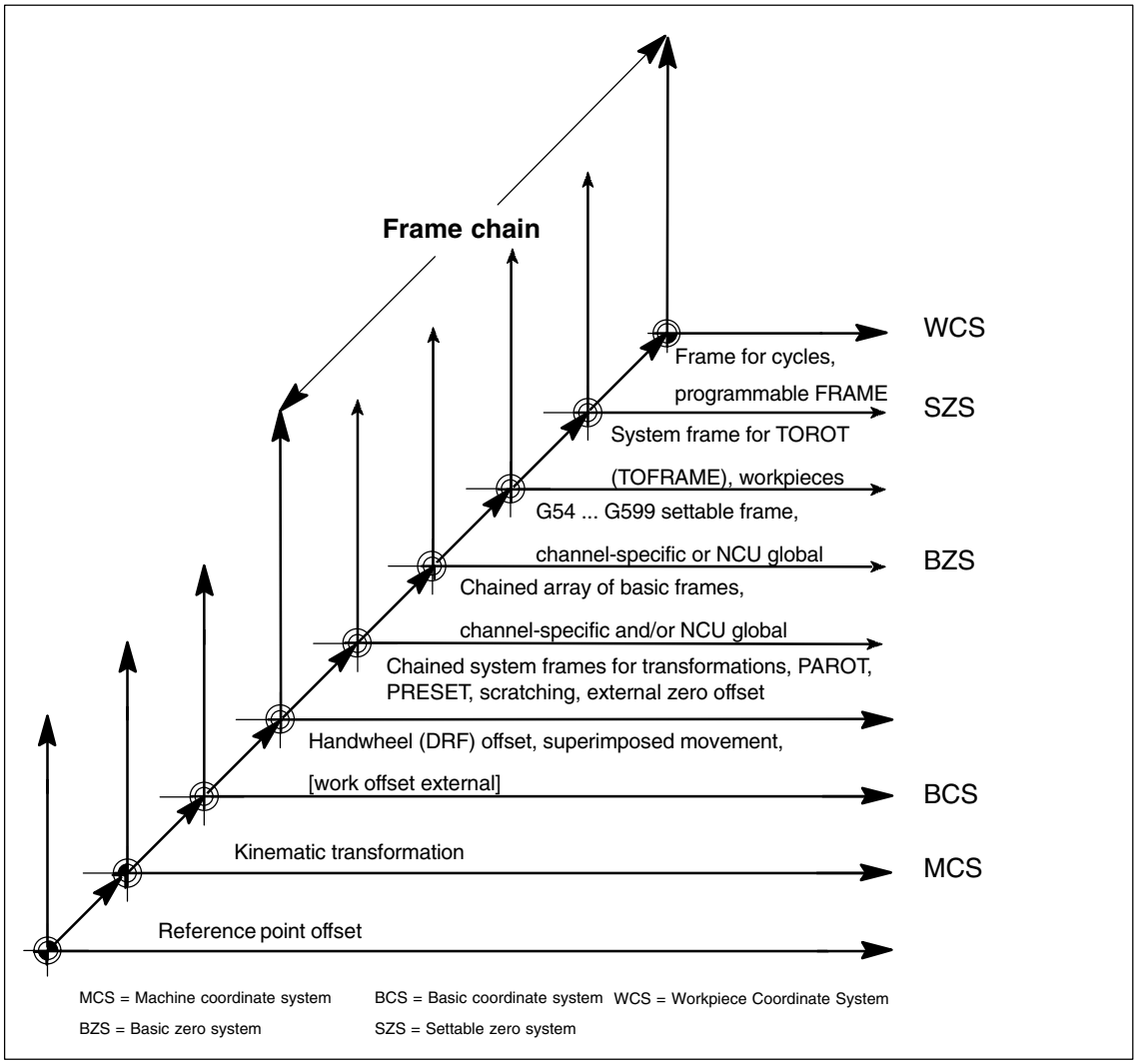


Fig. 2-12 Interrelationship between coordinate systems

© Siemens AG, 2003. All rights reserved

2.3.1 Machine coordinate system (MCS)

The machine coordinate system (MCS) is made up of all physically available machine axes.

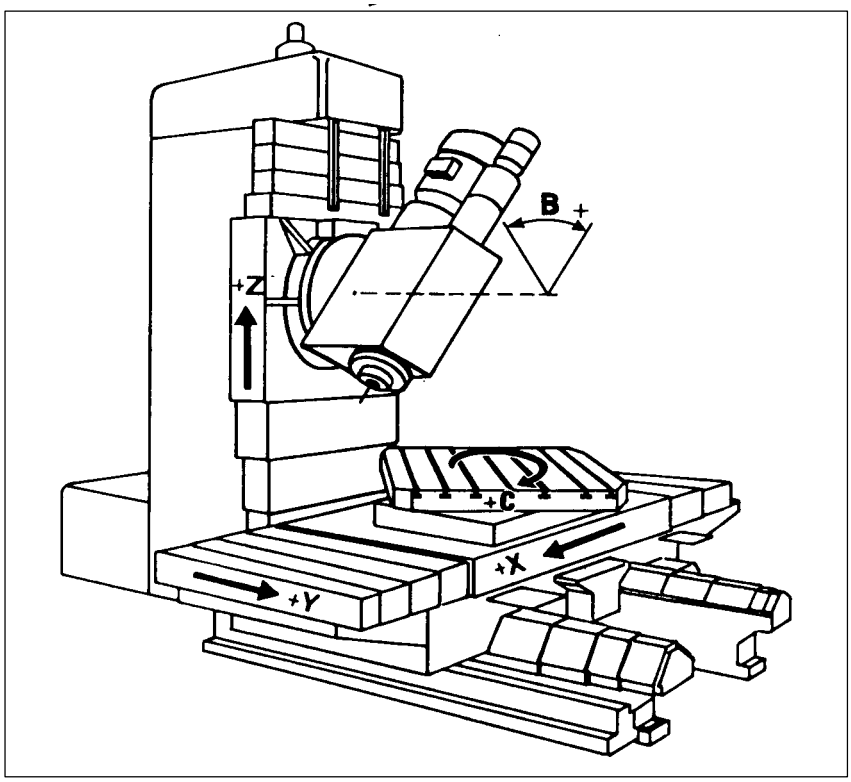


Fig. 2-13 MCS with machine axes X, Y, Z, B, C (5-axis milling machine)

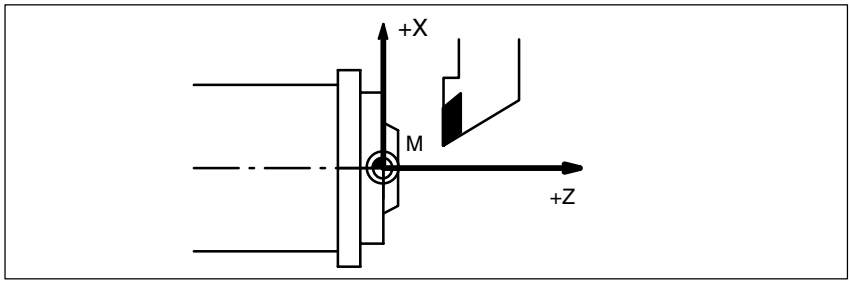


Fig. 2-14 MCS figure with machine axes X, Z (turning machine)

Axial preset offset

The "*Preset*" function can be used to define a new control zero in the machine coordinate system. The preset values act on machine axes. The axes do not move with "*Preset*".

Warning

After Preset, the reference points are invalid!
If possible do not use this function.

© Siemens AG, 2003. All rights reserved

2.3.2 Basic coordinate system (BCS)

The basic coordinate system (BCS) consists of three mutually perpendicular axes (geometry axes) as well as other special axes which are not interrelated geometrically.

Machine tools without kinematic transformation

The BCS and the MCS always coincide when the BCS can be mapped onto the MCS without kinematic transformation (e.g. TRANSMIT/face transformation, 5-axis transformation and up to three machine axes).

On such machines, machine axes and geometry axes can have the same names.

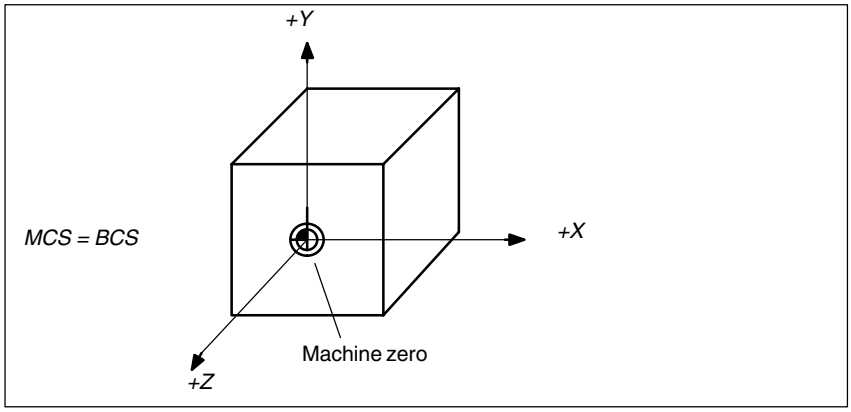


Fig. 2-15 MCS=BCS without kinematic transformation

Machine tools with kinematic transformation

The BCS and the MCS do not coincide when the BCS is mapped onto the MCS with kinematic transformation (e.g. TRANSMIT/face transformation, 5-axis transformation or more than three axes).

On such machines the machine axes and geometry axes must have different names.

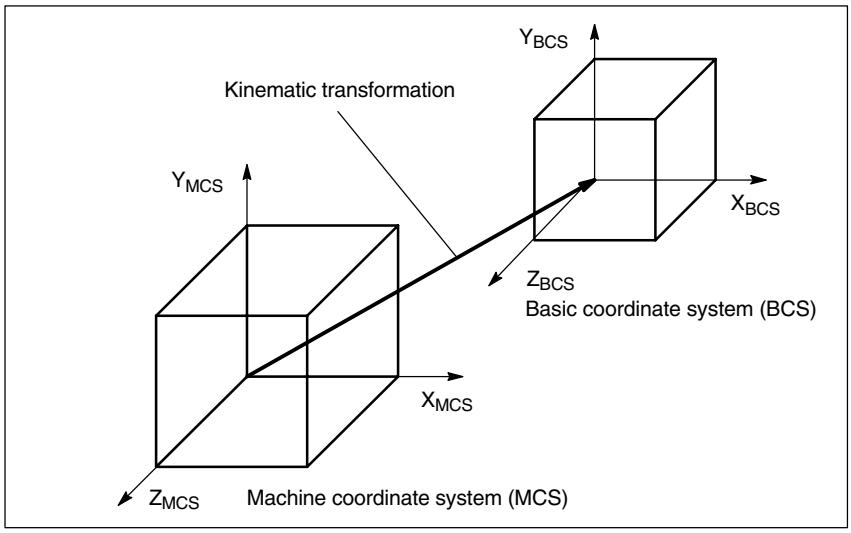


Fig. 2-16 Kinematic transformation between the MCS and BCS

© Siemens AG, 2003. All rights reserved

Machine kinematics

The workpiece is always programmed in a two or three-dimensional, right-angled coordinate system (WCS). However, such workpieces are being programmed ever more frequently on machine tools with rotary axes or linear axes not perpendicular to one another. Kinematic transformation is used to represent coordinates programmed in the workpiece coordinate system (rectangular) in real machine movements.

References: /FB/, F2, “5-axis transformation”
/FB/, M1, “Kinematic transformation”

2.3.3 Additive offsets

Work offsets external

The work offset external is an axial offset: Unlike with frames, no components for rotation, scaling and mirroring are possible.

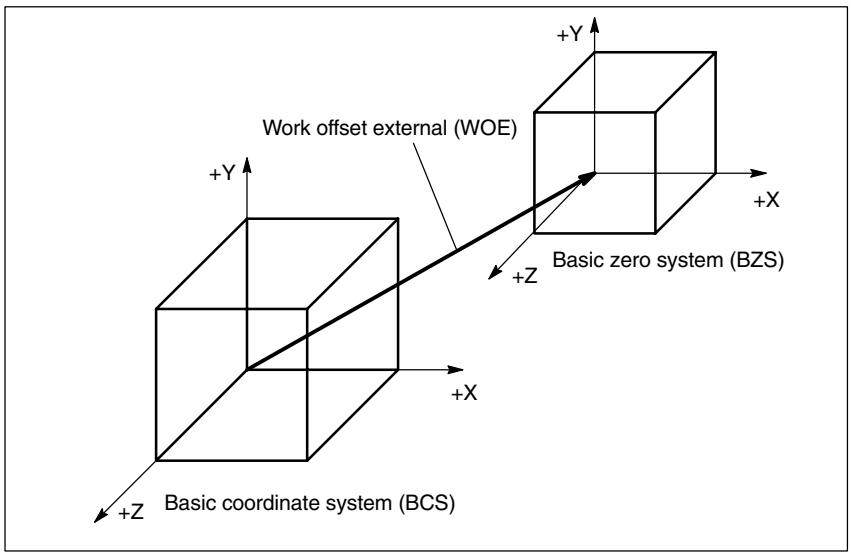


Fig. 2-17 Zero offset external between BCS and BZS

Setting the offset values

The offset values can be set:

- PLC
  - By describing system variables
- Via the operator panel
  - From menu “Current zero offsets”
- By the NC program
  - Assignment to system variables $AA_ETRANS[axis]

Activating the offset values

The 0/1 edge of the PLC signal “Accept external zero offset” (DB31, ... DBX3.0) activates the offset values defined previously.

© Siemens AG, 2003. All rights reserved

Effect of activation

The offset for an axis becomes active when the first motion block for this axis is executed after the offset is activated.

Example for possible sequence:

```
G0 X100
X150        ; A new work offset external is activated
            ; by the PLC during this movement
X200        ; due to G0 programming, the new work offset external
            ; is traversed at the end of the block (X200) if no
            ; velocity reserve exists (override 100%).
```

SW 6.1 and higher
The work offset external applied via system frame is applied immediately.
Channel-specific system frames can be configured via

MD 28082: MM_SYSTEM_FRAME_MASK.

Programming

Setting a new offset:

Via axis-specific system variables
$AA_ETRANS[axis]=$R_i$

Reading:

The following instruction reads the axis-specific **active** offset value.

R_i=$AA_ETRANS[axis]

Note

- The value read can be different from the value set if the value set has not been activated.
  The value read is a previously set value if the last value set has not yet been activated.
- The system frame for the work offset external exists only if it has been configured.

DRF offset

The DRF (Differential Resolver Function) offset allows an additional incremental zero offset to be set using the handwheel. It acts in the basic coordinate system on geometry and special axes. This function can be used, for example, to compensate for tool wear within a programmed block. The DRF offset is read

- via axis-specific system variable $AC_DRF[axis].

Overlaid movements

The overlaid movement for the programmed axis can be accessed only from

- synchronized actions via system variable $AA_OFF[axis].

Runup

After run-up (POWER ON) the last used offset values for the external zero offset are stored and become effective once more with a renewed activation signal.

In SW 6.1 and higher, system frames are retained according to MD 24008: CHSFRAME_POWERON_MASK on POWER ON.

© Siemens AG, 2003. All rights reserved

RESET/ end of program

The activated values remain active after RESET and program end.

Reset response of channel-specific system frames with SW 6.1 and higher as follows:
When MD 24006: CHSFRAME_RESET_MASK, Bit 1 = 1
system frame for work offset external is active after RESET.

When MD 24006: CHSFRAME_RESET_MASK, Bit 1 = 0
the work offset external in the active system frame is deleted in the data management.

In SW 6.3 and higher, the following frames are active after RESET:
System frame for
MD 24006: CHSFRAME_RESET_MASK, Bit 4 = 1 Workpiece zero points
MD 24006: CHSFRAME_RESET_MASK, Bit 5 = 1 Cycles

Suppression

The NC program instruction “SUPA” suppresses the work offset external while the block is being processed.

The command G74 (reference point approach) and the equivalent operator actions in reference point approach mode suppress the work offset external for the duration of reference point approach.

With G74, i.e. Automatic or MDA mode, the previously active work offset external automatically becomes active again with the next traverse motion in the block.

After a mode change from reference point approach mode, the VDI signal for the referenced axes must set for reactivation.

2.3.4 Basic zero system (BZS)

BZS

The basic zero system (BZS) is the basic coordinate system with a basic offset.

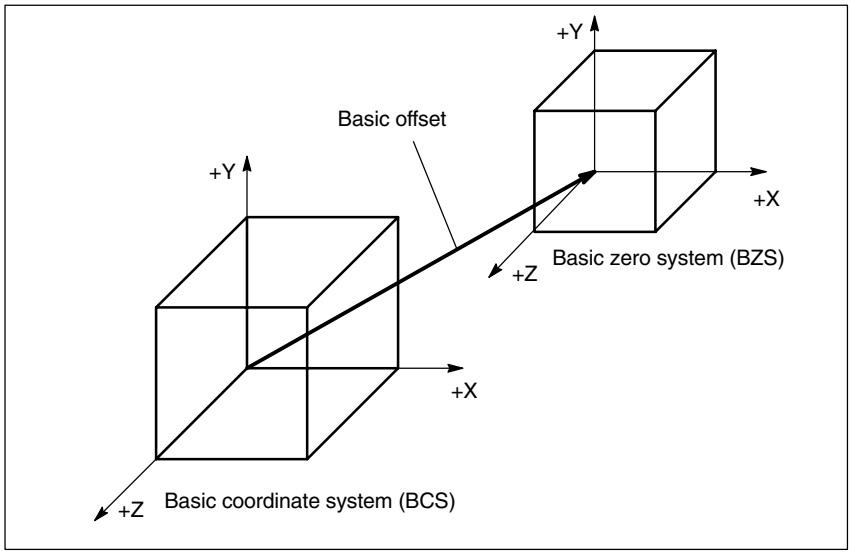


Fig. 2-18 Basic offset between BCS and BZS

© Siemens AG, 2003. All rights reserved

Basic offset

The basic offset defines the coordinate transformation between the BCS and BZS.
It can be used, for example, to define the pallet zero point.

The basic offset comprises:

- Zero offset
- DRF offset
- Overlaid movement
- Chained system frames
- Chained basic frames

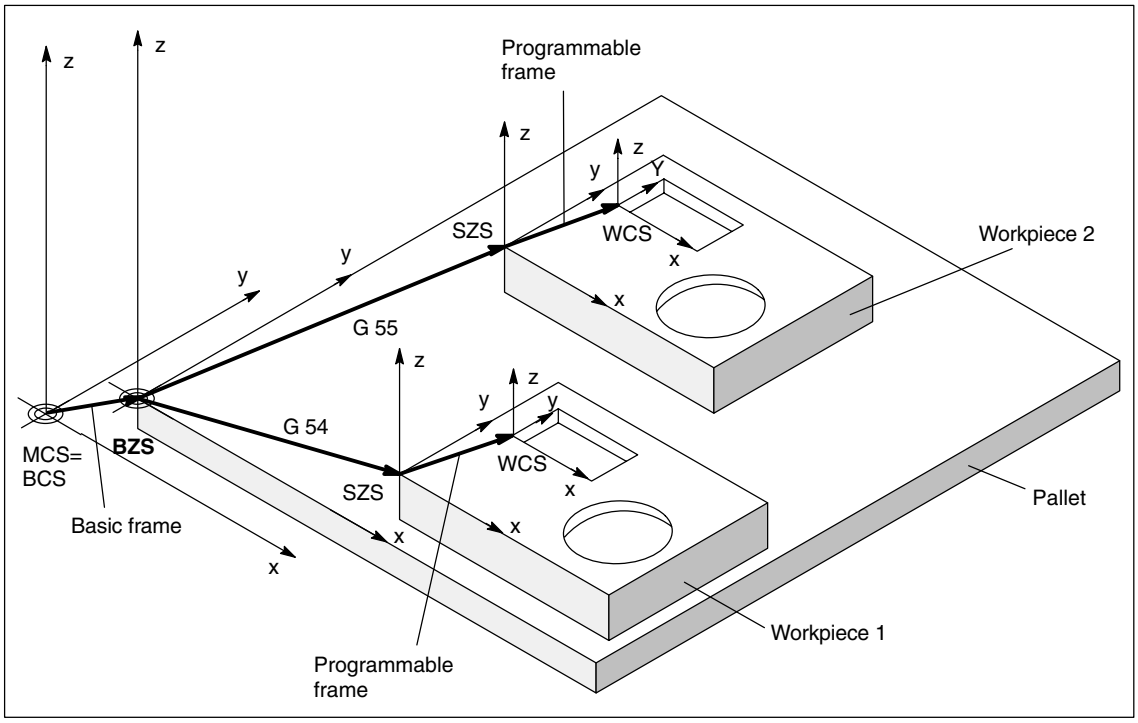


Fig. 2-19 Example of the use of the basic offset

The following applies:

- The user can change the basic offset from the parts program by means of an operator action and from the PLC.
- If the basic offset is to take effect immediately, an ASUB can be started via the PLC using FC9 in order to execute the appropriate G code.

Machine Manufacturer

Recommendation:
Use offsets starting at the 3rd basic offset. The 1st and 2nd basic offset are provided for setting the actual value and for the work offset external.

© Siemens AG, 2003. All rights reserved

2.3.5 Settable zero system (SZS)

SZS

The settable zero system (SZS) is the workpiece coordinate system WCS with a programmable frame (viewed from the perspective of the WCS). The workpiece zero is defined by settable FRAMES G54...G599.

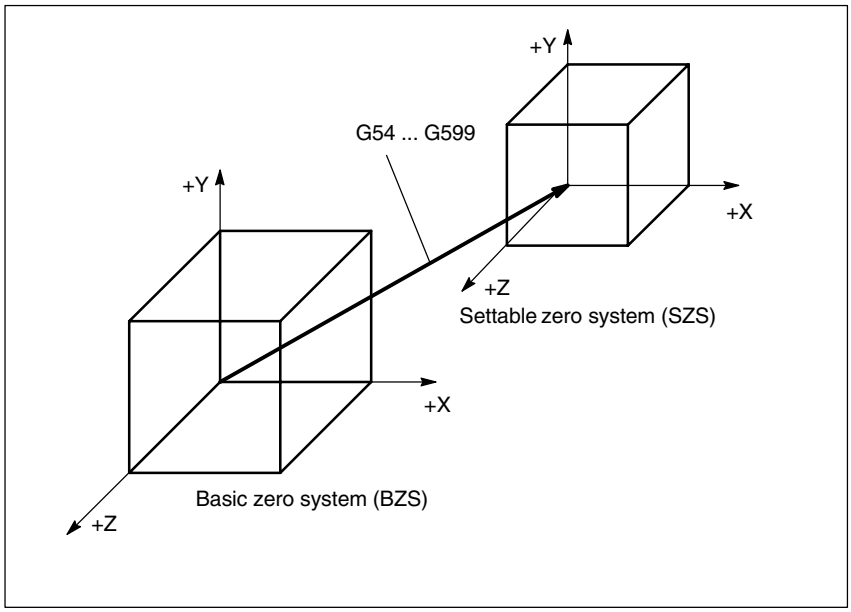


Fig. 2-20 Settable FRAME G54 ... G599 between BZS and SZS

Programmable offsets act on the settable zero system. All programmable offsets refer to the settable zero system.

SZS display at the MMC (SW 5 and higher)

The display machine data MD 9424: MA_COORDINATE_SYSTEM can be set to specify whether the WCS actual values must be displayed in the WCS or SZS. The coordinate system continues to be known as the WCS in the display.

- MD 9424: MA_COORDINATE_SYSTEM = 0: Display as before
- MD 9424: MA_COORDINATE_SYSTEM = 1: Display in SZS

Example:

Program	Display for 0:	Display for 1:
X100	100	100
X0	0	0
$P_PFRAME = CTRANS(X,10)	0	10
X100	100	110

© Siemens AG, 2003. All rights reserved

2.3.6 Configurable SZS (SW 6.4 and higher)

Application

The function of the SZS coordinate system is to display actual values and move the axes during a cycle interruption. Cycles utilize frames in the frame chain to perform their functions. They insert translations or rotations either in the

- programmed frame $P_PFRAME, or in the
- cycle system frame $P_CYCFRAME.

The WCS is therefore modified by cycles. A cycle interrupted by the user should thus also be executed in the programmed WCS. The SZS can be used to display the last valid coordinates.

MD 24030: FRAME_ACS_SET can then be set to define whether the SZS must be interpreted with or without the currently programmed frame $P_PFFRAME (see Figs. 2-21 and 2-22). The default setting is 1 and should not be changed.

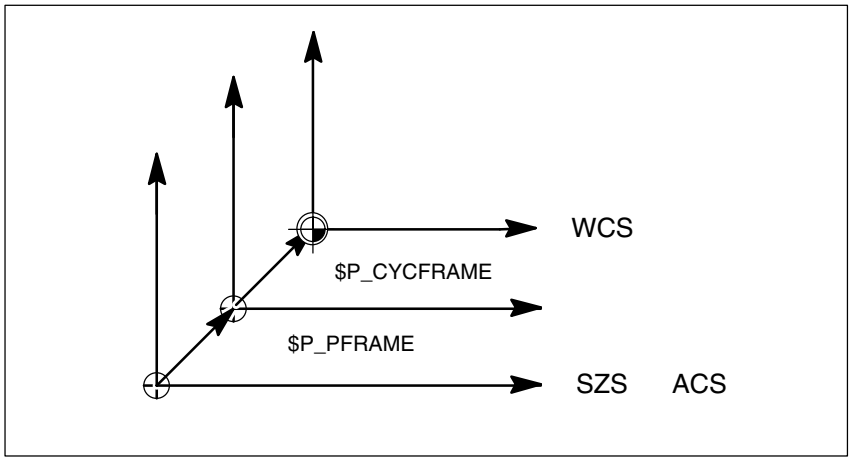


Fig. 2-21 SZS when MD 24030: FRAME_ACS_SET = 0

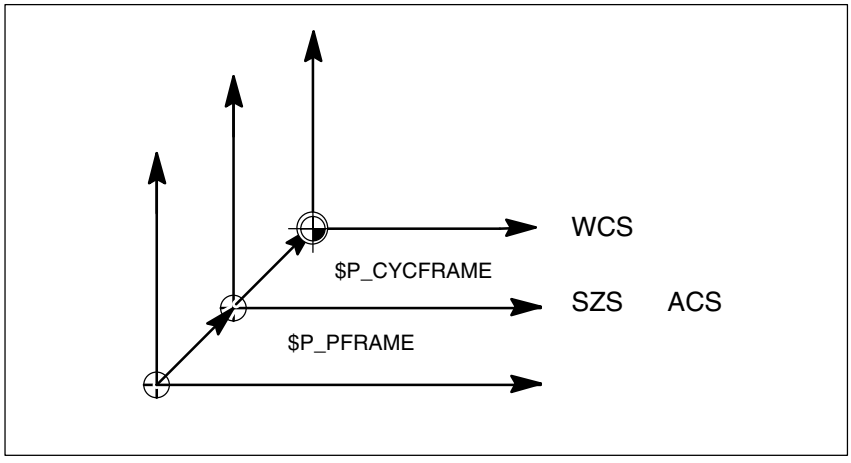


Fig. 2-22 SZS when MD 24030: FRAME_ACS_SET = 1

© Siemens AG, 2003. All rights reserved

Manual traverse in SZS (SW 6.4 and higher)

In SW 6.3 and lower, geometry axes traverse in the WCS in jog mode.
In SW 6.4 and higher, axes can be traversed manually in the SZS coordinate system as well. Using variable $AC_JOG_COORD it is possible to switch between manual traverse in the WCS or SZS.

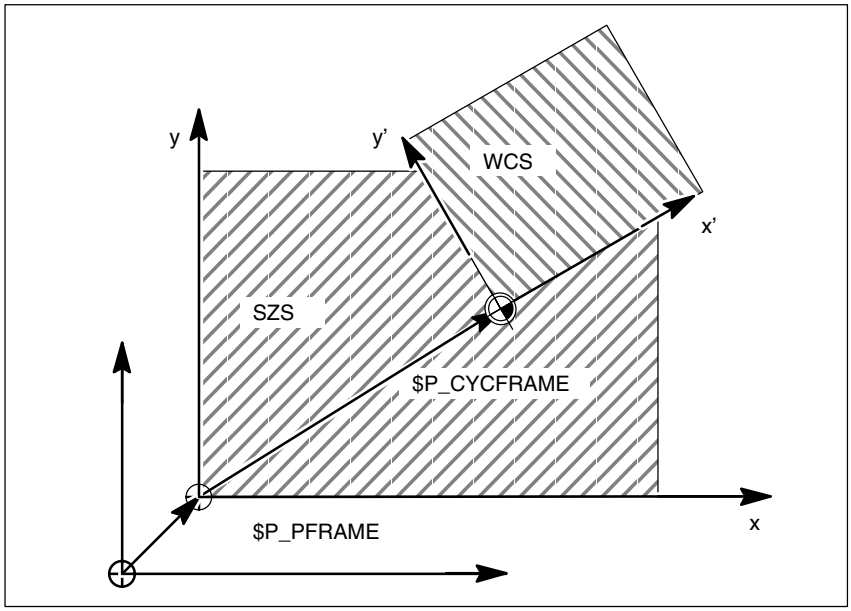


Fig. 2-23 Manual traverse in WCS or SZS

2.3.7 Workpiece coordinate system (WCS)

WCS

The workpiece coordinate system (WCS) is the programming basis.

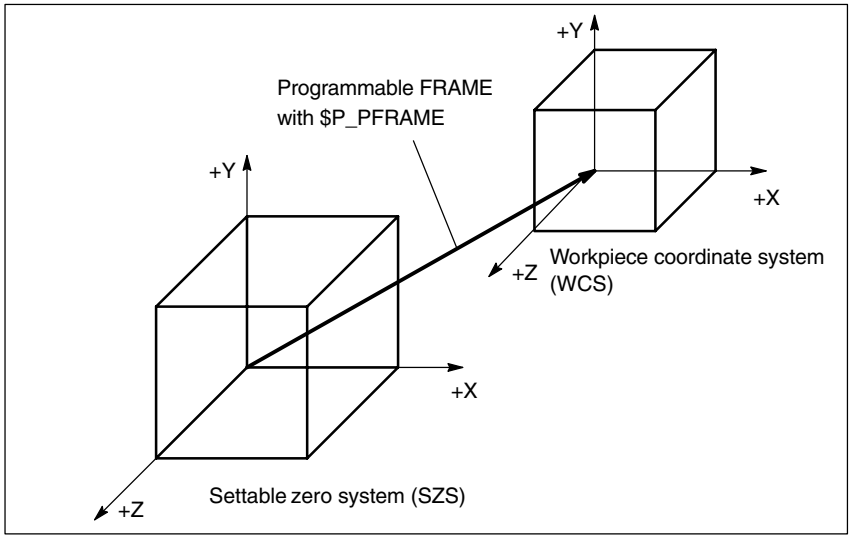


Fig. 2-24 Programmable FRAME between SZS and WCS

© Siemens AG, 2003. All rights reserved

2.4 Frames

2.4.1 Overview

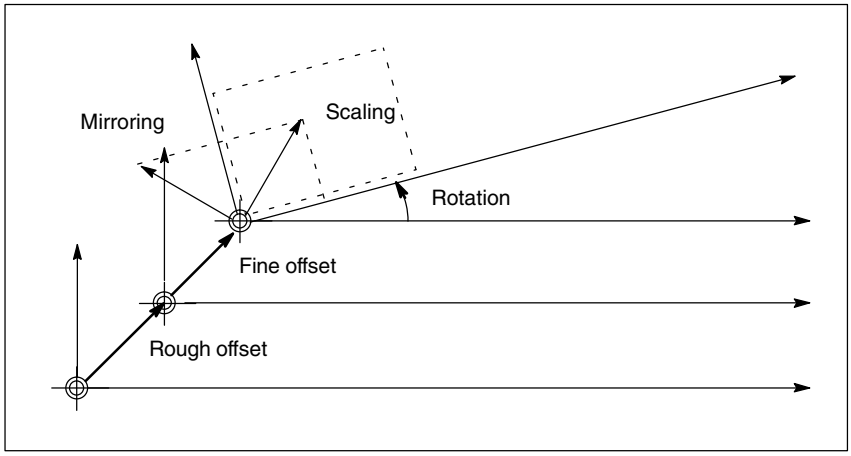


Fig. 2-25 Coordinate transformation by FRAME arithmetic

A FRAME is a closed calculation rule that translates one Cartesian coordinate system into another.

A FRAME consists of the following components:

- Offset
  Rough offset:
  –> Programmable with TRANS,
  –> ATRANS (additive translation component) and
  –> CTRANS (zero offset for several axes)
  –> G58 (axial zero offset).

  Fine offset:
  –> Programmable with CFINE and
  –> G59 (axial zero offset).
- Rotation –> Programmable with ROT, AROT and
  –> with ROTS, AROTS and CROTS in SW 5.3 and higher
- Scaling –> Programmable with SCALE and ASCALE
- Mirroring –> Programmable with MIRROR and AMIRROR

Note

For further explanations regarding the programming of FRAMES, please refer to:

References: /PG/, “Programming Guide: Fundamentals”

Features in relation to axes

The rough and fine offsets, scaling and mirroring can be programmed for geometry and special axes. A rotation can also be programmed for geometry axes.

© Siemens AG, 2003. All rights reserved

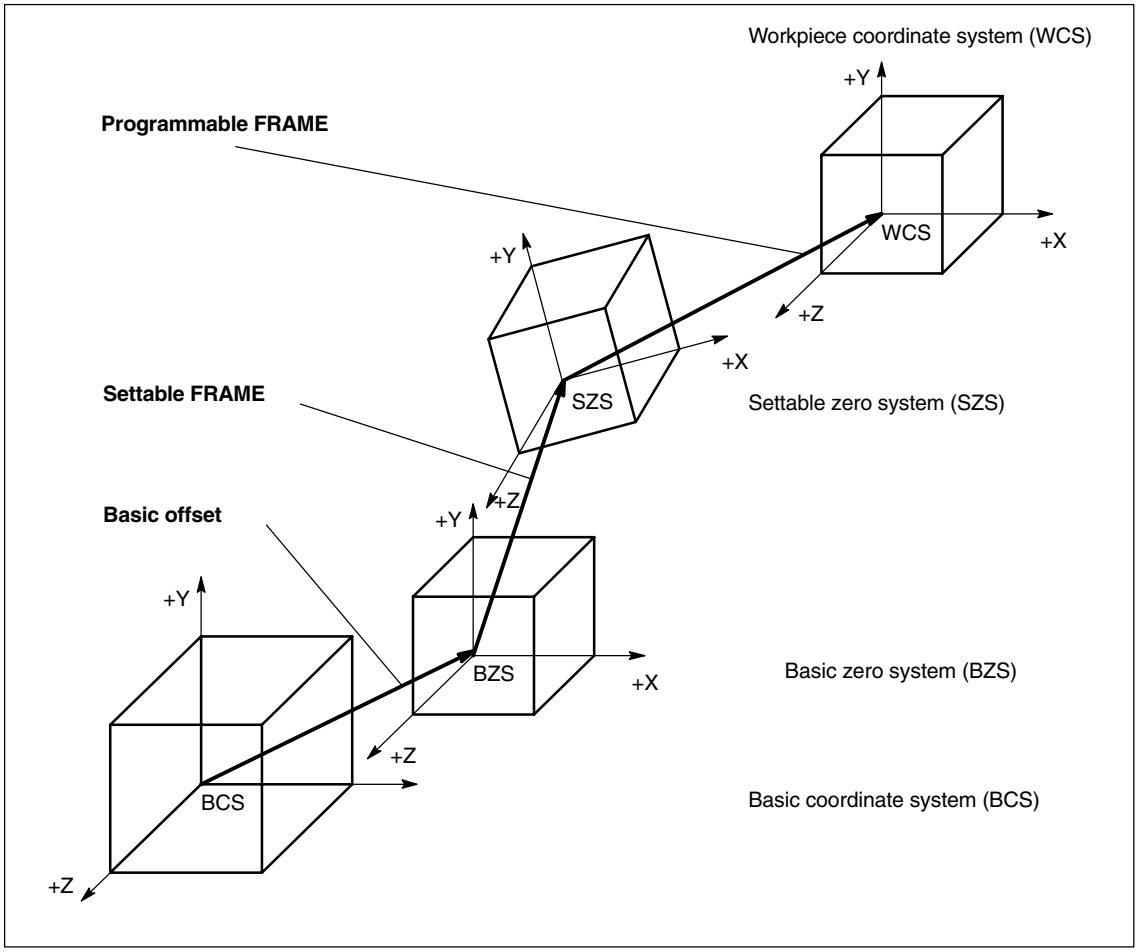


Fig. 2-26 Coordinate systems by frame transformation

2.4.2 Rough and fine offsets (translations)

The translation component of FRAMES comprises:

- Rough offset
  Defined by the tool setter. The access rights can be restricted via MMC input.
- Fine offset
  Can be defined by the operator via the MMC within certain input limits.
  The fine offset option is set with MD 18600: MM_FRAME_FINE_TRANS.

The rough and fine offsets together represent the total offset.

TRANS, ATRANS and CTRANS

The programmable offsets for all geometry axes and special axes are programmed with TRANS.

An additive programmable offset is programmed with ATRANS, and the zero offset for several axes with CTRANS.

© Siemens AG, 2003. All rights reserved

G58,G59
SW 5 and higher

Axial programmable zero offset

G58 and G59 can be programmed to replace the rough and fine offsets of the programmable frame on an axial basis. These functions can only be used if the fine offset is configured.

MD 24000 and MD 18600

G58 and G59 can be used only if
MD 24000: FRAME_ADD_COMPONENTS = 1 (TRUE)
(i.e. the separation absolute – additive is permitted). Otherwise, the alarm "18311 Channel %1 Block %2 Frame: Instruction not allowed" is output.

The function can be used for the programmable frame only in conjunction with a configured fine offset in MD 18600: MM_FRAME_FINE_TRANS. If G58 or G59 is used without a configured fine offset, alarm "18312 Channel %1 Block %2 Frame: Fine offset not configured" is output.

Rough offset

G58: The absolute component can be modified via TRANS, CTRANS; this resets the additive component ATRANS to zero.

G58 changes only the absolute translation component (rough offset) for the specified axis; the total of additively programmed translations (fine offset) is retained.

```
G58 X... Y... Z... A... ...          ; Modify absolute component
```

Fine offset

The fine offset can be configured in the following variants with MD 18600: MM_FRAME_FINE_TRANS:

0: The fine offset cannot be entered or programmed. G58 and G59 are not possible.

1: Fine offset for all frames, G58 and G59 can be entered or programmed.

G59 is used for axial overwriting of the additively programmed translations for the specified axes which were programmed with ATRANS.

```
G59 X... Y... Z... A... ...          ; Modify additive component
```

Example

```
N1   TRANS X10 Y10 Z10          ; Absolute default
N5   ATRANS X5 Y5               ; Total translation X15 Y15 Z10
N10  G58 X20                    ; Total translation X25 Y15 Z10
N15  G59 X10 Y10                ; Total translation X30 Y20 Z10
```

Table 2-2 Explanation of example

Absolute component		+ Additive component		= Total translation
X10 Y10 Z10	**N1**	X5 Y5	**N5**	X15 Y15 Z10
X20 Y10 Z10	new from **N10**	X5 Y5		X25 Y15 Z10
X20 Y10 Z10		X10 Y10	new from **N15**	X30 Y20 Z10

© Siemens AG, 2003. All rights reserved

The absolute component of the translation is stored in the rough offset component and the additive translation component is stored in the fine offset component.

The table below shows the effect of various program commands on the absolute and additive translation.

Command	Coarse or absolute offset	Fine or additive offset	Comment
TRANS X10	10	Unchanged	Absolute offset for X
ATRANS X10	Unchanged	Fine (old) + 10	Additive offset for X
CTRANS(X,10)	10	0	Offset for X; (if appropriate: Further axis, value)
CTRANS()	0	0	Deselection of offset
CFINE(X,10)	0	10	Fine offset in X
$P_PFRAME[X,TR] = 10	10	Unchanged	Progr. Offset in X
$P_PFRAME[X,FI] = 10	Unchanged	10	Progr. Fine offset in X
G58 X10	10	Unchanged	Overwrites absolute offset for X
G59 X10	Unchanged	10	Overwriting additive offset for X

Table 2-3 Effect on additive/absolute offset (translation)

2.4.3 Frame rotations

Orientations in space can be defined via frame rotations for specific applications as follows:

- Rotation with — ROT defines individual rotations for all geometry axes. It is possible to define any spatial orientation using three rotations.

SW 5.3 and higher:

- Solid angles with — ROTS, AROTS, CROTS for orientation of a plane in space. Specification of maximum two solid angles.
- Frame rotation with — TOFRAME defines a frame with a Z axis pointing in the tool direction.
- Rotation component — TOROT leaves the other components in the programmed frame unchanged. Only the rotation component is overwritten.

Rotation ROT

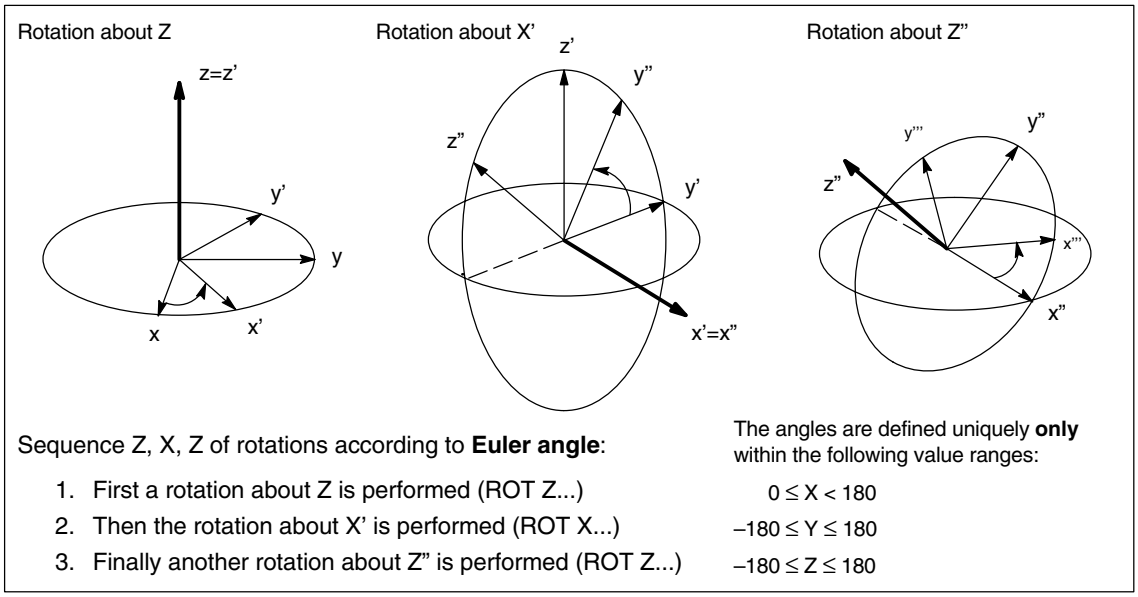
ROT defines the programmable individual rotations for all geometry axes. Any spatial orientation can be defined on the basis of three rotations. To ensure that written angles are returned unambiguously, it is absolutely essential to apply the defined value ranges.

MD 10600

If more than one rotation is programmed in a block, MD 10600: FRAME_ANGLE_INPUT_MODE (input mode of rotation for FRAME) is set to define the rotations about the three geometry axes. The order in which these rotations are programmed within the block is irrelevant.

© Siemens AG, 2003. All rights reserved

The rotations can be calculated according to **Euler angle** or **RPY**.



Sequence Z, X, Z of rotations according to **Euler angle**:

1. First a rotation about Z is performed (ROT Z...)
2. Then the rotation about X' is performed (ROT X...)
3. Finally another rotation about Z" is performed (ROT Z...)

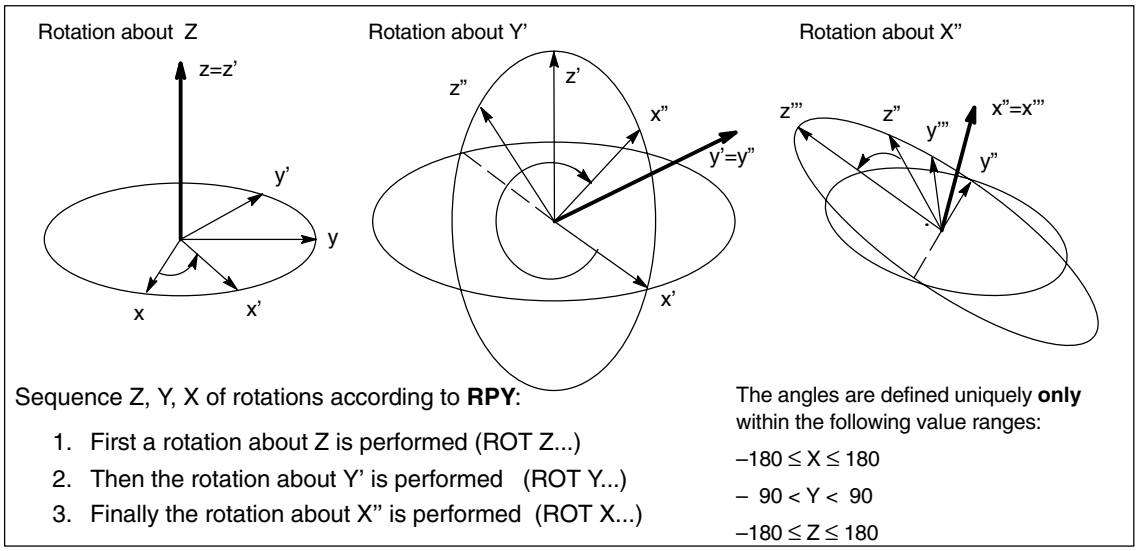
The angles are defined uniquely **only** within the following value ranges:

$0 \le X < 180$

$-180 \le Y \le 180$

$-180 \le Z \le 180$

Fig. 2-27 ROT with EULER angle when MD 10600: FRAME_ANGLE_INPUT_MODE = 2



Sequence Z, Y, X of rotations according to **RPY**:

1. First a rotation about Z is performed (ROT Z...)
2. Then the rotation about Y' is performed (ROT Y...)
3. Finally the rotation about X" is performed (ROT X...)

The angles are defined uniquely **only** within the following value ranges:

$-180 \le X \le 180$

$-90 < Y < 90$

$-180 \le Z \le 180$

Fig. 2-28 ROT with RPY when MD 10600: FRAME_ANGLE_INPUT_MODE = 1

© Siemens AG, 2003. All rights reserved

Note

When frame rotation components are read and written, the value range limits must be observed to ensure that the same results are obtained for read or write, or repeat write operations. When rotations that are larger than the specified angles are entered, these are converted to a mode of representation that does not exceed the specified range limits.

The setting RPY is recommended for rotations programmed in parts programs at the control (not with work preparation and/or postprocessor).

© Siemens AG, 2003. All rights reserved

SW 5.3 and higher

Frame rotations with solid angles

Linking of individual rotations

Where a frame is to be defined to describe a rotation around more than one geometry axis, this has so far been achieved through conversion of the solid angles into the angles of rotation of a series of individual rotations. A new rotation is hereby always performed in the already rotated coordinate system. This applies both when programming in a block, e.g. with ROT X.... Y... Z..., and where a frame is built up in several blocks, e.g. in the form

ROT, AROT

N10 ROT Y...
N20 AROT X...
N30 AROT Z...

In workpiece drawings, see Section 2.4 “Frames” oblique surfaces are generally described by way of solid angles, i.e. the angles which the intersection lines of the oblique plane form with the main planes (X–Y, Y–Z, Z–X planes), see Fig. 2-29.

ROTS, AROTS, CROTS

The commands ROTS, AROTS and CROTS are used to program rotations of the tool table/workpiece using solid angles.

For further details about programmable rotations ROT, AROT, ROTS, AROTS and CROTS, please see:
References: /PG/, “Programming Guide: Fundamentals”

Solid angles

The orientation of a plane in space is defined unambiguously by specifying two solid angles. The third solid angle is derived from the first two. Therefore, a maximum of 2 solid angles may be programmed, e.g. in the form ROTS X10Y15. If a third solid angle is specified, an alarm will be triggered.

It is permissible to specify a single solid angle. The rotations which are performed with ROTS or AROTS in this case are identical to those for ROT and AROT.

An extension of the existing functionality arises only in cases where exactly two solid angles are programmed.

The specification of the solid angle does not define the orientation of the two-dimensional coordinate system within the plane (i.e. the angle of rotation around the surface normal vector). The position of the coordinate system is thus determined so that the rotated first axis lies in the plane which is surrounded by the first and third axes of the non-rotated coordinate system, i.e.

- When programming X and Y the new X-axis lies in the old ZX plane.
- When programming Z and X the new Z-axis lies in the old YZ plane.
- When programming Y and Z the new Y-axis lies in the old XY plane.

If the required coordinate system does not correspond to this basic setting, then an additional rotation must be performed with AROT.

© Siemens AG, 2003. All rights reserved

MD 10600

The programmed solid angles are converted into the equivalent RPY or Euler angles at input, depending on MD 10600: FRAME_ANGLE_INPUT_MODE. These then appear also in the display.

Definition einer rotated plane

If exactly two solid angles are programmed, a rotated plane is defined as follows in SW 5.3 and higher:

- Both programmed axes define a plane,
- the non-programmed axis define the associated third axis of a righthand coordinate system.

It is thus also unambiguously defined which of the two programmed axes is the first and which is the second axis (the definition corresponds to the plane definition with G17/G18/G19). The angle programmed with the axis letter of an axis of the plane then gives the axis around which the other axis of the plane must be rotated to move this into the line of intersection which the rotated plane forms with the plane surrounded by the other and the third axis. Should one of the two programmed angles tend to zero, the defined plane becomes the plane which results when only one axis is programmed (e.g. also with ROT or AROT).

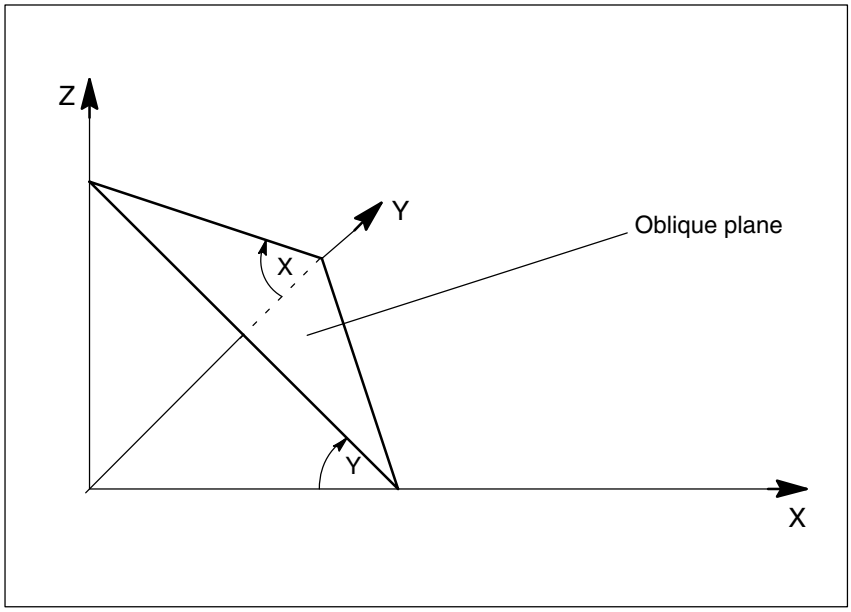


Fig. 2-29 Definition of a rotated plane with two solid angles

The diagram above shows an example where X and Y are programmed. Y here gives the angle by which the X axis must rotate around the Y axis to bring the X axis to the line of intersection formed by the oblique plane and the XZ plane. The same principle applies for the programmed value of the angle X.

Note

In the shown position of the oblique plane the value of Y is positive, that of X on the other hand negative.

© Siemens AG, 2003. All rights reserved

SW 5.3 and higher

Frame rotation in tool direction

TOFRAME

With the existing NC command TOFRAME it is possible to define a frame whose Z axis points in the tool direction. An already programmed FRAME is then overwritten by a frame which describes a pure rotation.

Any

- zero offsets,
- mirrorings or
- scaling

programmed in the previously active frame are deleted.

If a zero offset defining the reference point in the workpiece is to be kept, the NC command TOROT can be used.

TOROT

The NC command TOROT overwrites only the rotation component in the programmed frame, leaving the remaining components unchanged. The rotation defined by TOROT is the same as that defined with TOFRAME. TOROT is like TOFRAME independent of the availability of an orientational toolholder. This NC command is also especially useful for orientation transformations (3, 4, 5-axis transformations).

Note

NC command TOROT ensures consistent programming with active, orientational toolholders for every type of kinematic.
See also the description of NC command PAROT in

/FB/, W1, “Orientational Toolholders, Machine with Rotatable Table”.

Separate system frame in SW 6.1 and higher

The FRAMES generated by TOFRAME or TOROT can be written to their own, separate system frame $P_TOOLFR. Bit 3 in

MD 28082: MM_SYSTEM_FRAME_MASK must be set for this purpose.

The programmable frame is then retained unchanged. Changes occur when the programmable frame is processed elsewhere. For further details, see Subsection 2.4.7 “Channel-specific system frames”.

Activation

If a rotating frame is already active before NC command TOFRAME or TOROT is activated, the newly defined frame should deviate as little as possible from the old frame. This is the case, for example, if a defined frame needs to be modified slightly because the tool orientation cannot be set freely on account of Hirth-toothed rotary axes.

NC command TOFRAME or TOROT uniquely defines the Z direction of the new frame. The rotation around the Z axis is at first unimportant. With setting data SD 42980: TOFRAME_MODE this free rotation can be determined so that the newly defined frame deviates as little as possible from a previously active frame.

© Siemens AG, 2003. All rights reserved

SD 42980

This setting data determines the direction of the X or Y axis for frame definition using TOFRAME or TOROT.

In all cases where the setting data is unequal to zero, an active frame remains unchanged if the Z direction of the old and the new frame are identical. The values 0 to 3 have the following meanings:

SD 42980 = 0
Corresponds to the existing SW 5.2 and lower.
The orientation of the coordinate system is determined by the value of machine data MD 21110: X_AXIS_IN_OLD_X_Z_PLANE.

SD 42980 = 1
The new X direction is chosen to lie in the XZ plane of the old coordinate system. In this setting the angle difference between the old and new Y axis will be minimal.

SD 42980 = 2
The new Y direction is chosen to lie in the YZ plane of the old coordinate system. In this setting the angle difference between the old and new X axis will be minimal.

SD 42980 = 3
The value chosen is the mean value of the two settings which would have been chosen with 1 and 2.

Linking of frames

If the value 1000 is added to the above values of the setting data, then the tool frame is chained to an active basic frame and any settable frames. The behavior is thus compatible to earlier SW (before 5.3).

Characteristics and expansions

Value of setting data SD 42980 = 1 and SD 42980 = 2
These set frames are achieved by rotating the coordinate system about the new Z axis from any X and Y axis position

- until the desired setting is reached.

Value of setting data SD 42980 = 3
These set frames are achieved by executing a rotation whose amount equals the mean value of these two angles.

- This applies for the case that the old and new Z direction enclose an angle of less than 90 degrees.

When the frames are set by the value of setting data SD 42980 = 1

- the angle enclosed by the new and old X axes is less than 90 degrees.

The same applies for the Y axis (the axes concerned point “approximately” in the same direction) for frames by the setting data value SD 42980 = 2.

If the two Z directions form an angle of more than 90 degrees, however, the conditions of an angle < 90 degrees between the old and new axes can no longer be met simultaneously for both X and Y. In this case priority is given to the X direction, i.e. a mean value is taken from the direction for SD 42980 = 1 and the negative direction for SD 42980 = 2.

© Siemens AG, 2003. All rights reserved

Example

```
...
...
N90 $SC_TOFRAME_MODE=1
N100 ROT Z45
N110 TCARR=1 TCOABS T1 D1
N120 TOROT
...
...
```

N100 describes a rotation by 45 degrees in the XY plane. It is assumed that the toolholder activated in N110 rotates the tool by 30 degrees around the X axis, i.e. the tool lies in the YZ plane and is rotated by 30 degrees relative to the Z axis. As a result the Z axis of the frame newly defined in N120 also points in this direction (independently of the value in setting data SD 42980: *TOFRAME_MODE* in N90).

Fig. 2-30 illustrates the scenario for $SC_TOFRAME_MODE = 1: The old and new X axes X and X' coincide in the projection in the direction of the old Z axis. The old and new Y axes Y and Y' form an angle of 8.13 degrees (right angles are generally not retained in the projection!).

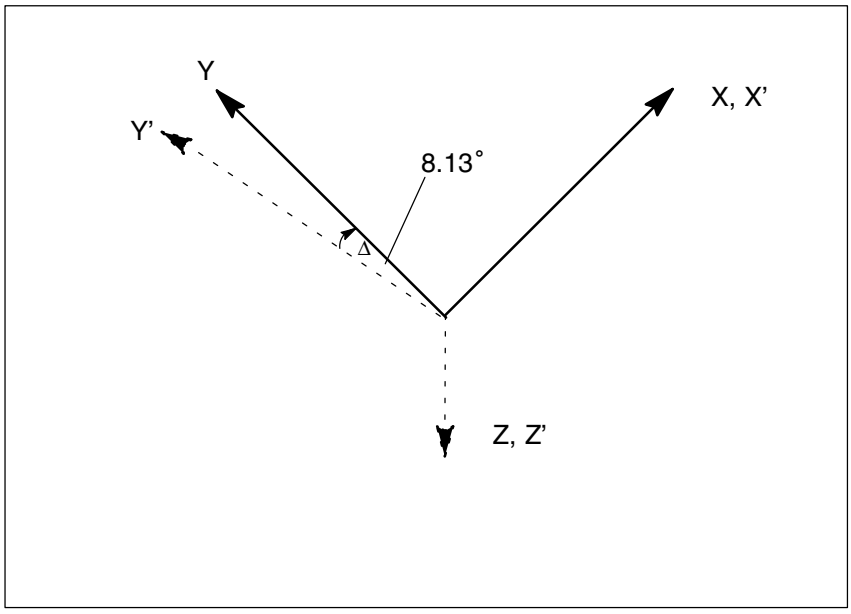


Fig. 2-30 Coordinate systems with TOROT and $SC_TOFRAME_MODE = 1

For SD 42980: TOFRAME_MODE = 2, similarly, Y and Y' would coincide and X and X' would form an angle of 8.13 degrees.

For SD 42980: TOFRAME_MODE = 3 the pairs X/X' and Y/Y' would each form an angle of 4.11 degrees.

Note

The named angles (8.13 and 4.11 degrees) are the angles which the **projections** of the axes form in the XY plane. They are **not the spatial angles** of these axes.

© Siemens AG, 2003. All rights reserved

2.4.4 SCALE

SCALE is used to program the programmable scale factors for all geometry axes and special axes.

ASCALE must be programmed if a new scaling is to be based on a previous scaling, rotation, translation or mirroring.

2.4.5 MIRROR

MIRROR is used to program the programmable mirrors for all geometry axes and special axes.

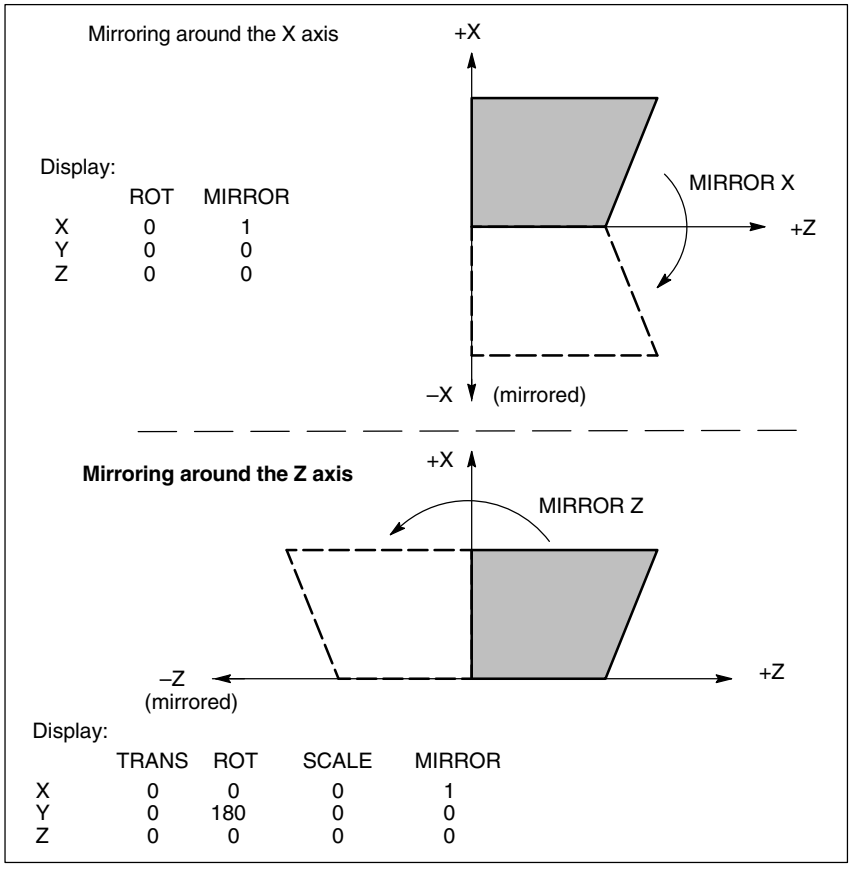


Fig. 2-31 MIRROR function in SW 1.1 and higher

AMIRROR must be programmed if a new mirroring is to be based on a previous mirroring, rotation, translation or scaling.

MIRROR (up to and including SW 4)

In SW 4 and lower, the specified value of the coordinate axis, e.g. the value 0 for AMIRROR X0, is not evaluated, but the AMIRROR has a toggle function, i.e. AMIRROR X0 activates the mirror and a further AMIRROR X0 deactivates it.

© Siemens AG, 2003. All rights reserved

MIRROR (SW 5 and higher)

The MD 10612: MIRROR_TOGGLE = 0 can be used to specify that the programmed values are evaluated. A value of 0, i.e. AMIRROR X0, deactivates the mirroring of the axis, and values not equal to 0 cause the axis to be mirrored if it is not already mirrored.

It is possible to read or write mirrors component by component independent of MD 10612: MIRROR_TOGGLE.

- A value = 0 means that the axis is subsequently not mirrored
  Example: $P_NCBFR[0,x,mi] = 0; x axis is not mirrored
- A value =1 means that the axis is always mirrored, regardless of whether it was already mirrored.
  Example: $P_NCBFR[0,x,mi] = 1; x axis is always mirrored

The MD 10610: MIRROR_REF_AX can be used to define the axis around which the mirroring operation is applied:

MD 10610 = 0: Mirroring is performed around the programmed axis.

MD 10610 = 1 or 2 or 3:
Depending on the input value the mirroring is mapped onto a mirroring of a specific reference axis and rotation of two other geometry axes.

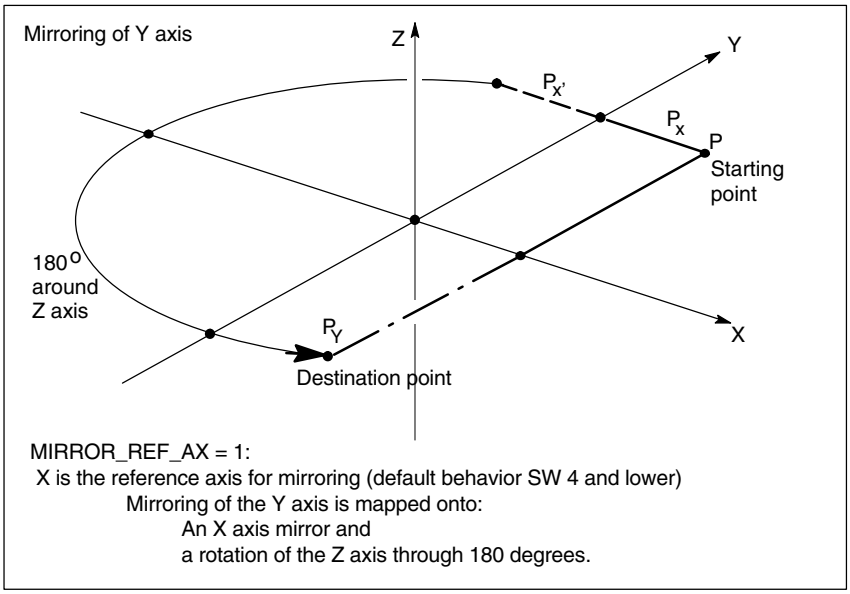


Fig. 2-32 Example of effect of MD 10610: MIRROR_REF_AX on mirroring of the Y axis

© Siemens AG, 2003. All rights reserved

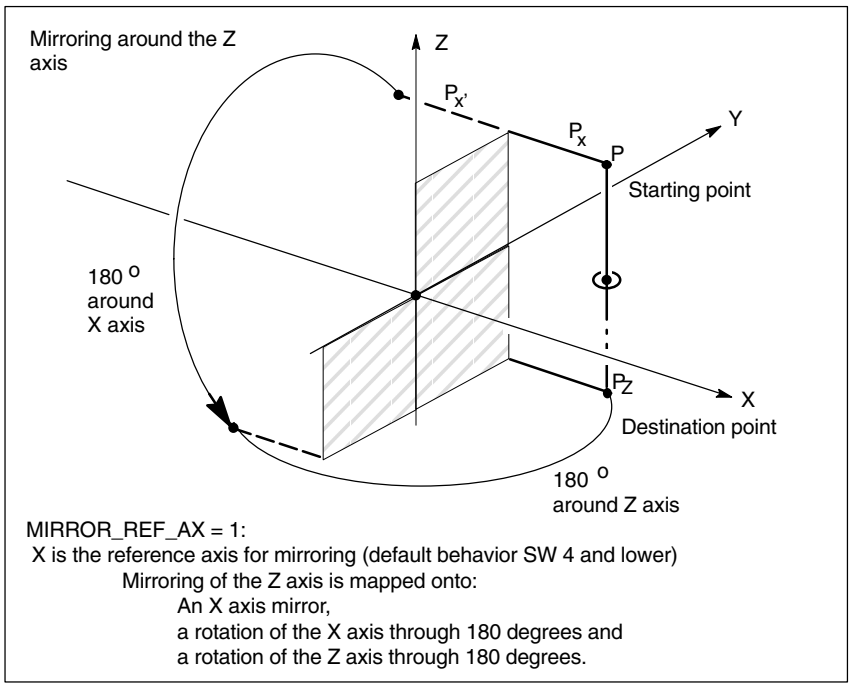


Fig. 2-33 Example of effect of MD 10610: MIRROR_REF_AX on mirroring of the Z axis

Adapting active frames

Prevent offset values and frames for axes

MD 32074: FRAME_OR_CORRPOS_NOTALLOWED can be used to define how offset values or an active frame actually affect the associated axis. Setting the individual bits can achieve the following

- Programmed zero offset (TRANS) do **not** affect indexing axes.
- Scale changes (SCALE) do **not** affect indexing axes.
- Change of direction (MIRROR) does **not** affect indexing axes.
- DRF offset does **not** affect an axis.
- External zero offset does **not** affect an axis.
- Online tool offset does **not** affect an axis.
- Synchronized action offset does **not** affect an axis.
- Compile cycle offset does **not** affect an axis.

Axial frames and tool length offsets are **not** considered for

- PLC axes for compatibility reasons if the bit is not set.
- Command axes if the bit is set.

© Siemens AG, 2003. All rights reserved

2.4.6 Frames and switchover of geometry axes (SW 5 and higher)

The geometry axis configuration can change in the channel on activation and deactivation of a transformation and on the GEOAX() command.

The machine data MD 10602: FRAME_GEOAX_CHANGE_MODE can be used to configure for all channels of the system whether the current complete frame is calculated again on the basis of the new geometry axes or whether the complete frame is deleted. Four modes can be set via:

1. MD 10602: FRAME_GEOAX_CHANGE_MODE = 0

   The current complete frame is deleted when geometry axes are switched over, when transformations are selected and deselected, and on GEOX(). The modified geometry axis configuration is not used until a new frame is activated.

2. MD 10602: FRAME_GEOAX_CHANGE_MODE = 1

   The current complete frame is calculated again when the geometry axes are switched over, and the translations, scales and mirrors of the new geometry axes are effective.

   The rotations of the geometry axes, which were programmed before the switchover, remain effective for the new geometry axes.

3. MD 10602: FRAME_GEOAX_CHANGE_MODE = 2

   The current complete frame is calculated again when the geometry axes are switched over, and the translations, scales and mirrors of the new geometry axes are effective.

   If rotations are active before the switchover in the current basic frame, the current settable frame or the programmable frame, the switchover is canceled with alarm 18313: “Frame: Geometry axis switchover not allowed”.

4. MD 10602: FRAME_GEOAX_CHANGE_MODE = 3

   The current frame is deleted when selecting and deselecting transformations. With GEOAX(), the current complete frame is calculated again, and the translations, scales and mirrors of the new geometry axes come into effect. The rotations of the geometry axes, which were programmed before the switchover, remain effective for the new geometry axes.

Workpiece geometry

The workpiece geometry is described by a coordinate system that is formed by the geometry axes. A channel axis is assigned to each geometry axis and a machine axis is assigned to each channel axis (unique assignment).

An axial frame exists for each machine axis and for each frame (basic frame, settable frame, programmable frame). When a new machine axis is assigned to a geometry axis, the axial frame components of the machine axis, such as translations (rough and fine), scales and mirrors of the appropriate frame are applied. The new geometry in the channel is then generated by the new contour frames resulting from the geometry axes (up to three in number).

The current valid frames are calculated again on the geometry axis switchover and a resulting complete frame is generated. The data management frames are not included unless they are activated.

For examples on the channel axis to be converted to a geometry axis and on the 5-axis orientation transformation, see **Section 6.3** “Examples”, Frames.

© Siemens AG, 2003. All rights reserved

2.4.7 Channel-specific system frames (SW 6.1 and later)

System frames

System frames are described by the following system functions:

1. Preset, scratching
2. Zero offset
3. Oblique machining with 3 + 2 axes
4. Frame rotations in tool direction
5. Workpiece reference points (SW 6.3 and higher)
6. Cycles (SW 6.3 and higher)

There are up to six system frames per channel whereby each occupies approx. 1KB SRAM and approx. 6KB DRAM per channel.

The system frame for preset actual value memory and scratching and the system frame for cycles are the default. The system frames are defined via the

MD 28082: MM_SYSTEM_FRAME_MASK in the channel

and are stored in the SRAM. System frames can be written and read in the program. Only system frames required for system functions should be configured. A non-configured system frame cannot be written; if it is nevertheless written, it will not be defined with the alarm 12550 “Channel %1 Block %2 Name %3” or it is denied with “Option does not exist”.

System frames in data management

All write operations to system frames must be executed using system functions. For cycle programmers, it has been made possible to write the frame using the variables below. All system frames in the data management are enabled either directly by the system function (TOROT, PAROT) or by a G function for “Settable zero offset” (G500, G54.. G599 statement).

Table 2-4 System frames per channel via $MC_MM_SYSTEM_FRAME_MASK

Bit	VB	System functions	Variable
0	1	For PRESET and scratching (**Set**-**Fr**ame)	$P_SETFR
1	0	For work offset external (**Ext**-**Fr**ame)	$P_EXTFR
2	0	With orientational toolholder (**Part**-**Fr**ame) TCARR and PAROT	$P_PARTFR
3	0	Frame rotation in tool direction (**Tool**-**Fr**ame) TOROT and TOFRAME	$P_TOOLFR
4	0	Frame for workpiece reference points (**W**ork-**P**iece-**F**rame)	$P_WPFR
5	1	Frame for cycles (**Cyc**le frame)	$P_CYCFR

VB: Presetting of bits in system frame

© Siemens AG, 2003. All rights reserved

2.4.8 Configuring the current system frames (SW 6.1 and higher)

The current system frames are the system frames active in the main run. An appropriate current system frame exists for each system frame in the data management. Only with the activation of the data management frame, the values are taken into account with regard to the preprocessing.

The current system frame can be read and written for the appropriate system function via the system variables in the parts program. The following current system frames exist:

- $P_PARTFRAME For TCARR and PAROT
- $P_SETFRAME For PRESET and scratching
- $P_EXTFRAME For work offset external
- $P_TOOLFRAME For TOROT and TOFRAME
- $P_WPFRAME For workpiece reference points (SW 6.3 and higher)
- $P_CYCFRAME For cycles (SW 6.3 and higher)

PRESET and scratching

This system frame is a standard frame and should always remain configured with bit 0 as this is the prerequisite setting for operating PRESET and scratching. This system frame is activated per default with

MD 28082: MM_SYSTEM_FRAME_MASK activated with **bit 0 = 1.**

Note

For the various measurement methods associated with scratching, please refer to:

References: /FB/, M5, “Measurements”, Workpiece and tool measurements

---

Work offset external via system frames

The work offset external can be managed and enabled via a system frame. This system frame is configured with

MD 28082: MM_SYSTEM_FRAME_MASK and **activated** with **bit 1 = 1.**

If the appropriate bit is set in the machine data, the system function will only use the system frame, disabling the existing functionality. The values of the work offset external are written directly to the reserved frame by the PLC (OPI) or in the parts program with $AA_ETRANS[axis].

The activation is carried out as previously via the IS “Accept external yero offset” (DB31, ... DBX3.0) by the PLC. The existing functionality of the work offset external is described in more detail in **Subsection 2.3.3** “Additive offsets”.

Sequence after change of IS level (DB31, ... DBX3.0)

If a level change of the axis signal from 0 to 1 is detected, the movement is stopped immediately, the preprocessing is reorganized and the current system frame, and the system frame in the data management, are written and activated with the axis value of $AA_ETRANS[axis]. As a result, the zero offset is first applied in order to then continue the interrupted motion.

With active G91 and MD 42440: FRAME_OFFSET_INCR_PROG = 0 the work offset external is always applied immediately with the approach block.

© Siemens AG, 2003. All rights reserved

The zero offset external acts on the absolute translation absolutely (coarse offset) of the current system frame. A multiple activation of an work offset external does not act additively, and only the rough portion of the translation is overwritten with the value $AA_ETRANS[axis].

Oblique machining with 3 + 2 axes

The system frame for TCARR and PAROT is configured with

MD 28082: MM_SYSTEM_FRAME_MASK and **activated** with **bit 2 = 1.**

MD 20184: TOCARR_BASE_FRAME_NUMBER is only evaluated if the system frame for TCARR and PAROT has **not** been configured.

With kinematic systems of the types P and M, TCARR will enter the table offset of the orientational toolholder (zero offset resulting from the rotation of the table) as a translation into the system frame.

PAROT converts the system frame such that a partoriented WCS results.

The system frames are stored in the SRAM and therefore remain stored after Reset. The system frames also remain active in the case of a mode change.

For the display, the G codes PAROT and TOROTOF, TOFRAME are each assigned to a separate G code group.

Note

For further explanations on the functions TCARR, TOROT and PAROT on machines with rotating work table, see:

References: /FB/, W1, “Tool offset”, orientable toolholder
/PGA/, “Tool compensations”

Frame rotations in tool direction

The system frame for TOROT and TOFRAME is configured with

MD 28082: MM_SYSTEM_FRAME_MASK and **activated** with **bit 3 = 1.**

This system frame is located before the programmable frame in the frame chain. The SZS coordinate system is located before the programmable frame.

When a system frame is configured for TOROT and TOFRAME, the programmable frame is retained unchanged. If the programmable frame is processed further, however, the following changes in relation to the programmed version occur with or without a system frame:

- Without system frames, any zero offsets, mirrorings or scalings in the previously active frame are deleted.

The following example shows another difference:
TRANS is programmed after TOROT. TRANS without specified parameters deletes the programmable frame.

- In the variant **without** system frame, the rotation component of the programmable frame caused by TOROT is also deleted.
- When TOROT is configured **in the** system frame, then the rotation component of the programmable frame caused by TOROT is not deleted.

© Siemens AG, 2003. All rights reserved

PAROTOF and TOROTOF

PAROTOF will delete the rotation in the system frame. The current $P_PARTFRAME and the data management frame are also deleted. TR = 0 additionally deletes the translation.

TOROTOF will delete the system frame for TOROT and TOFRAME. The current $P_PARTFRAME and the data management frame are also deleted.

Workpiece reference points in SW 6.3 and higher

Workpiece reference points can be set and read in the current system frame $P_WPFRAME if the frame has been activated via

MD 28082: MM_SYSTEM_FRAME_MASK when **bit 4 = 1**.

If the current system frame is not configured via MD 28082: MM_SYSTEM_FRAME_MASK, all previously set workpiece reference points are cleared in the current system frame $P_WPFRAME.

Frame for cycles in SW 6.3 and higher

Cycles can be set and read in the current system frame $P_CYCFRAME if it has been activated via

MD 28082: MM_SYSTEM_FRAME_MASK when **bit 5 = 1**

If the current system frame is not configured via MD 28082: MM_SYSTEM_FRAME_MASK, then the current system frame $P_CYCFRAME turns to a zero frame.

SW 6.4 and higher

For future system frame functions, in SW 6.4 and higher, bit 6 is reserved in machine data

MD 28082: MM_SYSTEM_FRAME_MASK.

For this reason, bit 6 must not be configured and activated.

© Siemens AG, 2003. All rights reserved

2.4.9 NCU global basic frames (SW 5 and higher)

Only one set of NCU global frames is used for all channels on each NCU. NCU global frames can be read and written from all channels. The NCU global frames are activated in the respective channel.

NCU global frames can be used to apply offsets, scale factors and mirroring operations to channel and machine axes.

With NCU global frames, there is no geometrical relationship between the axes. It is therefore not possible to perform rotations or program geometry axis identifiers.

$P_NCBFR[n] **NCU global basic frames**

Up to 16 NCU global basic frames can be configured.

The number of global basic frames is configured in MD 18602: MM_NUM_GLOBAL_BASE_FRAMES.
Channel-specific basic frames can also exist, as set in MD 18081: MM_NUM_BASE_FRAMES.

Global frames can be read and written from all channels of an NCU. When writing global frames, the user must ensure channel coordination. This can be achieved using wait markers (WAITMC), for example.

$P_UIFR[n] **NCU global settable frames**

All settable frames G500, G54...G599 can be configured either NCU-globally or channel-specifically.

All settable frames can be redefined as global frames with
MD 18601: MM_NUM_GLOBAL_USER_FRAMES.
MD 18601 > 0 No channel-specific settable frames
MD 28080: MM_NUM_USER_FRAMES is not evaluated.

Programming

Channel axis identifiers and machine axis identifiers can be used as axis identifiers in frame program commands. The programming of geometry axis identifiers is denied with an alarm.

Rotations cannot be used on global frames. The programming of a rotation is denied with alarm: “18310 Channel %1 Block %2 Frame: Rotation not allowed”.

It is possible to chain global frames and channel-specific frames. The resulting frame contains all frame components including the rotations for all axes. The assignment of a frame with rotation components to a global frame is denied with alarm “Frame: Rotation not allowed”.

© Siemens AG, 2003. All rights reserved

2.4.10 Channel-specific basic frames

$P_CHBFR[n]

The MD 28081: MM_NUM_BASE_FRAMES can be used to configure the number of the basic frames in the channel. The standard configuration is designed for at least one base frame per channel. A maximum of 16 basic frames per channel is possible. In addition to the 16 basic frames in the channel, a further 16 NCU global basic frames can exist.

System variable $P_CHBFR[n] can be used to read and write the basic frames. When a basic frame is written, the chained total basic frame is not activated until the execution of a G500, G54..G599 instruction. The variable is used primarily for storing write operations to the basic frame from MMC or PLC. These frame variables are saved by the data backup.

$P_UBFR

First basic frame in the channel
A write operation to the predefined variable $P_UBFR does not activate the basic frame with array index 0 simultaneously; the activation does not take place until the execution of a G500, G54..G599 instruction. The variable can also be read and written in the program.

$P_UBFR is identical to $P_CHBFR[0].

One basic frame always exists in the channel by default, so that the system variable is compatible with older versions. If there is no channel-specific basic frame, the alarm “Frame: Instruction not allowed” is output on a read or write access.

Programming

Settable frames or basic frames can be read and written by an operator action or from the PLC:

- via the parts program, or
- via the operator panel interface

via the operator panel interface.
The fine offset can also be used for global frames.
The suppression of global frames is performed in the same way as for channel-specific frames, using G53, G153, SUPA and G500
(see /PG/ and /PGA/ Programming Guide, Chapter Frames).

2.4.11 Frames active in the channel

$P_PARTFRAME
$P_SETFRAME
$P_EXTFRAME

Current system frames for TCARR and PAROT, PRESET and scratching, zero offset external (SW 6.1 and higher)
The current system frame can be read and written via these system variables in the parts program.

$P_NCBFRAME[n]

Current NCU global basic frames (n = 0 to 15)
System variable $P_NCBFRAME[n] can be used to read and write the array elements of the current global basic frame. The resulting complete basic frame is calculated in the channel as a result of the write operation.

© Siemens AG, 2003. All rights reserved

The modified frame is activated only in the channel in which the frame was programmed. If the frame is to be modified for all channels of an NCU, $P_NCBFR[n] and $P_NCBFRAME[n] must be written simultaneously. The other channels must then activate the frame, e.g. with G54. Whenever a basic frame is written, the complete basic frame is calculated again.

$P_CHBFRAME[n] **Current basic frames (n = 0 to 15) in the channel**

System variable $P_CHBFRAME[n] can be used to read and write the array elements of the current channel basic frame. The resulting complete basic frame is calculated in the channel as a result of the write operation. Whenever a basic frame is written, the complete basic frame is calculated again.

$P_BFRAME **Current first basic frame in the channel**

The current basic frame with array index 0, which is valid in the channel, can be read and written in the parts program using the predefined frame variable $P_BFRAME. The written basic frame is immediately included in the calculation.

$P_BFRAME is identical to $P_CHBFRAME[0]. The system varaible always has a valid default value. If there is no channel-specific basic frame, the alarm “Frame: Instruction not allowed” is output on a read or write access.

$P_ACTBFRAME **Current chained total basic frame**

The variable $P_ACTBFRAME is used to determine the chained total basic frame. The variable is readonly.

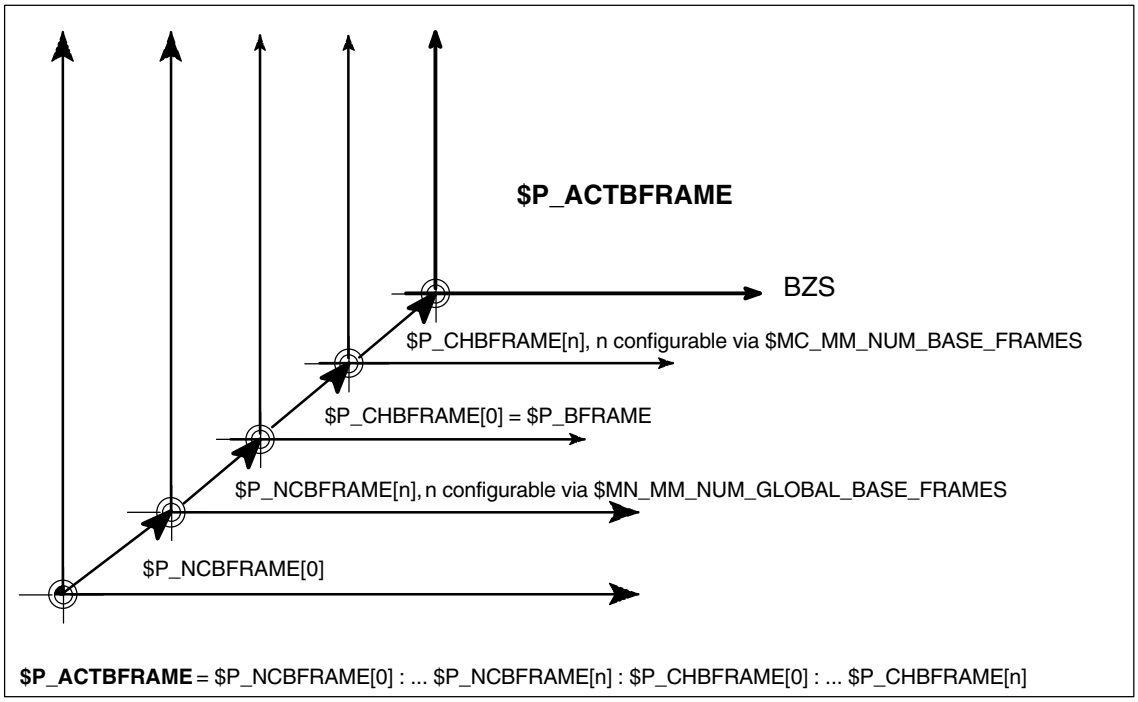


Fig. 2-34 Chained array of basic frames

© Siemens AG, 2003. All rights reserved

$P_CHBFRMASK
$P_NCBFRMASK

System variables $P_CHBFRMASK and $P_NCBFRMASK can be used to select which basic frames to include in the calculation of the “total” basic frame. The variables can only be programmed in the program and read via the operator panel interface. The value of the variables is interpreted as a bit mask and specifies which basic frame array element of $P_ACTBFRAME is included in the calculation.

$P_CHBFRMASK can be used to define which channel-specific basic frames are included, and $P_NCBFRMASK can be used to define which NCU global basic frames are included in the calculation.

When the variables are programmed, the total basic frame and the total frame are calculated again. After a reset and in the default setting, the value of $P_CHBFRMASK = $MC_CHBFRAME_RESET_MASK and $P_NCBFRMASK = $MN_NCBFRAME_RESET_MASK.

e.g.
$P_NCBFRMASK = ’H81’ ; $P_NCBFRAME[0] : $P_NCBFRAME[7]

$P_CHBFRMASK = ’H11’ ; $P_CHBFRAME[0] : $P_CHBFRAME[4]

For details of the relationship between coordinate systems and chained array of basic frames, please see:

Subsection 2.4.12 “FRAME chaining”

$P_IFRAME

Current settable frame

The predefined frame variable $P_IFRAME can be used to read and write the current settable frame, which is valid in the channel, in the parts program. The written settable frame is immediately included in the calculation.

In the case of NCU global settable frames, the modified frame acts only in the channel in which the frame was programmed. If the frame is to be modified for all channels of an NCU, $P_UIFR[n] and $P_IFRAME must be written simultaneously. The other channels must then activate the corresponding frame, e.g. with G54.

$P_TOOLFRAME
$P_WPFRAME

Current system frame for TAROT and TOFRAME (SW 6.1 and higher), tool reference points (SW 6.3 and higher)

The current system frame can be read and written in the parts program via these system variables.

$P_PFRAME

Current programmable frame

$P_PFRAME is the programmable frame produced as a result of programming TRANS/ATRANS, G58/G59, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR or by the assignment of CTRANS, CROT, CMIRROR, CSCALE to the programmable FRAME.

Current programmable frame variable which establishes the reference between the settable zero system (SZS) and the workpiece coordinate system (WCS).

In SW 6.1 and higher, frame instructions such as axis functions for a spindle can be programmed with command SPI(n). Example of a frame instruction:

$P_PFRAME[SP(1), TR] = 2.22

In SW 6.3 and higher, the programmable frame can be retained on Reset with MD 24010: PFRAME_RESET_MODE = 1. Preferred application for this functionality: Tool retraction from an oblique hole after a Reset.

© Siemens AG, 2003. All rights reserved

$P_CYCFRAME **Current system frame for cycles (SW 6.3 and higher)**
The current system frame can be read and written in the parts program via these system variables.

$P_ACTFRAME **Current total frame**
The resulting current total frame $P_ACTFRAME is now a chain of all system frames, basic frames, the current settable frame and the programmable frame. The current frame is always updated whenever a frame component is changed.

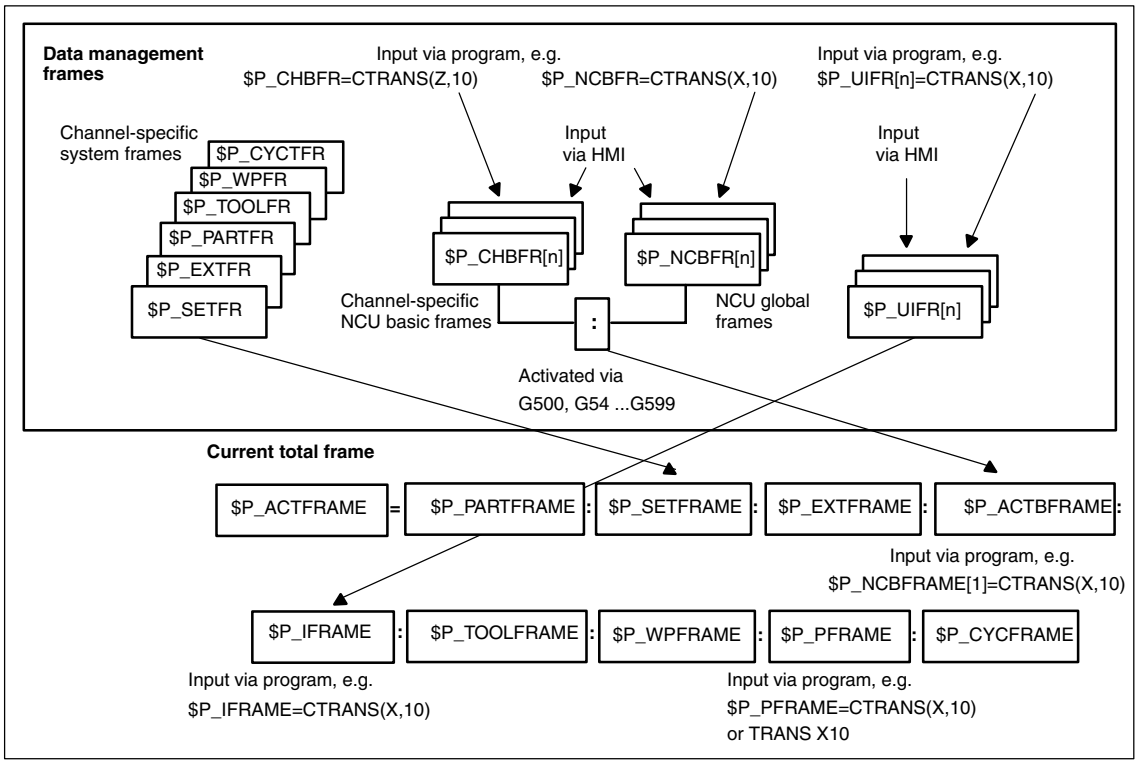


Fig. 2-35 Combination of complete frame in the channel

SW 6 and lower $P_ACTFRAME corresponds to $P_ACTBFRAME : $P_IFRAME : $P_PFRAME

SW 6.1 and higher
$P_ACTFRAME corresponds to
$P_SETFRAME : $P_EXTFRAME : $P_PARTFRAME :
$P_ACTBFRAME : $P_IFRAME : $P_TOOLFRAME : $P_PFRAME

SW 6.3 and higher
$P_ACTFRAME corresponds to
$P_SETFRAME : $P_EXTFRAME : $P_PARTFRAME :
$P_ACTBFRAME : $P_IFRAME : $P_TOOLFRAME : $P_WPFRAME :
$P_PFRAME : $P_CYCFRAME :

SW 6.4 and higher See Fig. 2-35 / 2-36 Frame chain with current frames
$P_ACTFRAME corresponds to
$P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME :
$P_ACTBFRAME : $P_IFRAME : $P_TOOLFRAME : $P_WPFRAME :
$P_PFRAME : $P_CYCFRAME :

© Siemens AG, 2003. All rights reserved

2.4.12 FRAME chaining

The current FRAME is composed of the total basic frame, the settable FRAME, the system frames and the programmable FRAME.

In **SW 6.1** and higher, the current complete frame results from the following formula:

$P_ACTFRAME= $P_SETFRAME : $P_EXTFRAME : $P_PARTFRAME : $P_ACTBFRAME : $P_IFRAME : $P_TOOLFRAME : $P_PFRAME

In **SW 6.4** and higher, the current complete frame results from the following formula:

$P_ACTFRAME= $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_ACTBFRAME : $P_IFRAME : $P_TOOLFRAME : $P_WPFRAME : $P_PFRAME : $P_CYCFRAME

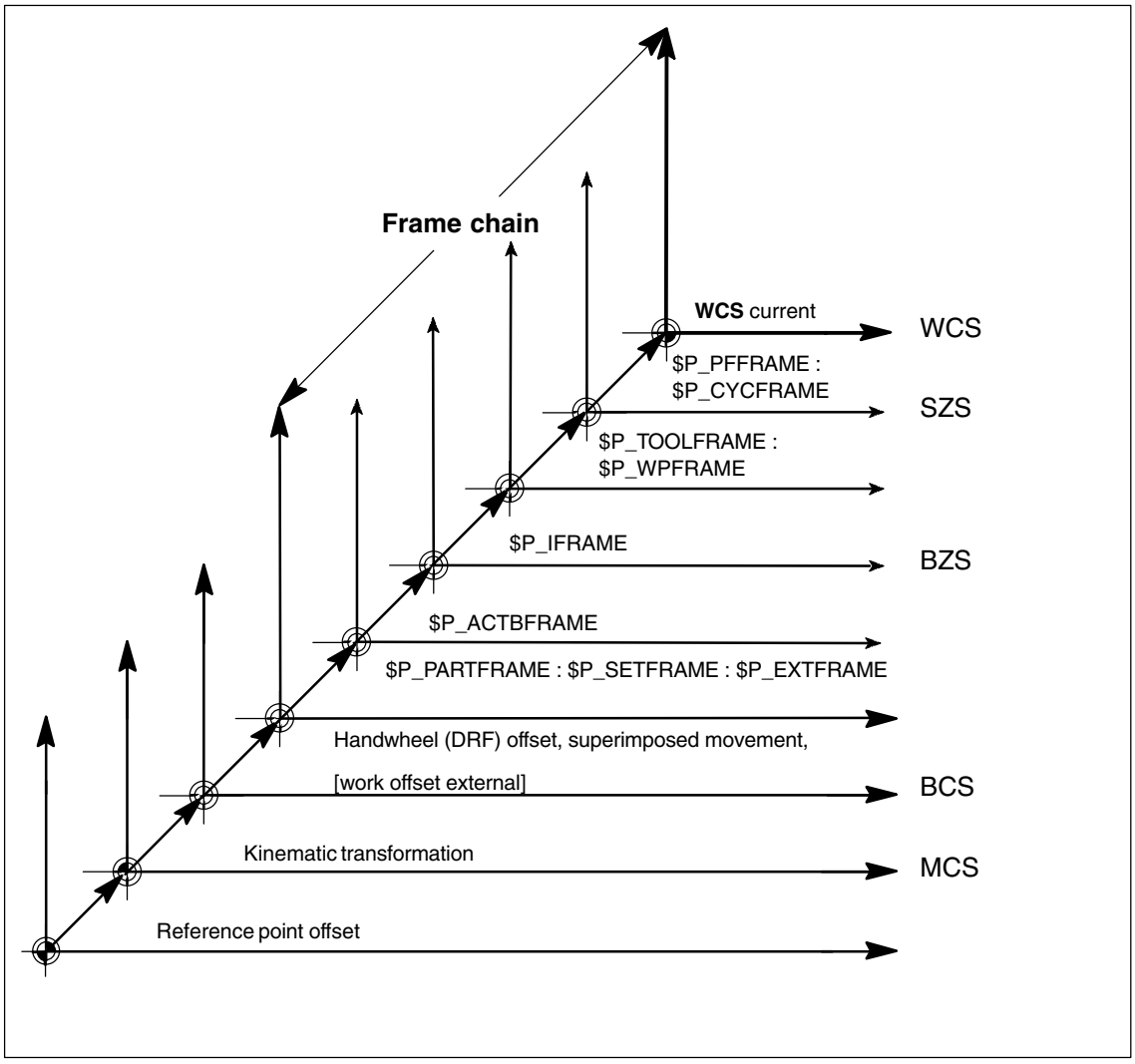


Fig. 2-36 Frame chain of active frames

© Siemens AG, 2003. All rights reserved

INVFRAME

Inverse frame

The parts program provides a function which calculates the inverse frame from another frame. The concatenation between a frame and its inverse frame always produces a zero frame.

FRAME INVFRAME(FRAME)
The inversion function can be used to transform a frame into another coordinate system. To enter a calculated measurement frame in the WCS into any of the other frames within a frame chain, for example, the following calculations need to be performed:

The new total frame is a chain of the old total frame and the calculated frame.

- $P_ACTFRAME$_{new}$ = $P_ACTFRAME$_{old}$: $AC_MEAS_FRAME

Target frame

The new frame in the frame chain is therefore:

- Target frame is **$P_SETFRAME**:
  $P_SETFRAME$_{new}$ = $P_ACTFRAME$_{old}$: $AC_MEAS_FRAME
  INVFRAME($P_ACTFRAME$_{old}$) : $P_SETFRAME$_{old}$
- Target frame is nth channel basic frame **$P_CHBFRAME[n]**:

  n = 0:
  TMP = $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_NCBFRAME[0..k]

  n <> 0:
  TMP = $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME :$P_NCBFRAME[0..k] : $P_CHBFRAME[0..n–1]

  k = $MN_MM_NUM_GLOBAL_BASE_FRAMES

  $P_CHBFRAME[n]$_{new}$ =
  INVFRAME(TMP) : $P_ACTFRAME$_{old}$: $AC_MEAS_FRAME :
  INVFRAME($P_ACTFRAME$_{old}$) : TMP : $P_CHBFRAME[n]$_{old}$
- Target frame is **$P_IFRAME**:
  TMP = $P_PARTFRAME : $P_SETFRAME : $P_EXTFRAME : $P_BFRAME

  P_IFRAME_{new}$ = INVFRAME(TMP) : $P_ACTFRAME$_{old}$: $AC_MEAS_FRAME :
  INVFRAME($P_ACTFRAME$_{old}$) : TMP : P_IFRAME_{old}$

Example

A frame calculated, for example, via a measuring function, must be entered in the current SETFRAME such that the new total frame is a chain of the old total frame and the measurement frame. The SETFRAME is calculated accordingly by means of frame inversions.

```
DEF INT RETVAL                  ;
DEF FRAME TMP                   ;

$TC_DP1[1,1]=120                ; Type
$TC_DP2[1,1]=20                 ; 0
$TC_DP3[1,1]= 10                ; (z) Length compensation vector
$TC_DP4[1,1]= 0                 ; (y)
$TC_DP5[1,1]= 0                 ; (x)
$TC_DP6[1,1]= 2                 ; Radius

T1 D1                           ;
g0 x0 y0 z0 f10000              ;
G54                             ;
```

© Siemens AG, 2003. All rights reserved

```
$AC_MEAS_VALID = 0                  ; Invalidate all input values

g1 x–1 y–3                          ; Approach 1st measuring point
$AC_MEAS_LATCH[0] = 1               ; Save measuring point 1

g1 x5 y–3                           ; Approach 2nd measuring point
$AC_MEAS_LATCH[1] = 1               ; Save measuring point 2

g1 x–4 y4                           ; Approach 3rd measuring point
$AC_MEAS_LATCH[2] = 1               ; Save measuring point 3

g1 x–4 y1                           ; Approach 4th measuring point
$AC_MEAS_LATCH[3] = 1               ; Save measuring point 4

                                    ; Set setpoint position of corner to (0, 0, 0)
$AA_MEAS_SETPOINT[x] = 0            ;
$AA_MEAS_SETPOINT[y] = 0            ;
$AA_MEAS_SETPOINT[z] = 0            ;

$AC_MEAS_CORNER_SETANGLE = 90; Input setpoint angle of intersection Φ
$AC_MEAS_WP_SETANGLE = 30           ; Input setpoint workpiece position angle α
$AC_MEAS_ACT_PLANE = 0              ; G17 is the measurement plane

                                    ; Select a tool
$AC_MEAS_T_NUMBER = 1               ;
$AC_MEAS_D_NUMBER = 1               ;

$AC_MEAS_TYPE = 4                   ; Set measurement type to corner 1

                                    ; Perform calculation
RETVAL = MEASURE()                  ;

if RETVAL <> 0
setal(61000 + RETVAL)
endif

if $AC_MEAS_WP_ANGLE <> 30          ; Scan known setpoint workpiece position angle α
setal(61000 + $AC_MEAS_WP_ANGLE)
endif

if $AC_MEAS_CORNER_ANGLE <> 90 ; Scan known setpoint angle of intersection Φ
setal(61000 + $AC_MEAS_CORNER_ANGLE)
endif

                                    ; Transform measured frame and write to
                                    ; $P_SETFRAME such that a total frame
                                    ; is produced that comprises a chain of the old
                                    ; total frame and the measurement frame.

$P_SETFRAME =
$P_ACTFRAME : $AC_MEAS_FRAME : INVFRAME($P_ACTFRAME) : $P_SETFRAME

$P_SETFR = $P_SETFRAME              ; Write system frame in data management

g1 x0 y0                            ; Approach the corner

                                    ; Machine rectangle rotated through 30 degrees
g1 x10                              ;
y10                                 ;
x0                                  ;
y0                                  ;

m30
```

© Siemens AG, 2003. All rights reserved

SW 6.3 and higher

Additive frame in frame chain

Measurements on the workpiece or calculations in the parts program and cycles generally produce a frame that is applied additively to the current total frame. The WCS and thus the programming zero must therefore be displaced and possibly rotated. This measured frame is available as a temporary frame and not yet actively included in the frame chain.

ADDFRAME

This temporary frame is included in the ADDFRAME(FRAME, STRING) function with the parameter settings shown in Table 2-5.

Table 2-5 Effect of INT ADDFRAME(FRAME, STRING)

Function value	Type	Meaning
Parameter 1:	FRAME	Additive measured or calculated frame
Parameter 2:	STRING	Strings for current frames:
		“$P–CYCFRAME”, ”$P_PFRAME”, ”$P_WPFRAME”, ”$P–TOOLFRAME”, ”$P_IFRAME”, ”$P–CHBFRAME[0..16]”, ”$P_NCBFRAME[0..16]”, ”$P–PARTFRAME”, ”$P_EXTFRAME”, ”$P_SETFRAME”,
		Strings for data management frames:
		”$P–CYCFR”, ”$P_WPFR”, ”$P–TOOLFR”, ”$P_UIFFR[0..99]”, ”$P–CHBFR[0..16]”, ”$P_NCBFR[0..16]”, ”$P–PARTFR”, ”$P_EXTFR”, ”$P_SETFR”,
Function value	INT	0: OK
		1: Specified target (string) is wrong
		2: Target frame is not configured
		3: Rotation in frame is not permitted

The ADDFRAME ( ) function calculates the target frame specified by the STRING. The target frame is calculated such that the new total frame comprises a chain of the old total frame and the transferred frame.

Example

ERG = ADDFRAME(TMPFRAME, ”$P_SETFRAME”)

The new total frame is calculated to be:

- $P_ACTFRAME$_{new}$ = $P_ACTFRAME$_{old}$: TMPFRAME

The new frame in the frame chain is therefore:

- Target frame is $P_SETFRAME:
  $P_SETFRAME$_{new}$ = $P_ACTFRAME$_{old}$: TMPFRAME : INVFRAME($P_ACTFRAME$_{old}$) : $P_SETFRAME$_{old}$

Dynamic response

If a current frame has been specified as a target frame, then the new total frame becomes active at the preprocessing stage.

If the target frame is a data management frame, then the frame is not operative until it is explicitly activated in the parts program.

The ADDFRAME () does not set any alarms, but returns the error codes via the return value. The cycle can react according to the error codes.

© Siemens AG, 2003. All rights reserved

2.4.13 Frames with G91 (SW 4 and higher)

Preconditions

Incremental programming with G91 is defined such that the compensation value is traversed additively to the incrementally programmed value when a zero offset is selected.

Applications

For applications such as scratching it is important that only the programmed path is traversed incrementally. (The activated zero offset should not be traversed).

Sequence

- Scratch workpiece with tool tip
- Save the actual position (set actual value) in the basic frame (basic offset) after reducing it by the tool offset
- Traverse incrementally from the zero position

Activation

With the setting data SD 42440: FRAME_OFFSET_INCR_PROG is used to set the following:

- Value = 1: Zero offset is traversed on FRAME and incremental programming of an axis (= default setting)
- Value = 0: Only the programmed path is traversed.

Example 1

For conventional response, zero offset is retracted:

```
SD 42440: FRAME_OFFSET_INCR_PROG = 1
                    ; G54 contains an offset of 25 in X
G90 G54             ;
G1 G91 X10          ; Offset is traversed → Position X35
```

Example 2

Zero offset is not retracted:

```
SD 42440: FRAME_OFFSET_INCR_PROG = 0
                    ; G54 contains an offset of 25 in X
G90 G54             ;
G1 G91 X10          ; → Position X10        ;The offset is now selected,
                                            but not retracted
G90 X50             ; → Position X75        ;Offset is traversed
```

Example 3

Response with FRAME suppression:

```
SD 42440: FRAME_OFFSET_INCR_PROG = 0
                    ; G54 contains an offset of 25 in X
G90 G54 X100        ; → Position X125
G53 X20             ; Deselect zero offset in this block → Position X20
G91 X100            ; Only a delta 100 is traversed here →  Position X120
G91 G53 X0          ; No movement (no delta is traversed)
```

© Siemens AG, 2003. All rights reserved

Tool offset

With the setting data SD 42442: TOOL_OFFSET_INCR_PROG it can be set whether the changed tool length is retracted with FRAME and incremental programming of an axis, or whether only the programmed path is traveled, see also /FB/, W1, “Tool offset, G91 extension”.

Condition

If the response is set such that the offset remains active beyond program end and RESET (MD 20110: RESET_MODE_MASK, Bit 5=1), and if an incremental path is programmed in the first parts program block, the offset is always retracted additively to the programmed path.
This applies up to and including SW 4.3. With higher software versions the corresponding setting is to be set with MD 20152 Index 7. See Table 2-14

Note

With this configuration, parts programs must always begin with absolute programming.

2.4.14 Suppression of frames with G53, G153 and SUPA (SW 6.1 and later)

The current frames can be suppressed as shown in Fig. 2-37 using the following statements:

- G53 Current zero offset (ZO)
- G153 Current frame, incl. basic frame
- SUPA Current ZO, incl. programmed offsets

Function	Behavior
G53	Nonmodal suppression of the frames: System frame for cycles Programmable frame System frame for TOROT and TORFRAME, workpieces Active settable frame
G153	Nonmodal suppression of the frames: System frame for cycles Programmable frame System frame for TOROT and TORFRAME, workpieces Active settable frame All channel-specific and NCU global basic frames
SUPA	As G153 and system frames for PRESET, scratching, work offset external, PAROT handwheel offsets (DRF), [ext. zero offset], superimposed movement
G500	Modal activation of the G500 frame. The G500 frame should be a zero frame.
DRFOF	Disabling (deletion) of the handwheel offsets (DRF)

© Siemens AG, 2003. All rights reserved

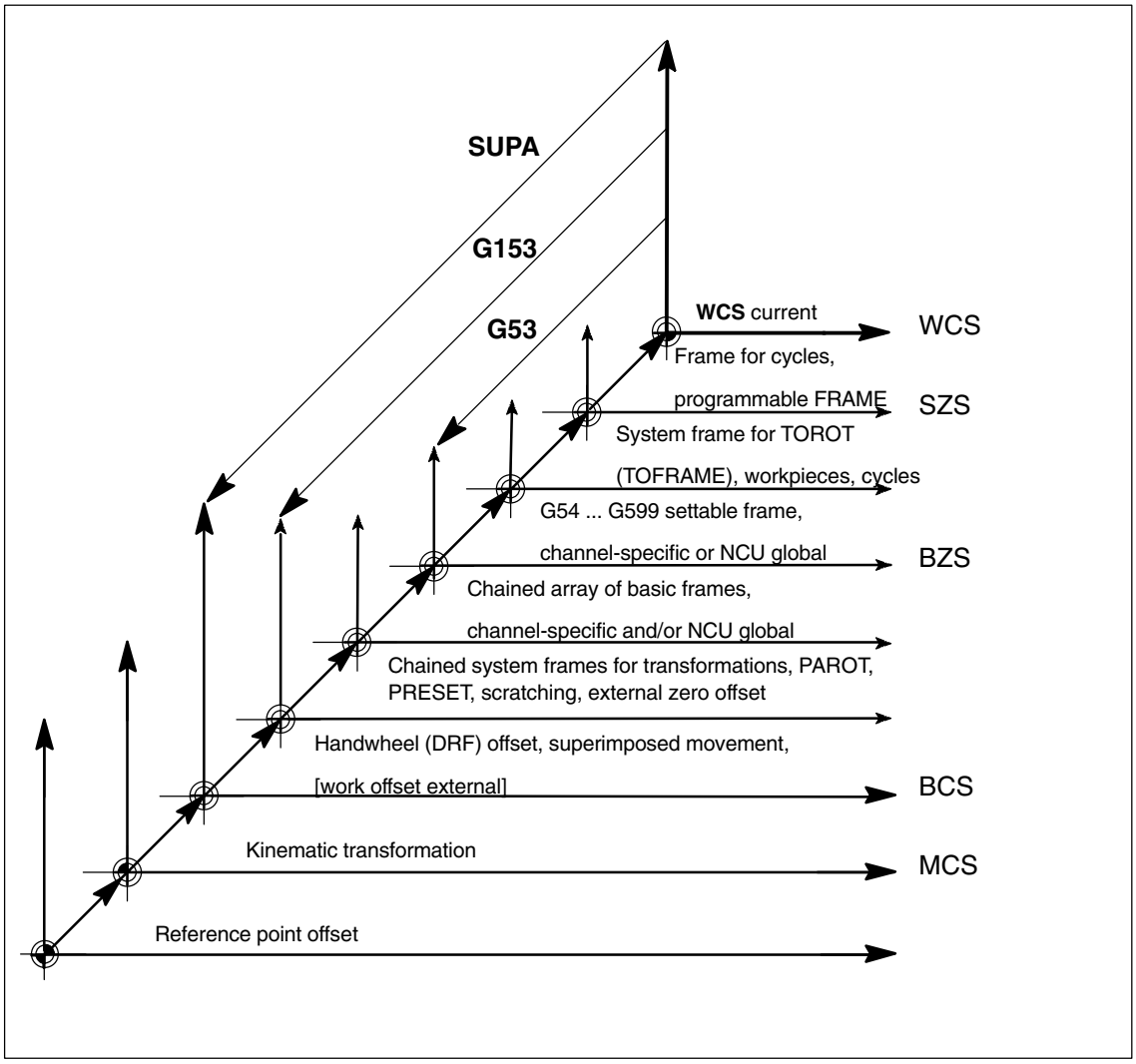


Fig. 2-37 Suppression of frames

Control frame suppression

Frame suppression commands SUPA, G153 and G53 cause a displacement in the

- WCS,
- SZS and possibly the
- BZS

when frame suppression is active.

With the machine data MD 24020: FRAME_SUPPRESS_MODE it is possible to change this characteristic for the position display and the predefined position variables. The following settings are available:

Bit 0: Positions for display (OPI) are without frame suppression.
Bit 1: Position variables are without frame suppression.

When the bit is set, the position for the display or the variables is calculated without frame suppression so that no further jumps in the position occur.

© Siemens AG, 2003. All rights reserved

2.4.15 Operating states and frames

Control system response Mode changes have no effect on FRAMES. All offsets are retained.

Table 2-6 Control system response to coordinate transformations

Coordinate transformation	Power ON	Reset, program end	Parts program start
Programmable FRAME	Deleted	Deleted	Deleted
Settable FRAMES	Permanent, depending on MD 20150: GCODE_RESET_VALUES (default G500)	Retained, depending on MD 20110: RESET_MODE_MASK / MD 20152: GCODE_RESET_MODE.	Retained, depending on MD 20112: START_MODE_MASK
Complete basic frame	Retained, depending on MD 20110: RESET_MODE_MASK, Bit 0 and Bit 14. Individual basic frames can be deleted with MD 10615: NCBFRAME_POWERON_MASK and MD 24004: CHBFRAME_POWERON_MASK.	Retained, depending on MD 20110: RESET_MODE_MASK, bits 0 and 14, and on MD 10613: NCBFRAME_ RESET_MASK, and on MD 24002: CHBFRAME_RESET_MASK.	Is retained
System frames (SW 6.1 and higher)	Retained, individual system frames can be deleted on Power ON depending on MD 24008: CHSFRAME_POWERON_MASK.	Retained, individual system frames can be deleted on Power ON depending on MD 24006: CHSFRAME_RESET_MASK and MD 20150: GCODE_RESET_VALUES	Is retained
Zero offset external (WOE)	Permanent, but has to be activated again. The system frame is retained.	Is retained System frame for ext. zero offset is active after RESET.	Is retained
DRF offset	Deleted	Is retained	Is retained

POWER ON The table below shows the response of the frames after POWER ON:

Frame	Behavior
Programmable frame	Deleted
Settable frames	Retained, depending on MD 20110: RESET_MODE_MASK
Complete basic frame	Retained, depending on MD 20110: RESET_MODE_MASK, bits 0 and 14. Individual basic frames can be deleted with MD 10615: NCBFRAME_POWERON_MASK and MD 24004: CHBFRAME_POWERON_MASK.
System frames (SW 6.1 and higher)	Retained, individual system frames can be deleted on Power ON depending on MD 24008: CHSFRAME_POWERON_MASK (system frames should be deleted in the data management as a priority).
Zero offset external (WOE)	Permanent, but has to be activated again. The system frame is retained.
DRF offset	Deleted

© Siemens AG, 2003. All rights reserved

Mode change

In jog mode, only the frame components of the current frame are referenced for geometry axes if a rotation is active. All other axial frames are ignored.

The response for PLC axes and command axes can be defined in MD 32074: FRAME_OR_CORRPOS_NOTALLOWED.

RESET, end of program

The reset response of basic frames is defined via MD 20110: RESET_MODE_MASK.

The system frames are retained in the data management after a Reset. Activation of individual system frames can be configured as follows via machine data **MD 24006: CHSFRAME_RESET_MASK**:

Bit 0: System frame for PRESET and scratching is active after Reset
Bit 1: System frame for ext. zero offset is active after RESET.
Bits 2 + 3: Are not evaluated.

SW 6.3 and higher

Bit 4: System frame for workpiece reference points is active after RESET.
Bit 5: System frame for cycles is active after RESET.

SW 6.4 and higher

Bit 6: Remaining behavior dependent on MD 20110: RESET_MODE_MASK

MD 20110: RESET_MODE_MASK	Effect
Bit 0 = 1 and bit 14 = 0	Chained total basic frame is deleted.
Bit 0 = 1 and bit 14 = 1	The complete basic frame results from MD24002: CHBFRAME_RESET_MASK and MD10613: NCBFRAME_RESET_MASK. MD24002: CHBFRAME_RESET_MASK: Bit 0 = 1: 1st channel basic frame is included in the chained total basic frame. Bit 7 = 1: 8th channel basic frame is included in the chained total basic frame. MD10613: NCBFRAME_RESET_MASK: Bit 0 = 1: 1st global NCU basic frame is included in the chained complete basic frame. Bit 7 = 1: 8th global NCU basic frame is included in the chained complete basic frame.

The table below shows the response of the frames after RESET, end of program:

Frame	Behavior
Programmable frame	Deleted
Settable frames	Retained, depending on MD 20110: RESET_MODE_MASK and MD 20152: GCODE_RESET_MODE.
Complete basic frame	Retained, depending on MD 20110: RESET_MODE_MASK bits 0 and 14, and on MD 10613: NCBFRAME_ RESET_MASK, and on MD 24002: CHBFRAME_ RESET_MASK.
System frames (SW 6.1 and higher)	Retained, individual system frames can be deleted on Power ON depending on MD 24006: CHSFRAME_RESET_ MASK and MD 20150: GCODE_RESET_VALUES[ ].
Zero offset external (WOE)	Retained.
DRF offset	Retained.

© Siemens AG, 2003. All rights reserved

Parts program start

MD 20112: START_MODE_MASK is not relevant to the basic frames and is not evaluated.

Frame	behavior
Programmable frame	Deleted
Settable frames	Retained, depending on MD 20112: START_MODE_MASK
Complete basic frame	Is retained
System frames (SW 6.1 and higher)	Retained
Zero offset external (WOE)	Is retained
DRF offset	Retained.

Block search, REPOS

There is no special treatment for global frames. If a frame is modified in an ASUB, it is retained in the program. On reapproach with REPOS, a modified frame is included, provided the modification was activated in the ASUB.

Return jump with SAVE

Subroutine return jump with SAVE

The SAVE attribute sets modal G functions to the same value at the end of subroutines that they had at the beginning. If this action results in a change to the

- G function group 8 (settable zero offset),
- G function group 52 (frame rotations of a rotational workpiece),
- G function group 53 (frame rotation in direction of tool),

then the relevant frames are restored.

The active basic frame is not altered on the return jump, the programmable zero offset is restored.

SW 6.1 and higher

The response of the settable zero offset and basic frame with restorable SAVE attribute on a subroutine return jump can be set via machine data MD 10617: FRAME_SAVE_MASK.

- Settable frames G54 to G599, MD 10617, bit 0 = 0:

  If the same G code is active on return from the subroutine as in the subroutine call, then the active settable frame is retained. If this is not the case, the settable frame at the instant the subroutine was called is reactivated.
- Basic frames $P_CHBFR[ ] and NCBFR[ ], MD 10617, bit 1 = 0:

  The active total basic frame is retained on return from a subroutine.
- Programmable frame:
  This frame is restored again on a subroutine return jump.
- The new SAVE attribute has no effect on system frames $P_SETFR, $P_EXTFR, $P_PARTFR, $P_TOOLFR, $P_WPFR and $P_CYCFR.
- The new SAVE attribute has no effect on system frames $P_SETFR, $P_EXTFR, $P_PARTFR, $P_TOOLFR in SW 6.3 and higher $P_WPFR, $P_CYCFR and in SW 6.4 and higher $P_TRAFR.

© Siemens AG, 2003. All rights reserved

2.5 Workpiece-related actual-value system

Definition

The term “workpiece-related actual-value system” designates a series of functions that permit the user:

- To use a workpiece coordinate system defined in machine data after poweron
  - Without additional operator actions
  - In JOG and AUTOMATIC modes
- To retain the valid settings for the following after end of program for the next parts program:
  - Active plane
  - Settable frame (G54G57),
  - Kinematic transformation,
  - Active tool length correction
- To switch between the following by operator action on the MMC
  - Workpiece coordinate system
  - Machine coordinate system
- To change the workpiece coordinate system by operator action (e.g. changing the settable frame or the tool length correction).

2.5.1 Use of workpiece-related actual-value system

Requirements, basic settings

The settings described in the previous Section have been made for the system. On MMC power-on the MCS is preset.

Switchover to WCS

The effects of switching over to the WCS via the operator panel are as follows:

- Axis positions are displayed with reference to the origin of the WCS.

Switchover to MCS

The effects of switching over to the MCS via the operator panel are as follows:

- Axis positions are displayed with reference to the origin of the MCS.

Interrelationships between coordinate systems

The following diagram shows the interrelationships between the machine coordinate system MCS and the workpiece coordinate system WCS for product level $\geq$ SW 2.

© Siemens AG, 2003. All rights reserved

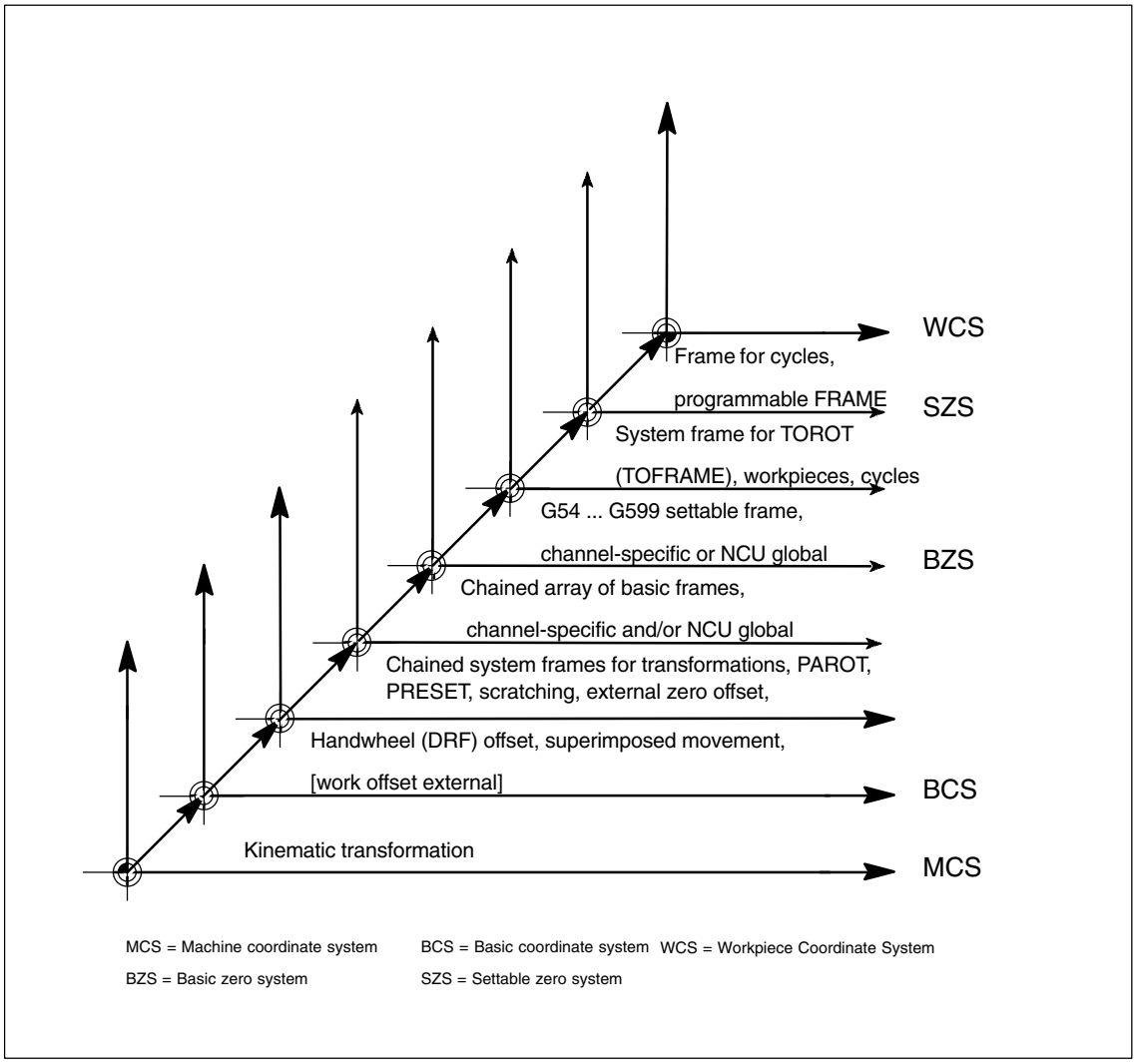


Fig. 2-38 Interrelationship between coordinate systems

References: /PG/, “Programming Guide: Fundamentals”
/FB/, W1, “Tool offset”
/FB/, H2, “Start-up”
/FB/, M1, “Kinematic transformation”
/FB/, M3, “Coupled motion and master/slave coupling”
/FB/, T3, “Tangential control”

© Siemens AG, 2003. All rights reserved

2.5.2 Special reactions

Overstoring

Overstoring in RESET state of:

- Frames (zero offsets)
- Active plane
- Activated transformation
- Tool offset

immediately affects the actualvalue display of all axes in the channel.

MMC inputs

If the values for:

- Active frame (zero offsets, operating area parameter)
- Active tool length compensation (operating area parameter)

are changed by operator action on the operator panel, the take effect in the display after the following steps have been followed:

- Press the RESET key
- Reselect
  - Zero offset by the parts program
  - Tool offset by the parts program
- Reset
  - Zero offset by overstoring (see above)
  - Tool offset by overstoring
- Parts program start

MD 9440
SW 4.3 and higher

If the MMC machine data for the operator panel
MD 9440: ACTIVATE_SEL_USER_DATA is set, then the value inputs are immediately activated in the Reset state.

When values are entered in the parts program execution stop state, they become effective when program processing continues.

Actual value reading

If the actual value in the WCS is read from $AA_IW after activation

- of a frame (zero offset) or
- of a tool offset,

the activated changes are already contained in the result read even if the axes have not yet been traversed with the activated changes.

The actual values in the settable zero system (SZS) can be read from the parts program for each axis using the variable $AA_IEN[axis].

The actual values in the basic zero system (BZS) can be read from the parts program for each axis using the variable $AA_IBN[axis].

© Siemens AG, 2003. All rights reserved

Actual-value display

The programmed contour is always displayed in the WCS. The following offsets are added to the MCS:

- Kinematic transformation
- DRF offset/work offset external
- Active frame
- Active tool offset of the current tool.

Changeover per PLC

In SW4.3 (840D) or SW2.3 (810D) and higher, the actual values can be displayed via the PLC in the WCS, SZS, BZS or MCS. The PLC can define which coordinate system corresponds to the workpiece coordinate system on a machine.

On MMC power-on the MCS is preset. With the signal DB19 DBB0.7 “MCS/WCS switchover”, it is also possible to switch from the PLC to the WCS.

Transfer to PLC

The auxiliary functions (D, T, M) are output to PLC (or not) when tool length compensation is selected, depending on MD 20110/20112 bit 1.

The associated signals are described in Chapter 5.

Note

If the WCS is selected by the PLC, it is still possible to switch between WCS and MCS for the mode by means of an operator action.
However, on a mode and/or area change, the WCS selected by the PLC is read and activated.

References: /FB/, K1, “Mode Group Channels, Program Operation, Reset Response”

■

© Siemens AG, 2003. All rights reserved

Supplementary Conditions 3

3.1 Axes

Availability

The “Screw interpolation 2D+6” function is not included in the export version of the SINUMERIK 840DE/810DE.

A total of 5 axes/spindles can be configured on the SINUMERIK 840D/840Di/810D. Up to 9 axes/spindles can be operated in the maximum possible configuration of the SINUMERIK 840Di. The scope of possible configurations for the SINUMERIK 840D is as follows:

- NCU572 hardware: Up to 9 axes/spindles are available up to SW 5.2.
- NCU572.3 hardware: Up to 12 axes/spindles are available from SW 5.3.
- NCU573.3 hardware: Up to 31 axes/spindles are available from SW 5.3.

Note

In SINUMERIK 840D, the following axes/spindles are permitted, depending on the HW/SW:

Per channel	Up to 12 axes/spindles are permitted
Per NCU	A maximum of 31 axes or a maximum of 20 spindles are permitted

For further information about other axis and spindle configurations, see
References: /BU/, “Order Document, Catalog NC 60”

Using DMP modules

If you are using DMP compact modules, when configuring axes with

- NCU 573.3, the number of axes, including the DMP modules, is limited to 31.

If, for example, a DMP compact module is used for 31axis software, 30 axes will be available.

Programming

An axis can be programmed only if it physically exists on the NCU. Attempts to program an axis of the wrong type is rejected with alarm “14092 Channel %1 block %2 axis %3 is wrong axis type”. In SW 5.1 and higher, for example, a link axis cannot be programmed as a channel axis.

© Siemens AG, 2003. All rights reserved

3.2 Coordinate systems

Availability

The “Zero offset external” function is available with SW 2 and higher for control systems with the NCU 570, NCU 571, NCU 572 and NCU 573 in the basic versions.
The basic frame and fine offset are available with SW 4 and higher.

3.3 Frames

NCU global frames

The “NCU global frames” function is available in the basic version with SW 5 and higher.

Frames with G91

The G91 function expanded via SD 42440: FRAME_OFFSET_INCR_PROG is available in SW 4 and higher for controls with NCU 571, NCU 572, NCU 573 and NCU 573.2 in the appropriate basic version.

3.4 Workpiece-related actual value system/reset response

Availability

The “Workpiece-related actual-value system” function is available in the basic version with SW 2 and higher. The option of defining the initial control setting for parts program start is available as a basic version function from SW 3. The control of all G codes with MD 20152: GCODE_RESET_MODE is possible in SW 5 and higher.

■

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

4.1 Operator panel machine data

Note

From SW 6.1 and higher: (MMC 100 is the same as HMI Embedded) and (MMC 103 is the same as HMI Advanced)

Note MD 9242 to MD 9245 (for HMI Advanced only)

9242 MD number	**MA_STAT_DISPLAY_BASE** Numerical basis for representation of moving joint STAT		
Default setting: 10	Minimum input limit: 2	Maximum input limit: 16	
Changes effective immediately	Protection level: 1/1	Unit: –	
Data type: signed ushort, USHORT	Applies from SW 6.1		
Significance:	This machine data defines the numerical system (bin, dec, hex) for representation of the “STAT” position of moving joints with special kinematics and robots. Possible settings are: 02: Representation as binary value with STAT = “B00001101” 10: Representation as decimal value with STAT = 13 16: Representation as hexadecimal value with STAT = ’H0D’		
Application example(s)	–		
Special cases, errors, ...			
Related to	–		

9243 MD number	**MA_TU_DISPLAY_BASE** Numerical basis for representation of rotary axis position TU		
Default setting: 10	Minimum input limit: 2	Maximum input limit: 16	
Changes effective immediately	Protection level: 1/1	Unit: –	
Data type: signed ushort, USHORT	Applies from SW 6.1		
Significance:	The availability in the Machine operating area is dependent on the access level. This machine data defines the numerical system (bin, dec, hex) for representation of the “TU” position of rotary axes with robots. Possible settings are: 02: Representation as binary value with TU = “B00001101” 10: Representation as decimal value with TU = 13 16: Representation as hexadecimal value with TU = ’H0D’		
Related to	–		

© Siemens AG, 2003. All rights reserved

9244 MD number	**MA_ORIAXIS_EULER_ANGLE_NAME** Display of orientation axes as Euler angle		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 1/1	Unit: –	
Data type: signed ushort, USHORT	Applies from SW 6.1		
Significance:	The availability in the Machine operating area is dependent on the access level. This machine data defines whether the axis identifier of the orientation axes is the Euler angle name or the channel axis name. Possible settings are: 0: Orientation axis name is the geo axis name from the channel block with index 3 to 5. 1: Orientation axis name is the name of the Euler angle from the general machine data.		
Related to	–		

9245 MD number	**MA_PREST_FRAMEIDX** Value storage scratching and PRESET		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 10	
Changes effective immediately	Protection level: 1/1	Unit: –	
Data type: signed ushort, USHORT	Applies from SW 6.1		
Significance:	Index of the basic frame to which the scratch and PRESET functions enter their values. The index must be within the limits specified by channel-specific machine data MD 28081: MM_NUM_BASE_FRAMES (number of basic frames). The machine data is not relevant if the values of scratch and PRESET are entered in the system frame and the system frame is active.		
Related to	–		

9247 MD number	**USER_CLASS_BASE_ZERO_OFF_PA** Availability of basic offset in Parameters operating area		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 1/1	Unit: –	
Data type: signed char, CHAR	Applies from SW 5.3		
Significance:	The machine data MD 9247: USER_CLASS_BASE_ZERO_OFF_PA can be used to set the access level from which the soft key “Base ZO” is displayed in the “Zero offset” window in the Parameters operating area. At the same time, the base frames also appear hidden/ unhidden in the Zero offset window and in the Active ZO + offsets window.		
Related to	–		

9248 MD number	**USER_CLASS_BASE_ZERO_OFF_MA** Availability of basic offset in Machine operating area		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 1/1	Unit: –	
Data type: signed char, CHAR	Applies from SW 5.3		
Significance:	The machine data MD 9248: USER_CLASS_BASE_ZERO_OFF_MA can be set to select the access level from which soft key “Base ZO” is displayed in the “Scratching” screen in the Machine operating area and/or G500 can be entered in the zero offset field. The “PRESET” function is also displayed depending on this MD. The operator can thus no longer change the values of the base zero offset without the appropriate access authorization.		
Related to	–		

© Siemens AG, 2003. All rights reserved

Note MD 9400 to MD 9402 (for HMI Embedded only)

9249 MD number	**USER_CLASS_VERT_MODE_SK** Provide protection for vertical area soft keys		
Default setting: 77777777	Minimum input limit: 0	Maximum input limit: 77777777	
Changes effective immediately	Protection level: 3/4	Unit: Hex	
Data type: BYTE	Applies from SW 6.1		
Significance:	The machine data MD 9249: USER_CLASS_VERT_MODE_SK can be set to provide appropriate protection for vertical area soft keys. **Note:** (SW 6.1 and higher) This functionality is available only with HMI Embedded.		
Related to	–		

9400 MD number	**TOOL_REF_GEO_AXIS1** Absolute dimension tool length compensation geometry axis 1		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	MD 9400: TOOL_REF_GEO_AXIS1 can be set to define the absolute dimension in the Tool Offsets or Calculate Offset screen of the Parameters area for the geometry axis. The appropriate geometry axis 1 is selected with the “toggle key”; the absolute dimension can then be edited via the numeric keypad. When the “OK” soft key is selected, the current position and this absolute dimension are computed for the selected tool parameter. The following applies: Position – absolute dimension = input value.		
Related to	–		

9401 MD number	**TOOL_REF_GEO_AXIS2** Absolute dimension tool length compensation geometry axis 2		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	MD 9401: TOOL_REF_GEO_AXIS2 can be set to define the absolute dimension in the Tool Offsets or Calculate Offset screen of the Parameters area for the geometry axis. The appropriate geometry axis 2 is selected with the “toggle key”; the absolute dimension can then be edited via the numeric keypad. When the “OK” soft key is selected, the current position and this absolute dimension are computed for the selected tool parameter. The following applies: Position – absolute dimension = input value.		
Related to	–		

9402 MD number	**TOOL_REF_GEO_AXIS3** Absolute dimension tool length compensation geometry axis 3		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	MD 9402: TOOL_REF_GEO_AXIS3 can be set to define the absolute dimension in the Tool Offsets or Calculate Offset screen of the Parameters area for the geometry axis. The appropriate geometry axis 3 is selected with the “toggle key”; the absolute dimension can then be edited via the numeric keypad. When the “OK” soft key is selected, the current position and this absolute dimension are computed for the selected tool parameter. The following applies: Position – absolute dimension = input value.		
Related to	–		

© Siemens AG, 2003. All rights reserved

9424 MD number	**MA_COORDINATE_SYSTEM** Coordinate system for actualvalue display		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5		
Significance:	By setting this MD it is determined whether the actual values for the WCS displayed in the three actual value windows (small, wide, large) are shown as before (MA_COORDINATE_SYSTEM=0) or in the settable zero system (SZS) including the programmed zero offsets (MA_COORDINATE_SYSTEM= 1). The coordinate system continues to be known as the WCS. Bit0 = 0 WCS Bit0 = 1 SZS (settable zero system)		
Related to	–		

9425 MD number	**MA_SCRATCH_DEFAULT_MODE** Tool offset computation for geometry axes with scratching		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 222222	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	MD 9425: MA_SCRATCH_DEFAULT_MODE can be parameterized to preset the tool offset directions for scratching in the Machine operating area. The directions of computation can be set as follows: – Wear for 3 geometry axes with following meaning: Without = 0, + = 1, – = 2 – 3 places basic offsets – 3 geometry axes 1 to 3 The appropriate value is selected through toggling in the Scratching screen. **Note:** (SW 6.1 and higher) This functionality is available only with HMI Embedded.		
Related to	–		

9440 MD number	**ACTIVE_SEL_USER_DATA** Active data (frames) are immediately operative after editing		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5		
Significance:	Active data (frames) are immediately operative after editing.		
Related to	–		

9449 MD number	**WRITE_TOA_LIMIT_MASK** Applicability of MD 9203 to edge data and location-dependent offsets		
Default setting: 7	Minimum input limit: 0	Maximum input limit: 7	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5.2		
Significance:	The applicability of machine data MD 9203: MD 9203: USER_CLASS_WRITE_FINE to WEAR can be set via this data: Bit0 = 1 Applied to edge data, wear values Bit1 = 1 Applied to SC data (locationdependent offsets, wear values) Bit2 = 1 Applied to EC data (locationdependent offsets, setup values) Default setting 7: Applied to all data **Note:** (SW 6.1 and higher) This functionality is available only with HMI Advanced.		
Related to	–		

© Siemens AG, 2003. All rights reserved

9450 MD number	**MM_WRITE_TOA_FINE_LIMIT** Limit value for wear fine		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective immediately	Protection level: 3/4	Unit: Length	
Data type: DOUBLE	Applies from SW 4.2		
Significance:	When entering the tool wear fine, the difference between the previous value and the new value may not exceed the limit entered here.		
Related to	MD 9202: USER_CLASS_WRITE_TOA_WEAR		

9451 MD number	**MM_WRITE_ZOA_FINE_LIMIT** Limit value for offsets fine		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective immediately	Protection level: 3/4	Unit: Length	
Data type: DOUBLE	Applies from SW 4.2		
Significance:	When entering the offset fine, the difference between the previous value and the new value may not exceed the limit entered here.		
Related to	MD 9210: USER_CLASS_WRITE_ZOA		

9459 MD number	**PA_ZOA_MODE** Display mode of zero offset		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 3/4	Unit: Length	
Data type: BYTE	Applies from SW 6.1		
Significance:	Display mode of zero offset as before or for HMI Embedded 0 Same screen mode as SW 5 and lower 1 Screen mode for HMI Embedded in SW 6.1 and higher **Note:** (SW 6.1 and higher) This functionality is available only with HMI Embedded.		
Related to	–		

© Siemens AG, 2003. All rights reserved

4.2 General machine data

10000 MD number	**AXCONF_MACHAX_NAME_TAB** Machine axis name		
Default setting: 1st machine axis ... X 2nd machine axis ... Y 3rd machine axis ... Z	Minimum input limit: one letter	Maximum input limit: 15 characters starting with a letter	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: STRING	Applies from SW 1.1		
Significance:	The name of the machine axis is entered in this MD		
Special cases, errors, ...	● The entered machine axis name must not conflict with the name and assignment of geometry axes (MD 20060: AXCONF_GEOAX_NAME_TAB, MD 20050: AXCONF_GEOAX_ASSIGN_TAB) and channel axes (MD 20080: AXCONF_CHANAX_NAME_TAB, MD 20070: AXCONF_MACHAX_USED). ● The entered machine axis name must not be the same as the names entered for Euler angles (MD 10620: EULER_ANGLE_NAME_TAB), names specified for directional vectors (MD 10640: DIR_VECTOR_NAME_TAB), names given to intermediate point coordinates in the case of CIP (MD 10660: INTERMEDIATE_POINT_NAME_TAB) and the names of interpolation parameters (MD 10650: IPO_PARAM_NAME_TAB). ● The machine axis name entered must not be given any of the following reserved address letters: – D Tool offset (D function) – E Reserved – F Feedrate (F function) – G Preparatory function – H Auxiliary function (H function) – L Subprogram call – M Miscellaneous function (M function) – N Subblock – P Subprogram number of passes – R Arithmetic parameters – S Spindle speed (S function) – T Tool (T function) ● Vocabulary words (e.g. DEF, SPOS etc.) and predefined identifiers (e.g. ASPLINE, SOFT) are also illegal. ● The use of axis designations consisting of a valid address letter (A, B, C, I, J, K, Q, U, V, W, X, Y, Z), followed by an optional numerical extension (1–99) gives slightly better block cycle times than a general designation. ● If no designation is assigned to a machine axis, the predefined name applies (“AXn” for the nth machine axis).		
Related to	MD 20060: AXCONF_GEOAX_NAME_TAB (geometry axis name in channel [GEO axis no.] MD 20080: AXCONF_CHANAX_NAME_TAB (channel axis name in channel [channel axis no.]		

© Siemens AG, 2003. All rights reserved

10600 MD number	FRAME_ANGLE_INPUT_MODE Input type for rotation with Frame		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	FRAME_ANGLE_INPUT_MODE is set to determine the properties of the rotations (ROT and AROT) around the three geometry axes if more than one rotation is programmed in a block. The order in which these rotations are programmed within the block is irrelevant. A computation of the rotations can be set according to: • Euler angle with FRAME_ANGLE_INPUT_MODE = 2 (see Fig. 2-27): Calculation of the rotation according to the Euler angle is performed in the following order: 1st Rotation about Z (ROT Z ...) 2nd Rotation about X (ROT X ...) 3rd Rotation about Y (ROT Y ...) • RPY with FRAME_ANGLE_INPUT_MODE = 1 (see Fig. 2-28): Calculation of the rotation according to RPY is performed in the following order: 1st Rotation about Z (ROT Z ...) 2nd Rotation about Y (ROT Y ...) 3rd Rotation about X (ROT X ...)		
Special cases, errors, ...	If parts programs are programmed with rotations on the control (not via work preparation and/or postprocessor), Siemens recommends the RPY setting. FRAME_ANGLE_INPUT_MODE = 0 corresponds to 1		
References	/PA/, “Programming Guide Fundamentals”		

10602 MD number	FRAME_GEOAX_CHANGE_MODE Frames and switchover of geometry axes		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: INT	Applies from SW 5.2		
Significance:	Geometry axes can be switched over in the following situations: – Selection and deselection of transformations – Geometry axis switchover GEOAX() The current complete frame is then calculated as follows: 0: The current complete frame is deleted. 1: The current complete frame is calculated again when the geometry axes are switched over. The translations, skaling and mirrorings of the new geometry axes are effective. The rotations of the old geometry axes are retained. 2: The current complete frame is calculated again when the geometry axes are switched over. The translations, skaling and mirrorings of the new geometry axes are effective. If rotations are active before the switchover in the current base frames, the current settable frame or the programmable frame, the switchover is canceled with alarm. 3: The current complete frame is deleted when selecting and deselecting transformations. With the GEOAX ( ) command, the frame is calculated again. The translations, skaling and mirrorings of the new geometry axes become effective. The rotations of the old geometry axes are retained.		

© Siemens AG, 2003. All rights reserved

10610 MD number	**MIRROR_REF_AX** Reference axis for mirroring (MIRROR) frame element		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 1.1 or 5.1		
Significance:	**SW 1.1 and higher** (see Fig. 2-31 for illustration) Mirroring (MIRROR and AMIRROR) about a certain axis (X) can always be mapped to: • Mirroring in this specific axis (X) • Mirroring in a different axis (Y) and rotation in a different axis (Z) • Rotation in 2 axes (X and Y) Mirroring functions in the control are always mapped by one specific method for system variables and display purposes. This is done by always referring (MIRROR and AMIRROR) to a defined reference axis. It does not matter, whether the programmed mirroring is active in the refernce axis or in a different axis. The reference axis must always be one of the three geometry axes.		
	SW 5.1 and higher: **0**: Mirroring **without** normalization Mirroring is always performed in the specified axis. **1, 2, 3**: Mirroring **with** normalization: Mirroring of a geometry axis can always be related to a defined reference axis. 1: **X** is the reference axis for mirroring (default response up to and including SW 4) Mirroring of the X axis is unique. Mirroring of the Y axis is mapped onto: An X axis mirror and a rotation of the Z axis through 180 degrees. See Fig. 2-32 Mirroring of the Z axis is mapped onto: An X axis mirror and a rotation of the X axis through 180 degrees and a rotation of the Z axis through 180 degrees. See Fig. 2-33 2: **Y** is the reference axis for mirroring Mirroring of the X axis is mapped onto: A Y axis mirror and a rotation of the Z axis through 180 degrees. Mirroring of the Y axis is unique Mirroring of the Z axis is mapped onto: A Y axis mirror and a rotation of the X axis through 180 degrees 3: **Z** is the reference axis for mirroring Mirroring of the X axis is mapped onto: A Z axis mirror and a rotation of the Z axis through 180 degrees. a rotation of the X axis through 180 degrees Mirroring of the Y axis is mapped onto: A Z axis mirror and a rotation of the X axis through 180 degrees. Mirroring of the Z axis is unique.		

10612 MD number	**MIRROR_TOGGLE** Mirror toggle		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: CHAR	Applies from SW 5.1		
Significance:	1: Programmed axis values are not evaluated. Toggle switch response: The command “MIRROR X0” activates the function; the next “MIRROR X0” deactivates it again. 0: Programmed axis values are always evaluated. <value> $\neq$ 0 The axis is mirrored if it is not already mirrored. <value> = 0 Mirroring is deactivated.		

© Siemens AG, 2003. All rights reserved

10613 MD number	NCBFRAME_RESET_MASK RESET response of the NCU global base frames		
Default setting: 0xFFFF	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: INT	Applies from SW 5.1		
Significance:	Bit mask for the reset response of the NCU global base frames calculated in the channel. The following applies: Bit0 = 1 and bit14 = 1 (MD 20110: RESET_MODE_MASK) The complete base frame on reset is formed by concatenating all NCU global base frame array elements whose bit is 1 in the bit mask. Bit0 = 1 and bit14 = 0 (MD 20110: RESET_MODE_MASK) The complete base frame is deselected on reset.		
Related to	–		

10615 MD number	NCBFRAME_POWERON_MASK POWER ON response of global base frames		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	This machine data defines whether global base frames are deleted on POWER ON. i.e. – Offsets set to 0. – Scalings set to 1. – Mirroring deactivated. The selection can be made separately for the individual base frames. Bit 0 corresponds to base frame 0, bit 1 to base frame 1, etc. Value=0: Base frame is retained on POWER ON. Value=1: Base frame is reset on POWER ON.		
Related to	–		

10650 MD number	IPO_PARAM_NAME_TAB Name of interpolation parameters		
Default setting: I, J, K	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: STRING	Applies from SW 1		
Significance:	List of designations of interpolation parameters The rules defined in $MC_AXCONF_CHANAX_NAME_TAB for axis designations apply for selecting the designations of interpolation parameters. The designations must be selected such that they do not conflict with other names (axes, Euler angle, normal vector, directional vector, interpolation parameters).		
Related to	INTERMEDIATE_POINT_NAME_TAB		
References	PA		

© Siemens AG, 2003. All rights reserved

10660 MD number	**INTERMEDIATE_POINT_NAME_TAB** Name of intermediate point coordinates for G2/G3		
Default setting: I1, J1, K1	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit:	
Data type: STRING	Applies from SW 1		
Significance:	List of designations of intermediate point coordinates. The rules defined in $MC_AXCONF_CHANAX_NAME_TAB for axis designations apply for selecting the designations of interpolation parameters. The designations must be selected such that they do not conflict with other names (axes, Euler angle, normal vector, directional vector, interpolation parameters).		
Related to	IPO_PARAM_NAME_TAB		
References	PA		

11640 MD number	**ENABLE_CHAN_AX_GAP** Channel axis gaps allowed		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 2	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	Value: 1 The machine data allows the configuration of channel gaps in machine data $MC_AXCONF_MACHAX_USED (assignment of value 0). The following MD assignment is thus allowed: $AXCONF_MACHAX_USED[0] = 1 ; 1st MA is 1st axis in channel $AXCONF_MACHAX_USED[1] = 2 ; 2nd MA is 2nd axis in channel $AXCONF_MACHAX_USED[2] = 0 ; **Channel gap** $AXCONF_MACHAX_USED[3] = 3 ; 3rd MA is 3rd axis in channel $AXCONF_MACHAX_USED[4] = 0 **CAUTION**: If a geometry axis is to be mapped onto a channel axis gap with $MC_AXCONF_GEOAX_ASSIGN_TAB[1]= 3, the controller response is the same as $MC_AXCONF_GEOAX_ASSIGN_TAB[1]= 0. The geometry axis is thus deleted! Transformation machine data cannot be assigned a channel axis number which is configured as a gap. Value: 0 Functionality same as SW 4, i.e. channel axis gap is denied with alarm 4000.		
References	Description of Functions: M1, F2		

18600 MD number	**MM_FRAME_FINE_TRANS** Fine offset for all settable FRAMES and the basic frame		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Modification effective after POWER ON; SRAM is formatted on modification	Protection level: 2 / 7	Unit: –	
Data type: INT	Applies from SW 4.3		
Significance:	1: Fine offset for settable frame; the definition of a base frame is possible by input on the MMC or via the program. 0: The fine offset cannot be entered or programmed. If fine offset is deactivated approx. 10 KB SRAM are saved (depending on MD 28080: MM_NUM_USER_FRAMES).		
Related to			
References	PA		

© Siemens AG, 2003. All rights reserved

18601 MD number	**MM_NUM_GLOBAL_USER_FRAMES** Number of globally predefined user frames		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 100	
Modification effective after POWER ON; SRAM is formatted on modification	Protection level: 2 / 7	Unit: –	
Data type: INT	Applies from SW 5.1		
Significance:	[Value]: The value is the number of array elements for the predefined array $P_UIFR[ ]. [Value] > 0: All settable frames are global. MD 28080: MM_NUM_USER_FRAMES is ignored. Battery-backed memory is reserved for this purpose.		
Related to			
References			

18602 MD number	**MM_NUM_GLOBAL_BASE_FRAMES** Number of global base frames		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 16	
Modification effective after POWER ON; SRAM is formatted on modification	Protection level: 2 / 7	Unit: –	
Data type: INT	Applies from SW 5.1		
Significance:	[Value]: The value is the number of array elements for the predefined array $P_NCBFR[ ]. Battery-backed memory is reserved for this purpose.		
Related to			

© Siemens AG, 2003. All rights reserved

4.3 Channel-specific machine data

4.3.1 Axes/coordinate system

20050 MD number	**AXCONF_GEOAX_ASSIGN_TAB** Assignment of geometry axis to channel axis		
Default setting: 1st geometry axis ... 1 2nd geometry axis ... 2 3rd geometry axis ... 3	Minimum input limit: 0 (0 means that the geometry axis is not assigned to any channel axis)	Maximum input limit: 5 (FM–NC/810D), 8 (840D)	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	In this MD you can enter to which channel axis the geometry axis is assigned. The assignment is made channel-specifically for all geometry axes. If no assignement is affected fo a geometry axis, this geometry axis does not exist and cannot be programmed (with the name defined in AXCONF_GEOAX_NAME_TAB). Note: The channel axes must be assigned to the geometry axes in continuous ascending sequence.		
Special cases, errors, ...	It is advisable to assign the (three) geometry axes to the first three channel axes.		

20060 MD number	**AXCONF_GEOAX_NAME_TAB** Geometry axis name in channel		
Default setting: 1st geometry axis ... X 2nd geometry axis ... Y 3rd geometry axis ... Z	Minimum input limit: one letter	Maximum input limit: 16 characters starting with a letter	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: STRING	Applies from SW 1.1		
Significance:	In this MD the names of the three geometry axes for the channel are entered separately. Geometry axes can be programmed in the parts program using the names specified here.		
Special cases, errors, ...	• The specified geometry axis name must not conflict with the designation and assignment of the machine and channel axis names. • The specified geometry axis name must not clash with the names for Euler angles (MD 10620: EULER_ANGLE_NAME_TAB), names specified for directional vectors (MD 10640: DIR_VECTOR_NAME_TAB), names given to intermediate point coordinates in the case of CIP (MD 10660: INTERMEDIATE_POINT_NAME_TAB) and the names of interpolation parameters (MD 10650: IPO_PARAM_NAME_TAB). • The specified geometry axis name must not include any of the following reserved address letters: – D Tool offset (D function) – E Reserved – F Feedrate (F function) – G Preparatory function – H Auxiliary function (H function) – L Subprogram call – M Miscellaneous function (M function) – N Subblock – P Subprogram number of passes – R Arithmetic parameters – S Spindle speed (S function) – T Tool (T function) • Vocabulary words (e.g. DEF, SPOS etc.) and predefined identifiers (e.g. ASPLINE, SOFT) are also illegal. • The use of axis designations consisting of a valid address letter (A, B, C, I, J, K, Q, U, V, W, X, Y, Z), followed by an optional numerical extension (1–99) gives slightly better block cycle times than a general designation. • Identical names may be given to geometry axes assigned to different channels.		
Related to	MD 10000: AXCONF_MACHAX_NAME_TAB (machine axis name [axis no.]) MD 20080: AXCONF_CHANAX_NAME_TAB (channel axis name in channel [channel axis no.])		

© Siemens AG, 2003. All rights reserved

20070 MD number	**AXCONF_MACHAX_USED** Machine axis name valid in channel		
Default setting: 1st channel axis ... 1 2nd channel axis ... 2 3rd channel axis ... 3 more channel axes ... 0	Minimum input limit: 0 (0 means that the geometry axis is not assigned to any channel axis)	Maximum input limit: 5 (FM–NC/810D), 31 (840D)	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The MD assigns a machine axis to a channel axis/special axis. The assignment is made channel-specifically for all channel axes. A machine axis that is not assigned to a channel is not active, i.e. the axis control is not computed, the axis is not shown on the screen and cannot be programmed in any channel. In SW 5 and higher, it is permissible, for reasons of configuration consistency, to assign no machine axes to a channel axis. In this case, the MD is set to 0 for the channel axis. MD 11640: ENABLE_CHAN_AX_GAP must be set to 1 (channel axis gaps allowed). In SW 5 and higher, machine data MD 20070: AXCONF_MACHAX_USED does not point directly to the machine axes created with MD 10000: AXCONF_MACHAX_NAME_TAB, but to the logical machine axis image defined with MD 10002: AXCONF_LOGIC_MACHAX_TAB. MD 10002: AXCONF_LOGIC_MACHAX_TAB points: – directly to a local machine axis on the NCU, – to the machine axis of another NCU – indirectly to an axis container with local or remote machine axes. If the default values AX1, ..., AX31 are entered in MD 10002: AXCONF_LOGIC_MACHAX_TAB, the NCU responds in the same way as up to SW 4, i.e. machine data MD 20070: AXCONF_MACHAX_USED points to the corresponding local machine axis.		
Special cases, errors, ...	• Every geometry axis must be assigned to a channel axis and a machine axis so that it can be programmed. • If a machine axis is assigned to several channels by means of AXCONF_MACHAX_USED, then the number of the channel from which the axis must be programmed must be entered in MD 30550: AXCONF_ASSIGN_MASTER_CHAN. • The list of entries must not contain any gaps up to SW 4 (see above for SW 5). In contrast, the machine axes used may contain gaps. e.g.: Permissible:AXCONF_MACHAX_USED [0] = 3 ; 3rd MA is 1st axis in channel AXCONF_MACHAX_USED [1] = 1 ; 1st MA is 2nd axis in channel AXCONF_MACHAX_USED [2] = 5 ; 5th MA is 3rd axis in channel AXCONF_MACHAX_USED [3] = 0 ; Last entry Error for SW 4, permissible for SW 5: AXCONF_MACHAX_USED [0] = 1 ; 1st MA is 1st axis in channel AXCONF_MACHAX_USED [1] = 2 ; 2nd MA is 2nd axis in channel AXCONF_MACHAX_USED [2] = 0 ; Gap in the list ... AXCONF_MACHAX_USED [3] = 3 ;... of the channel axes For the axes activated in the channel, axis designations must be assigned in the corresponding list locations of AXCONF_CHANAX_NAME_TAB.		
Application example(s)	This MD defines which axis can be used and exchanged in which channel for the “Axis/spindle” exchange function.		
Related to	MD 30550: AXCONF_ASSIGN_MASTER_CHAN (initial setting of channel for axis changeover) MD 20080: AXCONF_CHANAX_NAME_TAB (channel axis name in channel [channel axis no.]) MD 10002 :AXCONF_LOGIC_MACHAX_TAB MD 11640: ENABLE_CHAN_AX_GAP		
References	Description of Functions B3		

© Siemens AG, 2003. All rights reserved

20080 MD number	**AXCONF_CHANAX_NAME_TAB** Channel axis name/special axis name in channel		
Default setting: 1st machine axis ... X1 2nd machine axis ... Y1 3rd machine axis ... Z1	Minimum input limit: one letter or space	Maximum input limit: 15 characters starting with a letter	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: STRING	Applies from SW 1.1		
Significance:	The FM-NC/810D has five channel axes, the 840D has eight channel axes. In this MD you can set the name of the channel axis/special axis. The first three channel axes are normally occupied by three assigned geometry axes (see also MD 20050: AXCONF_GEOAX_ASSIGN_TAB). The remaining channel axes are also designated special axes. The channel axis/special axis is always displayed on the screen in the workpiece coordinate system with the name set in this MD.		
Special cases, errors, ...	● The specified channel axis name/special axis name must not conflict with the designation and assignment of the machine and geometry axis names. ● The specified channel axis name must not conflict with names of the Euler angle (MD 10620: EULER_ANGLE_NAME_TAB), names specified for directional vectors (MD 10640: DIR_VECTOR_NAME_TAB), names given to intermediate point coordinates in the case of CIP (MD 10660: INTERMEDIATE_POINT_NAME_TAB) and the names of interpolation parameters (MD 10650: IPO_PARAM_NAME_TAB). ● The specified channel axis name must not include any of the following reserved address letters: – D Tool offset (D function) – E Reserved – F Feedrate (F function) – G Preparatory function – H Auxiliary function (H function) – L Subprogram call – M Miscellaneous function (M function) – N Subblock – P Subprogram number of passes – R Arithmetic parameters – S Spindle speed (S function) – T Tool (T function) ● Vocabulary words (e.g. DEF, SPOS etc.) and predefined identifiers (e.g. ASPLINE, SOFT) are also illegal. ● The use of axis designations consisting of a valid address letter (A, B, C, I, J, K, Q, U, V, W, X, Y, Z), followed by an optional numerical extension (1–99) gives slightly better block cycle times than a general designation. ● For channel axes to which geometry axes are assigned (normally the first three channel axes), no special name must be entered in this MD.		

22532 MD number	**GEOAX_CHANGE_M_CODE** M code for replacement of geometry axes		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 99999999	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 4		
Significance:	Number of M code that is output at the VDI interface when a geometry axis is replaced. No M code is output if the machine data is set to one of values 0 to 6, 17, 30. There is no monitoring for conflicts between such a generated M code and other functions.		

© Siemens AG, 2003. All rights reserved

24000 MD number	FRAME_ADD_COMPONENTS Separate programming/modification of additive programmable frame components		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BOOL	Applies from SW 5.1		
Significance:	0: Additive translations programmed with ATRANS are stored in the frame together with the absolute translation (programmed with TRANS). G58, G59 are not possible. 1: The total of additive translations is stored in the fine offset of the programmable frame. The absolute and additiv translations can be modified independently of each other. G58, G59 are possible.		

24002 MD number	CHBFRAME_RESET_MASK Reset response of channel-specific base frames		
Default setting: 0xFFFF	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 5.1		
Significance:	Bit mask for the reset response of the channel-specific base frames calculated in the channel. The following applies: Bit0 = 1 and bit14 = 1 (MD 20110: RESET_MODE_MASK) The complete base frame on reset is formed by concatenating all channel-specific base frame array elements whose bit is 1 in the bit mask. Bit0 = 1 and bit14 = 0 (MD 20110: RESET_MODE_MASK) The complete base frame is deselected on reset.		

24004 MD number	CHBFRAME_POWERON_MASK POWER ON response of channel-specific base frames		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	This machine data defines whether channel-specific base frames are deleted on POWER ON. The selection can be made separately for the individual base frames. Bit 0 corresponds to base frame 0, bit 1 to base frame 1, etc. 0 Base frame is retained on POWER ON. 1 Base frame is deleted on POWER ON.		

24006 MD number	CHSFRAME_RESET_MASK Reset response of the channel-specific system frames		
Default setting: 0x1	Minimum input limit: 0x0	Maximum input limit: 0x0000007F	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.1, bit 6 in SW 6.4 and higher		
Significance:	Bit mask for the reset response of the channel-specific system frames calculated in the channel. The following applies: Bit0: System frame for PRESET and scratching is active after RESET. Bit1: System frame for work offset external is active after RESET. Bit2: Irrelevant, further behavior depends on : MD 20150: GCODE_RESET_VALUES[ ]. Bit3: Irrelevant, further behavior depends on : MD 20150: GCODE_RESET_VALUES[ ]. Bit4: System frame for workpiece ref. points is active after RESET (SW 6.3 and higher). Bit5: System frame for cycles is active after RESET (SW 6.3 and higher). Bit6: Reserved, reset response dependent on MD 20110: RESET_MODE_MASK (SW 6.4 and higher).		
Related to	MD 28082: SYSTEM_FRAME_MASK (configuring screen for channel-specific system frames)		

© Siemens AG, 2003. All rights reserved

24008 MD number	**CHSFRAME_POWERON_MASK** Delete system frames with POWER ON		
Default setting: 0x1	Minimum input limit: 0x0	Maximum input limit: 0x0000007F	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.1, bit 6 in SW 6.4 and higher		
Significance:	This machine data is used to define whether channel-specific system frames are reset in the data management in the case of POWER ON, i.e. offsets and rotations are set to 0, scalings to 1. Mirroring is deactivated. The selection can be made separately for the individual system frames. When bit = 1 is set, the following applies: Bit 0: System frame for PRESET and scratching is kept in the case of POWER ON. Bit 1: System frame for work offset external is kept in the case of POWER ON. Bit 2: System frame for TCARR and PAROT is kept in the case of POWER ON. Bit 3: System frame for TOROT and TOFRAME is kept in the case of POWER ON. Bit 4: System frame for workpiece reference points is retained on POWER ON (SW 6.3 and higher). Bit 5: System frame for cycles is retained on POWER ON (SW 6.3 and higher). Bit 6: System frame for transformation is deleted on POWER ON (SW 6.4 and higher).		
Related to	MD 28082: SYSTEM_FRAME_MASK (configuring screen for channel-specific system frames)		

24010 MD number	**PFRAME_RESET_MODE** RESET mode for programmable frame		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.3		
Significance:	0: Programmable frame is set on RESET according to MD 20110: RESET_MODE_MASK. 1: Programmable frame is retained on RESET.		

24020 MD number	**FRAME_SUPRESS_MODE** Positions for frame suppression		
Default setting: 0x0	Minimum input limit: 0	Maximum input limit: 0x0000003	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Bit screen for configuring positions for frame suppressions (SUPA, G153, G53). The following applies: 0: Positions for display (OPI) are without frame suppression. 1: Position variables are without frame suppression.		

24030 MD number	**FRAME_ACS_SET** SZS coordinate system setting		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	The following applies: 0: SZS is derived from the WCS transformed with $P_CYCFRAME and $P_PFRAME 1: SZS is derived from the WCS transformed with $P_CYCFRAME		

28081 MD number	**MM_NUM_BASE_FRAME** Number of channel-specific base frames		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 16	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	

© Siemens AG, 2003. All rights reserved

28081 MD number	**MM_NUM_BASE_FRAME** Number of channel-specific base frames	
Data type: DWORD		Applies from SW 5.1
Significance:	[Value]: The value is the number of array elements for the predefined array $P_CHBFR[ ]. Battery-backed memory is reserved for this purpose.	

28082 MD number	**MM_SYSTEM_FRAME_MASK** Configuration screen form for channel-specific system frames		
Default setting: 0x1	Minimum input limit: 0x0	Maximum input limit: 0x0000007F	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD		Applies from SW 6.1, bit 6 reserved in SW 6.4 and higher	
Significance:	Bit screen form for configuring channel-specific system frames taken into account in the channel. The following applies: Bit0: System frame for PRESET and scratching. Bit1: System frame for work offset external. Bit2: System frame for TCARR and PAROT. Bit3: System frame for TOROT and TOFRAME. Bit4: System frame for workpiece reference points (SW 6.3 and higher). Bit5: System frame for cycles (SW 6.3 and higher). Bit6: Reserved (SW 6.4 and higher).		

4.4 Axis-specific machine data

32074 MD number	**FRAME_OR_CORRPOS_NOTALLOWED** Adapting active frames		
Default setting: 0x0	Minimum input limit: 0	Maximum input limit: 0x3FF	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD		Applies from SW 4.1 as of SW 7.1 Bit 10 and 11	
Significance:	This machine data defines whether a frame is not permitted on traversing the associated axis and whether the main run offsets are permissible for this axis. Bit == 0: Frame or offset values are permitted. Bit 0 = 1 Programmed zero offset (TRANS) prohibited for indexing axis. Bit 1 = 1 Change of scale (SCALE) prohibited for an indexing axis. Bit 2 = 1 Change of direction (MIRROR) prohibited for an indexing axis. Bit 3 = 1 DRF offset prohibited for an axis. Bit 4 = 1 External zero offset prohibited for an axis. Bit 5 = 1 Online tool correction prohibited for an axis. Bit 6 = 1 Synchronized action offset prohibited for an axis. Bit 7 = 1 Compile cycles offset prohibited for an axis. Consider axial frame for PLC axes or command axes. Bit 8 = 1 Axial frame is considered for PLC axes and for PLC axes that are geometry axes, the tool length offset is considered. Bit 8 = 0 Axial frame and tool length offset are **not** considered for PLC axes (bit evalua tion therefore for reasons of compatibility). Bit 9 = 1 Axial frame and tool length offset are **not** considered for command axes. Bit 9 = 0 Axial frame is considered for command axes and forcommand axes that are geometry axes, the tool length offset is not taken into account.		
Special cases, errors, ...	Alarm 14420: Channel %1Block %2 Indexing axis %3 Frame not allowed. Remedy: Modify machine data CORR_FOR_AXIS_NOT_ALLOWED.		

© Siemens AG, 2003. All rights reserved

4.5 Channel-specific setting data

42440 SD number	**FRAME_OFFSET_INCR_PROG** Zero offsets are traversed on incremental programming		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 4.3		
Significance:	0: In the case of incremental programming of an axis, after changing the frame, only the programmed position delta is traversed. Zero offsets in FRAMES are only traversed with absolute position specification. 1: In the case of incremental programming of an axis, after changing the frame, changes in the zero offsets are traversed. (standard response up to SW 3)		
Related to	SD 42442: TOOL_OFFSET_INCR_PROG		

42980 SD number	**TOFRAME_MODE** Determination of the direction of X and Y axes for frame definition		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after RESET	Protection level: 7	Unit: –	
Data type: DWORD	Applies from SW 5.3		
Significance:	This setting data determines the direction of the X or Y axis for frame definition using TOFRAME or TOROT. With these frame definitions, the Z direction is defined unambiguously, and the rotation round the Z axis is first not fixed. This setting data can be used to define the free rotation such that the newly defined frame deviates from a previously active frame as little as possible. In all cases where the setting data is unequal to zero, an active frame remains unchanged if the Z directions of the old and the new frame are identical. 0: The orientation of the coordinate system is determined by the value of the machine data X_AXIS_IN_OLD_X_Z_PLANE. 1: The new X direction is chosen to lie in the XZ plane of the old coordinate system. In this setting the angle difference between the old and new Y axis will be minimal. 2: The new Y direction is chosen to lie in the YZ plane of the old coordinate system. In this setting the angle difference between the old and new X axis will be minimal. 3: The value chosen is the mean value of the two settings which would have been chosen with 1 and 2. If the value 1000 is additionally added to the values specified, the tool frame is concatenated with possibly active basic frames and settable frames. The behavior is thus compatible to earlier SW (before 5.3).		

■

© Siemens AG, 2003. All rights reserved

5 Signal Description

5.1 Axis/spindle-specific signals

5.1.1 Axes

None

5.1.2 Coordinate systems

None

5.1.3 Workpiece-related actual-value system

None

© Siemens AG, 2003. All rights reserved

5.2 Channel-specific signals

5.2.1 Axes

None

5.2.2 Coordinate systems

Signal from PLC to NCK channel

DB 31, ... **DBX3.0** Data block	**Zero offset** Signal(s)		
Edge evaluation: no		Signal(s) updated:	Signal(s) valid from SW: 2.0
Signal state 1 or signal transition 0 ——> 1	The values of the work offset external set axis-specifically, i.e. preselected, for one axis is taken as the **new** value for calculating the total zero offset between the basic and the workpiece coordinate systems.		
Signal state 0 or signal transition 1 ——> 0	The values of the work offset external set axis-specifically, i.e. preselected, for one axis is **not** taken as the new value for calculating the total zero offset between the basic and the workpiece coordinate systems. The previous value is still valid.		
Signal irrelevant for	$AA_ETRANS[axis] equals zero for all axes.		
Special cases, errors, ...	Signal zero after booting (POWER ON).		
Related to	$AA_ETRANS[axis]		

5.2.3 Workpiece-related actual-value system

DB 21–28	**Signals from NC channel to PLC**							
DBB	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
61								T fct. 1 change /K1/, /K2/
62								D fct 1 change /K1/, /K2/
118 to 119	T function 1 (binary) /S5/, /K2/							
129	D function 1 (binary) /S5/, /K2/							
DB 21–28	**Signals from NCK channel to PLC**							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
208	Number of active G function of G function group 1 (binary) /K1/, /K2/							
209	Number of active G function of G function group 2 (binary) /K1/, /K2/							

© Siemens AG, 2003. All rights reserved

DB 21–28	Signals from NCK channel to PLC							
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
210	Number of active G function of G function group 3 (binary) /K1/, /K2/							
211	Number of active G function of G function group 4 (binary) /K1/, /K2/							
212	Number of active G function of G function group 5 (binary) /K1/, /K2/							
213	Number of active G function of G function group 6 (binary) /K1/, /K2/							
214	Number of active G function of G function group 7 (binary) /K1/, /K2/							
215	Number of active G function of G function group 8 (binary) /K1/, /K2/							
...								
	Number of active G function of G function group n–1 (binary) /K1/							
	Number of active G function of G function group n (binary) /K1/							

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

6 Examples

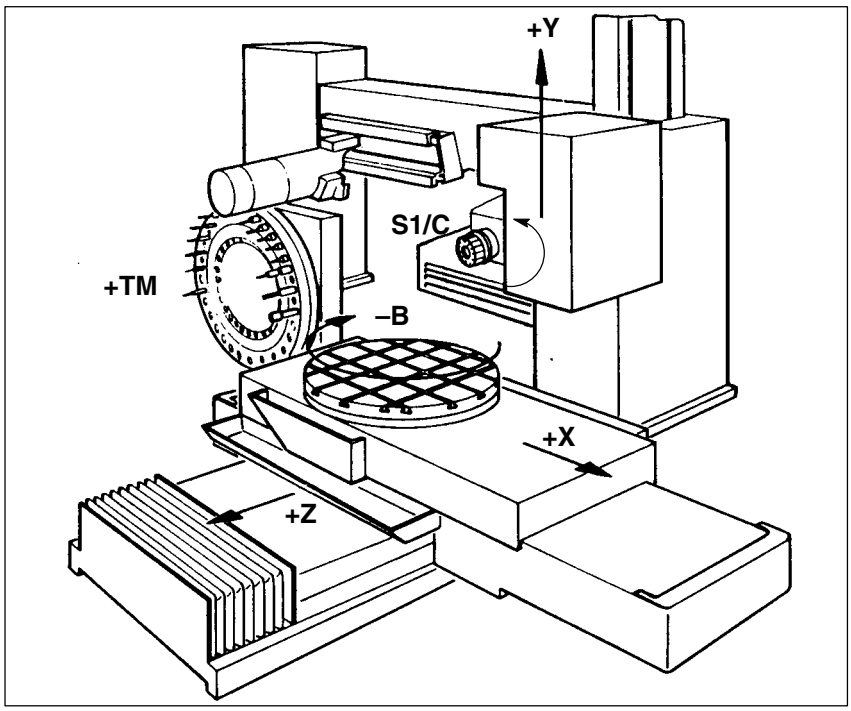
6.1 Axes

Example

Axis configuration for a three-axis milling machine with rotary table:

- 1st machine axis: X1 Linear axis
- 2nd machine axis: Y1 Linear axis
- 3rd machine axis: Z1 Linear axis
- 4th machine axis: B Rotary table (for turning for machining from different sides)
- 5th machine axis: TM Rotary axis for tool magazine (tool revolver)
- 6th machine axis: S1 (Spindle)

- Geometry axis: X (1st channel)
- Geometry axis: Y (1st channel)
- Geometry axis: Z (1st channel)
- Special axis: B (1st channel)
- Special axis: TM (1st channel)
- Spindle: S1/C (1st channel)



© Siemens AG, 2003. All rights reserved

Machine data

AXCONF_GEOAX_NAME_TAB[0] = X
AXCONF_GEOAX_NAME_TAB[1] = Y
AXCONF_GEOAX_NAME_TAB[2] = Z
AXCONF_CHANAX_NAME_TAB[0] = X
AXCONF_CHANAX_NAME_TAB[1] = Y
AXCONF_CHANAX_NAME_TAB[2] = Z
AXCONF_CHANAX_NAME_TAB[3] = B
AXCONF_CHANAX_NAME_TAB[4] = machine tool
AXCONF_CHANAX_NAME_TAB[5] = SP1
SPIND_ASSIGN_TO_MACHAX[AX1] = 0
SPIND_ASSIGN_TO_MACHAX[AX2] = 0
SPIND_ASSIGN_TO_MACHAX[AX3] = 0
SPIND_ASSIGN_TO_MACHAX[AX4] = 0
SPIND_ASSIGN_TO_MACHAX[AX5] = 0
SPIND_ASSIGN_TO_MACHAX[AX6] = 1
MD 30300: IS_ROT_AX[3] = 1
MD 30300: IS_ROT_AX[4] = 1
MD 30300: IS_ROT_AX[5] = 1
MD 30310: ROT_IS_MODULO[3] = 1
MD 30310: ROT_IS_MODULO[4] = 1
MD 30310: ROT_IS_MODULO[5] = 1
MD 30320: DISPLAY_IS_MODULO[3] = 1
MD 30320: DISPLAY_IS_MODULO[4] = 1
MD 20090: SPIND_DEF_MASTER_SPIND = 1

AXCONF_GEOAX_ASSIGN_TAB[0] = 1
AXCONF_GEOAX_ASSIGN_TAB[1] = 2
AXCONF_GEOAX_ASSIGN_TAB[2] =3
AXCONF_MACHAX_USED[0] = 1
AXCONF_MACHAX_USED[1] = 2
AXCONF_MACHAX_USED[2] = 3
AXCONF_MACHAX_USED[3] = 4
AXCONF_MACHAX_USED[3] = 5
AXCONF_MACHAX_USED[4] = 6
AXCONF_MACHAX_NAME_TAB[0] = X1
AXCONF_MACHAX_NAME_TAB[1] = Y1
AXCONF_MACHAX_NAME_TAB[2] = Z1
AXCONF_MACHAX_NAME_TAB[3] = B
AXCONF_MACHAX_NAME_TAB[4] = W1
AXCONF_MACHAX_NAME_TAB[5] = C1

MD No.

MD 10000: AXCONF_MACHAX_NAME_TAB[n]
MD 20050: AXCONF_GEOAX_ASSIGN_TAB[n]
MD 20060: AXCONF_GEOAX_NAME_TAB[n]
MD 20070: AXCONF_MACHAX_USED[n]
MD 20080: AXCONF_CHANAX_NAME_TAB[n]
MD 35000: SPIND_ASSIGN_TO_MACHAX[n]

© Siemens AG, 2003. All rights reserved

6.2 Coordinate systems

Example

Configure 1 global base frame for 1 NCU with 2 channels.
Each channel can describe the base frame, the other channel detects the change (after reactivation of the base frame). Both channels can read the base frame and each channel can activate the global base frame for itself.

Machine data:

```
$MN_AXCONF_MACHAX_NAME_TAB[0] = “X1”          ;Machine axis identifier
$MN_AXCONF_MACHAX_NAME_TAB[1] = “X2”
$MN_AXCONF_MACHAX_NAME_TAB[2] = “X3”
$MN_AXCONF_MACHAX_NAME_TAB[0] = “X4”
$MN_AXCONF_MACHAX_NAME_TAB[1] = “X5”
$MN_AXCONF_MACHAX_NAME_TAB[2] = “X6”
$MN_MM_NUM_GLOBAL_BASE_FRAMES = 1             ;NCU global base frame
$MC_MM_NUM_BASE_FRAMES = 1                    ;Channel base frame
```

Channel 1	Channel 2
$MC_AXCONF_CHANAX_NAME_TAB[0] = “X” $MC_AXCONF_CHANAX_NAME_TAB[1] = “Y” $MC_AXCONF_CHANAX_NAME_TAB[2] = “Z”	$MC_AXCONF_CHANAX_NAME_TAB[0] = “X” $MC_AXCONF_CHANAX_NAME_TAB[1] = “Y” $MC_AXCONF_CHANAX_NAME_TAB[2] = “Z”
$MC_AXCONF_MACHAX_USED[0] = “1” $MC_AXCONF_MACHAX_USED[1] = “2” $MC_AXCONF_MACHAX_USED[2] = “3”	$MC_AXCONF_MACHAX_USED[0] = “4” $MC_AXCONF_MACHAX_USED[1] = “5” $MC_AXCONF_MACHAX_USED[2] = “6”
$MC_AXCONF_GEOAX_NAME_TAB[0] = “X” $MC_AXCONF_GEOAX_NAME_TAB[1] = “Y” $MC_AXCONF_GEOAX_NAME_TAB[2] = “Z”	$MC_AXCONF_GEOAX_NAME_TAB[0] = “X” $MC_AXCONF_GEOAX_NAME_TAB[1] = “Y” $MC_AXCONF_GEOAX_NAME_TAB[2] = “Z”
$MC_AXCONF_GEOAX_ASSIGN_TAB[0] = “1” $MC_AXCONF_GEOAX_ASSIGN_TAB[1] = “2” $MC_AXCONF_GEOAX_ASSIGN_TAB[2] = “3”	$MC_AXCONF_GEOAX_ASSIGN_TAB[0] = “4” $MC_AXCONF_GEOAX_ASSIGN_TAB[1] = “5” $MC_AXCONF_GEOAX_ASSIGN_TAB[2] = “6”

Program X in channel 1	Program Y in channel 2
N.. N100 $P_NCBFR[0] = CTRANS(x, 10) ; Activate NCU global base frame N130 $P_NCBFRAME[0] = CROT(X, 45) ; current NCU global base frame is activated with : one rotation ; Alarm 18310, as rotations are not allowed with ; global frames. N...	N... ; NCU global base frame also active in 2nd channel N510 G500 X10 ; Activate base frame N520 $P_CHBFRAME[0] = CTRANS(x, 10) ; Current frame for channel 2 is activated with an offset

© Siemens AG, 2003. All rights reserved

6.3 Frames

Example 1

The channel axis is to become a geometry axis through geometry axis substitution. The substitution is to give the programmable frame a translation component of 10 in the X axis. The current settable frame is to be retained:
FRAME_GEOX_CHANGE_MODE = 1

```
$P_UIFR[1] = CROT(x,10,y,20,z,30)   ; Frame is retained after geometry axis
                                    ; substitution.
G54                                 ; Settable frame is activated.
TRANS a10                           ; Axial offset of a is also
                                    ; substituted.
GEOAX(1,a)                          ; a becomes X axis;
                                    ; $P_ACTFRAME=
                                    ; CROT(x,10,y,20,z,30):CTRANS(x10)
```

Several channel axes can become geometry axes on a transformation change.

Example 2

Channel axes 4, 5 and 6 become the geometry axes of a 5-axis transformation. The geometry axes are thus all substituted before the transformation.

The current frames are changed when the transformation is activated.

The axial frame components of the channel axes which become geometry axes are taken into account when calculating the new WCS. Rotations programmed before the transformation are retained. The old WCS is restored when the transformation is deactivated.

The most common application will be that the geometry axes do not change before and after the transformation and that the frames are to stay as they were before the transformation.

Machine data:

```
$MN_FRAME_GEOAX_CHANGE_MODE = 1

$MC_AXCONF_CHANAX_NAME_TAB[0] = ”CAX”
$MC_AXCONF_CHANAX_NAME_TAB[1] = ”CAY”
$MC_AXCONF_CHANAX_NAME_TAB[2] = ”CAZ”
$MC_AXCONF_CHANAX_NAME_TAB[3] = ”A”
$MC_AXCONF_CHANAX_NAME_TAB[4] = ”B”
$MC_AXCONF_CHANAX_NAME_TAB[5] = ”C”

$MC_AXCONF_GEOAX_ASSIGN_TAB[0] = 1
$MC_AXCONF_GEOAX_ASSIGN_TAB[1] = 2
$MC_AXCONF_GEOAX_ASSIGN_TAB[2] = 3

$MC_AXCONF_GEOAX_NAME_TAB[0] = ”X”
$MC_AXCONF_GEOAX_NAME_TAB[1] = ”Y”
$MC_AXCONF_GEOAX_NAME_TAB[2] = ”Z”

$MC_TRAFO_GEOAX_ASSIGN_TAB_1[0] = 4
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[1] = 5
$MC_TRAFO_GEOAX_ASSIGN_TAB_1[2] = 6

$MC_TRAFO_AXES_IN_1[0] = 4
$MC_TRAFO_AXES_IN_1[1] = 5
$MC_TRAFO_AXES_IN_1[2] = 6
$MC_TRAFO_AXES_IN_1[3] = 1
$MC_TRAFO_AXES_IN_1[4] = 2
```

© Siemens AG, 2003. All rights reserved

Program:

```
$P_NCBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
$P_CHBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
$P_IFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(z,45)
$P_PFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(x,10,y,20,z,30)

TRAORI   ; Trafo sets GeoAx(4,5,6)
         ; $P_NCBFRAME[0] = ctrans(x,4,y,5,z,6,cax,1,cay,2,caz,3)
         ; $P_ACTBFRAME = ctrans(x,8,y,10,z,12,cax,2,cay,4,caz,6)
         ; $P_PFRAME = ctrans(x,4,y,5,z,6,cax,1,cay,2,caz,3):
         ;                                              crot(x,10,y,20,z,30)
         ; $P_IFRAME = ctrans(x,4,y,5,z,6,cax,1,cay,2,caz,3):crot(z,45)

TRAFOOF; Deactivation of transformation sets GeoAx(1,2,3)
         ; $P_NCBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
         ; $P_CHBFRAME[0] = ctrans(x,1,y,2,z,3,a,4,b,5,c,6)
         ; $P_IFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(z,45)
         ; $P_PFRAME = ctrans(x,1,y,2,z,3,a,4,b,5,c,6):crot(x,10,y,20,z,30)
```

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

7

7.1 Interface signals

7.1.1 Axes

DB number	Bit, byte	Name	Refer-ence
Axis/spindle-specific			
31, ...	60.0	Spindle/no axis	S1

7.1.2 Coordinate systems

DB number	Bit, byte	Name	Refer-ence
Axis/spindle-specific			
31, ...	3.0	Zero offset external	

7.1.3 Workpiece-related actual-value system

DB number	Bit, byte	Name	Refer-ence
Channel-specific			
21, ...	61.0	T function modification	
21, ...	62.0	D function modification	
21, ...	118–119	T function	
21, ...	129	D function	
21, ...	208	Number of active function group 1	
21, ...	209	Number of active function group 2	
...	...	...	
21, ...	230	Number of active function group 29	

© Siemens AG, 2003. All rights reserved

7.2 Machine data

7.2.1 General machine data for operator panel

Number		Identifier	Name	Doc. Refer to
MMC machine data ($MM_ ...)				
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBEDDED (SW 6.1 and higher)	
9242		MA_STAT_DISPLAY_BASE	Numerical basis for display of moving joint STAT	IM4
9243		MA_TU_DISPLAY_BASE	Numerical basis for display of rotary axis position TU	IM4
9244		MA_ORIAXES_EULER_ANGLE_NAME	Display of orientation axes as Euler angle	IM4
9245		MA_PRESET_FRAMEIDX	Value storage scratching and PRESET	IM4
9247	9247	USER_CLASS_BASE_ZERO_OFF_PA	Availability of basic offset in Parameters operating area	IM2/IM4
9248	9248	USER_CLASS_BASE_ZERO_OFF_MA	Availability of basic offset in Parameters operating area	IM2/IM4
	9249	USER_CLASS_VERT_MODE_SK	Provide protection for vertical area soft keys	IM2
	9400	TOOL_REF_GEO_AXIS1	Absolute dimension tool length compensation geometry axis 1	IM2
	9401	TOOL_REF_GEO_AXIS2	Absolute dimension tool length compensation geometry axis 2	IM2
	9402	TOOL_REF_GEO_AXIS3	Absolute dimension tool length compensation geometry axis 3	IM2
9424	9424	MA_COORDINATE_SYSTEM	Coordinate system for actualvalue display	IM2/IM4
	9425	MA_SCRATCH_DEFAULT_MODE	Tool offset computation for geometry axes with scratching	IM2
9440	9440	ACTIVE_SEL_USER_DATA	Active data (frames) are immediately operative after editing	IM2/IM4
9449		WRITE_TOA_LIMIT_MASK	Applicability of MD 9203 to edge data and locationdependent offsets	IM4
9450	9450	MM_WRITE_TOA_FINE_LIMIT	Limit value for wear fine	IM2/IM4
9451	9451	MM_WRITE_ZOA_FINE_LIMIT	Limit value for offset fine	IM2/IM4
	9459	PA_ZOA_MODE	Display mode of zero offset	IM2

© Siemens AG, 2003. All rights reserved

7.2.2 Axes/coordinate systems

Number	Identifier	Name	Reference
General ($MN_ ...)			
10000	AXCONF_MACHAX_NAME_TAB	Machine axis name	
10600	FRAME_ANGLE_INPUT_MODE	Input type for rotation with Frame	
10602	FRAME_GEOAX_CHANGE_MODE	Frames and switchover of geometry axes	
10610	MIRROR_REF_AX	Reference axis for FRAME element mirroring	
10612	MIRROR_TOGGLE	Toggle mirroring	
10613	NCBFRAME_RESET_MASK	RESET response of global base frame	
10615	NCBFRAME_POWERON_MASK	POWER ON response of global base frames	
10650	IPO_PARAM_NAME_TAB	Name of interpolation parameters	
10660	INTERMEDIATE_POINT_NAME_TAB	Name of intermediate point coordinates for G2/G3	
11640	ENABLE_CHAN_AX_GAP	Channel axis gaps allowed	
18600	MM_FRAME_FINE_TRANS	Fine offset for all setable FRAMES and the basic frame	
18601	MM_NUM_GLOBAL_USER_FRAMES	Number of globally predefined user frames	
18602	MM_NUM_GLOBAL_BASE_FRAMES	Number of global base frames	
Channel-specific ($MC_ ...)			
20050	AXCONF_GEOAX_ASSIGN_TAB	Assignment geometry axis to channel axis	
20060	AXCONF_GEOAX_NAME_TAB	Geometry axis name in channel	
20070	AXCONF_MACHAX_USED	Machine axis number valid in channel	
20080	AXCONF_CHANAX_NAME_TAB	Channel axis name/special axis name in channel	
22532	GEOAX_CHANGE_M_CODE	M code for replacement of geometry axes	
22534	TRAFO_CHANGE_M_CODE	M code for transformation changeover	M1
24000	FRAME_ADD_COMPONENTS	Separate programming/modification of additively programmable frame components (SW 5.1 and higher)	
24002	CHBFRAME_RESET_MASK	RESET response of the channelspecific basic frames (SW 5.1 and higher)	
24004	CHBFRAME_POWERON_MASK	POWER ON response of the channelspecific basic frames (SW 5.2 and higher)	
24006	CHSFRAME_RESET_MASK	RESET response of the channelspecific system frames (SW 6.1 and higher)	
24008	CHSFRAME_POWERON_MASK	Delete system frames with POWER ON (SW 6.1 and higher)	
24010	PFRAME_RESET_MODE	RESET mode for programmable frame (SW 6.3 and higher)	
24020	FRAME_SUPPRESS_MODE	Positions for frame suppression (SW 6.4 and higher)	
24030	FRAME_ACT_SET	SZS coordinate system setting (SW 6.4 and higher)	
28081	MM_NUM_BASE_FRAMES	Number of channel-specific basic frames (SW 5.1 and higher)	
28082	MM_SYSTEM_FRAME_FRAMES	Configuration screen form for channel-specific system frames (SW 6.1 and higher)	
Axis-specific ($MA_ ...)			
32074	FRAME_OR_CORRPOS_NOTALLOWED	FRAME or HL offset are allowed	
35000	SPIND_ASSIGN_TO_MACHAX	Assignment spindle to machine axis	S1

© Siemens AG, 2003. All rights reserved

7.3 Setting data

Number	Identifier	Name	Reference
Channel-specific ($SC_ ...)			
42440	FRAME_OFFSET_INCR_PROG	Traversing of zero offset with incremental programming (SW 4.3 and higher)	
42980	TOFRAME_MODE	Determination of the direction of X and Y axes for frame definition (SW 5.3 and higher)	

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103 or HMI Advanced.

7.5 System variable

System variables	Name	Reference
$AC_DRF[axis]	DRF offset (differential resolver function)	
$AA_ETRANS[axis]	Offset value work offset external	
$AA_OFF[axis]	Overlaid motion for programmed axis	
$P_UIFR[n]	Settable frames, activated via G500, G54...G599	
$P_CHBFR[n]	Channel basic frames, activated via G500, G54...G599	
$P_SETFR	System frame for PRESET and scratching in data management (SW 6.1 and higher)	
$P_EXTFR	System frame for work offset external in data management (SW 6.1 and higher)	
$P_PARTFR	System frame for TCARR and PAROT in data management (SW 6.1 and higher)	
$P_TOOLFR	System frame for TOROT and TOFRAME in data management (SW 6.1 and higher)	
$P_WPFR	System frame for workpiece reference points (SW 6.3 and higher)	
$P_CYCFR	System frame for cycles (SW 6.3 and higher)	
$P_TRAFRAME	System frame for transformation (SW 6.4 and higher)	
$P_NCBFR[n]	NCU basic frames, activated via G500, G54...G599	
$P_UBFR	1st basic frame in channel activated after G500, G54...G599. Corresponds to $P_CHBFR[0].	
$P_SETFRAME	Current system frame for PRESET and scratching (SW 6.1 and higher)	

© Siemens AG, 2003. All rights reserved

System variables	Name	Refer-ence
$P_EXTFRAME	Current system frame for work offset external (SW 6.1 and higher)	
$P_PARTFRAME	Current system frame for TCARR and PAROT with orientational toolholder (SW 6.1 and higher)	
$P_TOOLFRAME	Current system frame for TOROT and TOFRAME (SW 6.1 and higher)	
$P_WPFRAME	Current system frame for workpiece reference points (SW 6.3 and higher)	
$P_CYCFRAME	Current system frame for cycles (SW 6.3 and higher)	
$P_TRAFRAME	Current system frame for transformation (SW 6.4 and higher)	
$P_CHBFRAME[n]	Current basic frame in channel, 0 to 15 NCU basic frames can be configured via MD28081: MM_NUM_BASE_FRAMES	
$P_NCBFRAME[n]	Current NCU basic frame, 0 to 15 NCU basic frames can be configured via MD18602: MM_NUM_GLOBAL_BASE_FRAMES	
$P_ACTBFRAME	Current chained total basic frame	
$P_BFRAME	Current first basic frame in the channel. Corresponds to $P_CHBFRAME	
$P_IFRAME	Current settable frame	
$P_PFRAME	Current programmable frame	
$P_ACTFRAME	Current total frame	
$P_UIFRNUM	Number of active frame $P_UIFR	
$P_NCBFRMASK	Bit mask for definition of NCU global basic frames	
$P_CHBFRMASK	Bit mask for definition of channel-specific basic frames	

$AA_ETRANS[X] is an axis-specific system variable of the DOUBLE type. The default setting in the system for this variable is zero. Values set by the user are activated by IS “Work offset external” (DB31–61, ... DBX3.0).

SW 5.1 and higher

$P_CHBFR[n] (basic frame in channel that is activated after G500, G54..G599)
$P_CHBFRAME[n] (current basic frame in channel)
$P_NCBFR[n] (NCU global basic frame that is activated after G500, G54..G599)
$P_NCBFRAME[n] (current NCU global basic frame)
$P_ACTBFRAME (current chained total basic frame)
$P_NCBFRMASK (bit mask for definition of NCU global basic frame).
$P_CHBFRMASK (bit mask for definition of channelsspecific basic frame)

For further details about frames and system variables, please see
Reference: /PGA/, “Programming Guide Advanced”
Chapter 6 Frames or Chapter 15 List of System Variables.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Communication (K4)

1	**Brief Description**		**1/K4/1-3**
	1.1	External data communications connector	1/K4/1-3
2	**Detailed Description**		**1/K4/2-5**
	2.1	Serial interfaces (RS232)	1/K4/2-5
	2.1.1	HMI Embedded with SW 6.1 and later (MMC 100)	1/K4/2-6
	2.1.2	HMI Advanced, SW 6.1 and higher (MMC 101–103)	1/K4/2-7
	2.2	Parallel interface (Centronics)	1/K4/2-8
	2.3	Universal Serial Bus USB	1/K4/2-9
	2.4	PS/2 keyboard/mouse interface	1/K4/2-9
	2.5	External floppy disk drive	1/K4/2-10
	2.6	MPI/DP interface (RS485)	1/K4/2-10
	2.6.1	Communication nodes and networking rules	1/K4/2-11
	2.6.2	Structuring and parameterization of logical address	1/K4/2-13
	2.7	General fundamentals about PROFIBUS DP	1/K4/2-15
	2.8	SINUMERIK 840Di with PROFIBUS DP drives	1/K4/2-16
	2.8.1	Cyclical communication with PROFIBUS DP interface	1/K4/2-16
	2.8.2	I/O address space	1/K4/2-16
	2.8.3	Structure of the logic I/O address space	1/K4/2-17
	2.8.4	Configure SINUMERIK 840Di for PROFIBUS DP drives	1/K4/2-19
3	**Supplementary Conditions**		**1/K4/3-21**
4	**Data Descriptions (MD, SD)**		**1/K4/4-23**
	4.1	XON character	1/K4/4-23
	4.2	XOFF character	1/K4/4-24
	4.3	End of transmission character	1/K4/4-24
	4.4	Special bits	1/K4/4-25
	4.5	Device type	1/K4/4-27
	4.6	Baud rate	1/K4/4-28
	4.7	Data bits	1/K4/4-28

© Siemens AG, 2003. All rights reserved

4.8 Parity 1/K4/4-29
4.9 Stop bits 1/K4/4-29
4.10 RS232 interface 1/K4/4-30
5 Signal Descriptions 1/K4/5-31
5.1 Serial interfaces 1/K4/5-31
5.2 Parallel interface 1/K4/5-32
6 Example 1/K4/6-33
6.1 Parameters for serial printer 1/K4/6-33
6.2 Parameters for archiving with PG/PC 1/K4/6-33
6.3 Parameters for DIN programs 1/K4/6-34
6.4 Tape input/output 1/K4/6-34
6.5 Import of machine data 1/K4/6-34
6.6 Import of binary data (FDD, MSD) 1/K4/6-35
7 Data Fields, Lists 1/K4/7-37
7.1 Machine data 1/K4/7-37

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

1.1 External data communications connector

Communication methods

The following communication equipment is available on the control depending on whether the PCU 20 or PCU 50 (MMC module) component is installed:

- Serial interfaces (RS 232) with
  HMI Embedded and PCU 20
  Conn. X9 for COM1/V.24/TTY and
  Connector X11 for COM2/V.24)
  HMI Advanced with PCU 50
  (COM1/V24/AG or TTY and
  COM2 for RS232).
  MMC 100 (connector X6)
  MMC 101–103 (connectors X6, X7)
- Parallel interface (Centronics) with
  HMI Advanced and PCU 50 (LPT1)
  MMC 101–103 (connector X8)
- I/O for USB (Universal Serial Bus) with
  HMI Embedded and PCU 20
  Connector X40 for USB interface
  HMI Advanced with PCU 50 OP010,
  OP012/15 two USB interfaces.
- PS/2 keyboard/mouse interface with
  HMI Embedded and PCU 20
  Connector X6 for PS/2 interface
  HMI Advanced with PCU 50 for two
  separate PS/2 interfaces.
- Serial interface (RS485) with
  MPI/DP (MultiPointInterface /
  PROFIBUS DP)
  HMI Embedded and PCU 20
  connector X800 MPI/DP interface
  HMI Advanced with PCU 50
  connector
  MPI/DP interface for connecting,
  e.g. a machine control panel.
- Ethernet for local network (LAN)
  HMI Embedded and PCU 20
  Connector X805 for RJ45 interface
  HMI Advanced with PCU 50 for RJ45
  Interface to (connector Ethernet).
  Connection option for network card
  MMC (AT design) in ISA adapter
  (as option only).
- Display interface (LVDS) with
  (TFT operator panel front)
  HMI Embedded and PCU 20
  Connector X400 for LVDS interface
  HMI Advanced with PCU 50
  LVDS interface for TFT display.
- VGA interface with
  (external monitor)
  HMI Advanced and PCU 50
  (PGA connector).
- Option for connecting an external floppy disk drive with HMI Advanced and PCU 50 (MMC 101–103).

© Siemens AG, 2003. All rights reserved

Serial RS232

The serial interfaces are provided as standard for the purpose of importing or exporting data to and from programming devices or PCs using data transfer programs (e.g. PCIN).
Example: Import of parts programs of machine data.

Parallel LPT1

The parallel interface is primarily intended as a printer port. A streamer can also be connected for data backup purposes (e.g. OEM_applications).

I/O for USB

HMI Embedded with PCU 20 (connector X40) and HMI Advanced with PCU 50 feature an optional USB interface (Universal Serial Bus) for serial data transfer, e.g. to a printer with USB port.

In the case of HMI Advanced with PCU 50, a second USB port, e.g. for a USB mouse, is fitted on the front of the operator panel (OP010, OP012 or OP015).

PS/2 keyboard/ mouse interface

A keyboard or mouse can be connected additionally with HMI Embedded and PCU 20 (connector X6).

Two individual PS/2 ports for a trackball keyboard and a mouse are available with HMI Advanced and PCU 50.

MPI/DP RS485

HMI Embedded with PCU 20 and HMI Advanced with PCU 50 can both communicate with the machine control panel (MCP) or with other I/O devices (nodes, stations) via the MPI/DP interface.

- Machine control panel (MCP) e.g. receive messages from PLC.
- PLC e.g. access to drive parameters
- SIMODRIVE 611 universal e.g. set parameters using SimoCom U Tool

PROFIBUS DP, which meets the requirements of European fieldbus standard EN 50170 Part 2, supports communication on this RS485 serial interface.

Floppy disk drive

With HMI Advanced and PCU 50 (MMC101–103) it is possible to connect a 3.5” floppy disk drive externally to facilitate data communication between the machine and, for example, the production planning department.

Network cards

HMI Embedded with PCU 20 and HMI Advanced with PCU 50 can be connected to a local network (LAN) via the Ethernet RJ45 interface.

Extension box as option for MMC
Network cards (e.g. Ethernet) can be inserted in the ISA_extension box (option for MMC). These cards can be used, for example, to import programs directly from the master computer to the NC. However, OEM_configuring of the MMC is required before such cards can be used.
The card functionality is not an integral component of the system.

Note
For OEM_users only.

■

© Siemens AG, 2003. All rights reserved

Detailed Description

2

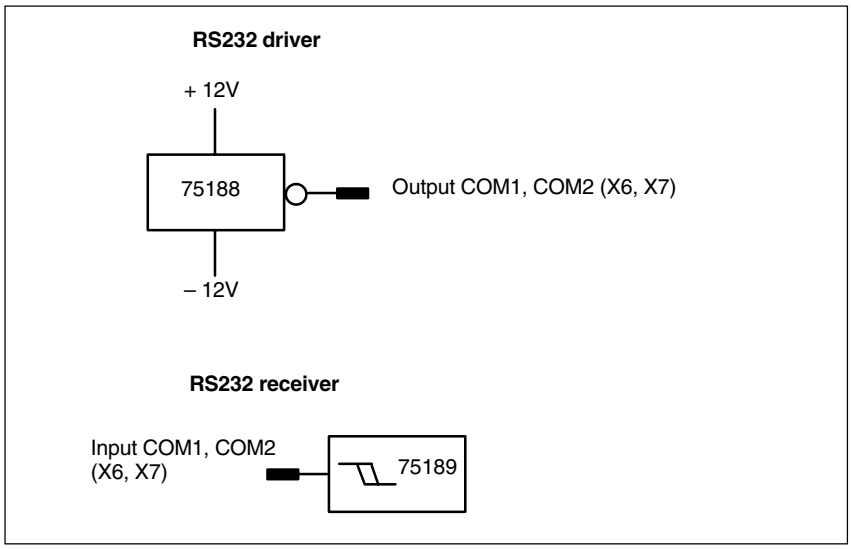
2.1 Serial interfaces (RS232)

The serial interfaces are available at the following connectors with HMI Embedded and PCU 20, HMI Advanced with PCU 50 and with the MMC modules:

• HMI Embedded	1st serial interface (RS 232)	Connector COM1/RS232
	2nd serial interface (RS 232)	Connector COM2/RS232
• HMI Advanced	1st serial interface (RS 232)	Connector COM1/RS232
	2nd serial interface (RS 232)	Connector COM2
	Parallel interface (Centronics)	Connector LPT1/Printer
• MMC 100	Serial interface (RS 232)	Connector X 6
MMC 101–103	1st serial interface (RS 232)	Connector X 6
	2nd serial interface (RS 232)	Connector X 7
	Parallel interface (Centronics)	Connector X 8

Maximum length RS232 leads of a maximum length of 30m can be connected.

Connection



© Siemens AG, 2003. All rights reserved

2.1.1 HMI Embedded with SW 6.1 and later (MMC 100)

The serial interface COM1 or COM2 with HMI Embedded and PCU 20 (X6 with MMC 100 COM1) can be set as follows in 3 parameter sets:

- RS232C user COM1/COM2 parameters for DIN programs
- RS232C printer COM1/COM2 parameters for serial printer
- RS232C PG/PC COM1/COM2 parameters for archiving with PG/PC.

These parameter sets are entered via the operator interface under SERVICES-DATA-RS232-SET. The preset values can be adapted to the individual terminal.
The operator inputs required to modify parameters are explained in
References: /BEM/, HMI Embedded Operator's Guide

COM1 or COM2

The serial interface to be used for initiation of a file transfer is selected in the following display machine data with HMI Embedded:
MD 9309: V24_USER_LINE,
MD 9319: V24_PRINTER_LINE and
MD 9329: V24_PG_PC_LINE

File types

The serial interface for HMI Embedded with PCU 20 (MMC 100) is used to transfer the following file types:

Abbreviation/ Extension	File type	Meaning of abbreviation
BIN	Binary files	**Bin**ary Files
BOT	SIMODRIVE 611D boot files	**BOO**T Files
COM	Comment file	**Com**ment File
CPA	Compiler configuring data	**C**ompiler **P**rojecting Dat**a**
DIR	Directory	**Dir**ectory
GIA	Gear interpolation data	**G**ear **I**nterpolation Dat**a**
GUD	User data (global)	**G**lobal **U**ser **D**ata
IKA	Interpolatory compensation	**I**nterpolative Compens**a**tion
INI	Initialization data	**Ini**tializing Data
LUD	User data (local)	**L**ocal **U**ser **D**ata
MPF	Parts programs	**M**ain **P**rogram **F**ile
OPT	Options	**Opt**ions
RPA	R parameter with value assignment	**R P**arameter **A**ctive
SEA	Addresses with value assignment	**Se**tting Data **A**ctive
SPF	Subprogram	**S**ub **P**rogram **F**ile
SYF	System files	**Sy**stem **F**iles
TEA	NC machine data	**Te**sting Data **A**ctive
UFR	Offset external (WOE)	**U**ser **Fr**ame
WPD	Workpiece directory	**W**ork **P**iece **D**irectory

© Siemens AG, 2003. All rights reserved

Parameterization

The following parameters can be altered:

Device types:	RTS/CTS or XON/XOFF
Baud rates:	300, 600, 1200, 2400, 4800, 9600
Parity:	None, even, odd
Data bits:	7 or 8
Stop bits:	1 or 2
Characters for	XON XOFF Text end
Special functions	Start with XON Program beginning with LF Block end with CR LF Stop with transmission end character Evaluate DSR signal Leader and trailer Tape format Time monitoring

Upper and lower case can be used for entering the HEX number. Display on the screen is in lower case, e.g. whether 1A or 1a is entered, the display is always 1a.

These parameters are stored internally as machine data and can be altered both via the parameter screen form and in the machine data area.
For a description of parameters/machine data, please refer to Chapter 4, Data Descriptions.

2.1.2 HMI Advanced, SW 6.1 and higher (MMC 101–103)

Interfaces COM1/RS232/AG and COM2 with HMI Advanced (X6 and X7 with MMC101–103) can be set in two parameter sets

- RS232C user COM1 parameters for DIN programs
- PG/PC COM2 parameters for archiving with PG/PC

Addresses

The 2 interfaces are permanently routed to the following addresses:

- COM1: 3F8–3FF
- COM2: 2F8–2FF

File types

Files of any type can be imported or exported via serial interfaces COM1/RS232/AG and COM2 of HMI Advanced (X6 and X7 of MMC 101–103). The operator inputs for selecting files for import and output are explained in

References: /BAD/ HMI Advanced Operator's Guide
/BA/ Operator's Guide.

© Siemens AG, 2003. All rights reserved

Parameterization

The following parameters can be altered via the operator interface under SERVICES-INTERFACES:

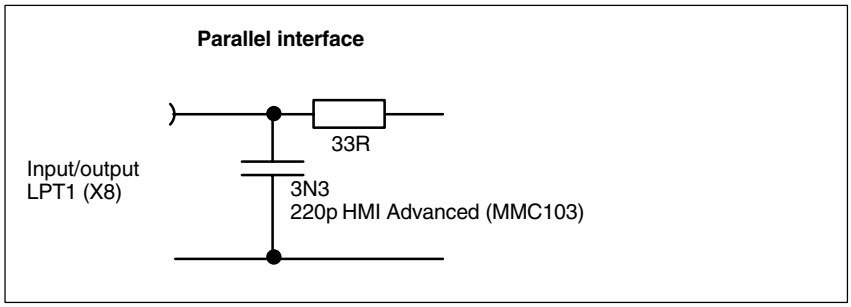
- Interface assignment (COM1, COM2)
- Protocol (Xon, Xoff, RTS/CTS)
- Parity
- Stop bit
- Number of data bits
- Baud rate
- Archive format (binary format (PC format), tape format with LF only, tape format with CR+LF)
- Leader/trailer
- Special transmission character
- Time monitoring.

2.2 Parallel interface (Centronics)

The parallel interface is available with HMI Advanced and PCU 50 (MMC 101 and 102/103) only and constructed as a bidirectional Centronics interface. It is primarily used as the printer interface.

• HMI Advanced with PCU 50 (MMC 101/102/103) (Centronics)	Parallel interface LPT1 (X8)

Connection



Maximum length

A connecting lead (e.g. for printer) of a maximum length of 2.5m may be connected to the interface.

Parameterization

The parallel interface cannot be parameterized.

Address

The parallel interface is permanently routed to address 3BC–3BE.

© Siemens AG, 2003. All rights reserved

2.3 Universal Serial Bus USB

USB printer or mouse interface

The Universal Serial Bus is an interface for serial data transmissions in both directions. It is used primarily for a printer or mouse with USB port with HMI Embedded with PCU 20 (connector X40) and HMI Advanced with PCU 50.

Maximum length

The maximum cable length must not exceed 5m.

Parameterization

The USB interface cannot be parameterized.

HMI Embedded with PCU 20 and HMI Advanced with PCU 50 feature a second USB interface on the operator panel front. The two USB interfaces use different USB channels.
The connector pin assignment is shown in Table 2-1 below.

Table 2-1 Assignment of USB interface (high current 500 mA)

Pin	Name	Type	Meaning
1	VCC	V	+ 5V supply voltage
2	– Data	I/O	USB data – (channeldependent)
3	+ Data	I/O	USB data + (channeldependent)
4	GND	V	System ground (reference potential)

2.4 PS/2 keyboard/mouse interface

PS/2-Trackball-keyboard interface

It is possible to operate an external trackball keyboard with PS/2 port on HMI Advanced with PCU 50 (keyboard connector).

Parameterization

The PS/2 interface cannot be parameterized. The connector pin assignment for a trackball keyboard is shown in Table 2-2 below:

Table 2-2 Assignment of PS/2 trackball keyboard interface

Pin	Type	Meaning
1	I/O	Keyboard data cable
2	I/O	Trackball data cable
3	V	System ground (reference potential)
4	V	5V supply voltage (current-limited)
5	I/O	Keyboard clockline
6	I/O	Trackball clockline

© Siemens AG, 2003. All rights reserved

PS/2 mouse interface

An external mouse with PS/2 port can be connected with HMI Embedded and PCU 20 (connector X6) and HMI Advanced with PCU 50 (mouse connector). The connector pin assignment for a PS/2 mouse is shown in Table 2-3 below:

Table 2-3 Assignment of PS/2 mouse interface

Pin	Type	Meaning
1	I/O	Data cable
2	–	Not assigned
3	V	System ground (reference potential)
4	V	5V supply voltage (current-limited)
5	I/O	Clockline
6	–	Not assigned

2.5 External floppy disk drive

HMI Advanced

A floppy disk drive interface (without power supply for the drive) is available at the 34-pin plug connector (X9 of MMC 101–103) floppy interface of HMI Advanced with PCU 50. The interface is specifically provided for the connection of a 3.5” diskette unit.

Maximum length

A connecting lead of a maximum length of 0.5m may be connected to the interface. The lead is included in the scope of supply of the diskette drive.

Parameterization

The interface cannot be parameterized.

2.6 MPI/DP interface (RS485)

MPI/DP operator components

An MPI/DP bus system can be operated via the serial RS485 interface of HMI Embedded with PCU 20 (connector X800) and HMI Advanced with PCU 50 (connector MPI/DP).

The PROFIBUS DP, which is compliant with the European field bus standard, supports communication on an MPI bus via a serial RS485 interface. Data can be transferred on this interface via a cable or optical fiber.

© Siemens AG, 2003. All rights reserved

MPI/DP Bus interface

The connector pin assignment for this MPI/DP interface is shown in Table 2-4 below:

Table 2-4 Assignment of MPI/DP interface of PCU 50

Pin	Name	Type	Meaning
1,2	NC	–	Not assigned
3	MPSS_A	I/O	Signal line A of MPI module
4	RTS_AS	I	Control signal for receive data stream. Signal 1 active if directly connected interface module is sending.
5	2M	V	Return line (GND) of 5V supply. Current load by load connected between 2P5 and 2M max. 90mA.
6	2P5	V	5V supply current load as for 2M
7	NC	–	Not assigned
8	XMPSS_B	I/O	Signal line B of MPI module
9	RTS_PG	I	RTS signal of MPI module
Shield		–	On connector housing

2.6.1 Communication nodes and networking rules

MPI/DP Bus nodes

HMI Embedded with PCU 20 (connector X800) and HMI Advanced with PCU 50 (connector MPI/DP) feature a serial RS485 interface which can operate as an MPI bus to allow communication between components

- handheld unit (distributor box connector X4) HHU
- machine control panel (connector X20) MCP
- NCU 561.2–573.2/3/4 (connector X101) NCU

be determined. Other nodes such as SIMODRIVE 611 digital drives, or stations such as a SIMATIC S7 PLC, a programming device (PG) or a handheld programming unit (HPU) as a start-up tool can be operated on this MPI bus. A CPU program must be set up in the PLC for every component coupled to the MPI/DP bus.

© Siemens AG, 2003. All rights reserved

Assignment of CPU programs

Each node on the MPI bus must have a bus address between 0 and 31. It is included in the network through the assignment of an MPI bus address to each CPU program.

Node	CPU program	MPI bus address
PG 740 programming device	PG	0
HMI Advanced operator panel	HMI	1
SIMATIC S7 PLC AS314	PLC-CPU	2
Machine control panel	MCP	6
SINUMERIK 840D NCU	NC	13
Handheld unit	HHU/HT 6	15
Handheld programming unit	HPU	11

Networking rules

The following **basic rules** for setting up an MPI network must be observed:

1. The bus line must be terminated **at both ends** by switching in/activating the terminating resistor in the MPI connector.

Note

- No more than two terminators may be activated.
- Bus terminating resistors are **integral components** of the HHU/HPU devices. For this reason, the bus terminators on the distributor boxes **must not** be activated.

---

2. **At least one terminator** must be connected to a **5V voltage supply**. This is automatically the case when the MPI connector with active terminating resistor is connected to a live device.
3. Spur lines (cables leading from bus segment to node) should be as short as possible and not exceed 5m. Unused spurs must be removed.

Note

Avoid using spur lines and piggyback connectors whenever possible.

---

4. Each MPI node must be inserted **before** it is activated. When an MPI node is removed, the link must be deactivated **before** the connector is removed.
5. One HHU and one HPU, or a maximum of two of these components, can be connected per bus segment. Alternatively, a machine control panel (MCP) can be combined with an HHU or a PHG.
6. The maximum permissible cable lengths are listed below:

   MPI (187.5 kbaud): Maximum total cable length 1000m (without repeaters)

   MPI (187.5 kbaud): Maximum total cable length 2000m (with repeaters)

   per bus segment: Maximum total cable length 200m

© Siemens AG, 2003. All rights reserved

Example

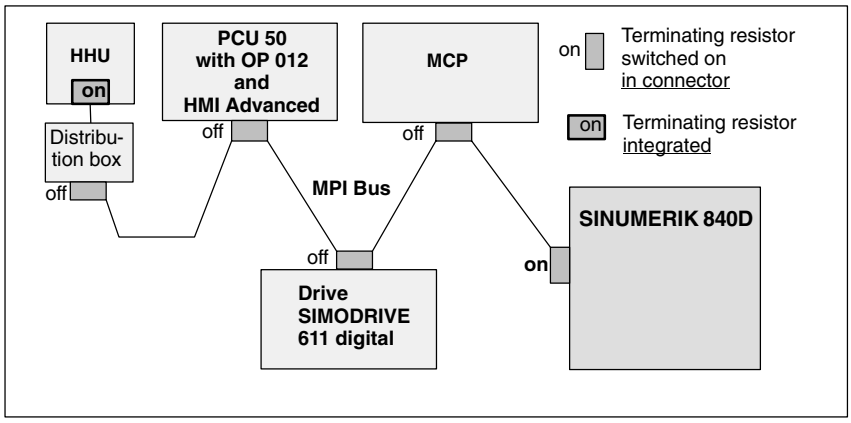


Fig. 2-1 MPI network with terminating resistors

Data transfer rate

The data transfer rate on the MPI/DP bus of the SINUMERIK 840D must be set to 1.5 Mbaud.

The same applies to all other nodes who data transfer rate is set manually directly on the device using appropriate DIP switches.

2.6.2 Structuring and parameterization of logical address

Bus addresses and logical address

Every MPI/DP bus node is networked via CPU programs. In this way, it is assigned a physical bus address (default address 6 for MCP, 15 for HHU, HPU 11, etc.). Bus addresses are logically addressed by means of GD circuit parameter settings.

The bus address in the relevant GD circuit is converted in the control system by the basic PLC program.

The table below shows the relationship between bus addresses and GD circuit.

Table 2-5 Bus address/GD circuit relationship

Bus addresses on MPI	GD circuit
15,14,13	1
12,11	2
10, 9	3
8, 7	4
6	8
5, 4	5

Several MPI/DP bus addresses can be grouped in a GD circuit. They are always physically addressed via GD circuits.

Every machine control panel must be addressed with its own GD circuit. The bus address, and thus also the relevant GD circuit, is set via the DIP-FIX switch (S3) on the rear panel. A 4-way DIP switch (S2) is provided on the HHU for this purpose.

© Siemens AG, 2003. All rights reserved

Parameterization

Logical addressing of components takes place in the basic PLC program and must always be parameterized to match the parameters of the relevant device.

The communication parameters of each node are defined and activated via the basic PLC program in function block FB1.

In addition to the first MCP, a second MCP or a HHU can be active simultaneously as an additional operator component. The PLC program must be adapted accordingly for the synchronization of more than one operator component.

Note

To be able to utilize a HHU alternately or simultaneously to the MCP on an automation system, the user (machine manufacturer) must adapt the PLC program accordingly.

MCP simulation is available for the HPU; this must be parameterized with FB1 as an MCP for the HPU.

Parameterization of PLC

The PLC program is modular in design. It comprises function blocks:

- Startup and synchronization (OB 100)
- Cyclical mode (OB 1)
- Process interrupt handling (OB 40)

The user (machine manufacturer) must call the relevant section of the basic program for these three function blocks. The communication parameters of the MCP are denoted MCPx... (where x = 1 or 2) in function block FB1.

Configurations

With HMI Embedded and PCU 20 and with HMI Advanced and PCU 50 it is possible to operate a PROFIBUS DP via the MPI/DP interface (RS 485) to permit communication with the following I/O devices:

• Machine control panel (MCP)	Cyclic communication: Setpoint/actual value transfer using process data e.g. receive message frames from PLC.
• PLC and SIMODRIVE 611 digital	Acyclic communication: Access to drive parameters e.g. data exchange with SIMATIC OP

Note

A detailed description of the basic PLC program and function block FB 1 can be found in:

References: /FB1/, Description of Functions: P3 Basic PLC Program Section: FB 1: RUN_UP Basic program, start-up section

For further information about starting up the MPI interface, please see:

References: /IAD/, Installation & StartUp Guide, MPI / OPI Settings
/HBI/, 840Di Manual, Chapter 5 MPI Components.

© Siemens AG, 2003. All rights reserved

2.7 General fundamentals about PROFIBUS DP

General

PROFIBUS DP is an international, open field bus standard which is defined in European field bus standard EN 50170 Part 2.

The PROFIBUS DP is optimized for high-speed, time-critical data transmission at field level. The components communicating via the PROFIBUS DP are categorized as either master or slave components. Cyclical and non-cyclic data exchanges between a master and the slaves assigned to it take place via the field bus.

The available modes of communication are described below:

- **Cyclical communication**
  Useful data for cyclical operations are referred to as Parameter Process Data Objects (PPO). They are divided into two areas within a message frame
  - Parameter area (PKW (parameter identifier value))
  - Process data area (PDA (process data)

  Cyclical DP communication using “SIMODRIVE 611 universal” drives offers the following functionality:
  - Cyclical standard DP operation
    (a new cycle is begun when the old cycle has finished)
  - Setpoint/actual value transfer using PDA
  - Isochronous functionality (isochronous mode)
- **Non-cyclic communication**
  Access to drive parameters
  - Parameter setting via the “SimoCom U” tool
  - Data exchange with SIMATIC operating panel (SIMATIC OP)

PROFIBUS DP nodes

In the content of PROFIBUS DP, a distinction is made between the following basic device types:

- **Master** or **DP master** (active nodes)
  Devices that constitute a master on the bus determine the data exchange on the bus and are designated as active nodes.
  With masters, two classes are distinguished:
  - DP master, class 1 (DPMC1)
    This term denotes central master devices that exchange information with the slaves within defined message cycles.
    Example: SINUMERIK 840Di, SIMATIC S7–CPU 315
  - DP master, class 2 (DPMC2)
    These are devices for configuring, start-up, operation and monitoring during running bus operation.
    Example: SimoCom U (SIMATIC Manger active)
- **Slave** or **DP slave** (passive nodes)
  These devices may only receive messages, acknowledge and transmit messages to the master upon its request.
  Example: SIMODRIVE 611 universal, SIMATIC S7 I/O modules

© Siemens AG, 2003. All rights reserved

2.8 SINUMERIK 840Di with PROFIBUS DP drives

2.8.1 Cyclical communication with PROFIBUS DP interface

Communication

The SINUMERIK 840Di control system is provided with a PROFIBUS DP interface for connecting the I/O devices. Communication between the SINUMERIK 840Di (NC and PLC) as the master and the slave components via PROFIBUS DP is characterized by:

- Configurable, equidistant DP cycle (cyclical communication)
- Synchronization of DP slaves via the DP master using a Global Control message in every DP cycle (isochronous mode)
- Independent maintenance of internal cycle by DP slaves in the event of communication failure (useful data backup)

For further information about PROFIBUS DP and SINUMERIK 840Di, see:
References: /HBI/, SINUMERIK 840Di Manual, Chapter 7

2.8.2 I/O address space

The I/O devices connected to PROFIBUS DP (node, stations) are placed in a logic I/O address space. Using this scheme of subrack/slot addressing, the form of addressing slave/slot is set up for the PROFIBUS DP areas.

DP Slaves

A **slave** is a PROFIBUS node that is supplied by a master. It is addressed from the bus.

PROFIBUS addresses are possible within the range between 0 and 125.

A max. of 126 slaves is theoretically possible on a master. The PROFIBUS address 126 is reserved for start-up and 127 is reserved as the broadcast address.

Slot

The individual (real or virtual) modules of a slave can be accessed via **slots** (1 slot = 1 module = 1 data block = an agreed functionality). A slot may contain input and/or output data.

A max. of 255 slots is possible per slave.

The slot number is an integer in a slave between 0 and 254 and starts at 0 in ascending order. Gaps are permitted. The slot with number 0 is reserved for the slave itself.

The diagnostic slot with a special code represents the slave itself. A slot is a consisten data area from the point of view of the bus configuration. A slave on PROFIBUS DP can have several slots of the same and/or different kind. Slots without I/Os are permitted, e.g. diagnostic slots.

© Siemens AG, 2003. All rights reserved

Example of slot assignment

The slot assignment is shown for the following PROFIBUS nodes:

- ET200 with I/Os:
  - An input module corresponds to a slot.
- The 611 universal 2-axis module consists of the following slots:

Table 2-6 Slot assignment of the 611 universal/2-axis module

Axes	Slots	Meaning
1st axis	Slot 4	No PKW
	Slot 5	Actual value (inputs)
	Slot 6	Setpoint (outputs)
	Slot 7	Axis separator
2st axis	Slot 8	No PKW
	Slot 9	Actual value (inputs)
	Slot 10	Setpoint (outputs)

2.8.3 Structure of the logic I/O address space

Logical address

The mapping of the cyclic data of the slaves in the logic address space is preset from a PROFIBUS SDB (system data block). For each slot, a logic basic address is defined in the address space under which the data of the slot are assigned. The assignment of logic base address and PROFIBUS address is via the PROFIBUS SDB (system data block). Furthermore, each slave in the logic address space is assigned a diagnostic address.

Configuration

The PROFIBUS addresses are defined in the configuration tool **HW Config** of the SIMATIC Manager. **HW Config** ensures that no overlaps of assigned address ranges occur for slots and diagnosis.

For SIMODRIVE 611 universal drives, the PROFIBUS address of the relevant slaves are generated using this Step 7 module “HW Config”.

For further information about configuring distributed I/O devices, see:
References: Getting Started with STEP7, Chapter 11 “Creating distributed I/Os using PROFIBUS DP”.

Addressing of the nodes

The nodes/slaves are addressed as follows:

- PLC slaves
  The addressing in the logic I/O address space is carried out via programming in the PLC user program.
- NC slaves
  The addressing in the logic I/O address space is carried out via programming in the NC machine data.

© Siemens AG, 2003. All rights reserved

Address space mapping

The coexistence of PLC and NC in a common I/O address space is ensured by parameterizing subareas for the individual functions. The NC can use address ≥ 272 only (compare the conditions of MD 13050: DRIVE_LOGIC_ADRESS). The joint use of input addresses of PLC and NC is possible.
Each slave can have several subareas of the address space. A possible address space distribution diagram is shown in Fig. 2-2.

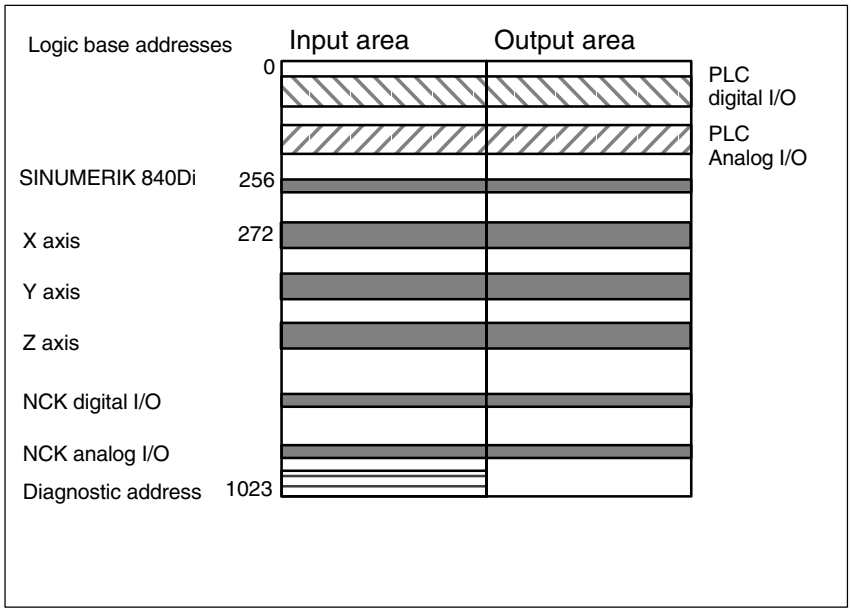


Fig. 2-2 Address space mapping (diagram)

Connected I/O devices

The I/O devices connected to PROFIBUS DP (such as SIMODRIVE 611 universal drives, I/O modules) will be allocated in a logic I/O address space. The addressing type slave/slot will be established for the PROFIBUS DP areas.

- A **DP slave** is a PROFIBUS node that is supplied from a master. It is addressed using a PROFIBUS address (e.g.: 12).
- The individual modules of a slave can be accessed via **slots**. A slot may contain input and/or output data.
- The whole I/O device (slave, e.g. drive SIMODRIVE 611 universal) is addressed as a **slot 0** on PROFIBUS.
- The assignment of the individual slots to the logic address space is carried out via a PROFIBUSSDB. The logic address space covers the range from 0 to 1023.

Note

By default, the PROFIBUS diagnostic address 1023 is assigned to the SINUMERIK 840Di and is therefore not available for other nodes on PROFIBUS.

© Siemens AG, 2003. All rights reserved

2.8.4 Configure SINUMERIK 840Di for PROFIBUS DP drives

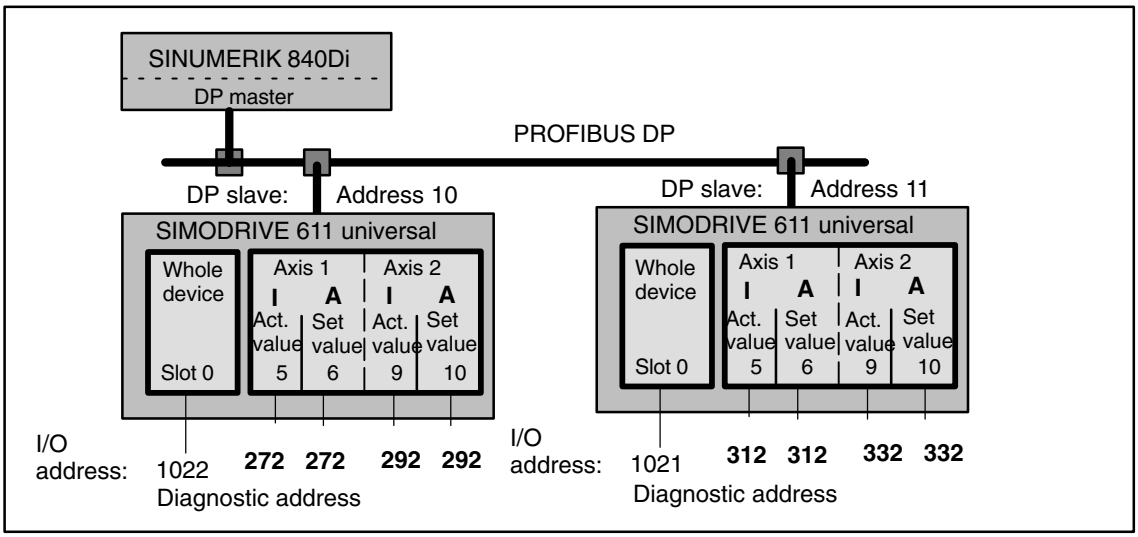


Fig. 2-3 Example of a PROFIBUS configuration with two drives 611 universal/2axis module

Address space for drives

Errors in the address space distribution (overlap of ranges) are detected during power up by comparing the machine data one with another and with the PROFIBUS SDB.

Example of MD with n drives:

The drives (axes and spindles) of the NC are allocated in the logic I/O address space using machine data as follows.

Axis MD	Default value	Meaning
CTRLOUT_SEGMENT_NR	= 5	PROFIBUS DP (not displayed)
CTRLOUT_MODULE_NR	= n	Index to DRIVE_LOGIC_ADDRESS
CTRLOUT_NR	= 1	Drive selection
ENC_SEGMENT_NR	= 5	PROFIBUS DP (not displayed)
ENC_MODULE_NR	= n	Index to DRIVE_LOGIC_ADDRESS
ENC_INPUT_NR	= 1	1st or 2nd encoder

MD 13050: (logic address for drive, also spindle)
These MD represent the joining element to describe the bus configuration in the PROFIBUS SDB (system data block).

DRIVE_LOGIC_ADDRESS[0]	=272	Base address for drive 1
DRIVE_LOGIC_ADDRESS[1]	=292	Base address for drive 2
DRIVE_LOGIC_ADDRESS[]	=272+n*20	Base address of drive n

MD 13060: (drive message type for drives connected to PROFIBUS)

DRIVE_TELEGRAM_TYPE[0]	=102	Message frame type
DRIVE_TELEGRAM_TYPE[1]	=102	Message frame type
DRIVE_TELEGRAM_TYPE[n]	=102	Message frame type

For different drives and different message frame types, the assigned address areas **can be of different sizes**.
Input and output areas have **optional, but the same** start addresses and must match the address of the PROFIBUS DP configuration.

© Siemens AG, 2003. All rights reserved

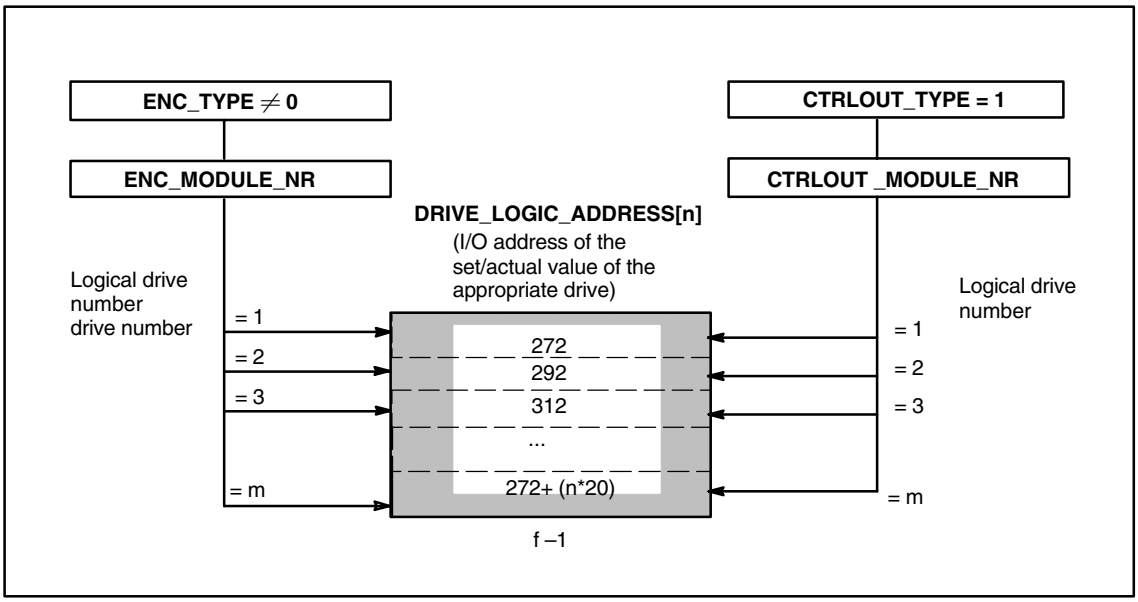


Fig. 2-4 Parameterization of address space for drives

Expansions in SW 6.3 and higher

For **non-Siemens drives**, the other bits in machine data

MD 13070: DRIVE_FUNCTION_MASK bits 4 to 8

are assigned a meaning. By configuring these new bits, you can adapt certain non-standardized PROFIBUS control and/or status bits for SIMODRIVE 611 universal included in the Profidrive profile.

Linear drives
To allow operation of linear drives, or drive types of non-Siemens slave devices (which do not, for example, support acyclic communication or signal a drive type), a new machine data

MD 13080: DRIVE_TYPE_DP has been added.

Rigidity control (DSC)
The new machine data make it possible to fine calibrate non-Siemens drives, including combinations with PROFIBUS DP and the DSC function:

MD 32644: STIFFNESS_DELAY_TIME DSC delay
MD 37602: PROFIBUS_OUTVAL_DELAY_TIME Setpoint delay time

Feedforward control variant

Upgrade on 840Di to SW 6.3

Note

If Feedforward control variant MD 32620: FFW_MODE = 3 has already been used on an 840Di, the start-up setting in MD 32810: EQUIV_SPEEDCTRL_TIME should be reconfigured when upgrading to SW 6.3 or later as the values Ti and To will automatically be taken into account. These values must be corrected in EQUIV_SPEEDCTRL_TIME.

■

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

Parameters

When transmitting data via the serial interfaces it is important to ensure that both communications partners have the same parameter settings: this applies in particular to the baud rate and number of stop bits.

Cable

The cable with the order no. 6FX2 0021AA011BF0 is used for the connection to a PC or PG.
If a printer with a serial input is to be connected, the cable must be equipped with a modem eliminator connector or a standard serial printer cable must be used.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

The three parameter sets for the serial interface X6 on the MMC 100 standard component are stored in three machine data groups which are described in more detail below.

Machine data	Parameter blocks for	Abbreviation
9300 – 9309	User	USER
9310 – 9319	Printer	PRINTER
9320 – 9329	PG/PC	PG_PC

With SW 5 and higher the three machine data groups are each expanded by data 9309, 9319 and 9329.

Note

From SW 6.1 and higher: (MMC 100 is the same as HMI Embedded)

4.1 XON character

9300 **9310** **9320** MD number	**V24_USER_XON** **V24_PRINTER_XON** **V24_PG_PC_XON** XON character		
Default setting: 11	Minimum input limit: 00 (suitable from 01)	Maximum input limit: FF (suitable up to 1F)	
Changes effective immediately	Protection level: 2	Unit: Hex	
Data type: BYTE	Applies from SW 1.1		
Significance:	XON character: This is the character with which transmission is initiated. It only has an effect with device type XON/OFF. If the special function “Start with XON” is activated, the program waits for an XON character from the connected device before starting. The device control character (DEVICE CONTROL 1 (XON) or DC 1 is 11H as standard. This is the default value. Input: As number in Parameters display under XON (hex).		

© Siemens AG, 2003. All rights reserved

4.2 XOFF character

9301 **9311** **9321** MD number	**V24_USER_XOFF** **V24_PRINTER_XOFF** **V24_PG_PC_XOFF** XOFF character		
Default setting: 13	Minimum input limit: 00 (suitable from 01)	Maximum input limit: FF (suitable up to 1F)	
Changes effective immediately	Protection level: 3/4	Unit: Hex	
Data type: BYTE	Applies from SW 1.1		
Significance:	XOFF character: This is the character with which transmission is stopped. It only has an effect with device type XON/OFF. The device control character 3 (DEVICE CONTROL 3 (XOFF) or DC 3 is 13H as standard. This is the default value. Input: As number in Parameters display under XOFF (hex).		

4.3 End of transmission character

9302 **9312** **9322** MD number	**V24_USER_EOF** **V24_PRINTER_EOF** **V24_PG_PC_EOF** Endoftransmission character		
Default setting: 1A	Minimum input limit: 00 (suitable from 01)	Maximum input limit: FF (suitable up to 1F)	
Changes effective immediately	Protection level: 3/4	Unit: Hex	
Data type: BYTE	Applies from SW 1.1		
Significance:	This is the character with which transmission is stopped. The default setting is 1A: DOS character for file end in text files. Input: As number in Parameters display under End of transmission. The character has an effect when “Stop with end of transmission character” is marked with a cross.		

© Siemens AG, 2003. All rights reserved

4.4 Special bits

9303 **9313** **9323** MD number	**V24_USER_CONTROLS** **V24_PRINTER_CONTROLS** **V24_PG_PC_CONTROLS** Special bits		
Default setting: 01001100 01001100 10000000	Minimum input limit: 00000000	Maximum input limit: 11111111	
Changes effective immediately	Protection level: 3/4	Unit: Empty field or checked bit field	
Data type: BYTE	Applies from SW 1.1		
Significance:	The special functions that can be activated in the Parameters display are stored in these special bits. If the bit is set: Special function is active. Stored in the machine data as a bit with following assignment: Bit 0 Start with XON Bit 1 Program start with LF Bit 2 Block end with CR LF Bit 3 Stop with end of transmission character Bit 4 Evaluate DSR signal Bit 5 Leader and trailer Bit 6 Tape format Bit 7 Time monitoring Input: Activation in Parameters display under Special functions An explanation of the special bits is given below.		

Bit 0 **Start with XON**

Active: Transmission is started when the character defined for XON appears in the data flow. This only applies if device type XON/XOFF has been set.

Inactive: Start independent of XON character.

Bit 1 **Program start with LF**

Not yet available.

Bit 2 **Block end with CR LF**

Active: CR characters (carriage return, hexadecimal OD) are inserted in tape format output.
CR characters are removed from input in tape format.

Inactive: Additional characters are not inserted.

Bit 3 **Stop with end of transmission character**

Active: End of transmission character is evaluated.

Inactive: End of transmission character is not evaluated (required for binary data transmission).

Bit 4 **Evaluate DSR signal**

Active: Transmission is interrupted if DSR signal (connection 6 on X6 or X7 connector (applies to MMC 101103 only) is missing.

Inactive: DSR signal has no effect.

© Siemens AG, 2003. All rights reserved

Bit 5 **Leader and trailer**

Active: Leader ignored during input
Output 120x0(hex) during output
(feed before and after data)
Inactive: Both leader and trailer are read in.
No leader 0(hex) on output.

Bit 6 **Tape format**

Active: Programs read in acc. to DIN 66025, e.g. SINUMERIK 3/8 programs:
SINUMERIK 3/8:
Start activated by % file name, %MPFxxx or %SPFxxx.
Inactive: Archives read in in SINUMERIK 840D/810D archive format.

Bit 7 **Time monitoring**

Active: If there are transmission problems, transmission is aborted after 10 seconds. Time monitoring is controlled by a timer which is reset on every transmitted character.
Inactive: Transmission is not aborted.

© Siemens AG, 2003. All rights reserved

4.5 Device type

9304 **9314** **9324** MD number	**V24_USER_RTS** **V24_PRINTER_RTS** **V24_PG_PC_RTS** Device type		
Default setting: 1 (XON/XOFF)	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 3/4	Unit: Bit	
Data type: BYTE	Applies from SW 1.1		
Significance:	Transmission control support exists for two different device types: XON/OFF and RTS/CTS. 1: XON/XOFF Control characters XON (DC1, DEVICE CONTROL 1) and XOFF (DC3) offer one type of transmission control. As soon as the buffer of the I/O device is full it transmits XOFF, as soon as it can receive data again it transmits XON. 0: RTS/CTS The signal RTS (Request To Send) controls the transmission of the data transmission equipment. Active: Data to be sent, passive: Do not exit send mode until all transferred data have been sent. Signal CTS (Clear To Send) flags the ready to send state of the data transmission equipment as the acknowledgment signal for RTS. Input: Selection in Parameters display under Device type.		

© Siemens AG, 2003. All rights reserved

4.6 Baud rate

9305 **9315** **9325** MD number	**V24_USER_BAUD** **V24_PRINTER_BAUD** **V24_PG_PC_BAUD** Baud rate		
Default setting: 5 9600 baud	Minimum input limit: 0	Maximum input limit: 6	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW		
Significance:	This is the modulation rate in baud, a unit of rate of data transmission. Input: By selection in Parameters display under Baud rate. 0: 300 baud 1: 600 baud 2: 1200 baud 3: 2400 baud 4: 4800 baud 5: 9600 baud 6: 19200 baud with SW 3.1 and higher		

4.7 Data bits

9306 **9316** **9326** MD number	**V24_USER_DATABITS** **V24_PRINTER_DATABITS** **V24_PG_PC_DATABITS** Data bits		
Default setting: 1 8 data bits	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW		
Significance:	Number of data bits for asynchronous transmission. Input: By selection in Parameters display under Data bits 0: 7 data bits 1: 8 data bits		

© Siemens AG, 2003. All rights reserved

4.8 Parity

9307 **9317** **9327** MD number	**V24_USER_PARITY** **V24_PRINTER_PARITY** **V24_PG_PC_PARITY** Parity		
Default setting: 0 No parity	Minimum input limit: 0	Maximum input limit: 2	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW		
Significance:	Parity bits are used to detect errors: The parity bits are added to the coded characters to make the number of places set to “1” an odd number (odd parity) or an even number (even parity). Input: By selection in Parameters display under Parity 0: No parity 1: Even parity 2: Odd parity		

4.9 Stop bits

9308 **9318** **9328** MD number	**V24_USER_STOPBIT** **V24_PRINTER_STOPBIT** **V24_PG_PC_STOPBIT** Stop bits		
Default setting: 0 1 stop bits	Minimum input limit: 0	Maximum input limit: 1	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Number of stop bits for asynchronous transmission. Input: By selection in Parameters display under Stop bits 0: 1 stop bits 1: 2 stop bits		

© Siemens AG, 2003. All rights reserved

4.10 RS232 interface

9309 **9319** **9329** MD number	**V24_USER_LINE** **V24_PRINTER_LINE** **V24_PG_PC_LINE** RS232 interface (COM1/COM2)		
Default setting: 1 COM1	Minimum input limit: 1	Maximum input limit: 2	
Changes effective immediately	Protection level: 3/4	Unit: –	
Data type: BYTE	Applies from SW 5		
Significance:	Selection of RS232 interface via which a file transfer must be initiated. Input: Selection in Parameters display under RS232 interface 1: COM1 2: COM2		

■

© Siemens AG, 2003. All rights reserved

Signal Descriptions

5

Hardware interfaces

For a description of the hardware design of the interfaces, please refer to
References: /BH/ Operator Components Manual

5.1 Serial interfaces

Assignment

The pin assignments of serial interfaces X 6 and X 7 (X 7 on MMC 101-103 only) comply with the standard which generally governs PC equipment: RS232 defines the meaning of the individual leads which are listed below together with their ID codes.

Connection	Designation	Meaning	Type of connection
1	DCD	Carrier detect	Input
2	RxD	Receive data	Input
3	TxD	Transmit data	Output
4	DTR	Data terminal ready	Output
5	GND	Signal ground	Signal ground
6	DSR	Data set ready	Input
7	RTS	Request to send	Output
8	CTS	Clear to send	Input
9	RI	Ring indicator	Input

Level

The level of the serial interface is governed by the V.28 standard. The standard ICs 75188 and 75189 which operate on ±12V are used as drives and receivers.

Control

The control circuit on the MMC 100 is compatible with interface controller type 16450.
The control switch on the MMC 101–103 is compatible with interface controllers 16450 and 16550.

© Siemens AG, 2003. All rights reserved

5.2 Parallel interface

Assignment

The pin assignments of the parallel interface X8 (MMC 101103 only) comply with the standard which generally governs PC equipment: 25-pin D subminiature female connector. The connector assignment is given in the table below.

Connection	Designation	Meaning	Type of connection
1	STROBE	Carrier detect	Output
2	DATA 0	Printer data bit 0	Bidirectional
3	DATA 1	Printer data bit 1	Bidirectional
4	DATA 2	Printer data bit 2	Bidirectional
5	DATA 3	Printer data bit 3	Bidirectional
6	DATA 4	Printer data bit 4	Bidirectional
7	DATA 5	Printer data bit 5	Bidirectional
8	DATA 6	Printer data bit 6	Bidirectional
9	DATA 7	Printer data bit 6	Bidirectional
10	ACKNLG	Acknowledge from printer	Input
11	BUSY	Busy from printer	Input
12	PE	Paper error	Input
13	SLCT	Select from printer	Input
14	ADF	Autofeed printer	Output
15	Error	Error from printer	Input
16	INIT	Initialize printer	Output
17	SLCTIN	Select to printer	Output
18	GND	Signal ground	Signal ground
19	GND	Signal ground	Signal ground
20	GND	Signal ground	Signal ground
21	GND	Signal ground	Signal ground
22	GND	Signal ground	Signal ground
23	GND	Signal ground	Signal ground
24	GND	Signal ground	Signal ground
25	GND	Signal ground	Signal ground

■

© Siemens AG, 2003. All rights reserved

6 Example

Both communications partners must have the same parameter settings before transmission can take place.

The parameters on the left in the examples are entered, for example, in selection fields on the operator panel under SERVICES/DATA/V.24_SET on the MMC 100. The parameters on the right are activated by entering an X in the checkboxes. An X means that the function is activated.

6.1 Parameters for serial printer

A printer with a serial interface is connected with a suitable cable (cable check to CTS).

Device type	RTSCTS		Start with XON
Baud rate	9600		Program start with LF
Stop bits	1	X	End of block with CR LF
Parity	none		Stop with end of transmission character
Data bits	8		Evaluate DSR signal
XON			Leader and trailer
XOFF		X	Tape format
End of trans.	0C (FormFeed)		Time monitoring

6.2 Parameters for archiving with PG/PC

Device type	RTSCTS		Start with XON
Baud rate	9600		Program start with LF
Stop bits	1		End of block with CR LF
Parity	none		Stop with end of transmission character
Data bits	8	X	Evaluate DSR signal
XON	00		Leader and trailer
XOFF	00		Tape format
End of trans.	00	X	Time monitoring

With this setting, files can be archived and read in in SINUMERIK 840D/810D format.
For transparent transmission (MSD, FDD files), “Stop with end of transmission character” must not be selected (PG does not stop automatically when saving).
Other settings are possible with ASCII data. They must correspond to those on the programmer. Cable 6FX 2002–1AA01– is used for this purpose.

© Siemens AG, 2003. All rights reserved

6.3 Parameters for DIN programs

Device type	RTSCTS		Start with XON
Baud rate	9600		Program start with LF
Stop bits	1	X	End of block with CR LF
Parity	none	X	Stop with end of transmission character
Data bits	8	X	Evaluate DSR signal
XON	11		Leader and trailer
XOFF	13	X	Tape format
End of trans.	1a		Time monitoring

With this setting, programs can be read in according to DIN or from System 3/8 (start with %).

6.4 Tape input/output

Leader and trailer must be marked with an X with tape reader and puncher.
If the tape reader is controlled via CTS, “Start with XON” must also be marked.
The tape reader can be stopped when the tape is inserted by pressing INPUT START/STOP.

Device type	RTSCTS		Start with XON
Baud rate	9600		Program start with LF
Stop bits	2	X	End of block with CR LF
Parity	none	X	Stop with end of transmission character
Data bits	8		Evaluate DSR signal
XON	00	X	Leader and trailer
XOFF	00	X	Tape format
End of trans.	00	X	Time monitoring

6.5 Import of machine data

The machine data file INITIAL.INI sets the machine default setting.
Field “Path from workpiece/archive” must be marked with an X on the RS232 interface before input is started.
This applies to data in archive format as well as in tape format.
The data are not activated until an NC Reset has been executed.

© Siemens AG, 2003. All rights reserved

6.6 Import of binary data (FDD, MSD)

Device type	RTSCTS	Start with XON
Baud rate	9600	Program start with LF
Stop bits	1	End of block with CR LF
Parity	none	Stop with end of transmission
Data bits	8	Evaluate DSR signal
XON		Leader and trailer
XOFF		Tape format
End of trans.	00	Time monitoring

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

7 Data Fields, Lists

7.1 Machine data

Number		Identifier	Name	Doc. Refer to
Operator panel data ($MM_ ...)		**User parameter (USER**		
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBEDDED (SW 6.1 and higher)	
	9300	V24_USER_XON	XON character	IM2
	9301	V24_USER_XOFF	XOFF character	IM2
	9302	V24_USER_EOF	Transmission character	IM2
	9303	V24_USER_CONTROLS	Special bits	IM2
	9304	V24_USER_RTS	Device type (linecontrolled)	IM2
	9305	V24_USER_BAUD	Baud rate (300, 600, 1200, 2400, 4800, 9600, 19200)	IM2
	9306	V24_USER_DATABITS	Data bits	IM2
	9307	V24_USER_PARITY	Parity bits	IM2
	9308	V24_USER_STOPBIT	Stop bits	IM2
	9309	V24_USER_LINE	RS232 interface (COM1/COM2)	IM2

Number		Identifier	Name	Doc. Refer to
Operator panel data ($MM_ ...)		**Printer parameters (PRINTER)**		
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBEDDED (SW 6.1 and higher)	
	9310	PRINT_USER_XON	Printer: XON character	IM2
	9311	PRINT_USER_XOFF	Printer: XOFF character	IM2
	9312	PRINT_USER_EOF	Printer: Transmission character	IM2
	9313	PRINT_USER_CONTROLS	Printer: Special bits	IM2
	9314	PRINT_USER_RTS	Printer: Device type (linecontrolled)	IM2
	9315	PRINT_USER_BAUD	Printer: Baud rate (300, 600, 1200, 2400, 4800, 9600, 19200)	IM2
	9316	PRINT_USER_DATABITS	Printer: Data bits	IM2
	9317	PRINT_USER_PARITY	Printer: Parity bits	IM2
	9318	PRINT_USER_STOPBIT	Printer: Stop bits	IM2
	9319	PRINT_USER_LINE	RS232 interface (COM1/COM2)	IM2

© Siemens AG, 2003. All rights reserved

Number		Identifier	Name	Doc. Refer to
Operator panel data ($MM_ ...)		**PG/PC parameters (PG_PC)**		
ADV	**EMB**	ADV ⇒ ADVANCED,	EMB ⇒ EMBEDDED (SW 6.1 and higher)	
	9320	V24_PG_PC_XON	PG: XON character	IM2
	9321	V24_PG_PC_XOFF	PG: XOFF character	IM2
	9322	V24_PG_PC_EOF	PG: Transmission character	IM2
	9323	V24_PG_PC_CONTROLS	PG: Special bits	IM2
	9324	V24_PG_PC_RTS	PG: Device type (linecontrolled)	IM2
	9325	V24_PG_PC_BAUD	PG: Baud rate (300, 600, 1200, 2400, 4800, 9600, 19200)	IM2
	9326	V24_PG_PC_DATABITS	PG: Data bits	IM2
	9327	V24_PG_PC_PARITY	PG: Parity bits	IM2
	9328	V24_PG_PC_STOPBIT	PG: Stop bits	IM2
	9329	V24_PG_PC_LINE	RS232 interface (COM1/COM2)	IM2

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

EMERGENCY STOP (N2)

1 Brief Description 1/N2/1-3

2 Detailed Description 1/N2/2-5

2.1 EMERGENCY STOP actuators 1/N2/2-6

2.2 EMERGENCY STOP sequence 1/N2/2-7

2.3 EMERGENCY STOP acknowledgment 1/N2/2-8

3 Supplementary Conditions 1/N2/5-11

4 Data Descriptions (MD, SD) 1/N2/5-11

5 Signal Descriptions 1/N2/5-11

5.1 General signals 1/N2/5-11

6 Example 1/N2/7-13

7 Data Fields, Lists 1/N2/7-13

7.1 Interface signals 1/N2/7-13

7.2 Machine data 1/N2/7-13

7.3 Alarms 1/N2/7-14

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

1 Brief Description

Standard EN 2922

In accordance with the basic safety requirements of the EC Machinery Directive, stipulated in Subsection 6.1.1 of EN 2922 with respect to EMERGENCY STOP, machines must be equipped with an EMERGENCY STOP facility.

Exceptions

An EMERGENCY STOP facility is not required on:

- Machines where an EMERGENCY STOP facility would not reduce the risk, either because the shutdown time would not be reduced or because the measures required to be taken would not be suitable for controlling the risk.
- Handheld or portable machines.

EMERGENCY STOP in the control system

The control system supports the machine manufacturer in implementing an EMERGENCY STOP function on the basis of the following features:

- The EMERGENCY STOP button is installed within easy reach on the machine control panel and is highlighted on a yellow background.
- A red EMERGENCY STOP button with positive opening action and automatic mechanical latching/locking.
- Activation of EMERGENCY STOP sequence in the NC via a PLC input.
- The EMERGENCY STOP procedure on the NC reduces the speed of all axes and spindles as quickly as possible.
- All machine functions controlled by the PLC can be brought into a safe state which can be configured by the machine manufacturer.
- The EMERGENCY STOP state is not canceled when the EMERGENCY STOP button is released.
  Resetting the control device does not restart the machine.
- After the EMERGENCY STOP state has been canceled, it is not necessary to reference axes or synchronize spindles (positions are automatically corrected).

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

!

Important

It is the duty of the machine manufacturer to observe national and international standards (see the notes on standards in the following paragraph). The SINUMERIK FM-NC and 840D/810D support the machine manufacturer in the implementation of the EMERGENCY STOP function in accordance with the specifications in this Description of Functions. The responsibility for the EMERGENCY STOP function (its triggering, procedure and acknowledgment) rests exclusively with the machine manufacturer.

Note

Particular reference should be made to the following standards for the EMERGENCY STOP function:
- EN 292 Part 1
- EN 292 Part 2
- EN 418
- EN 60204 Part 1: 1992 Section 10.7

VDE 0113 Part 1 is only valid for a transitional period and will be superseded by EN 60204.

EMERGENCY STOP Function

EN 418: EMERGENCY STOP is a function that

- Intended to prevent or diminish developing or existing risks to operating personnel, and damage to the machine or machined materials.
- Is triggered by a single human action in cases where the normal stop function is not suitable.

In the terms of EN 418, risks may arise from:

- Functional irregularities (machine malfunctions, unacceptable properties of the material to be machined, human error, etc.).
- Normal operation.

© Siemens AG, 2003. All rights reserved

2.1 EMERGENCY STOP actuators

EN 418 standards

EN 418 stipulates that EMERGENCY STOP actuating components must be designed such that they are easy to operate by operating personnel or others who may need to make use of them. The following list includes some possible types of actuator:

- Mushroom-head pushbutton switches
- Wires/cables, cords, rods
- Puller grips
- In special cases: foot-operated switches without protective covers.

All EMERGENCY STOP switches must be mechanically selflatching and installed within easy reach.

EMERGENCY STOP pushbutton

The Siemens machine control panel (MCP) for FM-NC and 840D/810D is equipped with a mushroom-head pushbutton switch with positive opening operation referred to below as the EMERGENCY STOP pushbutton.

Conditions for connection

For details about connecting the EMERGENCY STOP pushbutton, please see hardware configuring guide.
References: /BH/ Operator Components Manual

EMERGENCY STOP at NC

Actuation of the EMERGENCY STOP pushbuttons or a signal derived directly from the button must be taken to the control system (PLC) as a PLC input. In the PLC user program, this PLC input must be taken to IS “EMERGENCY STOP” (DB10 DBX56.1) in the NC.
The reset of the EMERGENCY STOP button, or a signal generated directly from its reset, must be communicated to the programmable controller (PLC) in the form of a PLC input signal. In the PLC user program, this PLC input must be taken to IS “Acknowledge EMERGENCY STOP” (DB10 DBX56.2) in the NC.

© Siemens AG, 2003. All rights reserved

2.2 EMERGENCY STOP sequence

EN 418 standard

Following actuation of the operating part of an EMERGENCY STOP switch, the EMERGENCY STOP must automatically act in the best possible way to avoid or reduce the risk.
"Best possible way" means that the smallest delay rate and the correct stop category (defined in EN 60204) can be specified according to the estimated risk.

Sequence in the NC

The predefined (in EN 418) sequence of internal functions implemented to reach the EMERGENCY STOP state is as follows in the control system:

1. Parts program execution is interrupted. All axes and spindles are braked along a braking ramp defined in MD 36610: AX_EMERGENCY_STOP_TIME. A steep braking ramp (short AX_EMERGENCY_TIME setting) results in rapid deceleration at maximum braking current (setpoint speed input = 0).
2. IS "Mode group ready" (DB11, ... DBX6.3) DBX6.3) is reset.
3. IS "EMERGENCY STOP active" (DB10 DBX106.1) is set.
4. Alarm 3000 is set.
5. On expiry of a delay that is set for specific axes/spindles in MD 36620: SERVO_DISABLE_DELAY_TIME (cutout delay servo enable), the controller enables (FM–NC: the controller enable relay) are disabled. It must be noted in this respect that the setting in SERVO_DISABLE_DELAY_TIME is at least as high as the setting for AX_EMERGENCY_STOP_TIME.
6. All axes and spindles are switched internally to follow-up mode (no position control active).

Sequence at the machine

The sequence of EMERGENCY STOP functions on the machine is determined solely by the machine manufacturer. Attention should be paid to the following points in connection to the procedure on the NC:

- The sequence of operations in the NC is started with IS "EMERGENCY STOP" (DB10 DBX 56.1). When the axes and spindles have come to a halt, the power supply must be interrupted, in compliance with EN 418.

© Siemens AG, 2003. All rights reserved

!

Important

The responsibility for interrupting the power supply rests with the machine manufacturer.

- The PLC I/Os (digital and analog I/Os) are not affected by the procedure on the NC. If individual outputs are required to attain a particular state or voltage level in the event of an EMERGENCY STOP, the machine manufacturer must include functions for this purpose in the PLC user program.
- The NCK I/Os (high-speed digital I/Os) are not affected by the procedure on the NC. If individual outputs must assume a specific state in the case of EMERGENCY STOP, the machine manufacturer must transfer this state to the NC via IS “DB10 DBB4 to 7” in the PLC user program.

!

Important

If the internal functions in the NC should not be executed in the predetermined sequence in the event of an EMERGENCY STOP, then IS EMERGENCY STOP (DB10 DBX56.1) must not be set at any time up to the point that an EMERGENCY STOP defined by the machine manufacture in the PLC user program is reached. As long as the EMERGENCY STOP interface signal has not been set and no other alarm exists, all interface signals are effective in the NC. Any EMERGENCY STOP state defined by the manufacturer (including axis/spindle-specific and channel-specific) can therefore be assumed.

2.3 EMERGENCY STOP acknowledgment

EN 418 standard

The EMERGENCY STOP switch may only be reset by manual actuation of the operating part. The reset of the EMERGENCY STOP operating part must not trigger a restart command.
It should not be possible to start the machine until all actuated EMERGENCY STOP operating parts have been individually and deliberately reset.

EMERGENCY STOP acknolwegement

The EMERGENCY STOP state is reset only if IS “Acknowledge EMERGENCY STOP” (DB10 DBX56.2) followed by IS “Mode group reset” (DB11, ... DBX0.7) are set. It must be noted in this respect that IS “Acknowledge EMERGENCY STOP” and IS “Reset” must be set (together) for a long enough period for IS “EMERGENCY STOP active” (DB10 DBX106.1) to be reset (see Fig. 2-1).

© Siemens AG, 2003. All rights reserved

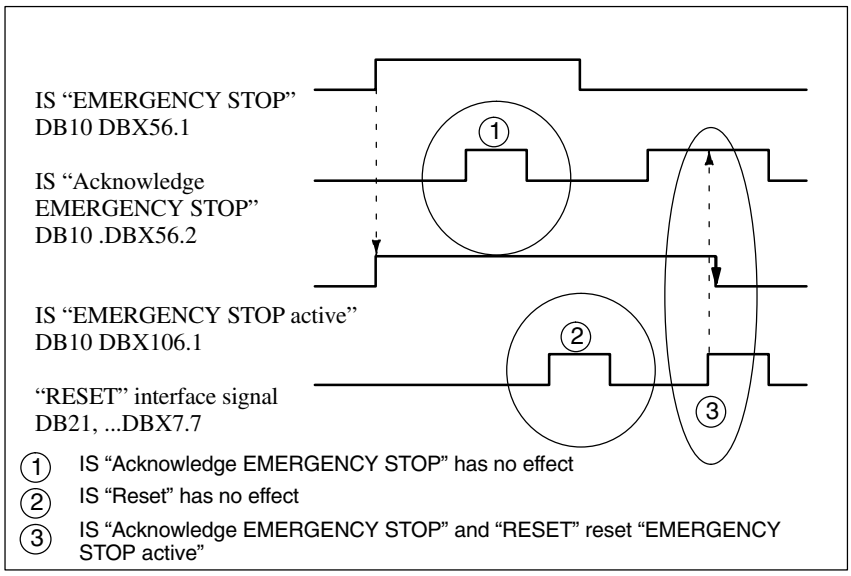


Fig. 2-1 Reset EMERGENCY STOP

Resetting the EMERGENCY STOP state has the following effects:

- The servo enable (FM-NC: controller enable relay) is enabled.
- Follow-up mode is canceled for all axes and position control mode resumed.
- IS “Position control active” is set.
- IS “Mode group ready” is set.
- IS “EMERGENCY STOP active” interface signal is reset
- Alarm 3000 is canceled.
- Parts program processing is interrupted for all channels.

PLC + NCK I/Os

The PLC user program must switch the PLC and NCK I/Os to the correct state for operation of the machine.

Reset

IS “Reset” (DB21, ... DBX7.7) alone cannot reset the EMERGENCY STOP status (see figure below).

Power OFF/ON

Power OFF/ON resets the EMERGENCY STOP state unless IS “EMERGENCY STOP” (DB10 DBX56.1) is still set.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

The following standards must be observed in all cases:

- EN 292
- EN 418
- EN 60204

4 Data Descriptions (MD, SD)

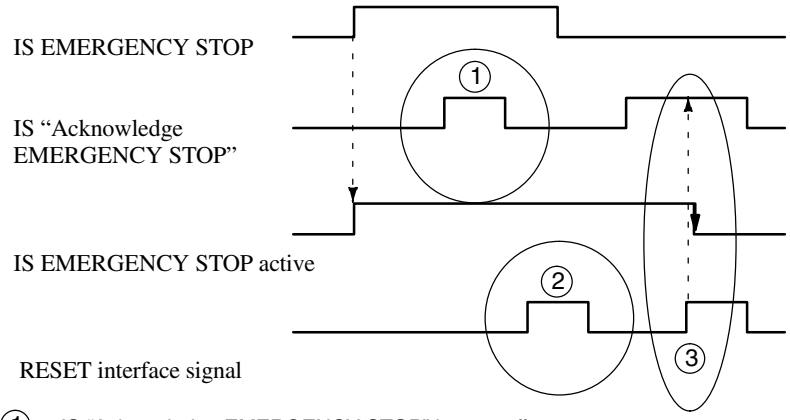
None

5 Signal Descriptions

5.1 General signals

DB 10 **DBX56.1** Data block	**EMERGENCY STOP** Signal(s) to NC (PLC ——> NC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The NC is switched to the EMERGENCY STOP state and the EMERGENCY STOP procedure is started on the NC (see Section 2.2).		
Signal state 0 or signal transition 1 ——> 0	• The NC is not in the EMERGENCY STOP state. • The EMERGENCY STOP state is (still) active, but can be reset with IS “Acknowledge EMERGENCY STOP” and “mode group reset” interface signals.		
Related to	IS “Acknowledge EMERGENCY STOP” (DB10 DBX56.2) IS “EMERGENCY STOP active” (DB10 DBX106.1)		

© Siemens AG, 2003. All rights reserved

DB 10 **DBX56.2** Data block	**Acknowledge EMERGENCY STOP** Signal(s) to NC (PLC ——> NC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The EMERGENCY STOP state is reset only if IS “Acknowledge EMERGENCY STOP” (DB10 DBX56.2) followed by IS “Mode group reset” (DB11, ... DBX0.7) are set. It must be noted in this respect that IS “Acknowledge EMERGENCY STOP” and IS “Reset” must be set (together) for a long enough period for IS “EMERGENCY STOP active” (DB10 DBX106.1) to be reset. Resetting the EMERGENCY STOP state has the following effects: ● The controller enable (FM-NC: the controller enable relay) is activated ● Follow-up mode is canceled for all axes and position control mode is activated ● IS “Position control active is set” ● IS “Mode group ready is set” ● IS “EMERGENCY STOP active is reset” ● Alarm 3000 is canceled ● Parts program processing is interrupted for all channels.		
	 ① IS “Acknowledge EMERGENCY STOP” has no effect ② IS “Reset” has no effect ③ IS “Acknowledge EMERGENCY STOP” and “RESET” reset “EMERGENCY STOP active”		
Special cases, errors, ...	IS “Reset” (DB21, ... DBX7.7) cannot reset the EMERGENCY STOP status.		
Related to	IS “EMERGENCY STOP” (DB10 DBX56.1) IS “EMERGENCY STOP active” (DB10 DBX106.1) IS “Mode group reset” (DB10 DBX0.7)		

DB 10 **DBX106.1** Data block	**EMERGENCY STOP active** Signal(s) to NC (PLC ——> NC)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The NC is in the EMERGENCY STOP state.	
Related to	IS “EMERGENCY STOP” (DB10 DBX56.1) IS “Acknowledge EMERGENCY STOP” (DB10 DBX56.2)	

■

© Siemens AG, 2003. All rights reserved

6 Example

None

7 Data Fields, Lists

7.1 Interface signals

DB number	Bit, byte	Name	Reference
General			
10	56.1	EMERGENCY STOP	
10	56.2	Acknowledge EMERGENCY STOP	
10	106.1	EMERGENCY STOP active	
Mode group-specific			
11, ...	0.7	Mode group reset	K1

7.2 Machine data

Number	Identifier	Name	Reference
Drive machine data ($MD_ ...)			
1404	PULSE_SUPPRESSION_DELAY	Time for pulse suppression	DB1
Axis-specific ($MA_ ...)			
36610	AX_EMERGENCY_STOP_TIME	Length of the braking ramp for error states	A3
36620	SERVO_DISABLE_DELAY_TIME	Cutout delay servo enable	A2

© Siemens AG, 2003. All rights reserved

7.3 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D/ Description of Functions Basic Machine (Part 1)

Transverse Axes (P1)

1 Brief Description 1/P1/1-3

2 Detailed Description 1/P1/2-5

2.1 Transverse axis definition 1/P1/2-5

2.2 Diameter programming 1/P1/2-6

2.3 Conversion of diameter values to internal radius values 1/P1/2-7

2.4 Conversion of internal radius values to diameter values 1/P1/2-7

3 Supplementary Conditions 1/P1/4-9

4 Data Descriptions (MD, SD) 1/P1/4-9

5 Signal Descriptions 1/P1/7-11

6 Example 1/P1/7-11

7 Data Fields, Lists 1/P1/7-11

7.1 Machine data 1/P1/7-11

7.2 Alarms 1/P1/7-11

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

1 Brief Description

Geometry axes can be defined as transverse axes via a machine data.
Measurements for transverse axes are programmed either in radius or diameter format.
The diameter programmed is activated/deactivated by means of G commands DIAMON, DIAMOF.

When diameter programming is selected:

- The setpoint/actual value display in the workpiece coordinate system (WCS) shows diameterrelated data.
- All offsets are entered, programmed and displayed in radius format.
- Programmed end positions are converted to internal radius values.
- Absolute interpolation parameters (e.g. I, J, K) for G2/G3 programming are converted to internal radius values.
- Measurements produced in the WCS with the aid of touch-trigger probes are stored in diameter format.
- Setpoints and actual values can be read in diameter format in the WCS with the aid of system variables.

When radius programming is selected, the above data are always entered, programmed, stored internally, read and displayed in radius format.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description

2

2.1 Transverse axis definition

General

"Longitudinal axis" is the term generally used in the "turning" technology to describe the Z axis; the term "transverse axis" is used to described the X axis. Measurements for a transverse axis are generally specified in diameter format (double the path dimension of other axes). G commands can be used to switch between diameter and radius programming.

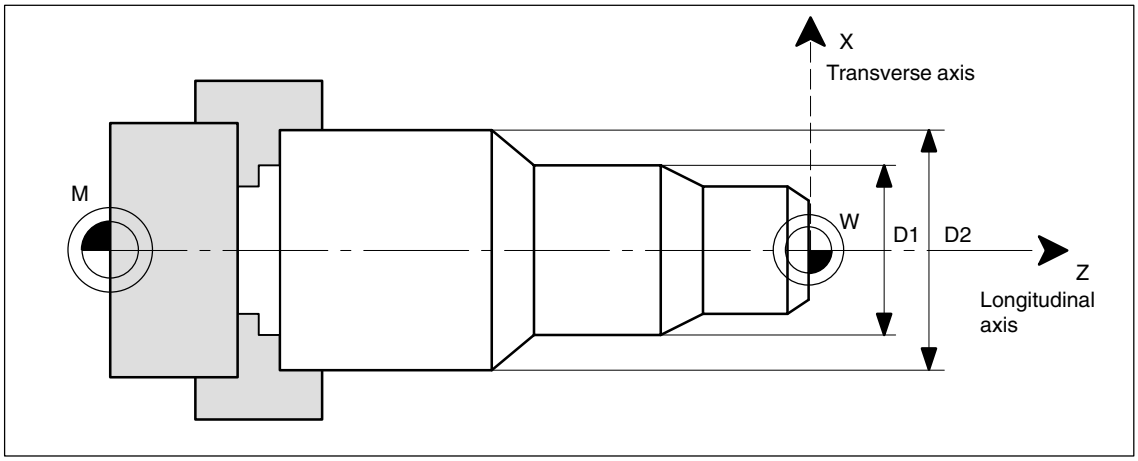


Fig. 2-1 Transverse axis with diameter specification

Geometry axis ⇒ Face axis

Channel-specific MD 20100: DIAMETER_AX_DEF (geometry axis with transverse axis function) can be used to define geometry axes as face axes. One transverse axis can be defined for each channel.

Transverse axis with G96/G961 and G97/G971

The geometry axis whose position determines the master spindle speed in connection with G96/G961 (constant cutting rate) and G97/G971 (constant speed) must be defined as a transverse axis or else alarm 10870 "No transverse axis defined" will be generated.

© Siemens AG, 2003. All rights reserved

2.2 Diameter programming

Activation and deactivation

Transverse axes can be programmed on the basis of diameter or radius.
The program commands “DIAMON” and “DIAMOF” can be used to activate or deactivate diameter programming for the transverse axis of a channel.
DIAMON and DIAMOF belong to G group 29 and are modally active. The initial setting is defined in MD 20150: GCODE_RESET_VALUES [28] (initial setting of G groups). If DIAMON is used to activate diameter programming for an axis which has not been defined as a transverse axis, then alarm 16510 “No transverse axis defined” is output.

JOG mode

If DIAMON is active, then the increments entered for machine functions INC (incremental dimension) and handwheel traverse in JOG are interpreted and traversed as diameter values for the associated transverse axis.

Setpoint/actual value display

If the “DIAMON” function is activated for a transverse axis, then the position, distance-to-go and REPOS offset are displayed as diameterrelated data when the workpiece coordinate system (WCS).
This information is always displayed as radius-related data in the machine coordinate system (MCS).
The format used in the service displays for axis, FDD and MSD is always radius.

Offsets

All offsets (e.g. tool offsets, programmable and settable frames, work offset external, DRF and preset offsets) are always entered, programmed and displayed as radius values (even if they are active in the transverse axis and command DIAMON is active).

Working area limitations, software limit switch, feed values

These data are always entered, programmed and displayed as radius values.

Positioning axes

If a geometry axis that has been defined as a transverse axis by means of machine data is operated as a positioning axis in the parts program in response to command POS [U] or POSA [U], then any diameter programming active for the axis will remain operative until it is deselected again.
When POSA [U] is active, the program advances to the next NC block even if the position has not yet been reached.

© Siemens AG, 2003. All rights reserved

2.3 Conversion of diameter values to internal radius values

Diameter values ⇒ Radius values

The following quantities for a transverse axis are converted to internal radius values (i.e. the programmed values are halved) when diameter programming is active:

- Programmed end positions, irrespective of whether they are absolute or incremental dimensions with G90/G91
- Absolute interpolation parameters (e.g. I, J, K) for G2/G3 programming
  Absolute interpolation parameters reference the coordinate origin of the WCS.
  References: /PG/, “Programming Guide: Fundamentals”
  (Interpolation parameters programmed relatively are not converted).

2.4 Conversion of internal radius values to diameter values

Radius values ⇒ Diameter values

In connection with the functions below, the internal radius values of a transverse axis are converted into diameter values (e.g. radius values are doubled) and stored when diameter programming is active:

- Measurements taken in the WCS with the functions “MEAS”, “MEASW”.
- Setpoints and actual values read in the WCS with the system variables $P_EP[x] or $AA_IW[x].

(For measurement or reading in the MCS, the calculated values are stored as radius values).

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

Availability of the function

The “Transverse axes” function is available as follows:

	SINUMERIK FMNC	SINUMERIK 810D	SINUMERIK 840D
Available as of	SW 1	SW 1	SW 2

■

4 Data Descriptions (MD, SD)

20100 MD number	**DIAMETER_AX_DEF** Geometry axis with transverse axis function		
Default setting: –	Minimum input limit: –	Maximum input limit: 16 characters	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: STRING	Applies from SW 1.1		
Significance:	The MD is used to define a geometry axis as a **transverse axis**. One transverse axis can be defined for each channel. The axis identifier of an active geometry axis that has been defined via channel-specific MD 20050: AXCONF_GEOAX_ASSIGN_TAB[n] or MD 24120: TRAFO_AX_GEOAX_ASSIGN_TAB_1[n] (SW 4 and higher) and MD 20060: AXCONF_GEOAX_NAME_TAB[n] must be specified. When space characters are entered or when the axis identifier is specified for an axis which is not defined as a geometry axis, activation of the diameter programming for Alarm 16510 “No face axis present” and alarm 10870 “No transverse axis defined” when G96/G961 and G97/G971 are programmed.		
Related to	MD 20050: AXCONF_GEOAX_ASSIGN_TAB[n] (assignment of geometry axis to channel axis) MD 20060: AXCONF_GEOAX_NAME_TAB[n] (geometry axis in channel) MD 24120: TRAFO_AX_GEOAX_ASSIGN_TAB_1[n] (assignment between geometry axis and channel axis for transformation 1)		

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

5 Signal Descriptions

None

6 Example

None

7 Data Fields, Lists

7.1 Machine data

Number	Identifier	Name	Refer-ence
Channel-specific ($MC_ ...)			
20050	AXCONF_GEOAX_ASSIGN_TAB[n]	Assignment of geometry axis to channel axis	K2
20060	AXCONF_GEOAX_NAME_TAB[n]	Geometry axis name in channel	K2
20100	DIAMETER_AX_DEF	Geometry axis with transverse axis function	
20150	GCODE_RESET_VALUES[n]	Reset G groups	K1

7.2 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

PLC Basic Program (P3)

1	**Brief Description**		**1/P3/1-5**
2	**Detailed Description**		**1/P3/2-7**
	2.1	Key data of PLC CPUs for FM-NC, 840D and 810D	1/P3/2-7
	2.2	Reserving resources (timer, FC, FB, DB, peripherals)	1/P3/2-14
	2.3	Starting up hardware configuration of PLC CPUs	1/P3/2-15
	2.4	Starting up the PLC program	1/P3/2-20
	2.4.1	Installing the basic program for FM NC, 810D, 840D	1/P3/2-20
	2.4.2	Application of basic program	1/P3/2-21
	2.4.3	Version codes	1/P3/2-23
	2.4.4	Machine program	1/P3/2-24
	2.4.5	Data backup	1/P3/2-24
	2.4.6	PLC series start-up, PLC archives:	1/P3/2-25
	2.4.7	Software upgrades	1/P3/2-27
	2.4.8	I/O modules (FM, CP modules)	1/P3/2-28
	2.4.9	Troubleshooting	1/P3/2-28
	2.5	Linking PLC CPUs to FM-NC, 810D, 840D	1/P3/2-30
	2.5.1	Properties of PLC CPUs	1/P3/2-30
	2.5.2	Interface on 810D and 840D with integrated PLC	1/P3/2-30
	2.5.3	PLC interface on FM-NC	1/P3/2-32
	2.5.4	Diagnostics buffer on PLC	1/P3/2-33
	2.6	Interface structure	1/P3/2-35
	2.6.1	PLC/NCK interface	1/P3/2-35
	2.6.2	PLC/MMC interface	1/P3/2-41
	2.6.3	PLC/MCP/HHU interface	1/P3/2-44
	2.7	Structure and functions of the basic program	1/P3/2-49
	2.7.1	Start-up and synchronization of NCK PLC	1/P3/2-50
	2.7.2	Cyclic operation	1/P3/2-50
	2.7.3	Time alarm processing (OB 35)	1/P3/2-52
	2.7.4	Process alarm processing (OB40)	1/P3/2-52
	2.7.5	Response to NC failure	1/P3/2-53
	2.7.6	Functions of basic program called from user-program	1/P3/2-54
	2.7.7	Symbolic programming of user program with interface DB	1/P3/2-56
	2.7.8	M decoding acc. to list	1/P3/2-58
	2.7.9	PLC machine data	1/P3/2-63

© Siemens AG, 2003. All rights reserved

2.7.10 Configurability of machine control panel, handheld unit 1/P3/2-66

2.8 SPL for Safety Integrated 1/P3/2-73

3 Supplementary Conditions and NC VAR Selector 1/P3/3-75

3.1 Supplementary conditions 1/P3/3-75
3.1.1 Programming and parameterizing tools 1/P3/3-75
3.1.2 SIMATIC documentation required 1/P3/3-77
3.1.3 Relevant SINUMERIK documents 1/P3/3-77

3.2 NC VAR selector 1/P3/3-78
3.2.1 Overview 1/P3/3-78
3.2.2 Description of Functions 1/P3/3-80
3.2.3 Startup, installation 1/P3/3-88

4 Block Descriptions 1/P3/4-89

4.1 FB 1: RUN_UP Basic program, start-up section 1/P3/4-89

4.2 FB 2: Read GET NC variables 1/P3/4-102

4.3 FB 3: PUT Write NC variables 1/P3/4-109

4.4 FB 4: PI_SERV General PI services 1/P3/4-115

4.5 FB 5: Read GUD variable GETGUD 1/P3/4-138

4.6 FB 7: PI_SERV2 General PI services 1/P3/4-143

4.7 FB 9: M : N operating unit changeover 1/P3/4-147

4.8 FB 10: Safety relay (SI relay) 1/P3/4-152

4.9 FB 11: Brake test 1/P3/4-155

4.10 FB 29: Signal recorder and data trigger diagnostics 1/P3/4-159

4.11 FC 2: GP_HP Basic program, cyclical section 1/P3/4-163

4.12 FC 3: GP_PRAL Basic program, interrupt-controlled section 1/P3/4-164

4.13 FC 5: GP_DIAG basic program, diagnostic alarm (only FM-NC) . . 1/P3/4-167

4.14 FC 7: TM_REV Transfer block for tool change with revolver 1/P3/4-168

4.15 FC 8: TM_TRANS Transfer block for tool management 1/P3/4-171

4.16 FC 9: ASUB Start-up of asynchronous subprograms 1/P3/4-179

4.17 FC 10: AL_MSG Error and operating messages 1/P3/4-182

4.18 FC 12: AUXFU Call interface for user with auxiliary functions 1/P3/4-184

4.19 FC 13: BHGDisp Display control for handheld unit 1/P3/4-185

4.20 FC 15: POS_AX Positioning of linear and rotary axes 1/P3/4-188

4.21 FC 16: PART_AX Positioning of indexing axes 1/P3/4-193

4.22 FC 17: Y Delta Star/delta switchover 1/P3/4-197

4.23 FC 18: SpinCtrl Spindle control 1/P3/4-200

4.24 FC 19: MCP_IFM Transmission of MCP signals to interface 1/P3/4-209

4.25 FC 21: Transfer PLC NCK data exchange 1/P3/4-217

4.26 FC 22: TM_DIR Direction selection for tool management 1/P3/4-224

© Siemens AG, 2003. All rights reserved

4.27 FC 24: MCP_IFM2 Transmission of MCP signals to interface PLC/NCK 1/P3/4-226
4.28 FC 25: MCP_IFT Transfer of MCP/OP signals to interface 1/P3/4-230
4.29 FC 26: HPU_MCP Transfer of HPU/HT6 signals to the interface . 1/P3/4-234
4.30 FC 19, FC 24, FC 25, FC 26 Source code description 1/P3/4-240

5 Signal/Data Descriptions 1/P3/5-243

5.1 Interface signals NCK/PLC, MMC/PLC, MCP/PLC 1/P3/5-243
5.2 Decoded M signals 1/P3/5-243
5.3 G functions 1/P3/5-244
5.4 Message signals in DB2 . . . . . 1/P3/5-247

6 Data Fields, Lists 1/P3/6-251

6.1 FB/FC assignment 1/P3/6-251
6.2 DB assignment 1/P3/6-251
6.3 Assigned timers 1/P3/6-253
6.4 Memory requirements of basic PLC program for FM-NC, 810D, 840D 1/P3/6-254

7 Useful Tips on Programming with STEP7 . . . . . 1/P3/7-257

7.1 Copying data 1/P3/7-258
7.2 ANY and POINTER 1/P3/7-258
7.2.1 Use of POINTER and ANY in FC if POINTER or ANY is available as parameter 1/P3/7-259
7.2.2 Use of POINTER and ANY in FB if POINTER or ANY is available as parameter 1/P3/7-260
7.2.3 POINTER or ANY variable for transfer to FC or FB 1/P3/7-261
7.3 Multi-instance DB 1/P3/7-263
7.4 Strings 1/P3/7-264
7.5 Determining offset addresses for data block structures 1/P3/7-265

8 Coupling two FM-NC Modules to one PLC 1/P3/8-267

8.1 Coupling two FM-NCs with one PLC 1/P3/8-268
8.2 Coupling with the components MMC/MSTT/OP 1/P3/8-270
8.3 Interface structure 1/P3/8-271
8.3.1 PLC/NCK interface 1/P3/8-271
8.3.2 PLC/MMC interface 1/P3/8-273
8.3.3 PLC/MCP/HHU interface 1/P3/8-273
8.4 Function of the basic program for two FM-NCs 1/P3/8-275
8.5 Special features of blocks when using two FM-NCs 1/P3/8-276
8.5.1 FB1: RUN_UP 1/P3/8-276
8.5.2 FB2...5: Read/write NC variables, PI services, GUD variables . . . 1/P3/8-278
8.5.3 FC 15, 16, 18: POS_AX, PART_AX, SpinCtrl 1/P3/8-279
8.5.4 FC 9: ASUB 1/P3/8-280

© Siemens AG, 2003. All rights reserved

8.6 Basic program error messages 1/P3/8-281
8.7 Memory space 1/P3/8-281

9 Machine Data, Alarms 1/P3/9-283

9.1 Machine data 1/P3/9-283
9.1.1 General machine data 1/P3/9-283
9.1.2 Channel-specific machine data 1/P3/9-285
9.2 Alarms 1/P3/9-285

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

General

The basic program organizes the exchange of signals and data between the PLC user program and the NCK (Numerical Control Kernel), MMC (Man Machine Communication) and MCP (Machine Control Panel) areas. A distinction is made between the following groups for signals and data:

- Cyclical signal exchange
- Event-driven signal exchange
- Messages

Cyclical signal exchange

The cyclically exchanged signals basically consist of bit fields.

- They contain **commands** transmitted from the PLC to the NCK (such as start or stop) and **status information** from the NCK (such as program running, interrupted, etc.).
- The bit fields are organized into signals for

  mode groups,
  channels,
  axes/spindles and
  general NCK signals.

The cyclical exchange of data is performed by the basic program at the start of the PLC cycle (OB1). This ensures that the signals of the NCK remain constant throughout the cycle.

Event-driven signal exchange NCK → PLC

PLC functions that have to be executed as a function of the workpiece program are triggered by auxiliary functions in the workpiece program. If the auxiliary functions are used to start execution of a block, the type of auxiliary function determines whether the NCK has to wait before executing the function (e.g. during a tool change) or whether the function is executed in parallel to machining of the workpiece (e.g. for tool preparation on milling machines with chain-type magazines).

The data transfer must be as fast and yet as reliable as possible, in order to minimize the influence on the NC machining process. Data transfer is therefore controlled by alarms and acknowledgments. The basic program evaluates the signals and data, acknowledges this to the NCK and transfers the data to the application interface at the start of the cycle. Where the data do not require user acknowledgment, this does not affect NC machining.

© Siemens AG, 2003. All rights reserved

Event-driven signal exchange PLC → NCK

An “event-driven signal exchange PLC –> NCK” takes place whenever the PLC passes a request to the NCK (e.g. traversal of an auxiliary axis). In this case, the data transfer is also controlled by acknowledgment. When performed from the user program, this type of signal exchange is triggered using a function block (FB) or function call (FC).

The associated FBs (Function Blocks) and FCs (Function Calls) are supplied together with the basic program.

Messages

User messages are acquired and conditioned by the basic program. A defined bit field is used to transfer the message signals to the basic program. The signals are evaluated there and entered in the PLC diagnostics buffer on the occurrence of the message events. Where an operator panel (e.g. MMC 100) is provided, the messages are transferred to and displayed on the OP.



Warning

The function of the machine is largely determined by the PLC program. Every PLC program in the working memory can be edited with the programming device.

■

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 Key data of PLC CPUs for FM-NC, 840D and 810D

The following table lists the performance data of the PLC and the scope of the basic program for the various control versions.

© Siemens AG, 2003. All rights reserved

Type of control / Feature		**810D and 840D**		
		Integrated PLC CPU314	Integrated PLC CPU3152–DP 6ES7 315–2AF00–0AB0	Integrated PLC CPU3152–DP Master/Slave 6ES7 315–2AF01–0AB0
1. Key CPU data				
Memory for user + basic program		64, 96, 128KB	64, 96 128, 160, 192, 224, 256, 288KB (dependent on option)	96 , 160, 224, 288, 352, 416, 480KB (dependent on option)
Data block memory		Like user memory	Like user memory	Up to 96 kbytes
Memory submodule		No	No	No
Bit memories		2048	2048	2048/ 4096 with PLC operating system 03.10.13 or later
Timers		128	128	128
Counters		64	64	64
Clock memories		8	8	8
Program/data blocks OB FC FC DB		 1, 10, 20, 35, 40, 80–82, 85, 87, 100, 121–122 1–127 1–127 1–127	 1, 10, 20, 35, 40, 80–82, 85–87, 100, 121–122 1–127 1–127 1–127	 1, 10, 20, 35, 40, 80–82, 85–87, 100, 121–122 0–255 0–255 1–399
Max. data block length		16KB	16KB	16KB
Max. block length FC, FB		16KB	24KB	24KB
Inputs/outputs (address capacity) – digital – analog		 768 64	 1024/1024 64	 1024/1024 64
Inputs/outputs (addressing) – digital – analog	Row 0 is integrated in the NC. Rows 1 to 3 are available for I/O devices	Since row 0 is not available for I/O devices: From I/O byte 32 From PI/PO byte 384	Through optional configuring of I/O devices: From I/O byte 0 From PI/PO byte 272	Through optional configuring of I/O devices: From I/O byte 0 From PI/PO byte 272
Execution time – bit instructions (I/O) – word instructions		 0.3 ms/kA 1–4 ms/kA	 0.3 ms/kA 1–4 ms/kA	 0.3 ms/kA 1–4 ms/kA
PDIAG (Alarm S,SQ)		No	No	Yes
PROFIBUS		Omitted	Master	Master/Slave
Number of PROFIBUS slaves			Min. 16, max. 64 Size SDB 2000 <= 8KB	Min. 16, max. 64 Size SDB 2000 <= 32KB
PBC (programmable block communication)		No	No	Yes
Consistent data to standard slave via SFC 14, 15		Omitted	26	26

© Siemens AG, 2003. All rights reserved

Type of control Feature		810D and 840D		
		Integrated PLC CPU314	Integrated PLC CPU315–2DP 6ES7 315–2AF00–0AB0	Integrated PLC CPU315–2DP Master/Slave 6ES7 315–2AF01–0AB0
2. I/O expansion				
I/O modules		24	24	24
PROFIBUS DP modules		Omitted	Yes	Yes
Interfaces (MPI)		1	1	1

Type of control Feature		840 Di	810D	840D
		Integrated PLC 315–2DP Master/Slave 6ES7 315–2AF03–0AB0	Integrated PLC 315–2DP Master/Slave 6ES7 315–2AF03–0AB0	Integrated PLC 314C–2DP Master/Slave 6FC5 314–6CF00 –0AB0
1. Key CPU data				
Memory for user + basic programs		64, 96, 128, 160, 192, 224, 256, 288KB	64, 96, 128, 160, 192, 224, 256, 288KB	96, 160, 224, 288, 352, 416, 480 KB (dependent on option)
Data block memory		Like user memory	Like user memory	to 96 KB
Memory submodule		No	No	No
Bit memories		4096	4096	4096
Timers		128	128	256
Counters		64	64	256
Clock memories		8	8	8
Program/data blocks OB FC FC DB		 1, 10, 20, 35, 40, 80–82, 85–87, 100, 121–122 0–255 0–255 1–399	 1, 10, 20, 35, 40, 80–82, 85–87, 100, 121–122 0–255 0–255 1–399	 1, 10, 20, 35, 40, 80–82, 85–87, 100, 121–122 0–255 0–255 1–399
Max. length of data block		16KB	16KB	16KB
Max. block length FC, FB		24KB	24KB	24KB
Inputs/outputs (address capacity) – digital – analog		 1024/1024 64	 1024/1024 64	 1024/1024 64
Inputs/outputs (addressing) – digital – analog	Row 0 is integrated in the NC. Rows 1 to 3 are available for I/O devices	Through optional configuring of I/O devices: From I/O byte 0 onwards from PI/PO byte 272 onwards PROFIBUS only	Through optional configuring of I/O devices: From I/O byte 0 onwards from PI/PO byte 272 onwards	Through optional configuring of I/O devices: From I/O byte 0 onwards from PI/PO byte 272 onwards
Execution time – Bit commands (I/O) – Word commands		 0.3 ms/kA 1–4 ms/kA	 0.3 ms/kA 1–4 ms/kA	 0.1 ms/kA 0.25–1.2 ms/kA
PDIAG (Alarm S,SQ)		Yes	Yes	Yes
PROFIBUS		Master	Master/Slave	Master/Slave

© Siemens AG, 2003. All rights reserved

2.1 Key data of PLC CPUs for FM-NC, 840D and 810D

Type of control / Feature		840 Di	810D	840D
		Integrated PLC 315–2DP Master/Slave 6ES7 315–2AF03–0AB0	Integrated PLC 315–2DP Master/Slave 6ES7 315–2AF03–0AB0	Integrated PLC 314C–2DP Master/Slave 6FC5 314–6CF00 –0AB0
Number of PROFIBUS slaves		At least 16, max. 64 Size SDB 2000 <= 32KB	At least 16, max. 64 Size SDB 2000 <= 32KB	At least 16, max. 32 Size SDB 2000 <= 32KB
PBC (programmable block communication)		Yes	Yes	Yes
Consistent data to standard slave via SFC 14, 15		26	26	32
2. I/O expansion				
I/O modules, central		PROFIBUS only	24	24
PROFIBUS DP modules		Yes	Yes	Yes
Interfaces (MPI)		1	1	1

Type of control / Feature		840 Di	810D	840D
		Integrated PLC 317–2DP Master/Slave 6FC5 317–2AJ10–0AB0		Integrated PLC 317–2DP Master/Slave 6FC5 317–2AJ10–1AB0
1. Key CPU data				
Memory for user + basic programs		786 KB		786 KB
Data block memory		max. 256 kByte		max. 256 kByte
Memory submodule		No		No
Bit memories		32768		32768
Timers		512		512
Counters		512		512
Clock memories		8	8	8
Program/data blocks OB FC FC DB		1, 10, 20–21, 32–35, 40, 55–57, 80, 82, 85–87, 100, 121–122; 0–2048 0–2048 1–2048		1, 10, 20–21, 32–35, 40, 55–57, 80, 82, 85–87, 100, 121–122; 0–2048 0–2048 1–2048
Max. length of data block		32KB		32KB
Max. block length FC, FB		64KB		64KB
Inputs/outputs (addressing volume in bytes) – digital / – analog Incl. reserved area – Process image **Caution:** The inputsoutputs above 4096 are reserved for integrated drives.		4096/4096 8192/8192 256/256		4096/4096 8192/8192 256/256

© Siemens AG, 2003. All rights reserved

Type of control Feature		840 Di	810D	840D
		Integrated PLC 317–2DP Master/Slave 6FC5 317–2AJ10–0AB0		Integrated PLC 317–2DP Master/Slave 6FC5 317–2AJ10–1AB0
Inputs/outputs (addressing) – digital – analog	Row 0 is integrated in the NC. Rows 1 to 3 are available for I/O devices	Through optional configuring of I/O devices: From I/O byte 0 onwards from PI/PO byte 272 onwards PROFIBUS only		Through optional configuring of I/O devices: From I/O byte 0 onwards from PI/PO byte 272 onwards
Execution time – Bit commands (I/O) – Word commands		<= 0.031 ms/kA 0.1 ms/kA		<= 0.103 ms/kA 0.1 ms/kA
PDIAG (Alarm S,SQ)		Yes		Yes
PROFIBUS		Master/Slave		Master/Slave
Number of PROFIBUS slaves		max. 125		max. 125
PBC (programmable block communication)		Yes		Yes
Consistent data to standard slave via SFC 14, 15		128		128
2. I/O expansion				
I/O modules, central		PROFIBUS only		24
PROFIBUS DP interfaces		1 (2)		1
Interfaces (MPI)		1 (0)		1

Note about number of PROFIBUS slaves

Since the content of SDB 2000 and other associated SDBs must be stored by the PLC operating system in internal data structures in the SRAM which can also be accessed by the PROFIBUS ASIC, SDBs that are less than 32KB may also be rejected on loading. It is not possible to specify the size of SDB 2000 exactly. It cannot be determined whether the configuration is legal until the SDB container has been loaded to the CPU. The values shown above must be taken as guide values only. If the configuration is illegal, a general reset request is issued when the SDBs are loaded. The cause of the configuring error can be found in the diagnostic buffer on completion of the general reset.

PLC versions

With SW 3.5 and higher on the 840D,
version 6 (version code 35.06.03) is installed with PLC 314 and
version 3 (version code 35.03.03) with PLC 315–2DP
or later.

These versions are compatible with the corresponding SIMATIC CPU300. All modules and software packages that are approved for these versions and CPUs on the SIMATIC are therefore suitable.
Modules that can generally only be installed in subrack 0 are the exception (modules FM NC and FM 357 are also exceptions).

The version code in the version display comprises the following information up to SW 3.6:

NC version, PLC version of SIMATIC-CPU, internal incrementation.

The version code in the version display comprises the following information with SW 3.6 and higher:

PLC version of SIMATIC-CPU, incrementation of transferred firmware, internal incrementation.

© Siemens AG, 2003. All rights reserved

Example of code in SW 3.6 and higher

PLC 315–2DP with order number 6ES7 315–2AF00–0AB0: 04.02.14
PLC 315–2DP with MLFB 6ES7 315–2AF01–0AB0: 03.10.23
PLC 314: 07.02.12

Version display on MMC

If you scroll to the end of the version display in NCK SW 3.7, you will see details of the PLC currently installed and the associated version of the PLC operating system.
e.g.
S7 PLC_315–2DP System 03.10.23
You can read off the module ID of the installed PLC module in the next column. The following PLC module IDs are currently in use:

Module ID	PLC module	Suitable PLC operating systems (appropriate SIMATIC MLFB)	Designation of software version/ operating system
0208	PLC 314	6ES7 314-1AE0-0AB0	07.02.12
1008	PLC 315–2DP with ASPC 2 Step C	6ES7 315-2AF00-0AB0	04.02.14
1100	PLC 315–2DP with ASPC 2 Step D	6ES7 315-2AF01-0AB0	03.10.23
1200	PLC 315–2DP with ASPC 2 Step E	6ES7 315-2AF01-0AB0 or 6ES7 315-2AF03-0AB0 FW1.2	03.10.23 12.30.09
1400	PLC 314C–2DP with IBC 16	6ES7 314-6CF00-0AB0 FW 1.0.2	10.60.17
2200	PLC 317–2DP with IBC 32	6ES7 317-2AJ10-0AB0 FW 2.1	20.71.01
MCI 1 (840Di)	PLC 315–2DP with ASPC 2 Step E	6ES7 315-2AF03-0AB0 FW 1.0	4.20.35
MCI 2 (840Di) 2100	PLC 317–2DP with IBC 32	6ES7 317-2AJ10-0AB0 FW 2.1	20.70.01

Feature		FM-NC		
		CPU 314	CPU 315	
1. Key CPU data				
	Memory for user + basic program	24KB	48KB	
	Memory submodule	Yes	Yes	
	Bit memories	2048	2048	
	Timers	128	128	
	Counters	64	64	
	Clock memories	8	8	
	Program/data blocks OB FC FC DB	 1, 10, 20, 35, 40, 80–85, 100, 120–123 1–127 1–127 1–127	 1, 10, 20, 35, 40, 80–85, 100, 120–123 1–127 1–127 1–127	
	Max. data block length (DB, FC, FB)	8KB	8KB	

© Siemens AG, 2003. All rights reserved

Feature		FM-NC		
		CPU 314	**CPU 315**	
	Inputs/outputs (address capacity) – digital – analog	 512 64	 512 64	
	Execution time – bit commands (I/O) – word commands	 0.3 ms/kA 1–4 ms/kA	 0.3 ms/kA 1–4 ms/kA	
2. I/O expansion				
	I/O modules	32	32	
	PROFIBUS L2DP modules	Omitted	Omitted	
	Interfaces (MPI)	1	1	
	Interfaces (OPI)	Omitted	Omitted	
	"FM-NC" NCUs per IM	1	1	

Type of control		**FM-NC**	**810 D**	**840D**
Interfaces (OPI)		**Omitted**	**Omitted**	**1**
3. Functions of basic program				
	Scope: Axes/spindles Channels Mode groups	 5 1 1	 5 2 1	 31 10 10
	Functions: Status/control signals M decoders (M00–99) G group decoders Aux. function distributors Alarm-driven aux. fnc. transfer Move axes/spindles from PLC Async. subprogram interface Error/operating messages MCP and handheld unit signals via NCK	 √ √ √ √ √ √ √	 √ √ √ √ √ √ √ √	 √ √ √ √ √ √ √ √ √
	Reading/writing of NC variables	√	√	√
	PI services	√	√	√
	Tool management		√	√
	Star/delta switchover		√	√
	Display control Handheld unit	√	√	√

See also Catalog ST 60.1 for PLC functions.

© Siemens AG, 2003. All rights reserved

2.2 Reserving resources (timer, FC, FB, DB, peripherals)

Timers

Timers T0 to T9 are reserved for the basic program.

FC, FB, DB

FC0 to FC29 and FB0 to FB29 are reserved for the basic program.

Data blocks DB1 to DB62 and DB71 to DB80 are reserved.
Data blocks from not activated channels, axes / spindles, tool management are free for the user.

FC30 to FC35 and DB81 to DB89 are reserved for ShopMill / ShopTurn instructions.

In the case of PLC 317–2DP, a further range of numbers is reserved in FC, FB and DB for Siemens applications.

For FC and FB respectively, 1000 to 1023 are reserved.
For DB, 1000 to 1099 are reserved.

I/O range:

The PLC 317 has an I/O address volume of 8192 bytes each for input and outputs. The address ranges from 4096/4096 are reserved here for integrated drives. Diagnostic address for modules can be placed in the top address range as suggested by STEP7. Address range 256 to 271 is reserved for the NC module and future extensions.

© Siemens AG, 2003. All rights reserved

2.3 Starting up hardware configuration of PLC CPUs

General procedure

The hardware configuration for the PLC CPUs used including the associated I/Os must be defined by means of STEP 7.
The following sequence of operations (STEP7 V2.1) illustrates how this process works:

1. Load tool box to PG/PC
2. Create a new project (File, New, Project)
3. Insert, Hardware, SIMATIC 300 station
4. Select SIMATIC 300 station1 with mouse
5. Open object with right-hand mouse key to start the HWConfig
6. Destination system, load to PG, the hardware equipment complement is read back from the central system
7. Configure distributed I/Os.
8. Add PLC basic program (see next section)

The addresses for the I/O modules can be changed if necessary (permissible only on certain PLC CPUs, e.g. PLC 315–2DP).

As an alternative, the entire hardware configuration can be entered manually (see also appropriate STEP 7 documentation).
Observe the information given under Note below.
With STEP7 Version 3 and higher, the hardware configuration of the SINUMERIK components must be defined with the entries in SIMATIC\RACK 300. The install or setup program of the basic program on the tool box diskettes is required for this purpose.

With STEP7 V5.1 SP2 and Toolbox 6.03.02 or later, the SINUMERIK components are stored under SIMATIC 300\SINUMERIK. The current hardware add-on for STEP 7 can also be found under eSupport.
Current path (19.05.2003): sinumerik_software –> 840d/810d/fm–nc –> patches & fixes –> plc –> Hardware_fuer_STEP7 –> step7_v6.x –> sinumerik_hw_step7.exe

NCU	MLFB	Comparable SIMATIC CPU order number included	Selection from STEP7 hardware catalog
CCU1 810D CPU	6FC5 410–0AA00–0AA0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
CCU2 810D CPU	6FC5 410–0AA01–0AA0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
CCU1 810DE CPU	6FC5 410–0AY01–0AA0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
CCU2 810D CPU	6FC5 410–0AX02–0AA0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 810DE Light CCU1 module with system software (export)	6FC5 410–0AY00–0AA0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 810D CCU2 module with system software (standard)	6FC5 410–0AX02–1AA0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840DE NCU 561.2 without system software	6FC5 356–0BB11–0AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 571 (export version)	6FC5 357–0BA10–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D NCU 571 (export version) with PROFIBUS DP	6FC5 357–0BA11–0AE0	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 571.2 (export version) with PROFIBUS DP	6FC5 357–0BA11–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840DE NCU 571.2 without system software	6FC5 357–0BB11–0AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC3152AF01
SINUMERIK 840D NCU 572	6FC5 357–0BA20–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 572	6FC5 357–0BA20–1AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 572	6FC5 357–0BA21–0AE0	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 572	6FC5 357–0BA21–1AE0	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 572.2 with PROFIBUS DP	6FC5 357–0BA21–1AE1	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D/DE NCU 572.2 without system software	6FC5 357–0BB21–0AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D/DE NCU 572.3 without system software	6FC5 357–0BB22–0AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 572 (export version)	6FC5 357–0BY20–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 572 (export version)	6FC5 357–0BY20–1AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 572 (export version) with PROFIBUS DP	6FC5 357–0BY21–0AE0	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 572 (export version) with PROFIBUS DP	6FC5 357–0BY21–1AE0	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 572.2 (export version) with PROFIBUS DP	6FC5 357–0BY21–1AE1	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 572 with digitizing	6FC5 357–0BA24–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 572.2 with digitizing and PROFIBUS DP	6FC5 357–0BA24–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D/DE NCU 572.2 without system software	6FC5 357–0BB24–0AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 572 (export version) with digitizing	6FC5 357–0BY24–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 572.2 (export version) with digitizing and PROFIBUS DP	6FC5 357–0BY24–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 573	6FC5 357–0BA30–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840DE NCU 573 (export version)	6FC5 357–0BY30–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 573 with digitizing	6FC5 357–0BA31–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 573 (export version) with digitizing	6FC5 357–0BY31–0AE0	6ES7 314–1AE01–0AB0	810D/840D with PLC314
SINUMERIK 840D NCU 573 with PROFIBUS DP	6FC5 357–0BA32–0AE1	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D NCU 573 (export version) with PROFIBUS DP	6FC5 357–0BY32–0AE1	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 573 with PROFIBUS DP	6FC5 357–0BA33–0AE0	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 573 (export version) with PROFIBUS DP	6FC5 357–0BY33–0AE0	6ES7 315–2AF00–0AB0	840D with PLC315–2AF00
SINUMERIK 840D NCU 573.2 (Pentium Pro) up to 12 axes with PROFI-BUS DP	6FC5 357–0BA32–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 573.2 (Pentium Pro) up to 31 axes with PROFI-BUS DP	6FC5 357–0BA33–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D/DE NCU 573.2 without system software	6FC5 357–0BB33–0AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D/DE NCU 573.2 Pentium II without system software	6FC5 357–0BB33–0AE1	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 573.2 (Pentium Pro) for digitizing with PROFIBUS DP	6FC5 357–0BA31–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D/DE NCU 573.2 without system software	6FC5 357–0BB31–0AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 573.2 (Pentium Pro) (export version) for 12 axes with PROFIBUS DP	6FC5 357–0BY32–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 573.2 (Pentium Pro) (export version) for 31 axes with PROFIBUS DP	6FC5 357–0BY33–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 573.2 (Pentium Pro) (export version) for digitizing with PROFIBUS DP	6FC5 357–0BY31–1AE0	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840Di	6FC5 220–0AA00–1AA0	6ES7 315–2AF03–0AB0	840Di with PLC315–2AF03
SINUMERIK 840Di with PK bus		6ES7 315–2AF03–0AB0	840Di with PLC315–2AF03, PK bus
SINUMERIK 840D NCU 572.3	6FC5 357–0BB22–0AE0	With BESY 03.10.23: 6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 572.3	6FC5 357–0BB22–0AE0	With BESY 12.30.07: 6ES7 315–2AF03–0AB0	810D/840D with PLC315–2AF03
SINUMERIK 840D NCU 573.3	6FC5 357–0BB33–0AE2	6ES7 315–2AF01–0AB0	810D/840D with PLC315–2AF01
SINUMERIK 840D NCU 573.3	6FC5 357–0BB33–0AE2	With BESY 12.30.07: 6ES7 315–2AF03–0AB0	810D/840D with PLC315–2AF03 (as of STEP7 V5.0 and Toolbox 05.03.05)
SINUMERIK 840D NCU 572.4	6FC5 357–0BB23–0AE0	6ES7 314–6CF00–0AB0	810D/840D with PLC314C–2DP
SINUMERIK 840D NCU 573.4	6FC5 357–0BB34–0AE1	6ES7 314–6CF00–0AB0	810D/840D with PLC314C–2DP
SINUMERIK 810D CCU3	6FC5 410–0AY03–0AA0	6ES7 315–2AF03–0AB0	810D/840D with PLC315–2AF03

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D NCU 571.3	6FC5 357–0BB11–0AE1	6ES7 315–2AF03–0AB0	810D/840D with PLC315–2AF03
SINUMERIK 840D NCU 561.3	6FC5356–0BB11–0AE1	6ES7 315–2AF03–0AB0	810D/840D with PLC315–2AF03 (as of STEP7 V5.0 and Toolbox 05.03.05)
SINUMERIK 840D NCU 571.4	6FC5357–0BB12–0AE0	6ES7 314–6CF00–0AB0	810D/840D with PLC314C–2 DP (as of STEP7 V5.1 SP3 and higher and Toolbox 06.03.02 and higher)
SINUMERIK 840D NCU 561.4	6FC5356–0BB12–0AE0	6ES7 314–6CF00–0AB0	810D/840D with PLC314C–2 DP (as of STEP7 V5.1 SP3 and higher and Toolbox 06.03.02 and higher)
SINUMERIK 840D NCU 573.5	6FC5357–0BB35–0A E0	6ES7 317–2AJ00–0AB0	810D/840D with PLC317–2 DP (as of STEP7 V5.2 SP1 and Toolbox 06.03.02)
SINUMERIK 840Di with MCI2	6FC5 222–0AA02–1AA0	6ES7 317–2AJ00–0AB0	840Di with PLC317–2 DP (as of STEP7 V5.2 SP1 and Toolbox 06.03.02)

Note

On the SINUMERIK 810D or 840D, SIMATIC row 0 is integrated in the NC. The components installed in this row are as follows:
The integrated PLC in slot 2 (PLC 314 or PLC 315–2DP)
An IM 360 in slot 3
The FM NCU in slot 4 (with a PLC 314, this FM NCU must also be defined when NC SW 3.5 or later is installed and when other MPI (K bus) nodes are contained in rows 1 to 3 (e.g. FM modules with K bus connection). The properties of the FM NCU must not be changed, as process interrupts (e.g. auxiliary functions) of the NCU may, in this case, no longer function.

© Siemens AG, 2003. All rights reserved

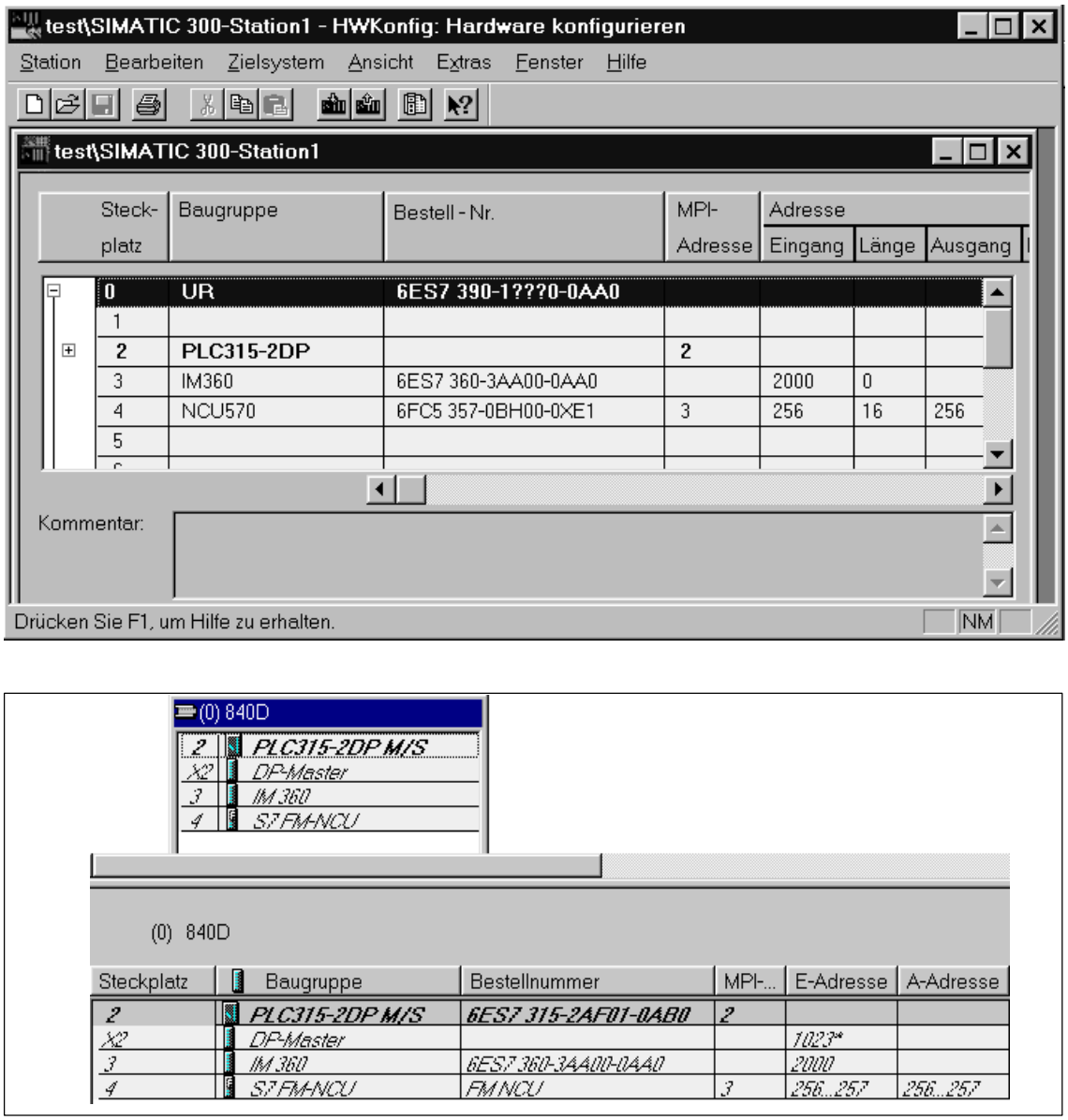


Fig. 2-1 Hardware configuration of row 0 on 810D, 840D

MCP HHU Configuration global data

(only valid for SINUMERIK 810D up to SW 3.x and FM-NC)
If the MCP or HHU has to be configured (deviation from standard), an additional SIMATIC 300 station should be added to the machine project for each operator component. Any type of CPU must be inserted in location 2 on row 0 in this station by means of the hardware configuration (HW config.). The desired MPI address of the operator component must be set as the MPI address. MPI network (1) can then be marked in the SIMATIC manager. The global data can then be activated via menu item Extras. The rest of the procedure is described in detail in the Installation and StartUp Guide.

Detailed information on the configuration of the machine control panel, hand-held unit is contained in Subsection 2.6.10 of this document.

© Siemens AG, 2003. All rights reserved

2.4 Starting up the PLC program

2.4.1 Installing the basic program for FM NC, 810D, 840D

A complete general reset of the NCU and the PLC is necessary before the initial start-up of the NCU component. To do this, turn switch S3 to position 1 and switch S4 to position 3. Then, switch on the control again. This action generates a “hard reset request” on the PLC. The memories of the PLC and NC are then initialized.

Installation

With SW 6.1 and higher, the installation is performed by a WINDOWS compliant setup program for the basic program, hardware selection in STEP7 (SINUMERIK 810D/840D option package) and the NC-Var Selector. To start the installation, run setup.exe in the main CD directory. You can then choose which components to install. After the installation you can select the basic program library directly from STEP 7 (gp8x0d61, 61 is the main basic program version). You can check the basic program version in the object properties of the library or in the comment field of the program folder.

With SW 3.7 or 4.2 and higher, the installation is performed by INSTALL.BAT (INSTALL1.BAT, INSTALL4.BAT) (doubleclick). This program installs the basic program and additional files for the appropriate STEP7 version. With the automatic installation in STEP7 V3 and higher, the TYP, GSD and Meta files in the hardware catalog are also augmented and updated. The hardware components of the SINUMERIK system are then also available for hardware configuration under STEP 7. You no longer need to unpack the files as described below.

The basic program is supplied in packed format as a **Project** for
STEP7 V1.x

or as a **library** for
STEP7 V2.x and its successor versions

delivered in packed form.

General

The source programs of the OBs including standard parameterization, interface symbols and DB templates for handheld unit and M decoding functions are enclosed in the SIMATIC project or SIMATIC library of the basic program.

STEP7 must be installed before the basic program.

STEP7 V1.x

The basic program is stored as a packed file with the name GP840D.EXE (or GP810D.EXE and GPFM-NC.EXE) in the main directory on the diskette. The basic program (GP840D.exe) must be copied to the main directory (root) of a drive (e.g. c:\) and then called. The project structure required for the basic program is generated automatically. The catalog name of the basic program is GP840Dxy.S7A. In this case, xy stands for the basic program version.

© Siemens AG, 2003. All rights reserved

Note

When STEP7 V1 is used, the GP840Dxy.S7A catalog must be copied to the root directory. Any existing catalog with the same name GP840Dxy.S7A must be deleted beforehand.

STEP7 V2.x, 3.x

The basic program is stored as a packed file with the name GP840D.EXE in directory S7V2.840 or S7V2.810 or S7V2 on the basic program diskette. The basic program (GP840D.exe) must be copied to subcatalog “S7LIBS” of STEP7 V2 (step7_v2) or subsequent versions thereof and then called. The library structure required for the basic program is generated automatically. The catalog name of the basic program is GP840Dxy. In this case, xy stands for the basic program version. The file MET.EXE must be copied into the basic catalog of STEP7 and called from there via the DOS window with MET.EXE O.

Note

The name GP840D specified above refers to the basic program of the SINUMERIK 840D. The basic program is named GP810D on the SINUMERIK 810D and GPFM-NC on the FM-NC. With effect from SW 4.2, the basic program for 810D and 840D is combined. It is now called GP8x0D.

2.4.2 Application of basic program

A new CPU program (e.g. “Turnma1”) must be set up in a project by means of the STEP7 software for each installation (machine).

Note

The catalog structures of a project and the procedure for creating projects and user programs are described in the relevant SIMATIC documentation.

STEP7 V1

A network to the PLC must be activated for the machine CPU program under menu items “Edit”, “Configuration”. This is done in the “Services”, “Parameterize” menu followed by selection of the MPI parameters.
Default settings:

- **“Networked”**
- **“MPI subnetwork number = 0”**
- **“CPU MPI Addr = 2”**

© Siemens AG, 2003. All rights reserved

The following must be copied into the CPU program for the machine-specific program files:

1. The basic program blocks (FCs, FBs, DBs, OBs, SFCs, SFBs and UDTs as well as SDB 210) must be copied (in menu “File”, “Manage project” in STEP7 program editor). SDB 210 is provided only on 810D and FM-NC systems.
2. File GPOB840D.AWL (or GPOB810D.AWL or GPOBFM-NC.AWL) and other STL (AWL) files if appropriate must also be copied from the basic program catalog into the CPU program. The OBs contained in this file are the basis for the user program with the associated basic program calls. Existing user blocks must be copied as STL files into the newly created CPU program (catalog name CPU1.S7D) and compiled.
3. We also recommend that the symbolic names are transferred with the files from the basic program package using the symbol editor.

STEP7 V2

The basic program blocks are copied using the SIMATIC manager via File/ Open/Library.
The following parts must be copied from the library:

AP-off: FCs, FBs, DBs, OBs, SFC, SFB, UDT and the SDB container.

SDB 210 is stored in the SDB container for 810D and FM-NC.
The SDB container exists solely for these control variants.

The source files (SO):

GPOB810D or GPOB840D or GPOBFM-NC,
possibly MDECLIST, BHG_DB and others

The symbol table (SY).

Compatibility with STEP7

No interdependencies exist between the basic program (including older program versions) and currently valid versions of STEP7 (STEP7 V1 to 3).

© Siemens AG, 2003. All rights reserved

2.4.3 Version codes

Basic program

The basic program version including the control type is output in the version display of the MMC with SW 4 (NCK, PLC) and higher.

In earlier versions of the basic program, the PLC version is stored in data block DB 17 on the data double word DBD 0. A HEX number must be set as the data format. The following example shows how V3.2 is displayed:

DB17.DBD0: 03**32**_0100

The program version is coded in display decades 3 and 4 (bold print). The two left-aligned decades contain the control type of this basic program. The other decades contain a development code.
The control type is coded as follows:

Left-aligned decade of DB17.DBD0 (byte 0)	Type of control
01	FM-NC
02	SINUMERIK 810D
03	SINUMERIK 840D (571, 572, 573)
04	SINUMERIK 840Di

User program

With SW 4 and higher, the user can also display the version code of his own user program on the MMC.

For this purpose, a data of type STRING containing a maximum of 54 characters must be defined in any data block. The data can contain a text of the user's choice. Parameter assignments for this string are made via a pointer in FB 1. Parameterization requires a symbolic definition of the data block.
For further details, see Block Description of FB 1.

© Siemens AG, 2003. All rights reserved

2.4.4 Machine program

The machine manufacturer creates the machine program using the library routines supplied with the basic program. The machine program contains the logic operations and sequences on the machine. The interface signals to the NC are also controlled in this program. More complex communications functions with the NC (e.g. read/write NC data, tool management acknowledgments, etc. are activated and executed via FCs and FBs of the basic program). The machine program can be created in different creation languages (e.g. STL, LAD, CSF, S7 HIGRAPH, S7GRAPH, SCL). The complete machine program must be generated and compiled in the correct sequence. This means that blocks that are called by other blocks must generally be compiled before the blocks which call them. If blocks that are called by other blocks are subsequently modified in the interface (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR) as the program is developed, then the call block and all blocks associated with it must be compiled again. This general procedure applies analogously to instance data blocks for FBs. If this sequence of operations is not observed, time-stamp conflicts occur when the data are retranslated into STEP7. In some cases, therefore, it may not be possible to retranslate blocks, creating problems, for example, with the “Block status” function. It is moreover advisable to generate blocks in ASCIISTL by means of the STEP7 editor when they have been created in the ladder diagram or in single statements (incremental mode).

2.4.5 Data backup

The PLC CPU does not store symbolic names, but only the data type descriptions of the block parameters (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR) and the data types of the global data blocks. Without the associated project for this machine, therefore, blocks cannot be retranslated meaningfully (e.g. for the Block status function or on subsequently required modifications to PLC CPU programs). It is therefore necessary to keep a backup copy of the STEP7 project located in the PLCCPU on the machine. This is extremely useful for servicing purposes and saves time and problems. If the STEP7 project exists and has been created according to the instructions given above, then symbols can be processed in the PLCCPU on this machine. It may also be advisable to store the machine source programs as STL files in case they are required for any future upgrade.

The source programs of all organization blocks and all instance data blocks should always be available.

© Siemens AG, 2003. All rights reserved

2.4.6 PLC series start-up, PLC archives:

After the blocks have been loaded to the PLC CPU, a series archive can be generated via the MMC operator interface to back up data on the machine. To ensure data consistency, this backup must be created immediately after block loading when the PLC is in the Stop state. It does not replace the SIMATIC project backup as the series archive saves binary data only, and does not back up, e.g. symbolic information. No CPU DBs (SFC 22 DBs) and SDBs created in the CPU and still not stored.

With toolbox version 06.03.03 and STEP 7 V5.1 or later, the PLC series archive can be generated directly from the appropriate SIMATIC project.
To do this, select menu commands “Options” –> “Settings” and the “Archive” tab in STEP 7. This contains an entry “SINUMERIK (*.arc)” which must be selected to create a series start-up file. After selection of the archive, select menu options “File”–> “Archive”. The relevant series archive will then be generated. If the project contains several programs, the program path can be selected. A series archive is set up for the selected program path. All blocks contained in the program path are incorporated into the archive, except for CPU DBs (SFC 22 DBs).

Automation:

The process of generating a series archive can be automated (comparable to command interface of STEP 7, V5.1 and higher). This automatic generation function is an extension of the command interface.
It offers the following functions:

The functions (shown here in VB script) are not available until server instantiations and Magic have been called:

```
Const S7BlockContainer = 1138689, S7PlanContainer = 17829889
Const S7SourceContainer = 1122308
set S7 = CreateObject(“Simatic.Simatic.1”)
rem Instantiate command interface of STEP7
Set S7Ext = CreateObject(“SimaticExt.S7ContainerExt”)
Call S7Ext.Magic(“”)
```

Functions:
Function **Magic**(bstrVal As String) As Long
Function **MakeSeriesStartUp**(FileName As String, Option As Long, Container As S7Container) As Long

Descriptions:

Function **Magic**(bstrVal As String) As Long

Call gives access to certain functions. The function must be called once after server instantiation. The value of bstrVal can be empty. This initiates a check of the correct Step7 version and path name in Autoexec. The functions are enabled when 0 is returned.
Return (–1) = incorrect Step7 version
Return (–2) = no entry in Autoexec.bat

Function **MakeSeriesStartUp**(FileName As String, Option As Long, Container As S7Container) As Long

© Siemens AG, 2003. All rights reserved

Parameter option:
0: Normal series startup file with general reset
Bit 0 = 1: Series startup file without general reset. When project contains SDBs, this option is inoperative. A general reset is then always executed.

Bit 1 = 1: Series startup file with PLC restart (supported with MMC SW 6.2 and higher)

Return value:
0 = OK
–1 = Function not available, call Magic function beforehand
–2 = File name cannot be generated
–4 = Parameter container invalid or block container empty
–5 = Internal error (save request rejected by Windows)
–6 = Internal error (problem with STEP7 project)
–7 = Write error on generating series startup file (e.g. floppy disk full)

Use in script:

```
If S7Ext.Magic(“”) < 0 Then
  Wscript.Quit(1)
End If
 Set Proj1 = s7.Projects(“new”)
 set S7Prog = Nothing
Set s7prog = Proj1.Programs.Item(1) ’if there is only one program
For i = 1 to S7Prog.Next.Count
      Set Cont = S7Prog.Next.Item(i)
’ Check block container
      If (Cont.ConcreteType = S7BlockContainer) Then
         Exit For
      End if
Next
Error = S7Ext.MakeSeriesStartUp(“f:\dh\arc.dir\PLC.arc”, 0, Cont)’Now error evaluation
```

© Siemens AG, 2003. All rights reserved

2.4.7 Software upgrades

Software upgrade

Whenever you update the PLC or NCK software, always first bring the PLC to its initial state. This initial clear state can be achieved by means of a general PLC reset. With a reset of this kind, all existing blocks are deleted.
Generally speaking, the new basic program must be incorporated when a new NC software version is installed. The basic programs blocks must be loaded into the user project for this purpose. OB 1, OB 40, OB 100, FC 12 and DB 4 should not be loaded if these blocks are already included in the user project. These blocks may have been modified by the user. The new basic program must be linked with the user program. This must be done by the following method:

1. Generate the text or source file of all user blocks before copying the basic program.
2. Then copy the new basic program blocks to this machine project (for a description, see Subsection “Application of basic program”)
3. All user programs *.awl must then be retranslated in the correct order! (See also Subsection “Machine program”).
   This newly compiled machine program must then be loaded to the PLC CPU by means of STEP7.

It is normally sufficient to recompile the organization blocks (OB) and the instance data blocks of the machine program, however. That means you only need to generate sources for the organization blocks and the instance data blocks (before upgrading).

General reset

A description of how to perform a general PLC reset is described in the Installation and StartUp Guide. However, a general reset does not delete the contents of the diagnostics buffer nor the node address on the MPI bus. Another possible general reset method is described below. This method must be used when the normal general reset process does not work.
The procedure is as follows:

No.	Action	Effect
1	Control system is switched off	
2	PLC switch setting 3 (MRES) and switch control on again or perform hardware reset.	LED labeled PS flashes slowly
3	Set PLC start-up switch to position 2 (STOP) and back to position 3 (MRES).	The LED labeled PS starts to flash faster
4	Set PLC start-up switch to position 2 or 0	

NC variables

The latest NC VAR selector can be used for each NC software version (even earlier versions). The variables can also be selected from the latest list for earlier NC software versions. The data content in DB 120 (default DB for variables) does not depend on the software version, i.e. selected variables in an older software version must not be reselected when the software is upgraded.

© Siemens AG, 2003. All rights reserved

2.4.8 I/O modules (FM, CP modules)

Special packages for STEP7 are generally required for more complex I/O modules. Some of these special packages include support blocks (FC, FB) stored in a STEP7 library. The blocks contain functions for operating the relevant module which are parameterized and called by the user program. In many cases, the FC numbers of the CP and FM module handling blocks are also included in the number range of the basic program for the FM-NC, 810D and 840D systems. How can these conflicts be resolved?
The block numbers of the basic program must not be altered. The block numbers of handling blocks can be assigned new, free numbers using STEP7. These new blocks (with new FC numbers) are then called in the user program with the parameter assignments required by the function.

2.4.9 Troubleshooting

This Section describes problems which may occur, their causes and remedies and should be read carefully before hardware is replaced.

© Siemens AG, 2003. All rights reserved

Table 2-1 Errors, cause/description and remedy

Serial no. error code	Error	Cause/description	Remedy
1	No connection via MPI to PLC.	The MPI cable is not inserted or is defective. The STEP7 software for the MPI card may also be incorrectly configured.	Test: Create a link with the programmer in the STEP7 editor by means of connection “Direct_PLC”. A number of node addresses must be displayed here. If they are not displayed, then the MPI cable is defective/not inserted. In Windows 95 versions of STEP7, the hardware must be checked in the control panel; the PC/programmer interface configuration should also be checked as well.
2	PLC cannot be accessed in spite of PLC general reset.	A system data block SDB 0 has been loaded with a modified MPI address. This has caused an MPI bus conflict due to dual assignment of addresses	Separate all MPI cables to other components. Create the link “Direct_PLC” with the programmer. Correct the MPI address.
3	All four LEDs on the PLC flash (DI disaster)	A system error has occurred in the PLC. **Measures:** The diagnostic buffer on the PLC must be read to analyze the system error in detail. To access the buffer, the PLC must be stopped (e.g. set switch S4 to position 2). The diagnostics buffer can then be read out with STEP7. Relay the information from the diagnostics buffer to the Hotline / Development Service. A general reset must be carried out if requested after the hardware reset. The diagnostics buffer can then be read with the PLC in the Stop state.	After the Reset or reloading of the PLC program, the system may run up normally again. Even in this case, the contents of the diagnostics buffer should be sent to the Development Office.

© Siemens AG, 2003. All rights reserved

2.5 Linking PLC CPUs to FM-NC, 810D, 840D

General

The AS 300 family is used as the PLC for all systems. The essential difference between the NCU variants lies in the method by which they are linked. On the 840 D and 810D, the PLC 314 CPU (user memory capacity up to 128KB) or PLC 315–2 DP (user memory capacity up to 288KB) is integrated as a submodule into the NC unit while the FM-NC is installed as an FM in the AS 314 system or AS 315 (–2DP).
The PLC CPU 315–2 DP also supports distributed I/Os on the PROFIBUS (L2DP). The relevant performance data for individual PLC CPUs can be found in the above table or in Catalog FT70.

2.5.1 Properties of PLC CPUs

The SINUMERIK 810D/840D/840Di PLC CPUs are based on standard SIMATIC CPUs in the S7300 family. As a result, they generally possess the same functions. Functional deviations are shown in the table above. Owing to differences in their memory system as compared to the S7 CPU, certain functions are not available (e.g. blocks on memory card, project on memory card).

Note

With the current SIMATIC CPUs, the PLC is not automatically started after voltage failure and recovery when a PLC Stop is initiated via software operation. In this instance, the PLC remains in the Stop state with an appropriate diagnostic entry for safety reasons. You can start the PLC only via software operation “Execute a restart” or by setting the switch to “Stop” and then “RUN”. This behavior is also integrated in the current versions of the SINUMERIK PLC.

2.5.2 Interface on 810D and 840D with integrated PLC

Physical interfaces

As the 810D and 840D systems have an integrated PLC, signals can be exchanged between the NCK and PLC directly via a dual port RAM (Fig. 2-2).

Exchange with operator panel and MCP

Data are generally exchanged with the operator panel (OP), machine control panel (MCP) and handheld unit (HHU) on the 840D via the operator panel interface (OPI[1)]), the COM module being responsible for data transport.

All devices specified above can also be operated on the multipoint interface (MPI) in the case of the 840 D. With the 810D, data communication with the operator panel (OP), machine control panel (MCP) and handheld unit (HHU) takes place only via the MPI.

© Siemens AG, 2003. All rights reserved

The programming device is connected directly to the PLC via the MPI (multi-point interface).

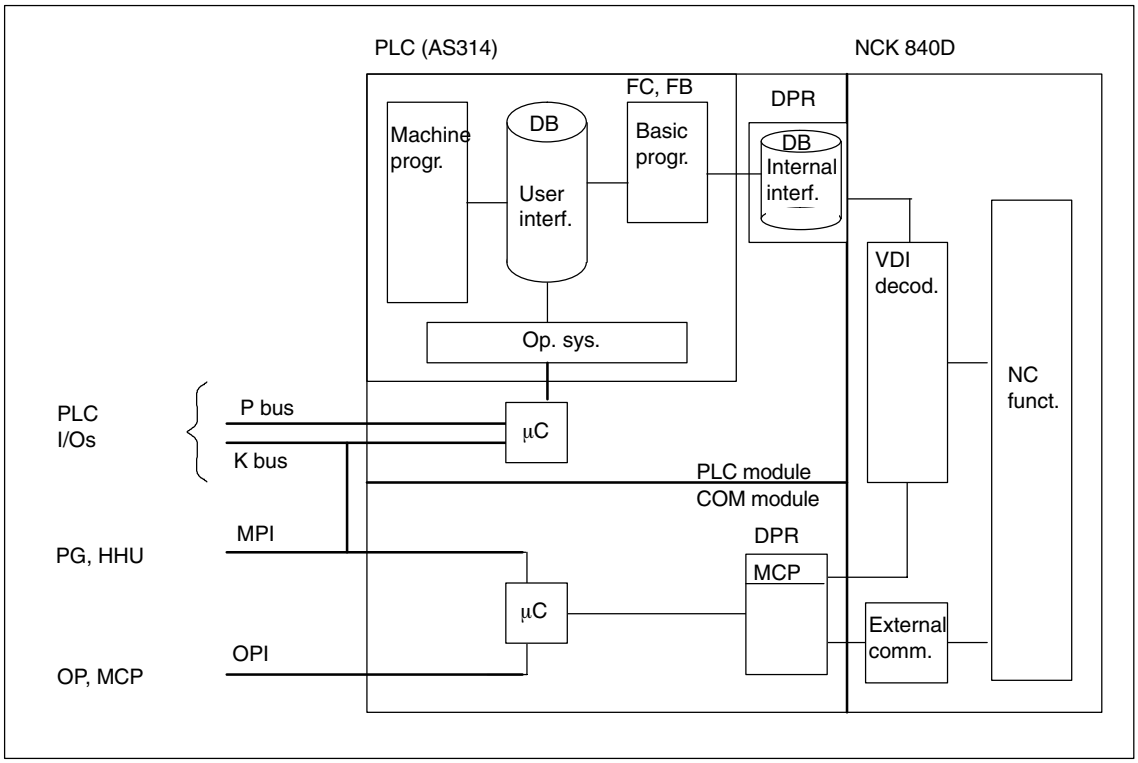


Fig. 2-2 NCK/PLC-coupling on 810D, 840D (integrated PLC)

1) OPI = Operator panel interface

PLC/NCK interface

The NCK/PLC data exchange (see Fig. 2-2) is organized by the basic program in the PLC. The **status information** (such as, for example, “Program running” stored in the internal DPR[1)] are copied to data blocks by the basic program at the beginning of the cycle (OB 1), which the user can then access (user interface). The user also enters NC **control signals** (e.g. NC start) in the interface data blocks, and these are also transferred to the NC at the start of the cycle.

Auxiliary functions are transferred to the PLC in dependence on the workpiece program. The basic program first evaluates these (alarm-driven) and then transfers them to the user interface at the start of OB1. If the relevant NC block contains auxiliary functions that require the interruption of the NC machining process (e.g. M06 for tool change), the basic program halts the execution of the block on the NC for one PLC cycle. The user can then use the “read disable” interface signal to halt the block execution until the tool change has been completed. If, on the other hand, the relevant NC block does not contain auxiliary functions requiring the interruption of the NC machining process (e.g. M08 for cooling on), the transfer of these “rapid” auxiliary functions is enabled directly in OB 40, so that the block execution is only marginally influenced by the transfer to the PLC.

© Siemens AG, 2003. All rights reserved

The evaluation and enabling of the **G functions** transferred from the NCK are also alarm-driven, however they are transferred directly to the user interface. Where a G function is evaluated at several points in the PLC program, differences in the information of the G function within one PLC cycle may arise.

In the case of **NC actions** which are triggered and assigned with parameters from the PLC (e.g. traverse concurrent axes), the triggering and parameter assignment is performed using FCs and FBs, not interface data blocks. The FCs and FBs belonging to the actions are supplied together with the basic program. The FCs must be loaded by the user and called in the PLC program of the machine manufacturer (machine program). For an overview of blocks FC, FB and DB, classed according to basic and extended functions, please refer to Section 6.4.

OP/PLC interface

Data are exchanged between the OP and PL via the OP/NC serial bus, COM module and C bus. The COM module transfers the data intact from one bus segment to another. It merely converts the baud rate. The OP is always the active partner (client) and the PLC is always the passive partner (server). Data transmitted or requested by the OP are read from and written to the OP/PLC interface area by the PLC operating system (timing: Cycle control point). From the viewpoint of the PLC application, the data are identical to I/O signals.

MCP/PLC interface HHU/PLC interface (840D only)

Data are exchanged between the MCP/PLC and HHU/PLC via the serial bus MCP, HHU/NC, COM module and NCK. The NCK transfers the MCP/HHU signals to and fetches them from the internal NC/NCK dual port RAM. On the PLC side, the basic program handles communication with the user interface. The parameters of the function call define the operand areas (e.g. I/O) and the start addresses.

---

1) DPR = DualPort RAM

MCP/PLC interface HHU/PLC interface (810D)

The data exchange between MCP/PLC, HHU/PLC takes place via the MPI interface on the PLC. The Communication with global data (GD)[1)] service is used for this purpose (see also STEP7 User's Guide). The PLC operating system handles the transfer of signals from and to the user interface. The STEP7 configuring tool **Communication configuration** is used to define both GD parameters as well as operand areas (e.g. I/O) and their initial addresses. In SW 2.2 and higher, data exchange is possible as on the 840D.

2.5.3 PLC interface on FM-NC

Physical interface

The FM-NC operating characteristics are the same as those of an AS-300 FM (function module) with C bus connection. Data can therefore be exchanged between the AS–300–CPU and FM-NC via the P bus and the C bus (Fig. 2-3).

© Siemens AG, 2003. All rights reserved

Communication with operator and machine control panels takes place via the MPI interface of the S7300 CPU. From an external viewpoint, the PLC CPU and the FM CPU are addressed directly.

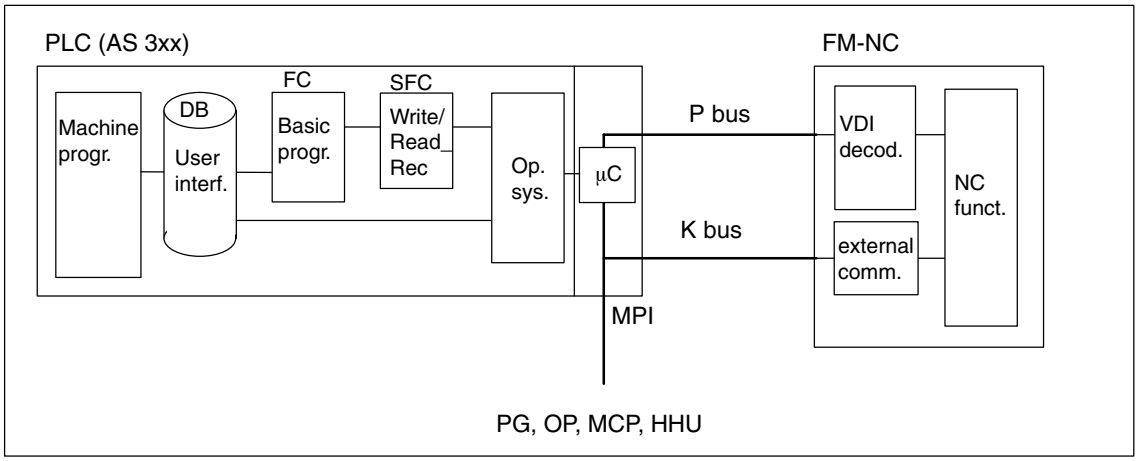


Fig. 2-3 NC/PLC-coupling on FM-NC

PLC/NCK interface

From the user's view, the interface to the NCK is identical to that on the 840D.

OP/PLC interface

Data are exchanged between the OP and PLC directly via the MPI interface on the PLC. As with the SINUMERIK 840D, the OP is always the active partner (client) and the PLC is always the passive partner (server). Data transmitted or requested by the OP are also read from and written to the OP/PLC interface area by the PLC operating system.

Interfaces MCP/PLC and HHU/PLC

Data are exchanged between the MCP/PLC and HHU/PLC via the MPI interface on the PLC. The Communication with global data (GD)[1)] service is used for this purpose (for details, see STEP7 User's Guide). The PLC operating system handles the transfer of signals from and to the user interface. The STEP configuring tool **Communication configuration** is used to define both GD parameters as well as operand areas (e.g. I/O) and their initial addresses.

1) IC (GD) = Implicit communication (global data)

2.5.4 Diagnostics buffer on PLC

General

The diagnostics buffer on the PLC, which can be read out using STEP7, contains diagnostic information about the PLC operating system. Data are also entered in the diagnostics buffer by the basic program and the Alarms/Messages function via FC10. A short guide to interpreting the alarms and messages generated by the basic program and stored in the diagnostics buffer is given below. A description text for these alarms/messages is entered in the diagnostic buffer.

© Siemens AG, 2003. All rights reserved

Alarms/ Messages

The event ID is the first item to be interpreted in the diagnostics buffer. The basic program generates the following alphanumeric combinations in the first two decades of the event ID:

A1 : Alarm set
A0 : Alarm deleted
B1 : Message set
B0 : Message deleted

The next two decades in the event ID corresponds to the two left-hand decades of the alarm/message number. This 2-decade number of the event ID must be converted from the displayed hex value to a decimal. Field info 1/2/3 must also be evaluated. In this case, only the lower 2 decades of info 1 and info 3 represent valid information about the message number. All the numbers in info 1/2/3 are hexadecimal numbers and must be converted to decimals before the message number can be compiled. The center decades are derived from info1 and the two right-hand decades from info3.
(e.g. event ID = B133, info 1/2/3 = 2900 0 9
represents a set message with number 510009).
The meaning of the message number is defined by the machine manufacturer. Only the two right-hand decades of the event ID with hex values 28, 29 indicate an error which has been generated by the basic program. The messages stored in the diagnostic buffer can be read out on the MMC with the associated message texts.

© Siemens AG, 2003. All rights reserved

2.6 Interface structure

Interface data blocks

The PLC user interfaces on the 840D, 810D and FM-NC are identical except for the data volume. Mapping in interface data blocks is required on account of the large number of signals. These are global data blocks from the viewpoint of the PLC program. During system start-up, the basic program creates these data blocks from current NC machine data (no. of channels, axes, etc.). The advantage of this approach is that the minimum amount of PLC RAM required for the current machine configuration is used.

2.6.1 PLC/NCK interface

General

The PLC/NCK interface comprises a data interface on one side and a function interface on the other. The data interface contains status and control signals, auxiliary functions and G functions, while the function interface is used to transfer jobs from the PLC to the NCK.

Data interface

The data interface is subdivided into the following groups:

- NCK-specific signals
- Mode group-specific signals
- Channel-specific signals
- Axis/spindle/drive-specific signals

© Siemens AG, 2003. All rights reserved

Function interface

The function interface is formed by FBs and FCs. Fig. 2-4 illustrates the general structure of the interface between the PLC and the NCK.

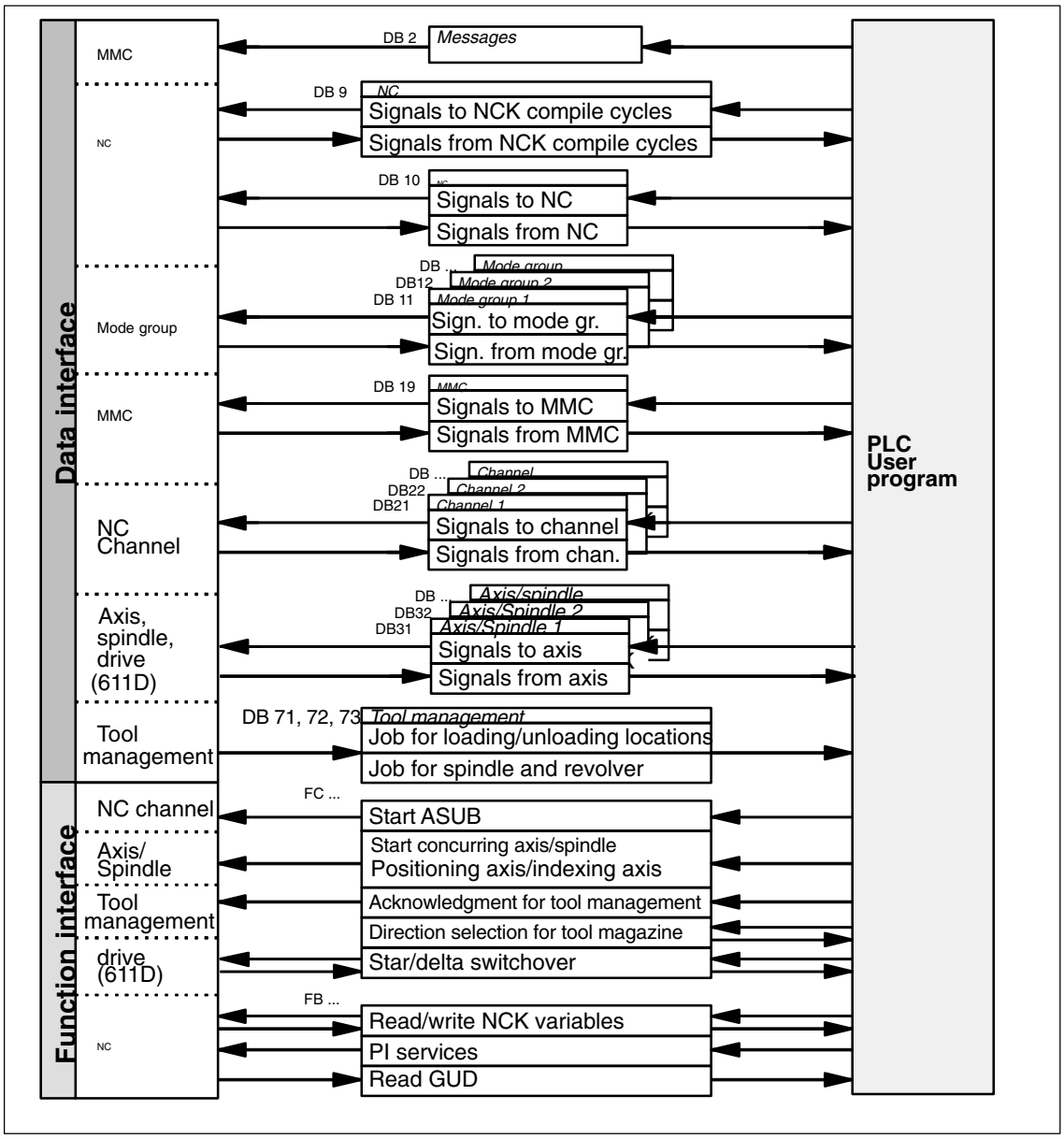


Fig. 2-4 PLC/NCK interface

Compile cycle signals

In addition to the standard signals exchanged between the PLC and NCK, an interface DB for compile cycles is also generated when required (DB 9). The signals, which are dependent on the compile cycles, are transmitted cyclically at the start of OB 1.

© Siemens AG, 2003. All rights reserved

Signals PLC/NC

The group of signals from the PLC to NC includes (Fig. 2-5):

- Signals for modifying the high-speed digital I/O signals of the NC
- Keyswitch and emergency stop signals

Signals NC/PLC

The group of signals from the NC to PLC includes:

- Actual values of the digital and analog I/O signals of the NC
- Ready and status signals of the NC

Also output in this group are the handwheel selection signals from the MMC and the status signals of the MMC.

The signals for handwheel selection are decoded by the basic program and entered in the machine/axis-specific interface.

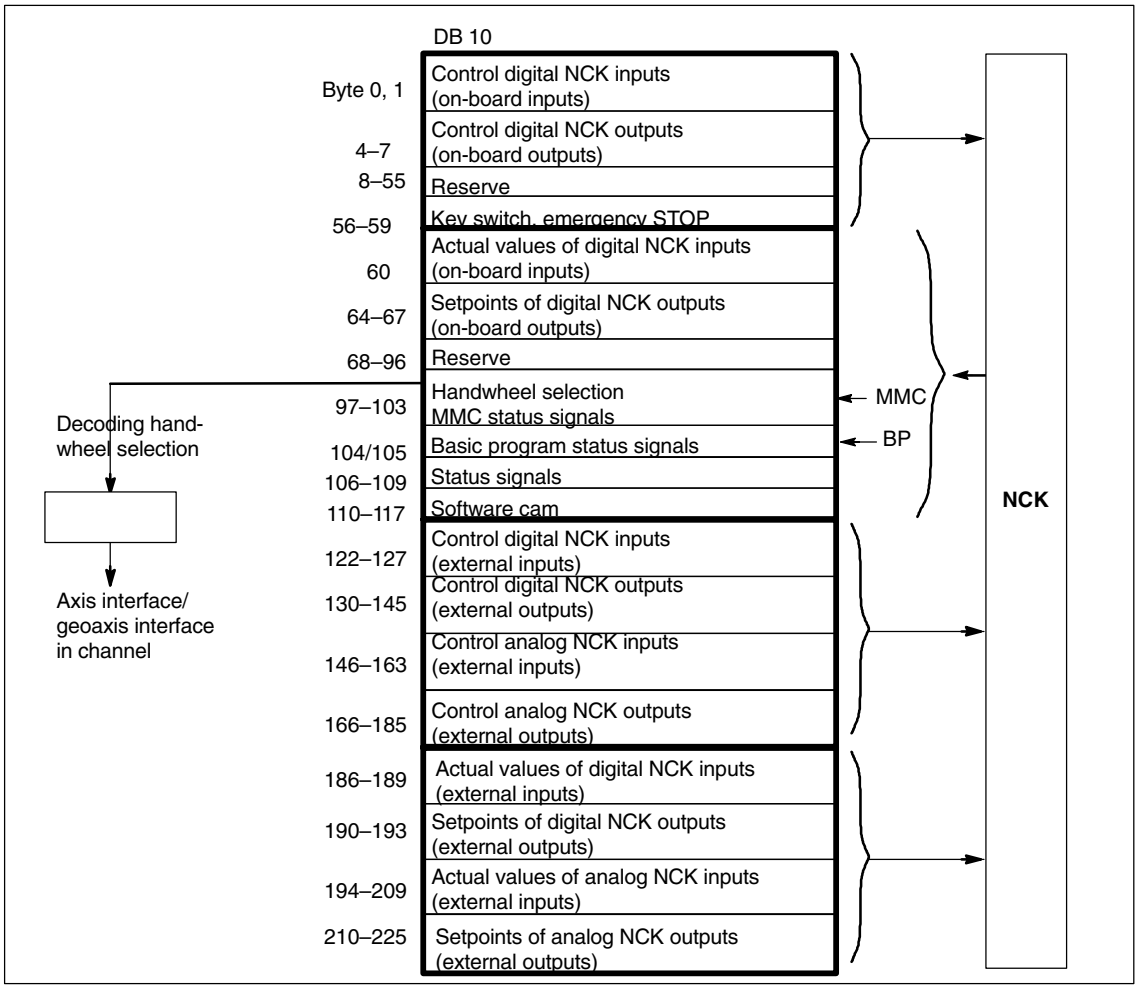


Fig. 2-5 PLC/NC interface

Digital/analog inputs/outputs of NCK

The following must be noted with respect to the digital and analog inputs and outputs of the NCK:

© Siemens AG, 2003. All rights reserved

Inputs:

- All input signals or input values of the NCK are also transferred to the PLC.
- The transfer of signals to the NC parts program can be suppressed by the PLC. Instead, a signal or value can be specified by the PLC.
- The PLC can also transfer a signal or value to the NCK even if there is no hardware for this channel on the NCK side.

Outputs:

- All signals or values to be output are also transferred to the PLC.
- The NCK can also transfer signals or values to the PLC even if there is no hardware for this channel on the NCK side.
- The values transferred by the NCK can be overwritten by the PLC.
- Signals and values from the PLC can also be output directly via the NCK I/O devices.

Note

When implementing digital and analog NCK I/Os, you must observe the information in the following references:

References: /FB/, A4, “Digital and Analog NCK I/Os”

Signals PLC/ mode group

The operating mode signals input from the machine control panel or the MMC are transferred to the operating mode group (mode group). The mode signals are valid for all NC channels of the mode group on the FM-NC, 810D and 840D. On 840D systems, several mode groups can optionally be defined in the NCK.

The mode group reports its current status to the PLC.

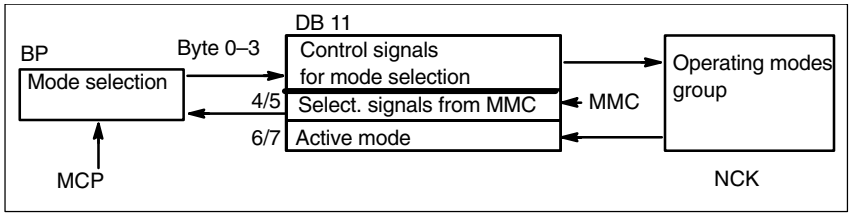


Fig. 2-6 PLC/mode group signals (e.g. mode group 1)

© Siemens AG, 2003. All rights reserved

Signals PLC/NCK channels

There are three groups of signals on the interface (see Fig. 2–6). These are:

- Control/status signals
- Auxiliary/G functions
- Tool management signals
- NCK functions.

The **control/status functions** are transmitted cyclically at the start of OB1. The signals[1] entered in the channel-specific interface from the MMC (the transfer of the MMC signals is performed by the PLC operating system) are also transferred at this time if the signals have been defined on the NC operator panel, not on the machine control panel.

The **auxiliary functions and G functions** are entered in the interface data blocks in two ways. First, they are entered with the change signals.

- The **M signals** M00–M99[2] (they are transferred from the NCK with extended address 0) are also decoded and the associated interface bits set for the duration of one cycle.
- In the case of the **G functions**, the group is additionally decoded and the G functions which are active in the relevant group are entered in the interface data block.
- The **S values** are additionally entered together with the related M signals (M03, M04, M05) in the spindle-specific interface. The axis-specific feedrates are also entered in the appropriate axis-specific interface.

When the **tool management** function is activated in the NCK, the assignment of spindle or revolver and the loading/unloading points are entered (DB71–73) in separate interface DBs.

The triggering and parameter assignment of **NCK functions** is performed by means of PLC function calls. The following function calls are available, for example:

- Position a linear axis or rotary axis,
- Position an indexing axis,
- Start a prepared asynchronous subprogram (ASUB),
- Read/write of NC variables,
- Update magazine assignment.

Part of the above functions are described in separate function documentation or in Chapter 4 of this documentation.

---

1) The MMC signals are entered via the operating system of the PLC
2) The M signals M0 to M99 are transferred by the NCK with the extended address 0.

© Siemens AG, 2003. All rights reserved

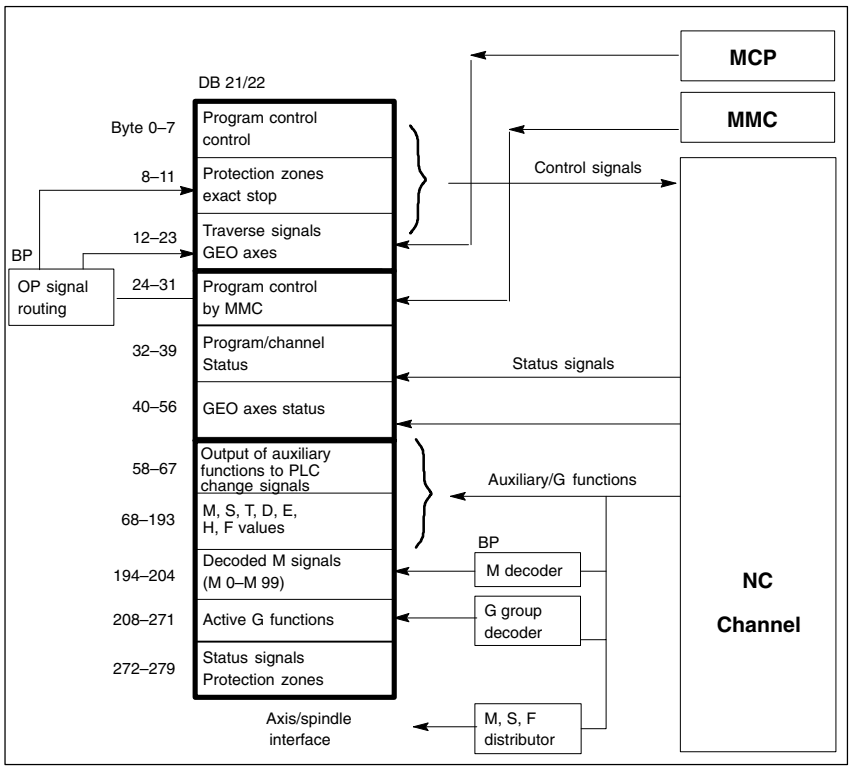


Fig. 2-7 PLC/NC channel interface

Signals PLC/axes, spindle, drive

The axis-specific and spindle-specific signals (Fig. 2-8, on following page) can be subdivided into the following groups:

- Shared axis/spindle signals
- Axis signals
- Spindle signals
- Drive signals

The signals are transferred cyclically at the start of OB 1, with the following exceptions: The exceptions include **INC mode of MMC, axial F value, M/S value.**

An **axial F value** is entered via the M, S, F distributor of the basic program if it is transferred to the PLC during the NC machining process.

The **M and S value** are also entered via the M, S, F distributor of the basic program if an S value requires processing together with the corresponding M value (M03, M04, M05).

© Siemens AG, 2003. All rights reserved

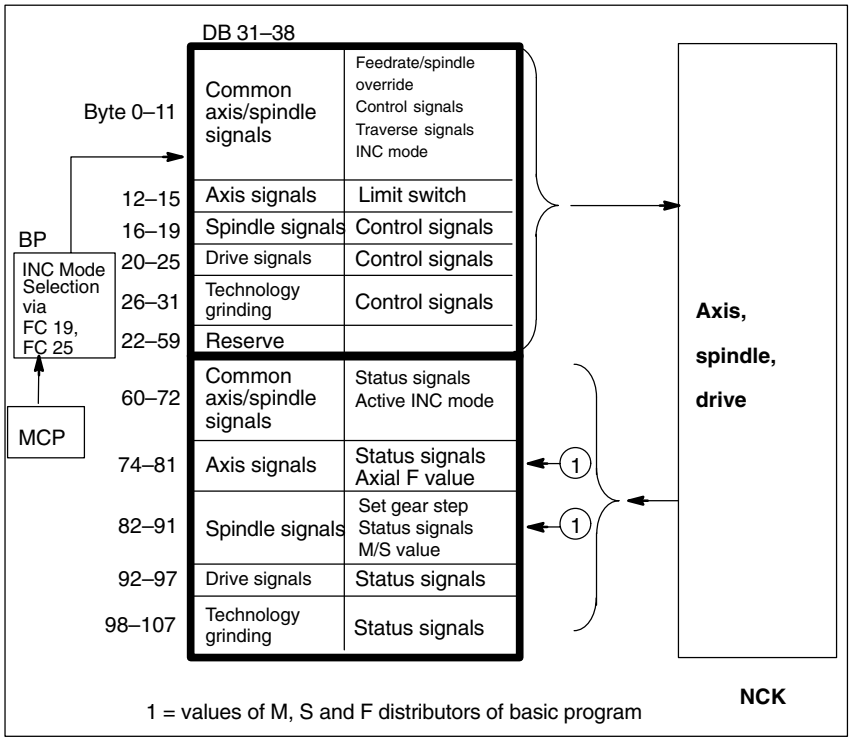


Fig. 2-8 Interface between PLC and axes/spindles/drives

2.6.2 PLC/MMC interface

General

The following groups of function are required for the PLC/MMC interface:

- Control signals
- Machine operation
- PLC messages
- PLC status display

Control signals

These are in some cases signals which are input via the machine control panel and which must be taken into account by the MMC. This group of signals includes, for example, display actual values in MCS or WCS, key disable, etc. These are exchanged with the MMC via a separate interface DB (DB19).

Machine operation

All operator inputs which lead to response actions on the machine are checked by the PLC. Operator actions are usually performed on the machine control panel. However, it is also possible to perform some operator actions on the MMC (e.g. mode selection, INC mode selection).

© Siemens AG, 2003. All rights reserved

The PLC operating system enters the operating signals arriving from the MMC straight into the interface data blocks (see Figs. 2-5, 2-6, 2-7, 2-8). In standard cases, the basic program, which decodes the operating signals, allows the operator actions to be performed **on the machine control panel (provided the operator actions are available) or on the MMC**. If required, the user can switch off the operation via MMC through a parameter “MMCToIF” of FB1.

PLC messages

The signaling functions are based on the system diagnostic functions integrated in the operating system of the AS 300. These have the following characteristics:

- The PLC operating system enters all important system states and state transitions in a **diagnostics status list**. Communication events and I/O module diagnostics data (for modules with diagnostic functions) are also entered.
- Diagnostics events which lead to a system stop are also entered with a time stamp in a **diagnostics buffer** (circular buffer) in the chronological order of their occurrence.
- The events entered in the diagnostics buffer are automatically transmitted to human machine interface (OS or MMC) via the MPI or via the OPI through the COM module, once these have issued a ready signal (message service). Transfer to the node ready is a function of the PLC operating system. Receipt and interpretation of the messages is executed by the MMC software.
- An SFC (system function call) can also be used to enter messages in the diagnostics buffer from the user program.
- The events are coded and entered in the diagnostics buffer. The message texts must be stored on the OP or MMC.

An FC (FC10) for message acquisition is prepared in conjunction with the basic program. This FB records the events, subdivides them into signal groups and reports them to the MMC via the diagnostics buffer.

The structure of message acquisition is outlined in Fig. 2-9. The features include the following:

© Siemens AG, 2003. All rights reserved

- Bit fields for events related to the VDI interface are combined in a single data block (DB2) with bit fields for user messages.
- The bit fields are evaluated at several levels by FC 10:
  - **Evaluation 1;** acquisition of **group signals**.
    A group signal is generated for each group of signals when at least one bit signal is set to "1". This signal is generally linked to the disable signal of the VDI interface (on modules with diagnostic functions). The group signals are acquired completely in cycles.
  - **Evaluation 2;** acquisition of **error messages**.
    It is predefined which signals of a group also generate an error message on changing from "0" to "1".
  - **Evaluation 3;** acquisition of **operating messages.**
    It is predefined which signals of a group generate an operating message.
- The scope of the user bit fields (user area) is defined as standard as 10 areas with 8 bytes each, but can also be adjusted to suit the requirements of the machine manufacturer via basic program parameters in FB 1.

Acknowledgment procedures

The following acknowledgment procedures are implemented for the error and operating messages:

Operating messages are intended for the display of normal operating states as information for the user. Acknowledgment signals are therefore not required for this type of message. An entry is made in the diagnostic buffer for incoming and outgoing messages. The MMC maintains an uptodate log of existing operating messages using the identifiers "operating message arrived" and "operating message departed".

Error messages are used to display error states on the machine which generally lead to a machine stoppage. Where several errors occur in rapid succession, it is important to be able to distinguish their order of occurrence for troubleshooting purposes. This is indicated, on the one hand, by the sequence of their entry in the diagnostics buffer and on the other, by the time stamp which every entry receives.

If the cause of the error disappears, the error message is only deleted if a the user has acknowledged the message (e.g. by pressing a key on the machine control panel). In response to this signal, the FC "message acquisition" examines which of the reported errors have disappeared and enters these in the diagnostics buffer with the entry "error departed". This enables the MMC to also maintain an up-to-date log of existing error messages. The time of day indicating the time of error occurrence remains available for messages which are still pending (in contrast to a received interrogation).

User program

The user PLC program merely needs to call the basic program block FC 10 with appropriate parameter settings in the cyclic program section (see Chapter 4) and to set or reset the bit fields in DB2. All further necessary measures are implemented by the basic program and MMC.

© Siemens AG, 2003. All rights reserved

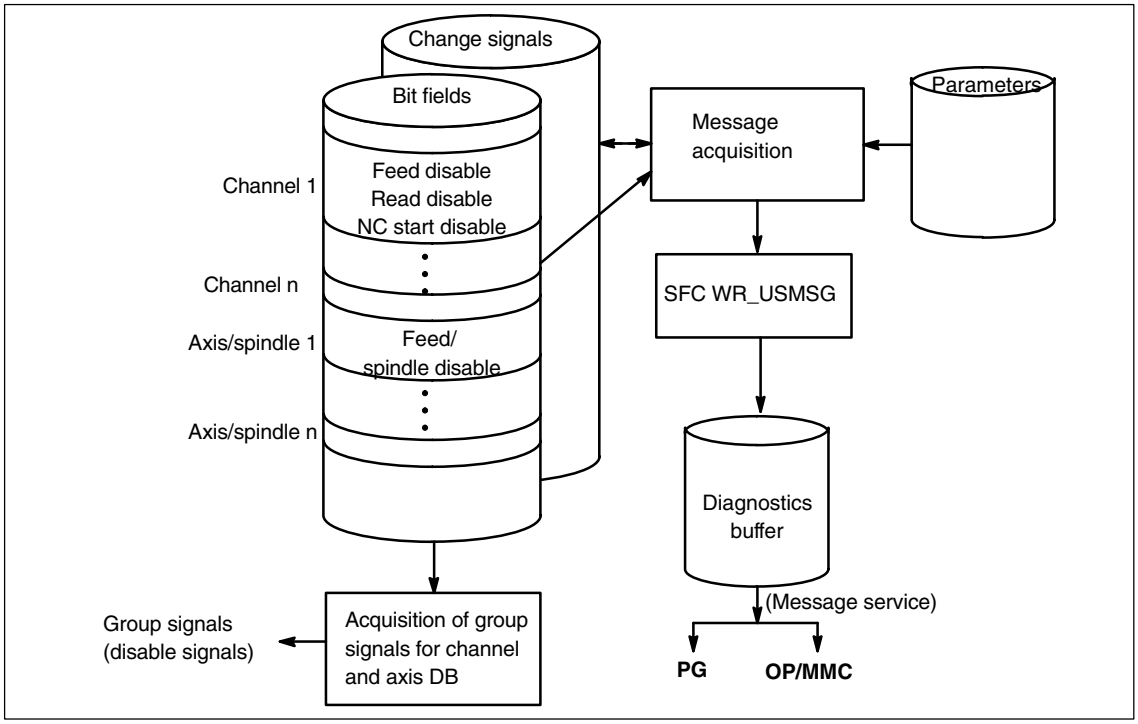


Fig. 2-9 Acquisition and signaling of PLC events

2.6.3 PLC/MCP/HHU interface

General

On the SINUMERIK 840D/810D and FM-NC, the machine control panel (MCP) is connected on the same bus that connects the OP with the NC. The advantage of this is that only one bus cable is required to connect the operator unit. The handheld unit (HHU) can be connected to the MPI of the PLC or to the operator panel interface (OPI) (840D only). However, since the OP bus on the 840D supports higher baud rates, two different types of bus topology are provided.

Topology 840D

On the 840D, the machine control panel is connected to the OPI bus segment (transmission rate 1.5 Mbaud) as an active node (Fig. 2–10). Where the connection of further keys and displays is required for customized operator panels, an additional keyboard (machine control panel without operating unit) can be used. 64 pushbuttons, switches, etc. and 64 display elements can be connected via ribbon cable.

The signals arriving from the machine control panel are copied by the COM module into the DPR (dual port RAM) for transfer to the NC. The NC in turn transmits them to the PLC (VDI task). The basic program of the PLC enters the incoming signals in the input image. The NC-related signals are generally distributed by the basic program to the VDI interface. This can be modified by the user if required.

The signals (displays) from the PLC to the machine control panel (MCP) are transferred in the opposite direction.

© Siemens AG, 2003. All rights reserved

The signals of the handheld unit (HHU) are transferred either via the operator panel interface in the same way as via the machine control panel or by means of the GD service (GD = Global Data) of the MPI interface. The PLC operating system enters the HHU data, for example, in the input image and transfers the display values, e.g. from the output image, back to the HHU. The corresponding parameters are set via the system data block SDB 210 which is generated with the STEP7 tool Communication configuration.

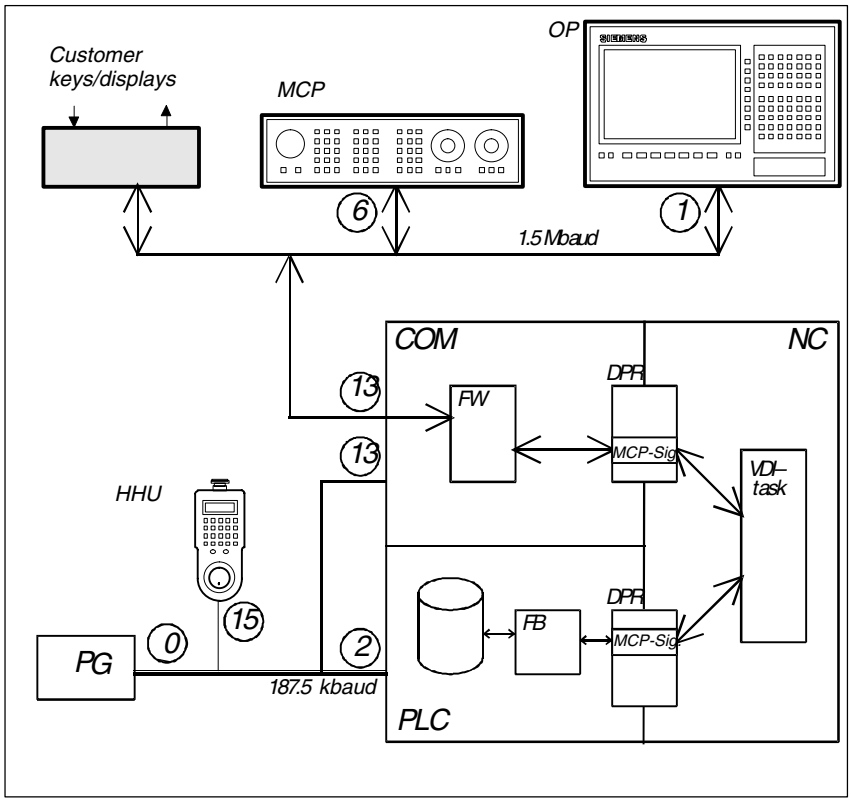


Fig. 2-10 Coupling of machine control panel on FM-NC

Topology FM–NC, 810D

On the FM-NC and 810D, the machine control panel, handheld unit and OP are connected to the MPI (multipoint Interface) of the AS 300 (Fig. 2-10). The transmission rate in this configuration amounts to 187.5kbaud. The PLC operating system copies the incoming signals straight to the user interface (e.g. input image) at the cycle control point. Transfer to the VDI interface is performed, as on the 840D, by the user program or by a standard routine of the basic program.

© Siemens AG, 2003. All rights reserved

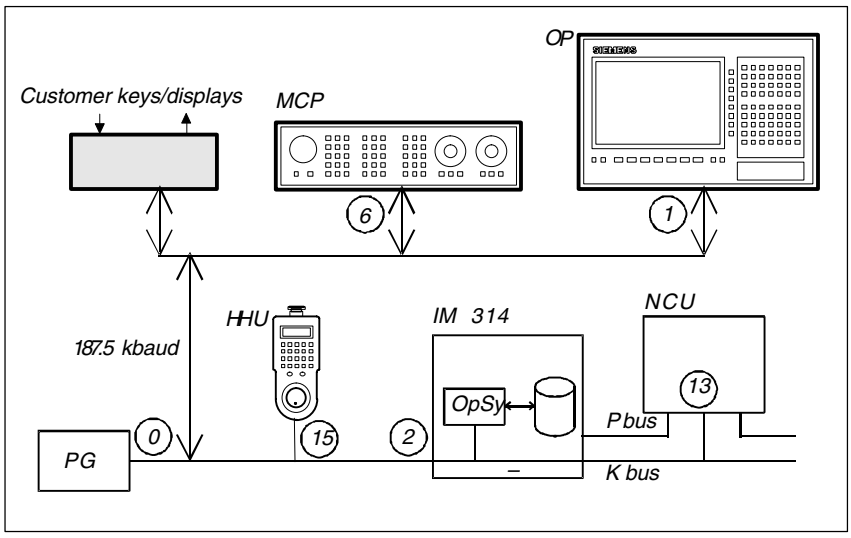


Fig. 2-11 Coupling of machine control panel on FM-NC

Bus addresses

The default bus addresses for the standard configurations are entered in Figs. 2-10 and 2-11. In addition to the bus addresses, the implicit communication service (global data) also requires the definition of a GD circle number.

The following should be taken into account when allocating the bus addresses (node no.):

Bus addresses 840D

The two bus segments on the 840D must be examined separately:

Table 2-2 **Operator panel bus segment:**

Bus station	Perm. setting range	Standard setting
Operator panel (OP)	MMC 100: 1 – 31 MMC 101–103: 1 – 31	1
Machine control panel/ keyboard interface	0–15	6 (Setting via DIP fix)
COM module	1 – 31	13
Programming device/PC (e.g. for start-up)	Permanent	0

Table 2-3 **PLC bus segment:**

Bus station	Setting range	Default setting	Note
PLC	1–31	2	
COM module	Fixed depending on PLC address	3	
Programming device/PC (e.g. for startup)	Permanent	0	
FMs on the K bus	1–31	CPU address + slot no.	Setting possible using S7 Configuration

© Siemens AG, 2003. All rights reserved

Bus addresses FM-NC

During start-up, the FMs with K bus connection are determined by the AS CPU and default Cbus addresses then assigned (node NO. FM= Addr. AS CPU + current no.). The FMs are then informed of these addresses via the P bus. The S7-Config configuration tool can be used to assign an individual node number to the AS 314.

Note

On the FM-NC, the K bus address in product versions 1 and 2 is set permanently to 13.
The hsa bus parameter (highest station address) is derived from the highest station address defined in the default setting or during configuration. This can only be allocated in increments of 15, 31, 653 and 126.
S7-config. derives ttr[2)] from parameter hsa. Since the setting options on the machine control panel are limited, the setting for ttr[1)] is fixed on the MCP. Since the ttr must be the same for all bus stations, extreme caution should be exercised that **none of the bus stations has an address > 15**.

---

1) Can be set with SW 3.2 and higher
2) ttr = permissible target rotation time (ttr = (hsa + 8) * 256 * bit time)

© Siemens AG, 2003. All rights reserved

The permissible bus addresses for the FM-NC are shown in the following table:

Bus station	Perm. setting range	Standard setting
Operator panel	1–15	1
Machine control panel/key-board interface	0–15	Setting via DIP fix
COM module	Fixed depending on PLC address	3
Programming device/PC (e.g. for startup)	Permanent	0

MCP interface in the PLC

The signals from the machine control panel are routed as standard via the I/O area (Fig. 2-12). A distinction is made between NC and machine-specific signals. NC-specific key signals are normally distributed by FC 19 to the various mode group, NCK, axis and spindle-specific interfaces. The reverse applies to the status signals, which are routed to the machine control panel interface.

Note

The FC 19 must be called in the PLC user program.

Customized keys, which can be used to trigger a wide range of machine functions, must be evaluated directly by the user program. The user program also routes the status signals to the output area for the LEDs.

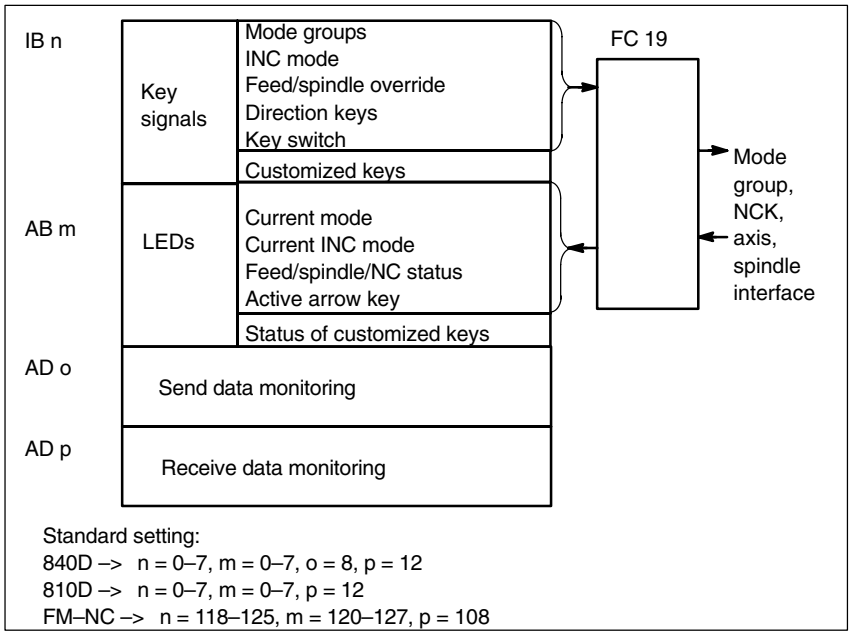


Fig. 2-12 Interface to and from machine control panel

© Siemens AG, 2003. All rights reserved

2.7 Structure and functions of the basic program

General

The program is modular in design, i.e. it is structured according to NCK functions (see Fig. 2-13). In the operating system, a distinction is made between the following levels of execution:

- Start-up and synchronization (OB 100)
- Cyclical mode (OB 1)
- Process interrupt handling (OB 40)

Each section of the basic program – as illustrated in Fig. 2-13 – must be called by the user in OBs 1, 40 and 100.

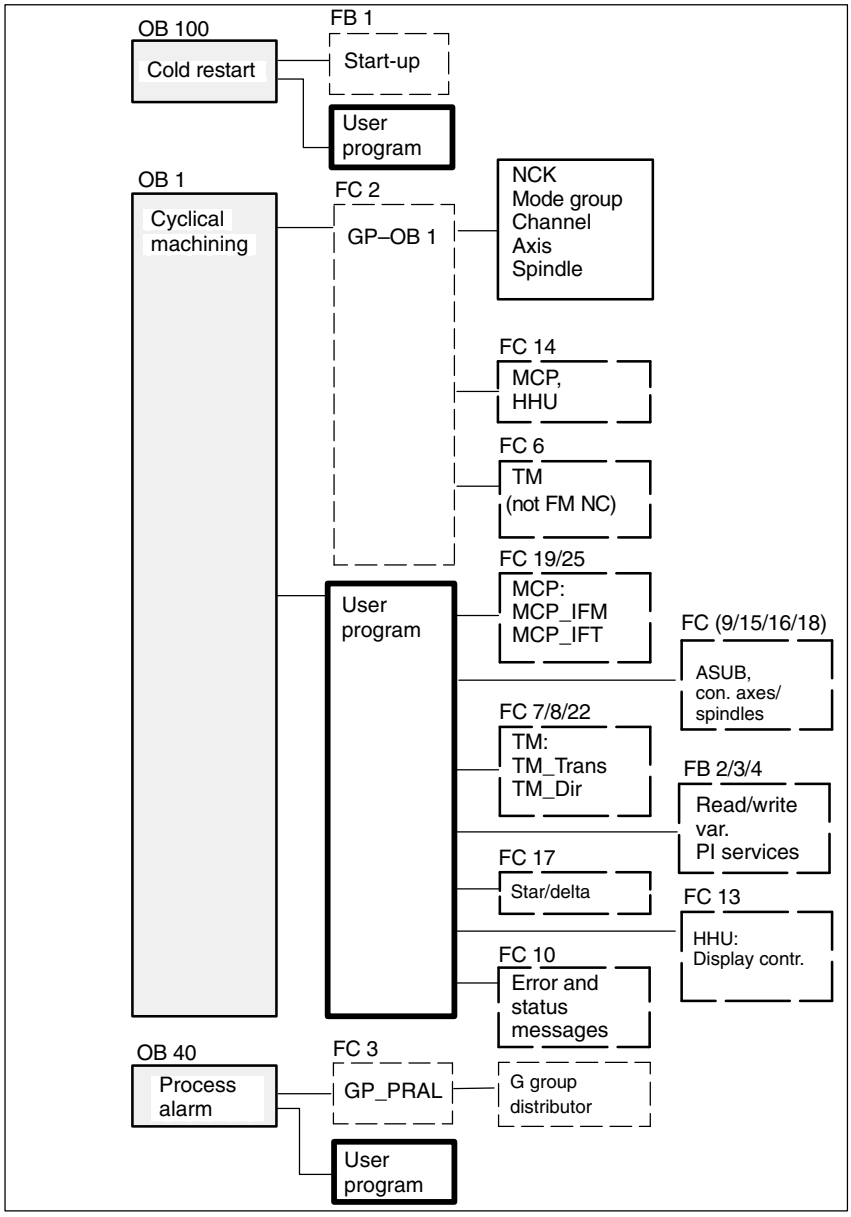


Fig. 2-13 Structure of PLC program

© Siemens AG, 2003. All rights reserved

2.7.1 Start-up and synchronization of NCK PLC

Loading the basic program

The basic program must be loaded with the S7 tool when the PLC is in the Stop state. This guarantees that all blocks in the basic program will be correctly initiated when they are next called. An undefined state may otherwise develop on the PLC (e.g. all PLC LEDs flashing).

Startup

The synchronization of NCK and PLC is performed during start-up. The system and user data blocks are checked for integrity and the most important basic program parameters are verified for plausibility. In cases of error, the basic program outputs an error identifier to the diagnostics buffer and switches the PLC to STOP.

A warm restart is not provided, i.e. following system initialization, the operating system runs organization block OB 100 and always commences cyclical execution at the start of OB 1.

Synchronization

The PLC is synchronized with the MMC and NCK during power-up.

Sign of life

Following an orderly start-up and the first complete OB1 cycle (basic setting cycle), the PLC and NCK continually exchange sign-of-life characters. If the signoflife signal from the NCK fails to arrive, the PLC/NCK interface is neutralized and the signal “NCK CPU ready” in DB 10 is set to zero (see Subsection 2.7.5).

2.7.2 Cyclic operation

General

The NCK-PLC interface is processed completely in cyclical mode. From a chronological viewpoint, the basic program runs ahead of the user program. In order to minimize the execution time of the basic program, only the control/status signals are transmitted cyclically; transfer of the auxiliary functions and G functions only takes place on request.

The following functions are performed in the cyclical part of the basic program:

- Transmission of the control/status signals
- Distribution of the auxiliary and G functions
- M decoding (M00–99)
- M, S, F distribution
- Transmission of the machine control panel signals via the NCK (on the 840D only)
- Acquisition and conditioning of the user errors and operating messages.

© Siemens AG, 2003. All rights reserved

Control/ status signals

A shared feature of the control and status signals is that they are bit fields. The basic program updates them at the start of OB 1.

The signals can be subdivided into the following groups:

- General signals
- Mode group-specific signals such as operating modes
- Channel-specific signals such as program and feed modifications
- Axis and spindle-specific signals such as feed disable

Auxiliary and G functions

The auxiliary and G functions have the following characteristics:

- Transfer to the PLC is block-synchronous (referred to a parts program block)
- Transfer is acknowledge-controlled
- The acknowledgment times have an immediate effect on the execution time of NC blocks containing auxiliary functions requiring acknowledgment.

The value range is presented in the following table:

Function	Structure		Value range		Data type	
	1st Value	2nd Value	1st Value	2nd Value	1st Value	2nd Value
G function		G function		0–255[1)]		Byte
M word	M group	M word	0–99	0–99.999.999	Word	DWord
S word	Spindle no.	S word	0–6	Floating point[2)]	Word	DWord
T word	Magazine no.	T word	0–99	0–65535	Word	Word
D word	–	D word	0–99	0–255	Byte	Byte
H word	H group	H word	0–99	Floating point	Word	DWord
F word	Axis no.	F word	0–18	Floating point	Word	DWord

The M, S, T, H, D and F values are passed from the NCK to the auxiliary/G function distributor, which outputs them together with the accompanying change signals to **CHANNEL DB** interface (see documentation “Lists of SINUMERIK 840D, 810D, FM-NC”). The function value and the extended address are transferred to the appropriate data word. The accompanying modification signal is activated to 1 for one PLC cycle. When the modification signal is reset, the acknowledgment is passed to the NCK. The acknowledgment of high-speed auxiliary functions is given by the basic program immediately the basic program detects the auxiliary function.

In addition to distribution of the auxiliary and G functions, selected signals are processed as described in the following paragraphs.

M decoder

M functions can be used to pass both switching commands and fixed point values. Decoded dynamic signals are output to the **CHANNEL DB** interface for the standard M functions (M00 to M99); the signal length = 1 cycle time.

---

1) Relative number passed for each G group
2) In STEP7 format (24 bit mantissa, 8 bit exponent).

© Siemens AG, 2003. All rights reserved

G group decoder

In the case of the G functions passed from the NCK, the groups are decided and each G number is entered in the corresponding interface byte of the **CHANNEL DB**, i.e. all active G functions are entered in the channel DBs. The entered G functions are retained even after the NC program has terminated or aborted.

Note

During system start-up, all of the G group bytes are initialized with the value “0”.

M, S, F distributor

The M, S, F, distributor is used to enter spindle-specific M words M(1 ...6)=[3,4,5], S words and F words for axial feeds in the appropriate **spindle and axis data blocks**. The criterion for distribution is the extended address, which is passed to the PLC for M words, S words and axial F words.

MCP signal transmission

On the 840D, the MCP signals are transferred to the NCK via the serial bus (MPI) and from there to the PLC. A function call from the basic program transfers the signals to the interface of inputs and outputs specified by basic program parameters. The status signals for controlling the LEDs on the machine control panel are passed in the opposite direction.

User messages

The system has already been described in Subsection 2.3.2. The acquisition and processing of the user error and operational messages is performed by an FC in the basic program.

2.7.3 Time alarm processing (OB 35)

General

The user must program **OB 35** for time-of-day alarm processing. The default time base setting of **OB 35** is 100ms. Another time base can be selected using the STEP7 application “S7 Configuration”. However, the OB 35 with a time base setting of less than approx. 15 ms must not be used without additional measures, since otherwise this would cause the CPU of the PLC to stop. The stop is caused by reading of the MMC system state list during run-up of the MMC. This reading process blocks the priority class control for approx. 8 to 12 ms. The OB 35 with a time base set to a rather lower value is then no longer processed correctly. If, however, small time base settings are required for OB 35, the stop can be prevented by programming OB 80 with at least the program command “BE”.

2.7.4 Process alarm processing (OB40)

General

A process alarm **OB40** (interrupt) can, for example, be triggered by appropriate configure I/Os or by certain NC functions (see also Section 4.12 Description of FC 3). Due to the different origin of the interrupt, the PLC user program must first interpret the cause of the interrupt in OB 40. The cause of the interrupt is included in the local data of OB 40 (see also SIMATIC STEP7 Description or STEP7 Online Help).

© Siemens AG, 2003. All rights reserved

2.7.5 Response to NC failure

General

During cyclical operation, the PLC continuously monitors the NC availability by querying the sign-of-life character. If the NCK is no longer reacting, then the NCK PLC interface is neutralized and IS **NCK CPU ready** in the **signals from NC group (DB 10.DBX 104.7)** is reset. The signals which are passed from the NCK to the PLC are initialized with default settings (as described below).

The PLC itself remains active so that it can continue to control machine functions.

Signals NCK → PLC

The signals which are transferred from the NCK to PLC are divided into the following groups:

- Status signals from the NCK, channels, axes and spindles
- Modification signals of the auxiliary functions
- Values of the auxiliary functions
- Values of the G functions

Status signals:
The status signals from the NCK, channels, axes and spindles are reset.

Change signals of the auxiliary functions:
The change signals of the auxiliary functions are also reset.

Values of the auxiliary functions:
The values of the auxiliary functions are retained so that it is possible to trace which were the last functions to be triggered by the NCK.

Values of the G functions:
The values of the G functions are reset (i.e. initialized with the value 0).

Signals PLC → NCK

The signals which are sent from the PLC to the NCK are divided into control signals and tasks that are transferred by FCs to the NCK.

Control signals:
The control signals from the PLC to the NCK are frozen; cyclical updating by the basic program is suspended.

Jobs from the PLC to the NCK:
The FCs and FBs, which are used to pass jobs to the NCK must no longer be processed by the PLC user program, as this could possibly lead to incorrect checkback signals. During run-up of the control, a job (e.g. reading NCK data) must not be activated in the user program until the **NCK CPU ready** is set.

© Siemens AG, 2003. All rights reserved

2.7.6 Functions of basic program called from user-program

General

In addition to the modules of the basic program, which are called at the start of OBs 1, 40 and 100, functions are also provided which can be called at a suitable point in the user program and supplied with parameters. These functions can be used, for example, to pass the following jobs from the PLC to the NCK:

- Traverse concurrent axes (FC 15, FC 16),
- Start asynchronous subprograms (ASUBs) (FC 9),
- Select NC programs and NC blocks (FB 4),
- Control of spindle (FC 18),
- Read/write variables (FB 2, FB 3).

Note

The following note will later help you to check and diagnose a function call (FCs, FBs of basic program). These are FCs and FBs which are controlled by a trigger signal (e.g. via parameter Req, Start, ...) and which supply an execution acknowledgment as an output parameter (e.g. via parameter Done, NDR, Error, ...). A variable compiled of other signals which produce the trigger for the function call should be set. Start conditions may be reset only as a function of the states of parameters Done, NDR and Error. This control mechanism may be positioned in front of or behind the function call. If the mechanism is placed after the call, the output variables can be defined as local variables (advantage: Reduction of global variables, flags, data variables and time-related advantages over data variables). The trigger parameter must be a global variable (e.g. flag, data variable).
Jobs that have been activated by the user program (Parameter Req, Start, ... := TRUE) must be reset to zero for the parameters named in OB 100. A POWER OFF/ON could result in a state in which jobs are still active.

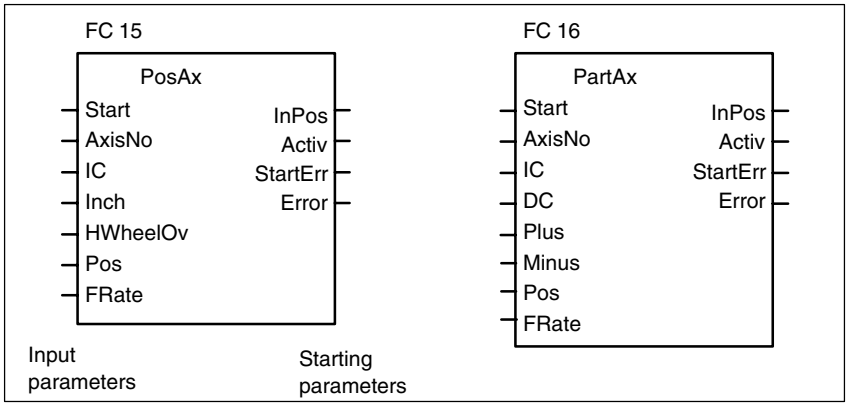
Concurrent axes

The distinguishing features of concurrent axes are as follows:

- They can be traversed either from the PLC or from the NC.
- They can be started from a function call on the PLC in all operating modes.
- The start is independent of NC block boundaries.

Function calls are available for positioning (FC 15) and indexing axes (FC 16).

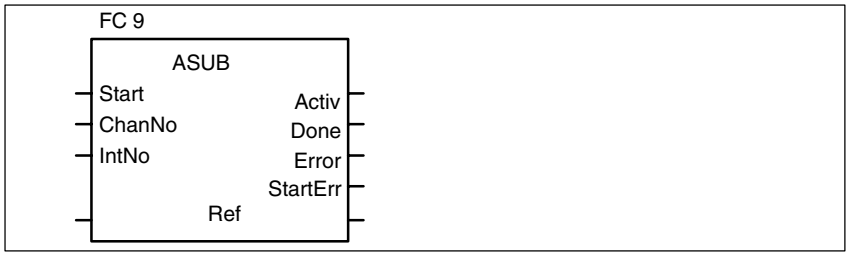
© Siemens AG, 2003. All rights reserved



ASUBs

Asynchronous subprograms (ASUBs) can be used to activate any selected function in the NC. Before an asynchronous subprogram can be started from the PLC, it must be ensured that it is available and prepared by the NC program or by FB 4 PI services (ASUB). ASUBs can only be started in MDA or Automatic mode with **running** parts program.

Once prepared in this way, it can be started at any time from the PLC. The NC program running on the channel in question is interrupted by the asynchronous subprogram. The asynchronous subprogram is started by FC 9.



Note

If an asynchronous subprogram has not been prepared by an NC program or by FB 4 (ASUB) (e.g. if no interrupt no. has been assigned), a start error is output.

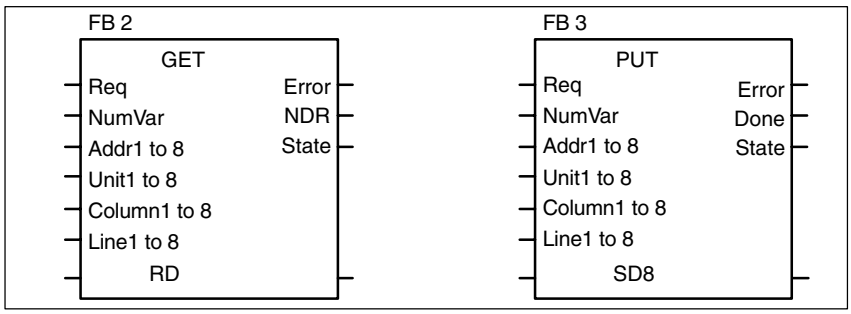
ReadNC variables

Variables of the NCK can be read with FB GET while values can be entered in NCK variables with FB PUT. The NCK variables are addressed via identifiers at inputs Addr1 to Addr8. The identifiers (symbols) point to address data which must be stored in a global DB. To allow generation of this DB, PC software is supplied with the basic program with which the desired variables can be selected from a table which is also supplied. The selected variables are first collected in a second, project-related list. Command **Generate DB** creates an *.STL file which must be linked to the program file for the machine concerned and compiled together with the machine program (see also Section 3.2).

© Siemens AG, 2003. All rights reserved

1 to 8 values can be read or written with a read or write job. If necessary, the values are converted (e.g. the NCK floating point values (64 bits) are converted to the PLC format (32 bits with 24 bit mantissa and 8 bit exponent) and vice versa). A loss of accuracy results from the conversion from 64 bits to 32 bit REAL. The maximum precision of 32 bit REAL numbers is approximately 10 to the power of 7.

FB 2 and FB 3 are described in Chapter 4 of this documentation.



2.7.7 Symbolic programming of user program with interface DB

General

Note

With basic program SW 3.2 and higher, files NST_UDT.STL and TM_UDT.AWL are stored on the basic program diskette supplied with the system.

The compiled UDT blocks from these two files are stored in the CPU program of the basic program.

A UDT is a data type defined by the user that can, for example, be assigned to a data block generated in the CPU.

Symbolic names of virtually all the interface signals are defined in these UDT blocks.

The UDT numbers 2, 10, 11, 19, 21, 31, 71, 72, 73 are used.

The assignments have been made as follows:

Table 2-4 UDT assignments

UDT number	Assignment to interface DB	Meaning
UDT 2	DB 2	Alarms/Messages
UDT 10	DB 10	NCK signals
UDT 11	DB 11	Mode group signals
UDT 19	DB 19	MMC signals
UDT 21	DB 21 to DB 30	Channel signal

© Siemens AG, 2003. All rights reserved

Table 2-4 UDT assignments

UDT number	Assignment to interface DB	Meaning
UDT 31	DB 31 to DB 61	Axis/spindle signals
UDT 71	DB 71	Tool management: Load/unload
UDT 72	DB 72	Tool management: Change in spindle
UDT73	DB 73	Tool management: Change in revolver

In order to program the interface signals as symbols, the data blocks of the interface must first be assigned symbolically with the symbol editor.

For example, symbol “AxisX” is assigned to operand DB31 with data type UDT 31 in the symbol file.

After this input, the STEP7 program can be programmed in symbols for this interface.

Note

Programs generated with an earlier software version that utilize the interface DBs described above can also be converted into symbol programs.

To do so, however, a fully qualified instruction is needed for data accessing in the earlier program (e.g. “U DB31.DBX 60.0” – this command is converted to “AxisX.E_SpKA” when the symbolics function is activated in the editor).

Description

Abbreviated symbolic names of the interface signals are defined in the two STL files NST_UDT.AWL and TM_UDT.AWL.

In order to create the reference to the names of the interface signals, the name is included in the comment after each signal.

The symbolic names, commands and absolute addresses can be viewed by means of a STEP7 editor command when the UDT block is opened.

Note

Unused bits and bytes are listed, for example, with the designation

“f56_3”.

“56” Specifies the relevant byte address of the relevant data block.

“3” This number corresponds to the bit number in this byte.

© Siemens AG, 2003. All rights reserved

English versions of the UDT interfaces are available in SW 4.4 and higher under NST_UDTB.AWL and TM_UDTB.AWL. Only the English versions will continue to be developed in future versions. The following procedure is necessary in order to convert the previously used German symbols into the English language:
If you are working without sources, you only need to compile the NST_UDTB.AWL and TM_UDTB.AWL blocks. The new symbols are then immediately visible.
If you are using sources, the sources must first be compiled with the previous UDTs (NST_UDT.AWL and TM_UDT.AWL). It is then necessary to compile NST_UDTB.AWL and TM_UDTB.AWL. After this step, you need to initiate a reverse compilation into the sources in accordance with the previous source setup. The interface symbols are then permanently converted to English.

2.7.8 M decoding acc. to list

Description of functions

When the **M decoding according to list** function is activated via the BP parameter of FB1 “ListMDecGrp”, up to 256 M functions with extended address can be decoded by the basic program.

The assignment between the M function with extended address and the bit to be set in the signal list is defined in the decoding list. The signals are grouped for this purpose.

The signal list contains 16 groups with 16 bits each as decoded signals.

There is only one decoding list and one signal list, i.e. this is a cross-channel function.

The M functions are decoded. Once they are entered in the decoding list, then the associated bit in the signal list is set.

When the bit is set in the signal list, the read-in disable in the associated NC channel is set simultaneously by the basic program.

The read-in disable in the channel is reset once the user has reset all the bits output by this channel and thus acknowledged them.

The output of an M function decoded in the list as a high-speed auxiliary function does not result in a read-in disable.

In the following, Fig. 2-14 shows the structure of the **M decoding according to list** function.

© Siemens AG, 2003. All rights reserved

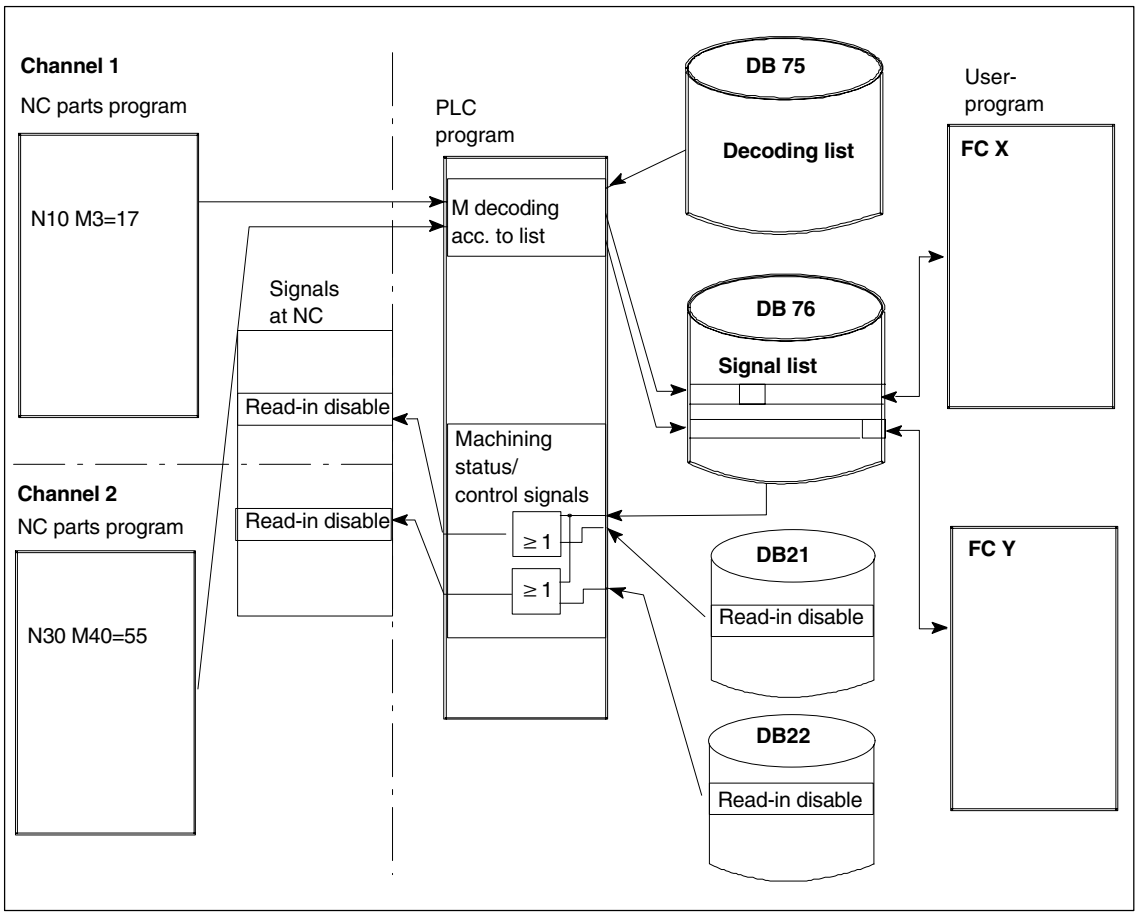


Fig. 2-14 M decoding acc. to list

Structure of the decoding list

There must be an entry in decoding list DB 75 for every group of M functions to be decoded.

A maximum of 16 groups can be created.

16 bits are available in each group in the list of decoded signals.

The assignment between the M function with extended address and the bit to be set in the signal list is specified via the first and last M functions in the decoding list.

The bit address is generated correspondingly from the first M function ("MFirstAdr") to the last M function ("MLastAdr") from bit 0 up to maximum bit 15 for each group.

Each entry in the decoding lists consists of 3 parameters, each of which are assigned to a group.

The source file for the decoding list (MDECLIST:AWL) is supplied with the basic program.

© Siemens AG, 2003. All rights reserved

Table 2-5 Assignment of groups

Group	Extended M address	First M address in group	Last M address in group
1	MSigGrp[1].MExtAdr	MSigGrp[1].MFirstAdr	MSigGrp[1].MLastAdr
2	MSigGrp[2].MExtAdr	MSigGrp[2].MFirstAdr	MSigGrp[2].MLastAdr
...	...	...	...
16	MSigGrp[16].MExtAdr	MSigGrp[16].MFirstAdr	MSigGrp[16].MLastAdr

Table 2-6 Type and value range for signals

Signal	Type	Value range	Notes
MExtAdr	Int	0 to 99	Extended M address
MFirstAdr	DInt	0 to 99,999,999	First M address in group
MLastAdr	Dint	0 to 99,999,999	Last M address in group

Example

```
DATA_BLOCK DB 75
TITLE =
VERSION: 0.0
  STRUCT
  MSigGrp : ARRAY  [1 .. 16 ] OF STRUCT
    MExtAdr :          INT ;
    MFirstAdr :        DINT ;
    MLastAdr :         DINT ;
    END_STRUCT;
  END_STRUCT ;
BEGIN
  MSigGrp[1].MExtAdr :=          0;
  MSigGrp[1].MFirstAdr :=        L#0;
  MSigGrp[1].MLastAdr :=         L#0;
  MSigGrp[2].MExtAdr :=          0;
  MSigGrp[2].MFirstAdr :=        L#0;
  MSigGrp[2].MLastAdr :=         L#0;


  MSigGrp[15].MExtAdr :=         0;
  MSigGrp[15].MFirstAdr :=       L#0;
  MSigGrp[15].MLastAdr :=        L#0;
  MSigGrp[16].MExtAdr :=         0;
  MSigGrp[16].MFirstAdr :=       L#0;
  MSigGrp[16].MLastAdr :=        L#0;
END_DATA_BLOCK
```

Signal list

Data block DB 76 is set up when the function is activated.

A bit is set in the appropriate group in DB 76 for an M signal decoded in the list.

© Siemens AG, 2003. All rights reserved

At the same time, a read-in disable is set (as described above) in the channel in which the M function has been output.

Activation of function

The number of groups to be evaluated is specified in basic program parameter ListMGDecGrp (see also description of FB1).

The decoding function is activated if this value is between 1 and 16.

Before activation of decoding, the decoding list DB75 must be transferred to the PLC followed by a restart.

© Siemens AG, 2003. All rights reserved

Example

The following table 2-7 contains the parameters for the programming example below:

Table 2-8 Example parameters

Group	Decoding list (DB 75)			Signal list
	Extended M address	First M address	Last M address in group	DB 76
1	2	1	5	DBX0.0 to DBX0.4
2	3	12	23	DBX2.0 to DBX3.3
3	40	55	55	DBX4.0

```
DATA_BLOCK DB 75
TITLE =
VERSION: 0.0
  STRUCT
    MSigGrp : ARRAY  [1 .. 16 ] OF STRUCT
      MExtAdr :          INT ;
      MFirstAdr :        DINT ;
      MLastAdr :         DINT ;
    END_STRUCT ;
  END_STRUCT ;
BEGIN
  MSigGrp[1].MExtAdr :=          2;
  MSigGrp[1].MFirstAdr :=        L#1;
  MSigGrp[1].MLastAdr :=         L#5;
  MSigGrp[2].MExtAdr :=          3;
  MSigGrp[2].MFirstAdr :=        L#12;
  MSigGrp[2].MLastAdr :=         L#23;
  MSigGrp[3].MExtAdr :=          40;
  MSigGrp[3].MFirstAdr :=        L#55;
  MSigGrp[3].MLastAdr :=         L#55;
END_DATA_BLOCK
```

© Siemens AG, 2003. All rights reserved

2.7.9 PLC machine data

General

The user has the option of storing PLC-specific machine data in the NC. He can process these data while the PLC is running up (OB100). This feature allows, for example, user options, machine expansions, machine configurations, etc. to be implemented.
The interface for reading the data is located in DB20. However, DB20 is set up by the basic program during power-up only when user machine data are used, i.e. sum of BP parameters UDInt, UDHex and UDReal is greater than zero. The size of the individual areas and thus also the total length of DB20 is set by PLC machine data (in general machine data: MAXNUM_USER_DATA_INT, MAXNUM_USER_DATA_HEX, MAXNUM_USER_DATA_FLOAT) and displayed for the user in BP parameters: UDInt, UDHex and UDReal. The BP stores data flush in DB20 in the following order: Integer MD, Hexa field MD, Real MD.

Integer and real values are stored in S7 format in DB20.
Hexadecimal data are stored in DB20 in the order in which they are input (use as bit fields).

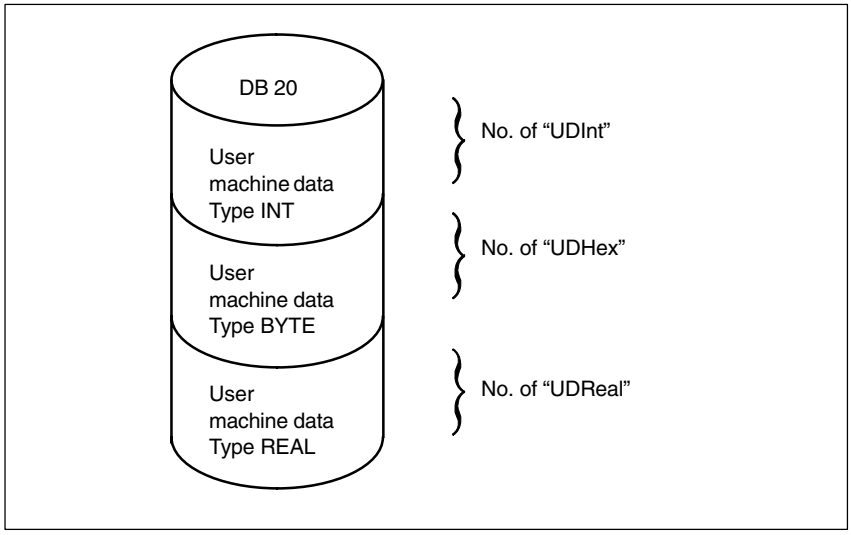


Fig. 2-15 DB 20

Note

If the number of PLC machine data used is increased later, then DB20 must be deleted beforehand. To prevent such extensions in use from having any effect on the existing user program, the data in DB20 should be accessed in symbolic form wherever possible, e.g. by means of a structure definition in the UDT.

© Siemens AG, 2003. All rights reserved

Table 2-9 Alarms

400120	Delete DB 20 in PLC and restart
Explanation	DB length is not the same as the required DB length
Reaction	Alarm display and PLC STOP
Remedy	Delete DB 20 followed by Reset
Continuation	After cold restart

Example

The project in the example requires 4 integer values, 2 hexadecimal fields with bit information and 1 real value.

Machine data:

```
14510 USER_DATA_INT[0]          123
14510 USER_DATA_INT[1]          456
14510 USER_DATA_INT[2]          789
14510 USER_DATA_INT[3]          1011
...
14512 USER_DATA_HEX[0]          12
14512 USER_DATA_HEX[1]          AC
                ...
14514 USER_DATA_FLOAT[0]        123.456
```

BP parameter (OB100):

```
Call fb 1, db 7(
                MCPNum :=       1,
                MCP1In :=       P#E0.0,
                MCP1Out :=      P#A0.0,
                MCP1StatSend := P#A8.0,
                MCP1StatRec :=  P#A12.0,
                MCP1BusAdr :=   6,
                MCP1Timeout :=  S5T#700MS,
                MCP1Cycl :=     S5T#200MS,
                NCCyclTimeout := S5T#200MS,
                NCRunupTimeout :=  S5T#50S;
```

BP parameters (scan during run):

```
                l gp_par.UDInt;     //=4,
                l gp_par.UDHex;     //=2,
                l gp_par.UDReal;    //=1 )
```

© Siemens AG, 2003. All rights reserved

While the PLC is powering up, DB20 has been set up with a length of 28 bytes:

DB 20

Table 2-10 DB 20

Address	Data
0.0	123
2.0	456
4.0	789
6.0	1011
8.0	b#16#12
9.0	b#16#AC
10.0	1.234560e+02

The structure of the machine data used is specified in a UDT:

```
TYPE UDT 20
  STRUCT
    UDInt :          ARRAY  [0 .. 3 ] OF INT ;
    UDHex0 :         ARRAY  [0 .. 15 ] OF BOOL ;
    UDReal :         ARRAY  [0 .. 0 ] OF REAL ;   //Description as field for
                                                  // subsequent extensions
  END_STRUCT ;
END_TYPE
```

Note

ARRAY OF BOOL are always send to even-numbered addresses. For this reason, an array range of 0 to 15 must generally be selected in the UDT definition or all Boolean variables specified individually.

Although only a REAL value is used initially in the example, a field (with one element) has been created for the variable. This ensures that extensions can be made easily in the future without the symbolic address being modified.

An entry is made in the symbol list to allow data access in symbolic form:

Alarms

Symbol	Operand	Data type
UData	DB 20	UDT 20

© Siemens AG, 2003. All rights reserved

Access operations in user program (list includes only symbolic read access operations):

```
...
    L    ”UData”.UDInt[0];
    L    ”UData”.UDInt[1];
    L    ”UData”.UDInt[2];
    L    ”UData”.UDInt[3];

    U    ”UData”.UDHex0[0];
    U    ”UData”.UDHex0[1];
    U    ”UData”.UDHex0[2];
    U    ”UData”.UDHex0[3];
    U    ”UData”.UDHex0[4];
    U    ”UData”.UDHex0[5];
    U    ”UData”.UDHex0[6];
    U    ”UData”.UDHex0[7];
                          ...
    U    ”UData”.UDHex0[15];

    L    ”UData”.UDReal[0];
...
```

2.7.10 Configurability of machine control panel, handheld unit

General

The communications system integrated in the NC permits a maximum of 2 machine control panels and one handheld unit to exchange data with the 810D (SW 4 and higher) and the 840D. An SDB 210 is not required to transfer the signals of these components. The information given below is based on the assumption that no SDB 210 has been installed for the components concerned.

The components are parameterized by calling basic program block FB 1 in OB 100.FB 1 stores its parameters in the associated instance DB (DB 7, symbolic name “gp_par”). Separate parameter sets are provided for each machine control panel and the handheld unit. The input/output addresses of the user must be defined in these parameter sets. These input and output addresses are also used in FC 19, FC 24, FC 25, FC 26 and FC 13. Addresses for status information, MPI or OPI (a GD parameter set must be set for the handheld unit rather than an MPI address) must also be defined (please also see Chapter 4, Description of FB 1). The time settings for timeout and cyclical forced retriggering can remain at the defaults.

Activation

Each component is activated either via the number of machine control panels (Parameter MCPNum) or, in the case of the handheld unit, parameter BHG := 2 (BHG := 1 signifies a link via the MPI interface in conjunction with an SDB 210). Whether a component is to be linked to the OPI or the MPI is determined by parameters MCPMPI and BHGMPI.

Handheld unit

The handheld unit addresses the MPI or OPI by means of a GD parameter set. These parameter values must be assigned according to the handheld unit settings. However, the parameter names on the handheld unit are the reverse of the parameter names in the basic program. All parameters of type Send on the handheld unit must be defined as type Rec (and type Rec as Send) in the basic program.

© Siemens AG, 2003. All rights reserved

Control signals

Parameters MCP1Stop, MCP2Stop and BHGStop can be used to stop communication with individual components (parameter setting = 1). This stop or activation of communication can be applied in the current cycle. However, the change in value must be implemented through the symbolic notation of the parameters and not by means of another FB 1 call.

Example of how to stop data exchange with the 1st machine control panel:

```
SET;
S                gp_par.MCP1Stop;
```

Setting of the parameters MCP1Stop, MCP2Stop, BHGStop also results in a suppression of the alarms 40260 to 40262

MPI switchover, OPI address

With SW 4 and higher, an existing connection with a machine control panel (MCP) or handheld unit (HHU) can be separated. Another MCP or HHU component already connected to the bus (with another MPI or OPI address) can then be activated. The following procedure must be followed to switch addresses:

1. Stop communication with component to be decoupled via parameter MCP1Stop or MCP2Stop or BHGStop = 1.
2. After checkback in DB10 byte 104 (relevant bits 0, 1, 2 set to 0). Change in the bus address or GD parameter set of this unit to that of the new component.
3. In this PLC cycle, communication with the new component can now be activated again by means of parameter MCP1Stop or MCP2Stop or BHGStop = 0.
4. Communication with the new component is taking place when the checkback in DB10 byte 104 (relevant bits 0, 1, 2 is set to 1).

As described in Section Control signals, all parameters must be programmed according to data type.

Configuration

There are basically two communications mechanisms for transferring data between the MCP/HHU and PLC. With the first mechanism, data are transported via the Comm module (840D/810D). The mechanism is parameterized completely via the MCP/HHU parameters in FB1.

In the second case, data are transferred via the PLC operating system (FM-NC) through evaluation of SDB210(Global Data). The mechanism is parameterized via STEP7–>Global Data. To allow the basic program to access these data and implement MCP/HHU failure monitoring, the addresses set via SDB210(Global Data) must be declared in the FB1 parameters in the basic program.

An overview of the various interfacing options as a function of the NC type used is given below. In each case, the parameter set of FB1 and the valid status information that are relevant for the respective data transmission method are specified.

© Siemens AG, 2003. All rights reserved

If an error is detected due to a timeout monitor, a corresponding entry is made in the diagnostics buffer of the PLCCPU (errors 400260 to 400262). In this case, the input signals from the MCP or from the handheld unit (MCP1In/MCP2In or BHGIn) are initialized with 0. If it is possible to resynchronize the PLC and MCP/HHU, communication is resumed automatically and the error message reset by the BP.

840D: OPI/MPI coupling

Communication starts from the PLC BP via the NCK and COMM mode, i.e. even a link via the MPI does not require an SDB210. The operation is parameterized via the relevant parameters in FB1

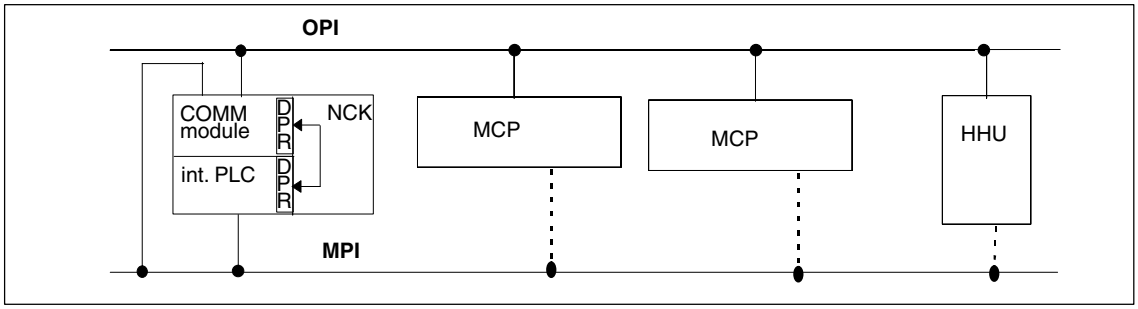


Fig. 2-16 840D: OPI/MPI coupling

Table 2-11 Relevant parameters (FB1)

MCP		HHU
MCPNum=1 or 2 (number of MCPs)		BHG=2 (transfer via COMM module)
MCP1In	MCP2In	BHGIn
MCP1Out	MCP2Out	BHGOut
MCP1StatSend	MCP2StatSend	BHGStatSend
MCP1StatRec	MCP2StatRec	BHGStatRec
MCP1BusAdr	MCP2BusAdr	BHGInLen
MCP1Timeout	MCP2Timeout	BHGOutLen
MCP1Cycl	MCP2Cycl	BHGTimeout
MCPMPI = FALSE (OPI), TRUE (MPI)		BHGCycl
MCP1Stop	MCP2Stop	BHGRecGDNo
		BHGRecGBZNo
		BHGRecObjNo
		BHGSendGDNo
		BHGSendGBZNo
		BHGSendObjNo
		BHGMPI = FALSE (OPI), TRUE (MPI)
		BHGStop

© Siemens AG, 2003. All rights reserved

Table 2-12 Status information

Available in	Bit No.	Description
MCP1StatSend MCP2StatSend HGStatSend	4	Syntax error in GD packet: Error in parameter set (FB1)
MCP1StatSend MCP2StatSend BHGStatSend	27	Sender: Time out
MCP1StatRec MCP2StatRec BHGStatRec	10	Receiver: Time out

An error entry is also generated in the PLC diagnostics buffer for the timeout monitors (bits 10 and 27), resulting in the following error messages on the MMC:

- 400260: MCP 1 failure or
- 400261: MCP 2 failure
- 400262: HHU failure

An MCP or HHU failure (SW 4 and higher) is detected immediately after a cold restart even if no data have yet been exchanged between the MCP/HHU and PLC. The monitoring function is activated as soon as all components have signaled “Ready” after power-up.

840D: MPI coupling for HHU (not for new development)

Communications for HHU by PLC operating system and parameterization via SDB210.

Communication for the MCP is controlled from the PLC BP via the NCK and COMM module as described above.

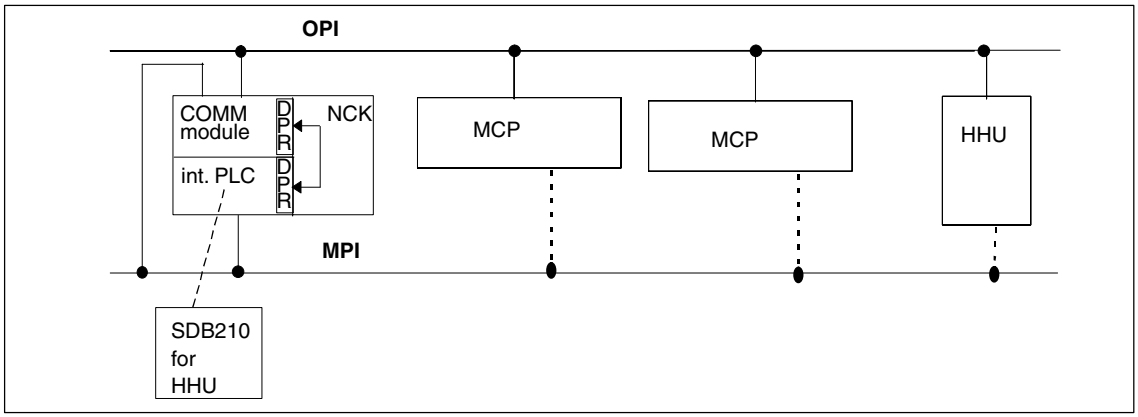


Fig. 2-17 840D: MPI coupling for HHU

Relevant parameters (FB1):

Communication between the PLC and HHU is implemented through configuring and subsequent loading of SDB210 (Global Data). To allow the basic program to access the HHU data and implement HHU failure monitoring, the addresses set via global data must be declared in FB1 parameters.

© Siemens AG, 2003. All rights reserved

Table 2-13 Relevant parameters (FB1)

MCP		HHU (parameterization via SDB210)
The interface is parameterized via the relevant parameters in FB1 (see Table 2-11)		BHG = 1 (transfer via SDB210)
		BHGIn (as parameterized in SDB210)
		BHGOut (as parameterized in SDB210)
		BHGStatRec (as parameterized in SDB210)
		BHGTimeout (as parameterized in SDB210)

Table 2-14 Status information (MCP1 and MCP2 see Table 2-12)

Available in	Bit No.	Description
BHGStatRec	10	Receiver: Time out,

An error entry is also generated in the PLC diagnostics buffer for the timeout monitors, resulting in the following error message on the MMC:

- 400262: HHU failure.

An HHU failure is onlz recognized if data exchange has taken place previously to the HHU. The first exchange of data with the HHU activates the monitoring function.

MPI coupling (SW 4 and higher)

Communication starts from the PLC BP via the NCK and COMM mode, i.e. even a link via the MPI does not require an SDB210. The operation is parameterized via the relevant parameters in FB1.

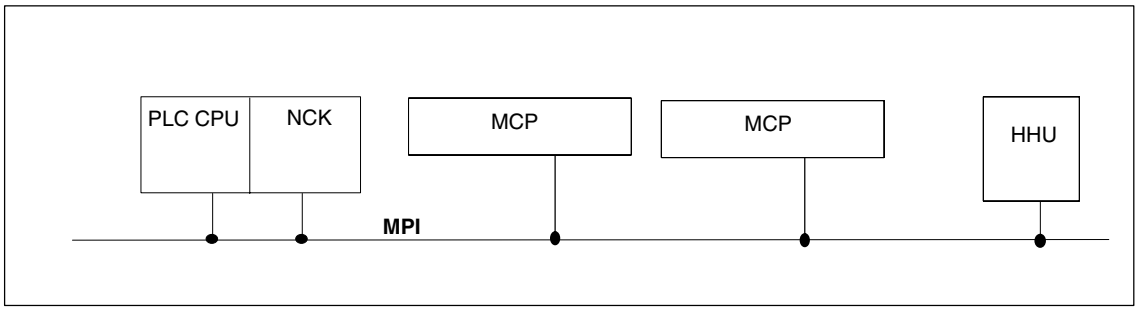


Fig. 2-18 MPI coupling

Table 2-15 Relevant parameters (FB1)

MCP		HHU
MCPNum=1 or 2 (number of MCPs)		BHG=2 (transfer via Comm-module)
MCP1In	MCP2In	BHGIn
MCP1Out	MCP2Out	BHGOut
MCP1StatSend	MCP2StatSend	BHGStatSend

© Siemens AG, 2003. All rights reserved

Table 2-15 Relevant parameters (FB1)

MCP		HHU
MCP1StatRec	MCP2StatRec	BHGStatRec
MCP1BusAdr	MCP2BusAdr	BHGInLen
MCP1Timeout	MCP2Timeout	BHGOutLen
MCP1Cycl	MCP2Cycl	BHGTimeout
MCPMPI = TRUE (MPI)		BHGCycl
MCP1Stop	MCP2Stop	BHGRecGDNo
		BHGRecGBZNo
		BHGRecObjNo
		BHGSendGDNo
		BHGSendGBZNo
		BHGSendObjNo
		BHGMPI = TRUE (MPI)
		BHGStop

Table 2-16 Status information

Available in	Bit No.	Description
MCP1StatSendMCP2StatSend BHGStatSend	4	Syntax error in GD packet: Error in parameter set (FB1)
MCP1StatSendMCP2StatSendB HGStatSend	27	Sender: Time out
MCP1StatRec MCP2StatRec BHGStatRec	10	Receiver: Time out

An error entry is also generated in the PLC diagnostics buffer for the timeout monitors (bits 10 and 27), resulting in the following error messages on the MMC:

- 400260: MCP1 failure or
- 400261: MCP2 failure.
- 400262: HHU failure.

An MCP or HHU failure (SW 4 and higher) is detected immediately after a cold restart even if no data have yet been exchanged between the MCP/HHU and PLC. The monitoring function is activated as soon as all components have signaled “Ready” after power-up.

FM–NC: MPI coupling and 810D (SW 4 and lower)

Communications for MCP and HHU by PLC operating system and parameterization via SDB210.

© Siemens AG, 2003. All rights reserved

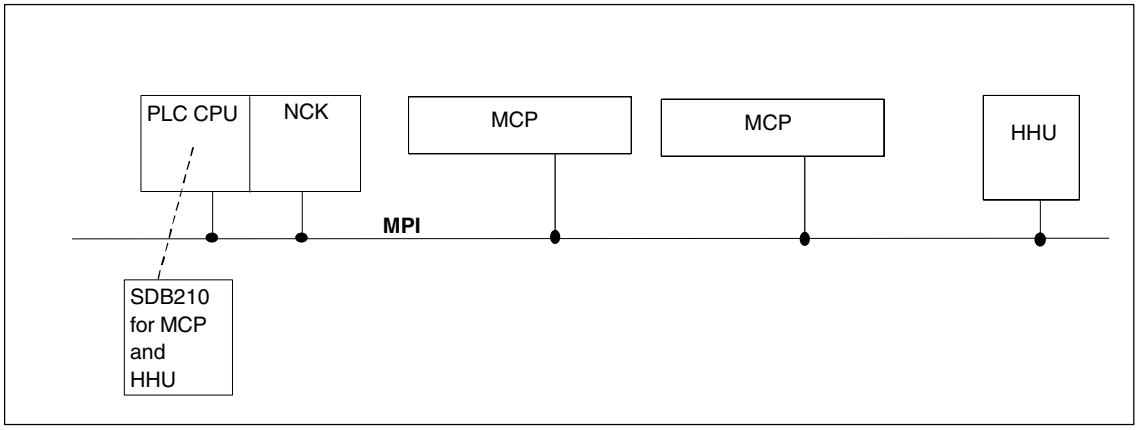


Fig. 2-19 MPI coupling and 810D

Relevant parameters (FB1):

Communication between the PLC and HHU is implemented through configuring and subsequent loading of SDB210 (Global Data). To allow the basic program to access the HHU data and implement HHU failure monitoring, the addresses set via global data must be declared in FB1 parameters.

Table 2-17 Relevant parameters (FB1), (all entries as parameterized in SDB210 global data)

MCP		HHU
MCPNum=1 or 2 (number of MCPs)		BHG=1 (MPI)
MCP1In	MCP2In	BHGIn
MCP1Out	MCP2Out	BHGOut
MCP1StatRec	MCP2StatRec	BHGStatRec
MCP1Timeout	MCP2Timeout	BHGTimeout

Table 2-18 Status information (MCP1 and MCP2 see Table 2-12)

Available in	Bit No.	Description
MCP1StatRec MCP2StatRec BHGStatRec	10	Receiver: Time out,

An error entry is also generated in the PLC diagnostics buffer for the timeout monitors, resulting in the following error message on the MMC:

- 400260: MCP1 failure or
- 400261: MCP2 failure.
- 400262: HHU failure.

An MCP/HHU failure is detected only if the unit has already been involved in a data exchange. The first exchange of data with the MCP/HHU activates the monitoring function.

© Siemens AG, 2003. All rights reserved

Note

SDB210 in BP of FM-NC

The project in the basic program supplied for the FM-NC already contains an SDB 210. The parameter settings of this block are *preprogrammed* and can be used only to link up *one* MCP. The bus address of the MCP must be set to “6”. Bus address 6 can only be used in conjunction with the SDB 210 integrated in the parts program.

If another MCP/HHU is to be connected or if this bus address cannot be used, then the user must create his own SDB210 using STEP7 global data.

2.8 SPL for Safety Integrated

See /FBSI/ Description of Functions for Safety Integrated

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Supplementary Conditions and NC VAR Selector

3

3.1 Supplementary conditions

3.1.1 Programming and parameterizing tools

Hardware

Programming devices or PCs with the following equipment are required for the PLCs installed on the FM-NC, 810D and 840D:

	Minimum	**Recommended**
Processor	80486	Pentium
RAM (MB)	32	64 or more
Hard disk, free capacity (MB)	200	> 400
Interfaces	MPI incl. Cable memory card	
Graphics	VGA or TIGA	SVGA
Mouse	Yes	
Operating system	Windows 95/98/NT with STEP7 Version 4	Windows 95/98/NT or later with STEP7 Version 5.1 or later

The **STEP7 package for the S7300** can be obtained and installed on equipment meeting the above requirements in cases where the package has not already been supplied with the programming device.

This package allows the following functions to be implemented (more detailed information about available functions can be found in the SIMATIC catalogs and the STEP7 documentation):

© Siemens AG, 2003. All rights reserved

Programming

- Editors and compilers for STL (complete scope of the language incl. SFB/ SFC calls), LAD, CSF
- Creation and editing of assignment lists (symbol editor)
- Data block editor
- Input and output of blocks ON/OFF line
- Insertion of modifications and additions ON and OFF line
- Transfer of blocks from programming device to the PLC and vice versa

Parameterizing

- Parameterizing tool **HW Config** for CPU and I/O device parameterization
- Configuration tool **Communication Configuration** or setting the CPU communication parameters
- Output of system data such as hardware and software version, memory capacity, I/O expansion/assignment

Testing and diagnostics (ONLINE)

- Variable status/forcing (I/Os, flags, data block contents, etc.)
- Status of individual blocks
- Display of system states (ISTACK, BSTACK, system status list)
- Display of system messages
- PLC STOP/complete restart/overall reset triggering from the programming device
- Compress PLC

Documentation

- Printout of individual or all blocks
- Allocation of symbolic names (also for variables in data blocks)
- Input and output of comments within each block
- Printout of test and diagnostics displays
- Hardcopy function
- Crossreference list
- Program overview
- Assignment plan I/O/M/T/C/B/P/D

© Siemens AG, 2003. All rights reserved

Archiving of utility routines

- Allocation of the output statuses of individual blocks
- Comparison of blocks
- Rewiring
- STEP 5 –> STEP 7 converter

Option packages

– Programming in S7-HIGRAPH, S7-GRAPH, SCL.
These packages can be ordered via SIMATIC Sales and Marketing Offices.

– Special packages for configuring modules
(e.g. CP3425 –> NCM package)

3.1.2 SIMATIC documentation required

References:

System description of SIMATIC S 7
List of S7–300 operations, CPU 314, CPU 315–2DP
Programming with STEP 7
STEP 7 User's Guide
STEP 7 Programming Guide: Design of user programs
STEP 7 Reference Manual: Statement List (STL)
STEP 7 Reference Manual: Ladder Diagram (LAD)
STEP 7 Reference Manual: System and Standard Functions
STEP 7 Manual: Conversion of STEP 5 Programs
STEP 7 General Index
Manual CPU 314, CPU 315–2DP

3.1.3 Relevant SINUMERIK documents

References: /IAF/, Installation and StartUp Guide FM-NC,
Chapter PLC Interface,
/IAD/, Installation and Start Up Guide 840 D, 611 D,
Chapter PLC Interface,
/IAG/, Installation and Start Up Guide 810 D, 611 D,
Chapter PLC Interface,
/BH/, Operator Components Manual (HW)
840 D/ FM-NC/ 810 D
/FB/, Descriptions of Functions FM-NC/840 D, 810 D
/LIS/, Lists 840 D/FM-NC/810 D
/FBP/, PLCCProgramming

© Siemens AG, 2003. All rights reserved

3.2 NC VAR selector

3.2.1 Overview

General

A catalog with an optional catalog name must be set up via the Windows-Explorer. The selected data of the VAR selector (data.VAR and data.AWL (STL)) must be stored in this catalog. An “Insert”, “External source” into the STEP7 machine project for the “Data.AWL” file must then be carried out via the STEP7 manager (V3 and higher). The source container must be selected in the manager for this purpose. This action stores this file in the project structure. Once the file has been transferred, these AWL (STL) files must be compiled with STEP7.

The PC application **NC VAR selector** fetches the addresses of required NC variables and conditions them for accessing in the PLC program (FB 2/FB 3). This enables the programmer to select NC variables from the entire range of NC variables, to store this selection of variables, to edit them by means of a code generator for the STEP7 compiler and finally to store them as an ASCII file (*.AWL) in the machine CPU program. Fig. 3–1 illustrates this process.

Note

The latest NC VAR selector can be used for each NC software version (even earlier versions). The variables can also be selected from the latest list for earlier NC software versions. The data content in DB 120 (default DB for variables) does not depend on the software version, i.e. selected variables in an older software version must not be reselected when the software is upgraded.

© Siemens AG, 2003. All rights reserved

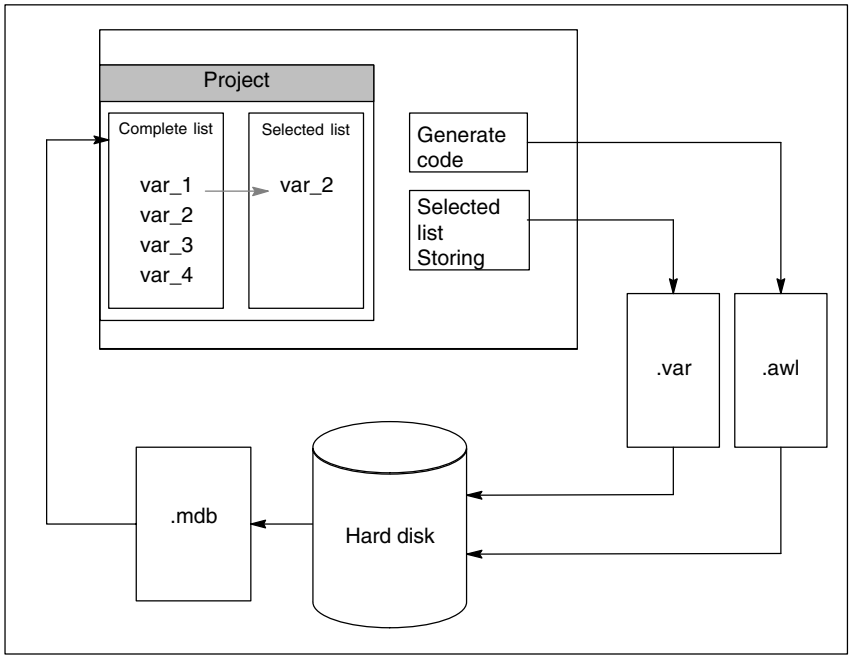


Fig. 3-1 NC VAR Selector

After the **NC VAR selector** application has been started, a list of variables of an NC variant is selected (hard disk –> File Ncv.mdb) and all the variables contained in this list displayed in a window. SW 6.3 includes the variable lists ncv*.mdb categorized according to:
Variables of NC including machine and setting data: ncv_NcData.mdb
Machine data of the 611D drive: ncv_611d.mdb
Machine data of the 611D linear drive: ncv_611dLinear.mdb
Machine data of the 611D drive, Performance 2: ncv_611d_P2.mdb
Machine data of the 611D linear drive, Performance 2: ncv_611d_P2Linear.mdb
Machine data of the hydraulic drive: ncv_Hydraulics.mdb

The user can also transfer variables to a second list (another window). This latter selection of variables can then be stored in an ASCII file or edited as a STEP7 source file (.awl) and stored.

Once he has generated a PLC data block by means of the STEP7 compiler, the programmer is able to read or write NCK variables via the basic program function blocks “PUT” and “GET” using the STEP7 file.

The list of selected variables is also stored as an ASCII file (file extension .var).

The variable list supplied with the **NC VAR selector** tool is appropriately adapted to the current NC software version. This list does not contain any variables (GUD variables) defined by the user. These variables are dealt with by the basic program by function block FB 5 (see Chapter 4 of this description).

© Siemens AG, 2003. All rights reserved

Note

The latest version of the **NC VAR selectors** is capable of processing all previous NC software versions. It is not therefore necessary to install different versions of the **NC VAR selectors** in parallel.

System features, supplementary conditions

The PC application **NC VAR Selector** requires Windows 95 (or later operating system).
Assignment of names to variables is described in

References: /LIS/, Lists, Section Variables
or also in the help file of the varaibles
(integrated in NC VAR selector)

3.2.2 Description of Functions

Overview

Fig. 3-2 illustrates how the NC VAR selector is used within the STEP7 environment.

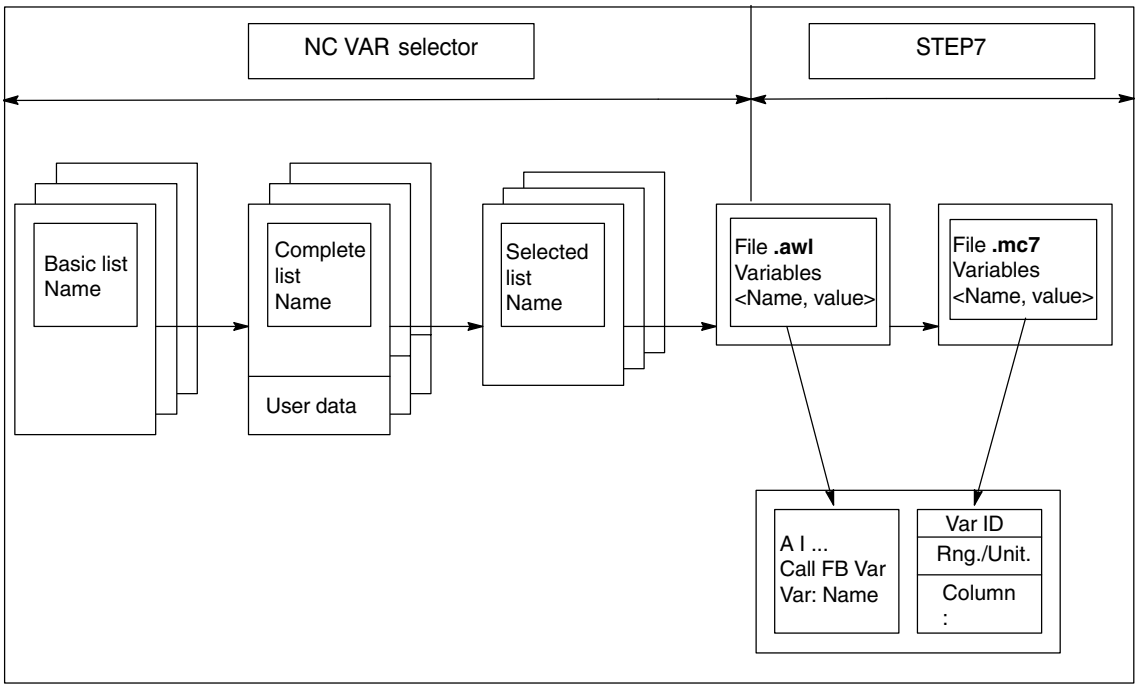


Fig. 3-2 Application of NC VAR selector in the STEP7 environment

The NC VAR selector is used to generate a list of selected variables from a list of variables and then to generate an **.awl** file that can be compiled by the STEP7 compiler.

© Siemens AG, 2003. All rights reserved

Note

An *.awl file contains both the names or ALIAS names and the data for the address parameters of the NC variables.
A data block generated from the file contains only the address parameters (10 bytes per parameter).

- The generated data blocks must always be stored in the machine-specific file storage according to STEP 7 specifications.
- To ensure that the parameterization of the GET/PUT (FB 2/3) blocks with respect to NC addresses can be implemented with symbols, the freely assignable, symbolic name of the generated data block must be included in the STEP 7 symbol list.

Basic display/ basic menu

After the NC VAR selector has been selected (started), the basic display with all input options (upper menu bar) appears on the screen (Fig. 3-3). All other displayed windows are placed within the general window.

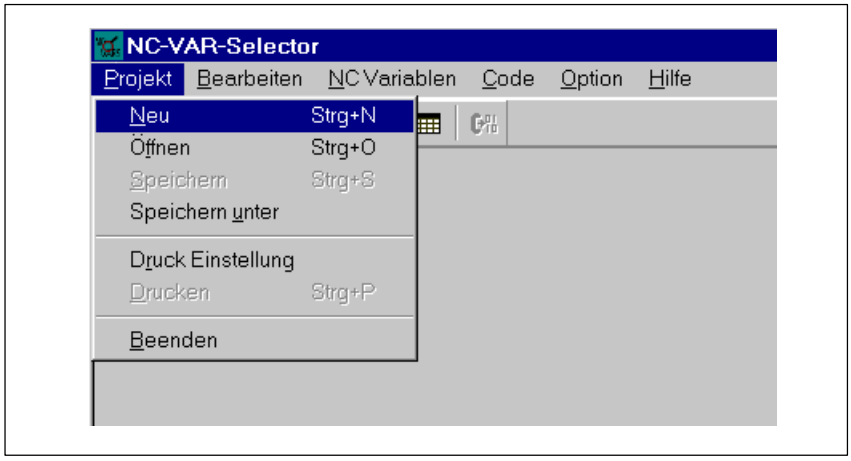


Fig. 3-3 Basic display with basic menu

Menu item Project

All operator actions associated with the project file (file of selected variables) are performed under this menu item (Fig. 3-3).

End of the application

The application can be terminated by selecting option **End** under menu item **Project**.

© Siemens AG, 2003. All rights reserved

Creating a new project

A new project (new file for selected variables) can be set up under menu item **Project**.

A window is displayed for the selected variables when **NEW** is selected (Fig. 3-4). The file selection for the NC variable list is then displayed after a prompt (applies only if the NC variable list is not already open).

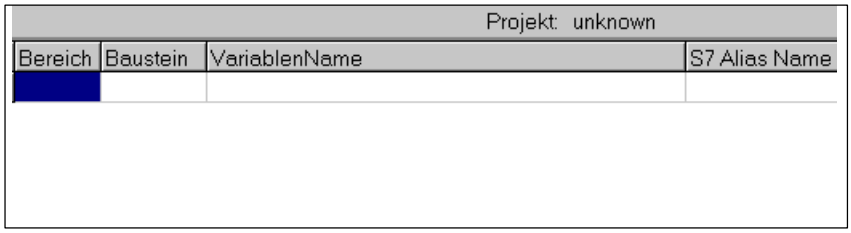


Fig. 3-4 Window with selected variables for new project

The selected variables are displayed in a window.

Opening an existing project

A project which already exists (variables have already been selected) can be opened by selecting **Open** under menu item **Project** (see Fig. 3–5). A file selection window is displayed allowing the appropriate project with extension .**var** to be selected.

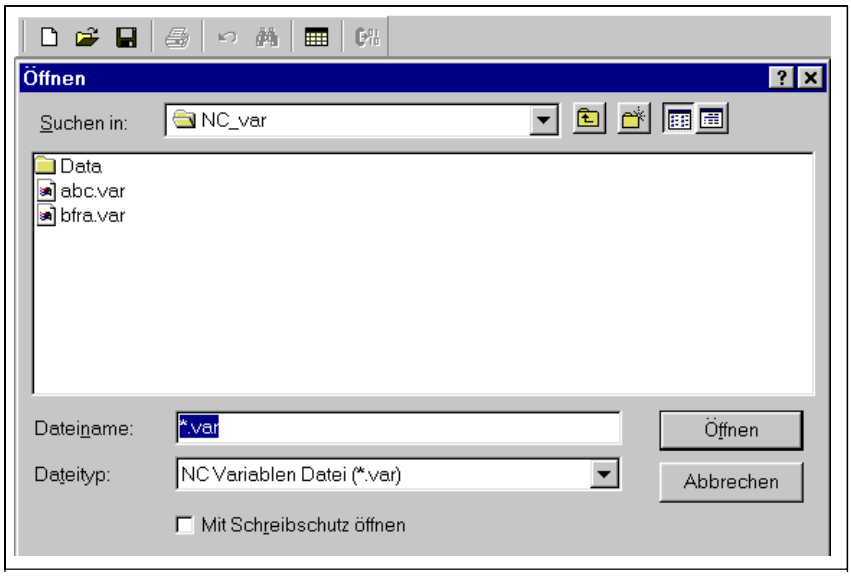


Fig. 3-5 Selection window for existing projects

If, after selection of the project, new variables are to be added, a complete list of NCK variables must be selected (see: Selecting complete list). No complete list need be called if the user only wishes to delete variables from the project.

Saving a project

The variable list is stored via menu commands **Project, Save** or **Save as ...**.

Save stores the variable list under a path which is already specified. If the project path is not known, then the procedure is as for **Save as ...**.

© Siemens AG, 2003. All rights reserved

Save as ... displays a window in which the path for the project to be stored can be specified.

Printing a project

The command **Print** under menu item **Project** can be selected to print a project file. The number of lines per page are selected via menu command “Print Setting” (default is 77 lines).

Menu item Edit

All operator actions such as

- transfer variables
- delete variables
- change alias names
- search for variables

can be executed directly in this menu. These actions can also be canceled again under Edit.

Undoing actions

Operator actions relating to the creation of the project file (transfer variables, delete variables, change alias names) can be undone in this menu.

Menu option NC variables

The basic list of all variables is saved in NC Var Selector path Data\Swxy (xy stands for SW no., e.g. SW 5.3:=xy=53). This list can be selected as an NC variables list. In SW 6.3 and higher, variable lists are available according to topic (see previous chapter).

Selection of a NC variable list

A list of all the NC variables of an NC version is now selected and displayed via menu commands **NC Variable List, Select** (see Fig. 3-6).

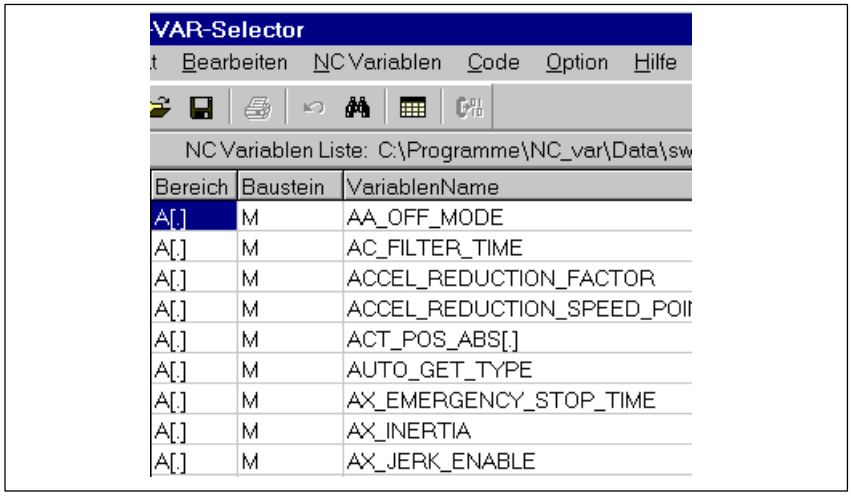


Fig. 3-6 Window with selected Complete List

© Siemens AG, 2003. All rights reserved

The field variables (e.g. axis area, T area data, etc.) are indicated by means of brackets ([.]). Additional information must be specified here. When the variables are transferred to the project list, the additional information required is requested.

Displaying subsets

When any table field is doubleclicked (with the exception of variable fields!), a window is displayed in which filter criteria can be preset (Fig. 3-6).

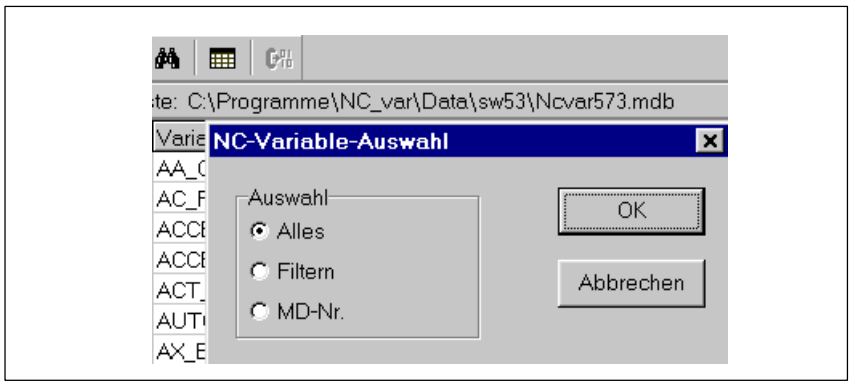


Fig. 3-7 Window with filter criteria for displaying list of variables

There are three options available:·

- Display all data
- Input area, block and name (incl. combinations)
- Display MD/SE data number

It is also possible to use the following wildcards:
* to extend search criterion to any length,

Example of search criteria

Search criterion Name: CHAN*	This is found:	CHAN_NAME chanAlarm chanStatus channelName chanAssignment

Select variables

A variable is selected by means of a simple mouse click and transferred to the window of selected variables with a double click.

This action can also be undone again under menu item **Edit**.

© Siemens AG, 2003. All rights reserved

Alias name

The variables names provided can be up to 32 characters in length. To make variables clearly identifiable in the data block to be generated, several ASCII characters are added to the selected name. However, the STEP7 compiler recognizes only 24 ASCII characters as an unambiguous STEP7 variable. Since it cannot be precluded that variable names can only be differentiated by the last 8 character positions, **Alias** names are used for names which are too long. When a variable is selected, the length of the STEP7 name to be used is therefore checked. If it is longer than 24 characters, the user must input an additional name which is then used as the alias. **In this case, the user must ensure that the alias name is unambiguous.**

The alias input can always be activated by the user in the **Option** menu. An alias name can then be entered every time a variable is transferred.

It is also possible to edit alias names at a later point in time by doubleclicking on the S7 variable name field. This action can also be undone again under menu item **Edit**.

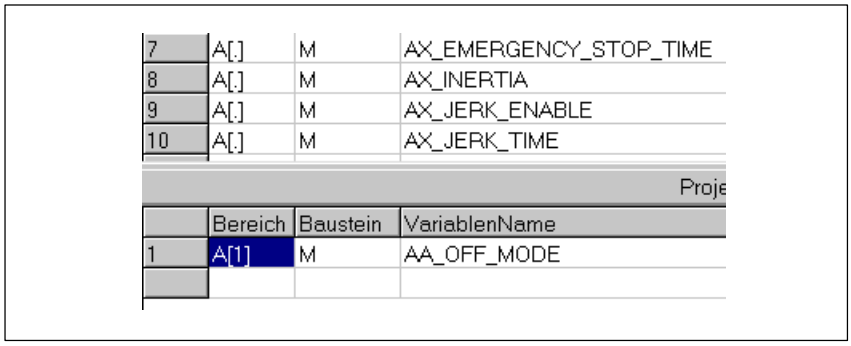


Fig. 3-8 Screen with complete list and selected variables

Scrolling

A scroll bar is displayed if it is not possible to display all variables in the window. The remaining variables can be reached by scrolling (page up/down).

Variables in multi-dimensional structures

If variables are selected from multi-dimensional structures, then the column and/or line number as well as the area number must be entered so that the variables can be addressed (see Fig. 3-8). The required numbers can be found in the NC variables documentation. (Manufacturer Documentation **Lists**, Section Variables)

By entering a zero (0) as the block number or the line or column index, it is possible to use the variable in the S7 PLC as a pointer to these data. When reading or writing these data via the functions “PUT” and “GET”, the optional parameters “UnitX”, “ColoumnX” and “LineX” must be filled with the necessary information.

© Siemens AG, 2003. All rights reserved



Fig. 3-9 Entry field for line, column and block no.

Deleting variables

Variables are deleted in the window of selected variables by selecting the appropriate variables (single mouse click) and pressing the **Delete** key. No deletion action is taken with the doubleclick function. It is possible to select several variables for deletion (see under Selection of variables).

This action can also be undone again under menu item **Edit**.

Note

Deleting of variables results in a change of the absolute addresses of the pointer structures to the variables. When changing the Variable Selection, it is therefore absolutely necessary to **generate** one or several **text files of all user blocks prior to the change**. Only thus, it can be ensured that the assignment of the variables in FB “GET” or FB “PUT” is still correct even after recompilation.

Storing a selected list

Once variables have been selected, they can be stored under a project name. The files are stored on a project-specific basis.

A window is displayed (Fig. 3-10) for the file to be stored. The project path and name for the file must be selected in the window.

© Siemens AG, 2003. All rights reserved

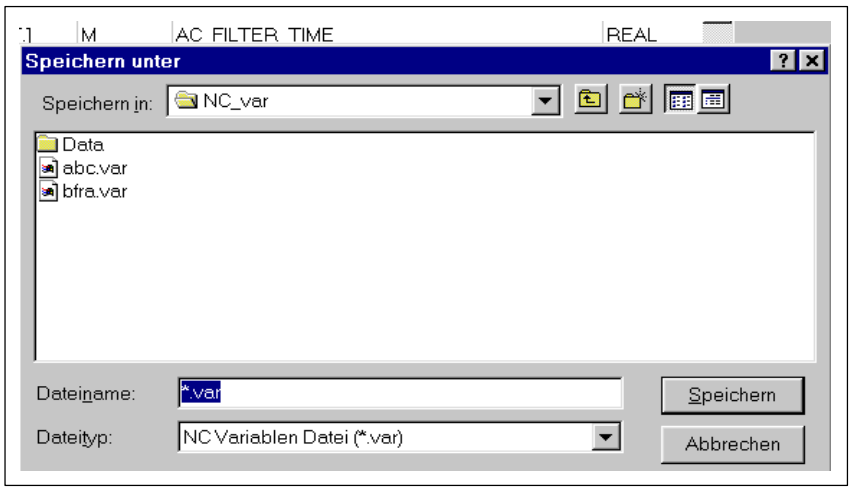


Fig. 3-10 Window for project path and name of file to be stored

Code generation

This menu item contains three selection options:

1. Settings (input of data block number to be generated) and other settings
2. Generate (create data block)
3. In the STEP7 project (transferring the data block to a STEP7 project)

Settings

Under this menu item, the DB number and the symbol for this DB number for which the code is created is entered. Under the “Unit System” tab, a selection is made how the the unit system variables are calculated in the PLC. Unter the “Generate” tab, the project creation is defined for the relevant target system.

Generate

Under this menu item, the STEP 7 file from the selected variable list with extension **.awl** is set. A file is generated when **Select** is clicked:

An **.awl** file that can be used as an input for the STEP 7 compiler.

A window is opened in which path and name for the .awl file to be generated must be specified for the file to be saved.

In a STEP7 project

The generated AWL file is transferred to a selectable SIMATIC project (program path) and compiled. Furthermore, the symbol can also be transferred. This function will be available as of STEP7 Version 5.1 and NC Var Selector 6.04.05. This process take some time due to the time require by STEP7. Before transfer of a new AWL file, the file window of the AWL file must be closed in the LAD/FUP/AWL Editor.

© Siemens AG, 2003. All rights reserved

Menu item Option

The following can be selected under menu item **Option**

- The current language
- The mode for the alias input (always / >24 characters).

Menu item Help

The following submenu points

- The Operator's Guide
- The Description of Variables

can be selected here for further reference.

The copyright and the version number can also be displayed.

3.2.3 Startup, installation

The Windows application **NC Var selector** is installed using the SETUP program supplied with the package.

■

© Siemens AG, 2003. All rights reserved

Block Descriptions

4

4.1 FB 1: RUN_UP Basic program, start-up section

Description of functions

The synchronization of NCK and PLC is performed during start-up. The data blocks for the NC/PLC user interface are created with reference to the NC configuration defined in the machine data and the most important parameters verified for plausibility. In the event of an error, FB 1 passes an error identifier to the diagnostics buffer and switches the PLC to the STOP state.

To enable an orderly start-up of the control, it is vital to synchronize the NCK and PLC, as these systems have their own types of powerup procedure. In the start-up routine, the CPUs therefore perform “subsidiary start-up functions” and exchange ID information to ensure that the procedure is functioning correctly. Since the start-up procedure is asynchronous, it is unavoidable that one CPU will have to “wait” in certain circumstances until the other has “caught up”. This is automatically managed by the basic program.

The PLC 314 and PLC 315-2DP only know the start-up type cold restart. A warm restart is not provided, i.e. following system initialization, the operating system runs organization block OB 100 and always commences cyclical execution at the start of OB1.

The user need only supply the FB 1 parameters that are relevant to his application. The preset values in the associated instance DB 7 do not need to be assigned. The block can only be called in OB 100.

Starting parameters

The output parameters in FB 1 provide the PLC user with information about the control system configuration. These data can also be accessed in the cyclic program section. There are 2 possible methods of access:
1. By means of direct access to data block DB 7 (instance of FB 1) in symbolic form (e.g. Lgp_par.MaxChan, in this case gp_par is the symbolic name of DB7).
2. Assignment of a flag, data element to the relevant parameter when FB 1 is parameterized (e.g. MaxChan:=MW 20). Information about the maximum number of channels can then be scanned in flag word 20 in the rest of the user program.

© Siemens AG, 2003. All rights reserved

Note

An additional SDB 210 must be generated via the STEP7 application Communication Configuration for the **operator components** which are connected at the **MPI**. The corresponding procedure is explained in the Installation and StartUp Guide.



Caution

The following applies only if a pointer parameter MCP1In, MCP1Out, MCP1StatSend, MCP1StatRec (or the parameter of the 2nd machine control panel and handheld unit) is to be assigned to a data block (the default settings for these pointers are inputs and outputs).

The following applies for the assignment of pointer parameters MCP1In, MCP1Out, MCP1StatSend, MCP1StatRec (and the parameters of the 2nd machine control panel and the handheld unit) to data block elements:

1. The In parameter (e.g. MCP1In) for this operator component **must** be parameterized to a data block.

2. The parameterized DB number must be the same with the other parameters if the other pointers are parameterized to data blocks (the other parameters can point to inputs, outputs or flags).

3. It is note essential to parameterize all pointers of an operator component to data blocks.

Declaration 810D, 840D

```
FUNCTION_BLOCK FB 1
VAR_INPUT
  MCPNum:        INT:=1;              //0: no MCP
                                      //1: 1 MCP (default)
                                      //2: 2 MCP
  MCP1In:        POINTER;             //Start addr. input signals MCP 1
  MCP1In:        POINTER;             //Start addr. output signals MCP 1
  MCP1StatSend:  POINTER;             //Status DW for transmission MCP 1
  MCP1StatRec:   POINTER;             //Status DW for receiving MCP 1
  MCP1BusAdr:    INT:=6;              //default
  MCP1Timeout:   S5TIME:= S5T700MS;
  MCP1Cycl:      S5TIME:= S5T200MS;
  MCP2In:        POINTER;             //Start addr. input signals MCP 2
  MCP1In:        POINTER;             //Start addr. output signals MCP 2
  MCP2StatSend:  POINTER;             //Status DW for transmission MCP 2
  MCP2StatRec:   POINTER;             //Status DW for receiving MCP 2
  MCP2BusAdr:    INT;
  MCP2Timeout:   S5TIME:= S5T700MS;
  MCP2Cycl:      S5TIME:= S5T200MS;
  MCPMPI:        BOOL:= FALSE;
  MCP1Stop:      BOOL:= FALSE;
  MCP2StopI:     BOOL:= FALSE;
  MCP1NotSend:   BOOL:= FALSE;
  MCP2NotSend    BOOL:= FALSE;
```

© Siemens AG, 2003. All rights reserved

```
    BHG:              INT;                //Handheld unit interface
                                          //0: No HHU
                                          //1: HHU at MPI
                                          //2: HHU at OPI
    BHGIn:            POINTER;            //Transmit data of handheld unit
    BHGOut:           POINTER;            //Receive data of handheld unit
    BHGStatSend:      POINTER;            //Status DW for transmission HHU
    BHGStatRec:       POINTER;            //Status DW for reception HHU
    BHGInLen:         BYTE:= B#16#6;      //Input 6 bytes
    BHGOutLen:        BYTE:= B#16#14;     //Output 20 bytes
    BHGTimeout:       S5TIME:= S5T700MS;
    BHGCycl:          S5TIME:= S5T#100MS;
    BHGRecGDNo:       INT:= 2;
    BHGRecGBZNo:      INT:= 2;
    BHGRecObjNo:      INT:= 1;
    BHGSendGDNo:      INT:= 2;
    BHGSendGBZNo:     INT:= 1;
    BHGSendObjNo:     INT:= 1;
    BHGMPI:           BOOL:= FALSE;
    BHGStop:          BOOL:= FALSE;
    BHGNotSend:       BOOL:= FALSE;

    NCCyclTimeout:    S5TIME:= S5T#200MS;
    NCRunupTimeout:   S5TIME:= S5T#50S;
    ListMDecGrp:      INT:=0;
    NCKomm:           BOOL:= FALSE;
    MMCToIF:          BOOL:= TRUE;
    HWheelMMC:        BOOL:= TRUE;        //Handwheel selections via MMC
    MsgUser:          INT:= 10;           //No. of user areas in DB2
    UserIR:           bool:= FALSE;       //User programs in OB40,
                                          //Note local data expansion !

    IRAuxfuT:         bool:= FALSE;       //Evaluate T function in OB40
    IRAuxfuH:         bool:= FALSE;       //Evaluate H function in OB40
    IRAuxfuE:         bool:= FALSE;       //Evaluate DL function in OB40
    UserVersion:      Pointer;            //Pointer to string variable that is output in
                                          //version display
END_VAR
VAR_OUTPUT
    MaxBAG:           INT;
    MaxChan:          INT;
    MaxAxis:          INT;
    ActivChan:        ARRAY[1..10] OF BOOL;
    ActivAxis:        ARRAY[1..31] OF BOOL;
    UDInt: INT;
    UDHex: INT;
    UDReal: INT;
END_VAR
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters 810D, 840D

The following table shows all formal parameters of the function RUN_UP for 810D, 840D.

Signal	Type	Type	Value range	Notes
MCPNum	I	Int	0 to 2	Number of active MCPs 0: No MCP installed
MCP1In MCP2In	I	Pointer	I0.0 to I120.0 or F0.0 to F248.0 or DBn DBX0.0 to DBXm.0	Start address for input signals of the relevant machine control panel
MCP1Out MCP2Out	I	Pointer	Q0.0 to Q120.0 or F0.0 to F248.0 or DBn DBX0.0 to DBXm.0	Start address for output signals of the relevant machine control panel
MCP1StatSend MCP2StatSend	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn DBX0.0 to DBXm.0	Start address for status double word for transmission to machine control panel: DW#16#08000000:Time out, otherwise 0
MCP1StatRec MCP2StatRec	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn DBX0.0 to DBXm.0	Start address for status double word for data reception by machine control panel: DW#16#00040000:Timeout, otherwise 0
MCP1BusAdr MCP2BusAdr	I	Int	1...15	Bus address of machine control panel
MCP1Timeout MCP2Timeout	I	S5time	Recommendation: 700 ms	Cyclic sign-of-life monitoring for machine control panel
MCP1Cycl MCP2Cycl	I	S5time	Recommendation: 200 ms	Time reference for cyclic updating of signals to machine control panel
MCPMPI	I	Bool		1: All machine control panels connected to MPI bus (without GD parameterization), SW 4.x and higher
MCP1Stop MCP2Stop	I	Bool		0: Start transfer of machine control panel signals 1: Stop transfer of machine control panel signals
MCP1NotSend MCP2NotSend	I	Bool		0: Send and receive operation activated 1: Receive machine control panel signals only (available soon) (SW 4 and higher)
HHU	I	Int		HHU interface 0: No HHU 1: HHU to MPI with SDB 210 configuration (for SW 3.x) 2: HHU to OPI or MPI if FB1 parameter BHGMPI is also set to TRUE (SW 4.x and higher).
BHGIn	I	Pointer	I0.0 to I124.0, F0.0 to F252.0 or DBn DBX0.0 to DBXm.0	Start address Receive data of PLC from HHU Receive data of PLC from HHU
BHGOut	I	Pointer	Q0.0 to Q124.0, F0.0 to F252.0 or DBn DBX0.0 to DBXm.0	Start address Transmit data of PLC to HHU Transmit data of PLC to HHU
BHGStatSend	I	Pointer	Q0.0 to Q124.0, F0.0 to F252.0 or DBn DBX0.0 to DBXm.0	Start address for status double word for transmitting data to HHU: DW#16#08000000: Time out, otherwise 0

© Siemens AG, 2003. All rights reserved

Signal	Type	Type	Value range	Notes
BHGStatRec	I	Pointer	Q0.0 to Q124.0, F0.0 to F252.0 or DBn DBX0.0 to DBXm.0	Start address for status double word for receiving data from HHU: DW#16#00040000: Time out, otherwise 0
BHGInLen	I	Byte	HHU default: B#16#6 (6 Byte)	Quantity of data received from handheld unit
BHGOutLen	I	Byte	HHU default: B#16#6 (20 Byte)	Quantity of data transmitted to handheld unit
BHGTimeout	I	S5time	Recommendation: 700 ms	Cyclical sign-of-life monitoring for handheld unit
BHGCycl	I	S5time	Recommendation: 100 ms	Time reference for cyclical updating of signals to handheld unit
BHGRecGDNo	I	Int	HHU default: 2	Receive GD circle no.
BHGRecGBZNo	I	Int	HHU default: 2	Receive GI no.
BHGRecObjNo	I	Int	HHU default: 1	Object number for receive GI
BHGSendGDNo	I	Int	HHU default: 2	Transmit GD circle no.
BHGSendGBZNo	I	Int	HHU default: 1	Transmit GI no.
BHGSendObjNo	I	Int	HHU default: 1	Object number for transmit GI
BHGMPI	I	Bool		1: Handheld unit coupled to MPI (without SDB 210 config.) Parameter HHU must be set to 2.
BHGStop	I	Bool		0: Transmission of Start handheld unit signals 1: Stop transmission of handheld unit signals
BHGNotSend	I	Bool		0: Send and receive operation activated 1: Receive handheld unit signals only (available soon) (SW 4 and higher)
NCCyclTimeout	I	S5time	Recommendation: 200 ms	Cyclical sign-of-life monitoring NCK
NCRunupTimeout	I	S5time	Recommendation: 50 s	Power-up monitoring NCK
ListMDecGrp	I	Int	0...16	Activation of extended M group decoding. 0 = not active 1 ...16: number of M groups
NCKomm	I	Bool		PLC NC communications services (FB 2/3/4/5/7: Put/Get/PI_SERV/GETGUD) True: Disable active
MMCToIF	I	Bool		Transmission of MMC signals to interface (modes, program control, etc.) True: Disable active
HWheelMMC	I	Bool		True: Handwheel selection via MMC False: Handwheel selection via user program
MsgUser	I	Int	0...25	Quantity user areas for messages (DB2)
UserIR	I	Bool		Local data expansion OB40 required for processing of signals from user
IRAuxfuT	I	Bool	Default, false	Evaluate T function in OB40
IRAuxfuH	I	Bool	Default, false	Evaluate H function in OB40
IRAuxfuE	I	Bool	Default, false	Evaluate DL function in OB40

© Siemens AG, 2003. All rights reserved

Signal	Type	Type	Value range	Notes
UserVersion	I	Pointer		Pointer to string variable. The associated string variable is output in the version display (max. 41 characters) (SW 4 and higher)
MaxBAG	O	INT	1..10	Number of mode groups (SW 4 and higher)
MaxChan	O	INT	1..10	Number of channels (SW 4 and higher)
MaxAxis	O	INT	1..31	Number of axes (SW 4 and higher)
ActivChan	O	ARRAY[1..10] OF BOOL		Bit string for active channels (SW 4 and higher)
ActivAxis	O	ARRAY[1..31] OF BOOL		Bit string for active axes (SW 4 and higher)
UDInt	O	Int		Number of integer machine data in DB20 (SW 4 and higher)
UDHex	O	Int		Number of hexadecimal machine data in DB20 (SW 4 and higher)
UDReal	O	Int		Number of real machine data in DB20 (SW 4 and higher)

Declaration 810D

```
FUNCTION_BLOCK FB 1

VAR INPUT
  MCPNum:          INT:=1;              //0: No MCP
                                        //1: 1 MCP (default)
                                        //2: 2 MCPs
  MCP1In:          POINTER;
  MCP1Out:         POINTER;
  MCP1StatRec:     POINTER;
  MCP1Timeout:     S5TIME:= S5T700MS;
  MCP2In:          POINTER;
  MCP2Out:         POINTER;
  MCP2StatRec:     POINTER;
  MCP2Timeout:     S5TIME:= S5T700MS;

  BHG:             INT:= 0;             //Handheld unit interface
                                        //0: No HHU
                                        //1: HHU at MPI
  BHGIn:           POINTER;             //Transmit data of handheld unit
  BHGOut:          POINTER;             //Receive data of handheld unit
  BHGStatRec:      POINTER;             //Status DW for reception HHU
  BHGTimeout:      S5TIME:=S5T700MS;

  NCCyclTimeout:   S5TIME:=S5T#200MS;
  NCRunupTimeout:  S5TIME:=S5T#50S;
  ListMDecGrp:     INT:= 0;
  NCKomm:          BOOL:= FALSE;
  MMCToIF:         BOOL:= TRUE;
  HWheelMMC:       BOOL:= TRUE; //Handwheel selections via MMC
  MsgUser:         INT:=10;             //No. of user areas in DB2
  UserIR:          bool:= FALSE         //User programs in OB40,
                                        //Note local data expansion !

  IRAuxfuT:        bool:= FALSE;        //Evaluate T function in OB40
  IRAuxfuH:        bool:= FALSE;        //Evaluate H function in OB40
  IRAuxfuE:        bool:= FALSE;        //Evaluate E function in OB40
  UserVersion:     Pointer;             //Pointer to string variable that is output in
                                        //version display
```

© Siemens AG, 2003. All rights reserved

```
END_VAR
VAR OUTPUT
  MaxBAG:        INT;
  MaxChan:       INT;
  MaxAxis:       INT;
  ActivChan:     ARRAY[1..10] OF BOOL;
  ActivAxis:     ARRAY[1..31] OF BOOL;
  UDInt: INT;
  UDHex: INT;
  UDReal: INT;
END_VAR
```

Explanation of the formal parameters 810D and 840D

The Table below shows all formal parameters of the function RUN_UP for the 810D and 840D.

Signal	Type	Type	Value range	Notes
MCPNum	I	Int	0 to 2	Number of active MCPs 0: No MCP installed
MCP1In MCP2In	I	Pointer	I0.0 to I120.or0 F0.0 to F248.0 or DBn.DBX0.0 to DBXm.0	Start address for input signals of the relevant machine control panel 1)
MCP1Out MCP2Out	I	Pointer	Q0.0 to Q120.0 or F0.0 to F248.0 or DBn.DBX0.0 to DBXm.0	Start address for output signals of the relevant machine control panel 1)
MCP1StatSend MCP2StatSend	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for data transmission from machine control panel: DW#16#00080000:Timeout, otherwise 0 1)
MCP1StatRec MCP2StatRec	I	Pointer	Q0.0 to Q124.0, F0.0 to F252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for data reception by machine control panel: DW#16#00040000:Timeout, otherwise 0 1)
MCP1BusAdr MCP2BusAdr	I	Int	1 ... 15	Bus address of machine control panel
MCP1Timeout MCP2Timeout	I	S5time	Recommendation: 700 ms	Cyclic sign-of-life monitoring for machine control panel
MCP1Cycl MCP2Cycl	I	S5time	Recommendation: 200 ms	Time reference for cyclic updating of sig- nals to machine control panel
MCPMPI	I	Bool		1: All machine control panels connected to MPI bus (without GD parameterization)
MCP1Stop MCP2Stop	I	Bool		0: Start transfer of machine control panel signals 1: Stop transfer of machine control panel signals
MCP1NotSend MCP2NotSend	I	Bool		0: Send and receive operation activated 1: Receive machine control panel signals only (SW 4 and higher)
HHU	I	Int		Handheld unit interface 0 - No HHU 1 - HHU to MPI with SDB 210 configuration
BHGIn	I	Pointer	I0.0 to I124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address Receive data of PLC from HHU data from handheld unit 2)

© Siemens AG, 2003. All rights reserved

Signal	Type	Type	Value range	Notes
BHGOut	I	Pointer	Q0.0 to Q124.0, F0.0 to F252.0 or DBn.DBX0.0 to DBXm.0	Start address Transmit data of PLC to HHU to HHU 2)
BHGStatSend	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for sending data to handheld unit: DW#16#08000000: Timeout, otherwise 0 2)
BHGStatRec	I	Pointer	Q0.0 to Q124.0, F0.0 to F252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for receiving data from handheld unit: DW#16#00040000: Timeout, otherwise 0 2)
BHGInLen	I	BYTE	HHU default: B#16#6 (6 Byte)	Quantity of data received from handheld unit
BHGOutLen	I	Byte	HHU default: B#16#6 (10 Byte)	Quantity of data transmitted to handheld unit
BHGTimeout	I	S5time	Recommendation: 700 ms	Cyclical sign-of-life monitoring for handheld unit
BHGCycl	I	S5time	Recommendation: 100 ms	Time reference for cyclical updating of signals to handheld unit
BHGRecGDNo	I	Int	HHU default: 1	Receive GD circle no.
BHGRecGBZNo	I	Int	HHU default: 2	Receive GI no.
BHGRecObjNo	I	Int	HHU default: 1	Object number for receive GI
BHGSendGDNo	I	Int	HHU default: 2	Transmit GD circle no.
BHGSendGBZNo	I	Int	HHU default: 1	Transmit GI no.
BHGSendObjNo	I	Int	HHU default: 1	Object number for transmit GI
BHGMPI	I	Bool		1: Handheld unit coupled to MPI (without SDB 210 config.) Parameter HHU must be set to 2.
BHGStop	I	Bool		0: Start transfer of HHU signals 1: Stop transfer of HHU signals
BHGNotSend	I	Bool		0: Send and receive operation activated 1: Receive HHU signals only (SW 4 and higher)
NCCyclTimeout	I	S5time	Recommendation: 200 ms	Cyclical sign-of-life monitoring Monitoring NCK
NCRunupTimeout	I	S5time	Recommendation: 50 s	Power-up monitoring NCK
ListMDecGrp	I	INT	0...16	
NCKomm	I	Bool		PLC NC communications services (FB 2/3/4/5/7: Put/Get/PI_SERV/GETGUD) 1: Active
MMCToIF	I	Bool		Transmission of MMC signals to interface (operating modes, program control, etc.) True: Active
HWheelMMC	I	Bool		True: Handwheel selection via MMC False: Handwheel selection by user prog.
MsgUser	I	Int	0...25	Quantity user areas for messages (DB2)

© Siemens AG, 2003. All rights reserved

Signal	Type	Type	Value range	Notes
UserIR	I	Bool		Local data expansion OB40 required for processing of signals from user
IRAuxfuT	I	Bool		Evaluate T function in OB40
IRAuxfuH	I	Bool		Evaluate H function in OB40
IRAuxfuE	I	Bool		Evaluate DL function in OB40
UserVersion	I	Pointer		Pointer to string variable. The associated string variable is output in the version display (max. 54 characters)
MaxBAG	O	INT	1..10	Number of mode groups
MaxChan	O	INT	1..10	Number of channels
MaxAxis	O	INT	1..31	Number of axes
ActivChan	O	ARRAY[1..10] OF BOOL		Bit string for active channels
ActivAxis	O	ARRAY[1..31] OF BOOL		Bit string for active axes
UDInt	O	Int		Number of integer machine data in DB20
UDHex	O	Int		Number of hexadecimal machine data in DB20
UDReal	O	Int		Number of Real machine data in DB20

1) To ensure BP monitoring of the MCP(s), the addresses must be specified as set in SDB 210 on the 810D. The start address is set via SDB 210 on the 810D. On the supplied SDB 210, the start address is specified as IB 0 for input signals and QB 0 for output signals. If a different start address is required, then it must be programmed by means of the STEP7 Communication Configuration package.
2) To ensure that the handheld unit is monitored by the BP, the addresses must be specified on the 810D as they are set in SDB 210.

Declaration FM-NC

```
FUNCTION_BLOCK FB 1
  MCPNum:        INT:= 1;            //0: No MCP
                                     //1: 1 MCP (default)
                                     //2: 2 MCP
  MCP1In:        POINTER;
  MCP1Out:       POINTER;
  MCP1StatRec:   POINTER;
  MCP1Timeout:   S5TIME:= S5T700MS;

  MCP2In:        POINTER;
  MCP2Out:       POINTER;
  MCP2StatRec:   POINTER;
  MCP2Timeout:   S5TIME:= S5T700MS;

  BHG:           INT:= 0;            //Handheld unit interface
                                     //0: No HHU
                                     //1: HHU at MPI
  BHGIn:         POINTER;            //Transmit data of handheld unit
  BHGOut:        POINTER;            //Receive data of handheld unit
  BHGStatRec:    POINTER;            //Status DW for reception HHU
  BHGTimeout:    S5TIME:= S5T700MS;
```

© Siemens AG, 2003. All rights reserved

```
NCLaddr1:        INT:= 320;          //I/O address of first FM-NC (default)
NCLaddr2:        INT:= 0;            //I/O address of second FM-NC (default)
NCKSigDB2:       INT:= 0;            //NCK signal data block of 2nd FM-NC
NCCyclTimeout:   S5TIME:= S5T#200MS;
NCRunupTimeout:  S5TIME:= S5T#5MS;
ListMDecGrp:     INT:= 0;
NCKomm:          BOOL:= FALSE;

MMCToIF:         BOOL:= TRUE;
HWheelMMC:       BOOL:= TRUE;        //Handwheel selections via MMC
MsgUser:         INT:= 10;           //No. of user areas in DB2
UserVersion:     DWORD:= 0;          //V no. of user program (default)
UserDate:        DWORD:= 0;          //V date of user program (default)
UserTime:        DWORD:= 0;          //V. time of user program (default)
END_VAR
```

Explanation of the formal parameters FM-NC

The following table shows all formal parameters of the function RUN_UP for the FM-NC.

Signal	Type	Type	Value range	Notes
MCPNum	I	Int	0 to 2	Number of active MCPs 0: No MCP installed
MCP1In MCP2In	I	Pointer	I0.0 to I120.or0 F0.0 to F248.0 or DBn.DBX0.0 to DBXm.0	Start address for input signals of the relevant machine control panel 1)
MCP1Out MCP2Out	I	Pointer	Q0.0 to Q120.0 or F0.0 to F248.0 or DBn.DBX0.0 to DBXm.0	Start address for output signals of the relevant machine control panel 1)
MCP1StatRec MCP2StatRec	I	Pointer	Q0.0 to Q124.0, F0.0 to F252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for data reception by machine control panel: control panel: DW#16#00040000: Timeout, otherwise 0 1)
MCP1Timeout MCP2Timeout	I	S5time	Recommendation: 700 ms	Cyclic sign-of-life monitoring for machine control panel
HHU	I	Int		Handheld unit interface 0: No HHU 1: HHU at MPI
BHGIn	I	Pointer	I0.0 to I124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address Receive data of PLC from HHU data from handheld unit 2)
BHGOut	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address Transmit data of PLC to HHU to HHU 2)
BHGStatRec	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for receiving from handheld unit: DW#16#00040000: Timeout, otherwise 0 2)
BHGTimeout	I	S5time	Recommendation: 700 ms	Cyclical sign-of-life monitoring for handheld unit
NCLaddr1	I	INT	320 (default)	I/O address of 1st FM-NC
NCLaddr2	I	INT	0 (default)	I/O address of 2nd FM-NC
NCKSigDB2	DB	INT	DB 81 to DB 127	Signal DB of second FM–NC
NCCyclTimeout	I	S5time	Recommendation: 200 ms	Cyclical sign-of-life monitoring NCK
NCRunupTimeout	I	S5time	Recommendation: 5 min	Power-up monitoring NCK
ListMDecGrp	I	INT	0	

© Siemens AG, 2003. All rights reserved

Signal	Type	Type	Value range	Notes
NCKomm	I	Bool		PLC NC communications services (FB 2/3/4/5:Put/Get/PI_SERV/GETGUD) 1: Active
MMCToIF	I	Bool		Transmission of MMC signals to interface (modes, program control, etc.) True: Active
HWheelMMC	I	Bool		True: Handwheel selection via MMC False: Handwheel selection by user prog.
MsgUser	I	Int	0...25	Quantity user areas for messages (DB2)
UserVersion	I	Dword		User version
UserDate	I	Dword		User date
UserTime	I	Dword		User time

1)
For monitoring of MCPs by the BP, the addresses must be specified for the FM-NC as they are set in SDB210. The start address is set via SDB210 for the FM-NC. The start address for input signals is specified as 118 and for output signals as 120 on the supplied SDB210. If another start address is desired, then it must be input via the STEP7 package Communications Configuration.
2)
To ensure that the handheld unit is monitored by the BP, the addresses as they are set in SDB 210 must be specified in the case of the FM-NC.

Monitoring MCP/HHU (for 810D, 840D and FM-NC)

The following status information regarding communication with the machine control panels is output in the event of an error:

Available in:	Bit No.	Description
MCP1StatRec MCP2StatRec BHGStatRec	10	Receiver: Time out
Only SINUMERIK 840D: MCP1StatSend MCP2StatSend BHGStatSend	27	Sender: Time out

In addition, an error entry is generated in the diagnostics buffer of the PLC, resulting in the following error messages on the MMC:

- 400260: MCP 1 failure or
- 400261: MCP 2 failure.
- 400262: HHU failure.

In this case, the input signals from the MCP or from the handheld unit (MCP1In/MCP2In or BHGIn) are initialized with 0. If it is possible to resynchronize the PLC and MCP/HHU, communication is resumed automatically and the error message reset by the BP.

© Siemens AG, 2003. All rights reserved

810D call

A call example for the FB 1 in OB 100 is given below. This example is part of the diskette with the basic program for 810D.

```
ORGANIZATION_BLOCK OB 100
VAR_TEMP
  OB100_EV_CLASS :          BYTE;
  OB100_STRTUP :            BYTE;
  OB100_PRIORITY :          BYTE;
  OB100_OB_NUMBR :          BYTE;
  OB100_RESERVED_1 :        BYTE;
  OB100_RESERVED_2 :        BYTE;
  OB100_STOP :              WORD;
  OB100_RESERVED_3 :        WORD;
  OB100_RESERVED_4 :        WORD;
  OB100_DATE_TIME :         DATE_AND_TIME;
END_VAR
BEGIN
  Call fb 1, db 7(
        MCPNum :=          1,
        MCP1In :=          P#E0.0,
        MCP1Out :=         P#A0.0,
        MCP1StatSend.=     P#A8.0,
        MCP1StatRec :=     P#A12.0,
        MCP1BusAdr :=      14,
        MCP1Timeout :=     S5T#700MS,
        MCPMPI:=           TRUE,
        NCCyclTimeout :=   S5T#200MS,
        NCRunupTimeout := S5T#50S);
//INSERT USER PROGRAM HERE
END_ORGANIZATION_BLOCK
```

840D call

A call example for the FB 1 in OB 100 is given below. This example is part of the diskette with basic program for 840D.

```
ORGANIZATION_BLOCK OB 100
VAR_TEMP
  OB100_EV_CLASS :          BYTE;
  OB100_STRTUP :            BYTE;
  OB100_PRIORITY :          BYTE;
  OB100_OB_NUMBR :          BYTE;
  OB100_RESERVED_1 :        BYTE;
  OB100_RESERVED_2 :        BYTE;
  OB100_STOP :              WORD;
  OB100_RESERVED_3 :        WORD;
  OB100_RESERVED_4 :        WORD;
  OB100_DATE_TIME :         DATE_AND_TIME;
END_VAR
BEGIN
  Call fb 1, db 7(
        MCPNum :=          1,
        MCP1In :=          P#E0.0,
        MCP1Out :=         P#A0.0,
        MCP1StatSend :=    P#A8.0,
        MCP1StatRec :=     P#A12.0,
        MCP1BusAdr :=      6,
        MCP1Timeout :=     S5T#700MS,
        MCP1Cycl :=        S5T#200MS,
        NCCyclTimeout :=   S5T#200MS,
        NCRunupTimeout := S5T#50S);
//INSERT USER PROGRAM HERE
END_ORGANIZATION_BLOCK
```

© Siemens AG, 2003. All rights reserved

Call example for FM-NC

A call example for the FB 1 in OB 100 is given below. This example is an integral part of the diskette with the basic program for SINUMERIK FM-NC.

```
ORGANIZATION_BLOCK OB 100

VAR_TEMP
  OB100_EV_CLASS :          BYTE;
  OB100_STRTUP :            BYTE;
  OB100_PRIORITY :          BYTE;
  OB100_OB_NUMBR :          BYTE;
  OB100_RESERVED_1 :        BYTE;
  OB100_RESERVED_2 :        BYTE;
  OB100_STOP :              WORD;
  OB100_STRT_INFO:          DWORD;
  OB100_DATE_TIME :         DATE_AND_TIME;
END_VAR
BEGIN
  Call fb 1, db 7(
            MCPNum :=          1,
            MCP1In :=          P#E118.0,
            MCP1Out :=         P#A120.0,
            MCP1StatRec :=     P#A108.0,
            MCP1Timeout :=     S5T#700MS,
            NCCyclTimeout :=   S5T#200MS,
            NCRunupTimeout :=  S5T#5M,
            NCLaddr1 :=        320,
            NCLaddr2 :=        0,
            NCKSigDB2 :=       0,
            UserVersion :=     DW#16#0,
            UserDate :=        DW#16#0,
            UserTime :=        DW#16#0);
  //INSERT USER PROGRAM HERE
END_ORGANIZATION_BLOCK
```

© Siemens AG, 2003. All rights reserved

4.2 FB 2: Read GET NC variables

Description of functions

The PLC user program can read variables can be read from the NCK area using FB GET.

FB 2 also includes an Instance DB from the user area. (multi-instance capability in SW 3.7 and higher).

When FB 2 is called with a positive signal edge change at control input “Req” a job is started which reads the NCK variables referenced by ADDR1ADDR8 and then copies them to the PLC operand areas referenced by RD1 to RD8. Successful completion of the read process is indicated by a logical “1” in status parameter NDR.

The read process lasts for several PLC cycles (normally 1-2). The block can be called up in cyclic mode only.

Any errors are indicated by Error and State.

In order to reference the NCK variables, all required variables are first selected with the “NC Var Selector” tool (see also Section 3.2 of this documentation) and generated as an STL source in a data block. A name must then be assigned to this DB in the signal list.
“DB name.S7” name is transferred as the actual parameter of the NCK variable address (Addr1 to Addr8) when FB3 is called.

Variable addressing

For some NC variables, it is necessary to select area no. and/or line or column from the NCVAR selector. A basic type can be selected for these variables, i.e. area/column/line are preset to 0.
A basic type can be selected for these variables, i.e. area/column/line are preset to “0”.
The contents of the area number, line and column specified by the NC VAR selector are checked for a “0” in the FB. If a “0” is present, the value is transferred to the input parameter. Before calling the GET FB, the user must supply the desired parameter (UnitX/ColumnX/LineX).

!

Important

After communication between the PLC and NC (read/write NC variables, FB2, 3, 5, or PI general services, FB4) has been aborted by power OFF, the start jobs must be deleted in the first OB1 run after cold restart or reset (signal: Req = 0).

FB 2 can read NC variables only if basic program parameter NCKomm =“1” has been set (in OB 100: FB 1, DB7).
When **channel-specific** variables are read, only the variables of **one** channel may be addressed in a job (FB 2 call) via Addr1 to Addr8.

In areas V and H, different logic axis numbers must not be assigned in one job. (Failure to observe this rule results in Error:= TRUE, State:= W#16#02).

© Siemens AG, 2003. All rights reserved

NCK variables within **one** group can be combined in a job:

	Range				
Group 1	C[1]	N	B	A	T
Group 2	C[2]	N	B	A	T
Group 3	V[.]	H[.]			

The same rules apply to channels 3 to 10 as illustrated as examples in the above table in groups 1 and 2.

Note

Especially when reading several long strings, the number of usable variables can be less than 8.

Declaration

```
FUNCTION_BLOCK FB 2
VAR_INPUT
   Req :          BOOL ;
   NumVar :       INT ;
   Addr1 :        ANY ;
   Unit1 :        BYTE ;
   Column1 :      WORD ;
   Line1 :        WORD ;
   Addr2 :        ANY ;
   Unit2 :        BYTE ;
   Column2 :      WORD ;
   Line2 :        WORD ;
   Addr3 :        ANY ;
   Unit3 :        BYTE ;
   Column3 :      WORD ;
   Line3 :        WORD ;
   Addr4 :        ANY ;
   Unit :         BYTE ;
   Column4 :      WORD ;
   Line4 :        WORD ;
   Addr5 :        ANY ;
   Unit5 :        BYTE ;
   Column5 :      WORD ;
   Line5 :        WORD ;
   Addr6 :        ANY ;
   Unit6 :        BYTE ;
   Column6 :      WORD ;
   Line6 :        WORD ;
   Addr7 :        ANY ;
   Unit7 :        BYTE ;
   Column7 :      WORD ;
   Line7 :        WORD ;
   Addr8 :        ANY ;
   Unit :         BYTE ;
   Column8 :      WORD ;
   Line8 :        WORD ;
   FM-NCNo :      int;1)
END_VAR
VAR_OUTPUT
```

© Siemens AG, 2003. All rights reserved

```
    Error :         BOOL;
    NDR :           BOOL;
    State :         WORD;
END_VAR
```

1) On FM-NC only

© Siemens AG, 2003. All rights reserved

```
VAR_IN_OUT
  RD1 :          ANY;
  RD2 :          ANY;
  RD3 :          ANY;
  RD4 :          ANY;
  RD5 :          ANY;
  RD6 :          ANY;
  RD7 :          ANY;
  RD8 :          ANY;
END_VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the function GET.

Signal	Type	Type	Value range	Notes
Req	I	Bool		Job start with positive signal edge
NumVar	I	Int	1 to 8 (corresponds to use of Addr1 to Addr8)	Number of variables to be read
Addr1 to Addr8	I	Any	[DBName].[VarName]	Variable identifiers from **NC Var selector**
Unit1 to Unit8	I	Byte		Area address, optional for variable addressing
Column1 to Column8	I	Word		Column address, optional for variable addressing
Line1 to Line8	I	Word		Line address, optional for variable addressing
FM-NCNo[1)]	I	Int	0, 1, 2	0, 1=1 NCU, 2=2 NCU's
Error	O	Bool		Negative acknowledgment of job or execution of job impossible
NDR	O	Bool		Job successfully executed. Data are available
State	O	Word		See error identifiers
RD1 to RD8	I/O	Any	P#Mm.n BYTE x... P#DBnr.dbxm.n BYTE x	Target area for read data

1) On FM-NC only

Error identifiers

If it was not possible to execute a job, the failure is indicated by "logic 1" on status parameter error. The error cause is coded at the block output State:

State		Meaning	Note
WORD H	WORD L		
1 to 8	1	Access error	In high byte number of Var in which error occurred
0	2	Error in job	Incorrect compilation of Var. in a job
0	3	Negative acknowledgment, job not executable	Internal error, possible remedy: NC Reset
1 to 8	4	Insufficient local user memory available	Read var. is longer than specified in RD1 (..RD8); in high byte number of var in which error occurred

© Siemens AG, 2003. All rights reserved

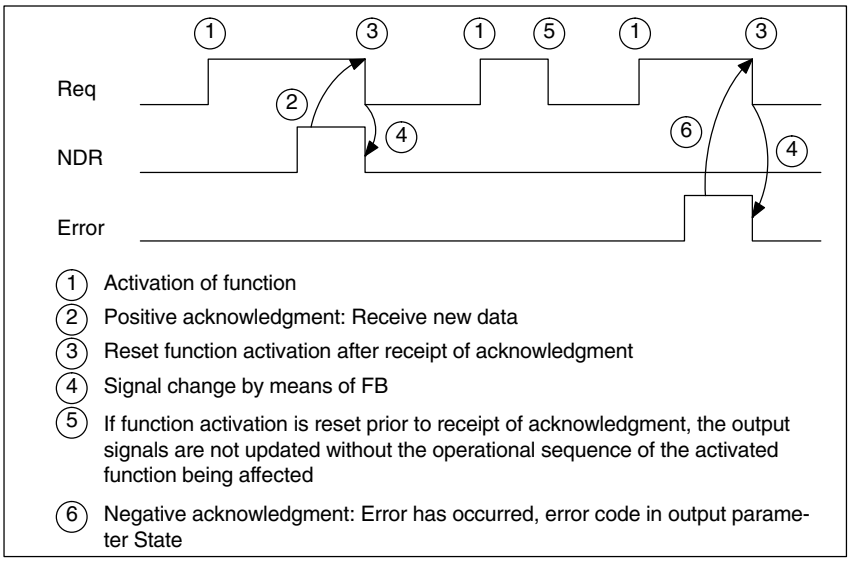
State		Meaning	Note
WORD H	WORD L		
0	5	Format conversion error	Error on conversion of var. type double: Var. is not within S7 REAL area
0	6	FIFO full	Job must be repeated since queue is full
0	7	Option not set	BP parameter “NCKomm” is not set
1 to 8	8	Incorrect target area (RD)	RD1 to RD8 may not be local data
0	9	Transmission occupied	Job must be repeated
1 to 8	10	Error in variable addressing	Unit or column/line contains value 0
0	11	Address of variable invalid	Check Addr (or variable name), area, unit
0	12	NumVar = 0	Check parameter NumVar

Configuration steps

The following steps are required for reading NC variables:

- Select variables with the NCVAR selector,
- Store the selected variables in a file *.VAR in the desired project catalog (*.S7D),
- Generate a STEP7 source file *.STL,
- Generate a DB with the associated address data,
- Enter the symbol for the generated DB in the symbol table so that it is possible to access the address parameters symbolically in the user program,
- Set FB2 parameters.

Pulse diagram



© Siemens AG, 2003. All rights reserved

Call example

Reading of three channel-specific machine data from channel 1, whose address specification are stored in DB120.

Select data with NC VAR selector and store in file DB120.VAR; then create file DB120.AWL:

Range	Block	Name	Type	No.	Byte	S7 Name
C[1]	M	MD 20070: AXCONF_MACHAX_USED[1]	char	20070	1	C1AxConfMachAxUsed1
C[1]	M	MD 20070: AXCONF_MACHAX_USED[2]	char	20070	1	C1AxConfMachAxUsed2
C[1]	M	MD 20090: SPIND_DEF_MASTER_SPIND	int	20090	1	C1SpindDefMasterSpind

S7 (ALIAS) names have been selected in order to

A. incorporate the channel designation into the name and

B. to remove the characters [ ] which are not legal in a STEP 7 symbol.

Entry of the name in the S7 SYMBOL table (e.g. NCVAR for DB120):

Symbol	Operand	Data type
NCVAR	DB120	DB120

File DB120.AWL must be compiled and transferred to the PLC.

Parameterization of FB 2 with instance DB 110:

```
DATA_BLOCK DB 110                    //Unassigned user DB, as instance for FB 2
FB 2
BEGIN
END_DATA_BLOCK
Function FC "VariablenCall" : VOID
  I 7.7;                             //Unassigned key on MCP
  S      M 100.0:                    //Activate Req
  A      F 100.1;                    //Done message NDR
  R      M 100.0:                    //End job
  A      I 7.6;                      //Acknowledge error manually
  A      M 102.0;                    //Error is present
  R      M 100.0:                    //End job
  Call fb 2, db 110(
            Req :=      F 100.0,
            NumVar :=   3,                  //Read 3 variables
            Addr1 :=    NCVAR.C1AxConfMachAxUsed1,
            Addr2 :=    NCVAR.C1AxConfMachAxUsed2,
            Addr3 :=    NCVAR.C1SpindDefMasterSpind,
            Error :=    F102.0,
            NDR :=      F100.1,
            State :=    FW104,
            RD1 :=      P#DB99.DBX0.0 BYTE 1,
            RD2 :=      P#DB99.DBX1.0 BYTE 1,
            RD3 :=      P#M110.0 INT 1);
```

© Siemens AG, 2003. All rights reserved

Example: Variable addressing

Reading of two R parameters from channel 1, whose address specifications are stored in DB120 as the basic type. The R parameter number is parameterized via parameter lineX.

```
DATA_BLOCK DB 120
VERSION : 0.0

STRUCT

  C1_RP_rpa0_0:
  STRUCT
  SYNTAX_ID :        BYTE := B#16#82;
  area_&_unit :      byte := B#16#41;
  column :           word := W#16#1;
  line :             word := W#16#0;
  block type :       byte := B#16#15;
  NO. OF LINES :     BYTE := B#16#1;
  typ :              byte := B#16#F;
  length :           byte := B#16#8;
  END_STRUCT ;

END_STRUCT ;

BEGIN
END_DATA_BLOCK

  CALL FB 2, DB 110 (
              Req :=          M 0.0,
              NumVar :=       2,
              Addr1 :=        “NCVAR” C1_RP_rpa0_0,
              Line1 :=        W#16#1,
              Addr1 :=        “NCVAR” C1_RP_rpa0_0,
              Line2 :=        W#16#2,
              FM-NCNo :=      1,                  //On FM-NC only
              Error :=        F 1.0,
              NDR :=          F 1.1,
              State :=        FW 2,
              RD1 :=          P#M 4.0 REAL 1,
              RD2 :=          P#M 24.0 REAL 1);
```

Data types

The data types of the NCK are listed in the NCVAR selector with the variables. The tables below give the assignments to the S7 data types.

Table 4-1 Classification of data types

NCK data type	S7 data type
double	REAL
float	REAL
long	DINT
integer	DINT
uint_32	DWORD
int_16	INT
uint_16	WORD
unsigned	WORD
char	CHAR or BYTE
string	STRING
bool	BOOL

© Siemens AG, 2003. All rights reserved

4.3 FB 3: PUT Write NC variables

Description of functions

The PLC user program can write variables can be read from the NCK area using FB PUT.

Every FB 3 call must be assigned a separate instance DB from the user area. (multiinstance capability in SW 3.7 and higher).

When FB 3 is called with a positive signal edge change at control input Req, a job is started to overwrite the NC variables referenced by Addr1 to Addr8 with the data of the PLC operand areas locally referenced by SD1 to SD8. Successful completion of the write process is indicated by a logical “1” in status parameter Done.

The write process lasts for several PLC cycles (normally 1-2). The block can be called up in cyclic mode only.

Any errors are indicated by Error and State.
In order to reference the NCK variables, all required variables are first selected with the “NC Var Selector” tool and generated as an STL source in a data block. A name must then be assigned to this DB in the signal list. “DB name.S7 name” is transferred as the actual parameter of the NCK variable address (Addr1 to Addr8) when FB3 is called.

Variable addressing

For some NC variables, it is necessary to select area no. and/or line or column from the NCVAR selector. A basic type can be selected for these variables, i.e. area/column/line are preset to 0.
A basic type can be selected for these variables, i.e. area/column/line are preset to “0”.
The contents of the area number, line and column specified by the NC VAR selector are checked for a “0” in the FB. If a “0” is found, then the value from the input parameter is accepted. Before calling the PUT FB, the user must supply the desired parameter (UnitX/ColumnX/LineX).

Machine data, GUD

In order to define machine data and GUDs without a password, the protection levels of the data you want to access must be redefined to the lowest level. The procedure is described in the Installation Guide (in the section describing the protection level concept) and in the Programming Guide Advanced (protection levels for user data).

! **Important**

After communication between the PLC and NC (read/write NC variables, FB2, 3, 5, or PI general services, FB4) has been aborted by POWER OFF, the start jobs must be deleted in the first OB1 run after cold restart or reset (signal: Req = 0).

FB 3 can write NC variables only if basic program parameter NCKomm has been set to “1” (in OB100: FB 1, DB7). When **channel-specific** variables are written, only variables from **one** and the same channel may be addressed via Addr1 to Addr8 in a job (FB 3 call).

In areas V and H, different logic axis numbers must not be assigned in one job. (Failure to observe this rule results in Error:= TRUE, State:= W#16#02).

© Siemens AG, 2003. All rights reserved

NCK variables within **one** group can be combined in a job:

	Range				
Group 1	C[1]	N	B	A	T
Group 2	C[2]	N	B	A	T
Group 3	V[.]	H[.]			

The same rules apply to channels 3 to 10 as illustrated as examples in the above table in groups 1 and 2.

Note

Especially when reading several long strings, the number of usable variables can be less than 8.

Declaration

```
FUNCTION_BLOCK FB 3
VAR_INPUT
  Req :        BOOL ;
  NumVar :     INT ;
  Addr1 :      ANY ;
  Unit1 :      BYTE ;
  Column1 :    WORD ;
  Line1 :      WORD ;
  Addr2 :      ANY ;
  Unit2 :      BYTE ;
  Column2 :    WORD ;
  Line2 :      WORD ;
  Addr3 :      ANY ;
  Unit3 :      BYTE ;
  Column3 :    WORD ;
  Line3 :      WORD ;
  Addr4 :      ANY ;
  Unit :       BYTE ;
  Column4 :    WORD ;
  Line4 :      WORD ;
  Addr5 :      ANY ;
  Unit5 :      BYTE ;
  Column5 :    WORD ;
  Line5 :      WORD ;
  Addr6 :      ANY ;
  Unit6 :      BYTE ;
  Column6 :    WORD ;
  Line6 :      WORD ;
  Addr7 :      ANY ;
  Unit7 :      BYTE ;
  Column7 :    WORD ;
  Line7 :      WORD ;
  Addr8 :      ANY ;
  Unit :       BYTE ;
  Column8 :    WORD ;
  Line8 :      WORD ;
  FM-NCNo:     INT;          //On FM-NC only
END_VAR
```

© Siemens AG, 2003. All rights reserved

```
VAR_OUTPUT
   Error :          BOOL;
   Done :           BOOL;
   State :          WORD;
END_VAR
VAR_IN_OUT
   SD1 :            ANY;
   SD2 :            ANY;
   SD3 :            ANY;
   SD4 :            ANY;
   SD5 :            ANY;
   SD6 :            ANY;
   SD7 :            ANY;
   SD8 :            ANY;
END_VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the function PUT.

Signal	Type	Type	Value range	Notes
Req	I	Bool		Job start with positive signal edge
NumVar	I	Int	1 to 8 (corresponds to use of Addr1 to Addr8)	Number of variables to be written
Addr1 to Addr8	I	Any	[DBName].[VarName]	Variable identifiers from **NC Var selector**
Unit 1 to Unit 8	I	Byte		Area address, optional for variable addressing
Column 1 to Column 8	I	Word		Column address, optional for variable addressing
Line 1 to Line 8	I	Word		Line address, optional for variable addressing
FM-NCNo[1)]	I	Int	0, 1, 2	0, 1=1 NCU, 2=2 NCU's
Error	A	Bool		Negative acknowledgment of job or execution of job impossible
Done	A	Bool		Job successfully executed.
State	A	Word		See error identifiers
SD1 to SD8	I/O	Any	P#Mm.n BYTE x... P#DBnr.dbxm.n BYTE x	Data to be written

1) On FM-NC only

Error identifiers

If it was not possible to execute a job, the failure is indicated by "logic 1" on status parameter error. The error cause is coded at the block output State:

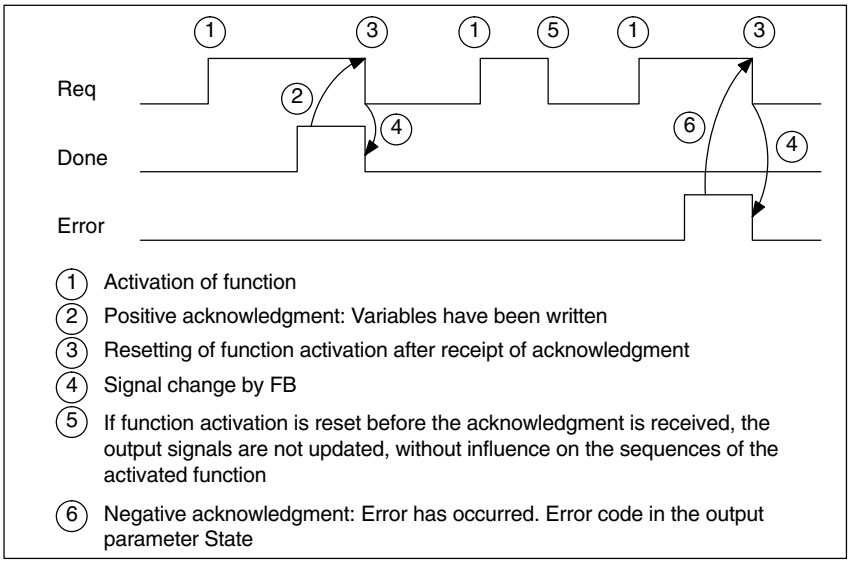
© Siemens AG, 2003. All rights reserved

State		Meaning	Note
WORD H	WORD L		
1 to 8	1	Access error	In high byte number of Var in which error occurred
0	2	Error in job	Incorrect compilation of Var in a job
0	3	Negative acknowledgment, job not executable	Internal error, possible remedy: Check job, NC Reset
1 to 8	4	Data areas or data types do not match or string is empty	Check data to be written in SD1 to SD8; in high byte number of the Var in which error occurred
0	6	FIFO full	Job must be repeated since queue is full
0	7	Option not set	BP parameter “NCKomm” is not set
1 to 8	8	Incorrect target area (RD)	SD1 to SD8 may not be local data
0	9	Transmission occupied	Job must be repeated
1 to 8	10	Error in variable addressing	Unit or column/line contains value 0
0	11	Variable addr. invalid or var. is read only	Check Addr (or variable name), area, unit
0	12	NumVar = 0	Check parameter NumVar

Configuration steps

To write NC variables, the same configuration steps are required as for reading NC variables. It is useful to store the address data of all NC variables to be read or written in a DB.

Pulse diagram



© Siemens AG, 2003. All rights reserved

Call example

Writing of three channels-pecific machine data of channel 1:

Select the three data with NCK VAR selector and store in the file DB120.VAR:

Range	Block	Name	Type	Byte	S7 Name
C[1]	RP	rpa[5]	double	4	rpa_5C1RP
C[1]	RP	rpa[11]	double	4	rpa_11C1RP
C[1]	RP	rpa[14]	double	4	rpa_14C1RP

Entry NCVAR for DB 120 with the S7 SYMBOL Editor:

Symbol	Operand	Data type
NCVAR	DB120	DB120

File DB120.AWL must be compiled and transferred to the PLC.

Call and parameterization of FB 3 with instance DB 111:

```
DATA_BLOCK DB 111                    //Unassigned user DB, as instance for FB 3
FB 3
BEGIN
Function FC “VariablenCall” : VOID
END_DATA_BLOCK
  I 7.7;                             //Unassigned key on MCP
  S      M 100.0:                    //Activate Req
  A      M 100.1;                    //Done message
  R      M 100.0:                    //End job
  A      I 7.6;                      //Acknowledge error manually
  A      M 102.0;                    //Error is present
  R      M 100.0:                    //End job
  Call fb 3, db 111(
            Req :=          F 100.0,
            NumVar :=       3,                  //Write 3 variables
            Addr1 :=        NCVAR.rpa_5C1RP,
            Addr2 :=        NCVAR.rpa_11C1RP,
            Addr3 :=        NCVAR.rpa_14C1RP,
            FM-NCNo :=      1,                  //On FM-NC only
            Error :=        F102.0,
            Done :=         F100.1,
            State :=        FW104,
            SD1 :=          P#DB99.DBX0.0 REAL 1,
            SD2 :=          P#DB99.DBX4.0 REAL 1,
            SD3 :=          P#M110.0 REAL 1);
```

© Siemens AG, 2003. All rights reserved

Example: Variable addressing

Writing of two R parameters of channel 1, whose address specifications are stored in DB 120 as the basic type. The R parameter number is parameterized via parameter lineX.

```
DATA_BLOCK DB 120
VERSION : 0.0

STRUCT

   C1_RP_rpa0_0:
   STRUCT
   SYNTAX_ID :        BYTE := B#16#82;
   area_&_unit :      byte := B#16#41;
   column :           word := W#16#1;
   line :             word := W#16#0;
   block type :       byte := B#16#15;
   NO. OF LINES :     BYTE := B#16#1;
   typ :              byte := B#16#F;
   length :           byte := B#16#8;
   END_STRUCT ;

END_STRUCT ;

BEGIN
END_DATA_BLOCK

CALL FB 3, DB 122 (
                   Req :=          M 10.0,
                   NumVar :=       2,
                   Addr1 :=        “NCVAR”C1_RP_rpa0_0,
                   Line1 :=        W#16#1,
                   Addr1 :=        “NCVAR”C1_RP_rpa0_0,
                   Line3 :=        W#16#2,
                   FM-NCNo :=      1,                  //On FM-NC only
                   Error :=        F 11.0,
                   Done :=         F 11.1,
                   State :=        FW 12,
                   SD1 :=          P#M 4.0 REAL 1,
                   SD2 :=          P#M 24.0 REAL 1);
```

© Siemens AG, 2003. All rights reserved

4.4 FB 4: PI_SERV General PI services

Description of functions

The FB PI_SERV can be used to starts program instance services in the NCK area. The possible services are described in this section. The request via the PI service initiates the execution of a program section in the NCK which performs a specific function (e.g. searching for empty locations in a magazine with tool management).
A separate instance DB from the user area must be assigned to each FB 4 call. The multi-instance capability can also be applied in SW 3.7 and higher. The instructions are provided in the STEP7 documentation.
The specified service is referenced via parameter PIService. The selected PI service is supplied by means of the user-assignable additional input variables with different data types (Addr1 ... Addr4 for strings, WVar1 ... WVar10 for integers or Word variables). A job is started when FB 4 is called by means of a positive edge change at control input Req. Successful execution of the job is displayed by means of a logic “1” in status parameter Done. Any errors are indicated by Error and State.
Data block “PI” (DB 16) contains internal descriptions of the available PI services. A name must then be assigned to this DB in the signal list. “DBName.PIName” is transferred as the actual parameter for PIService when FB4 is called.

Execution of the PI Service lasts for several PLC cycles (normally 1-2). The block can be called up in cyclic mode only.

Note

After communication between the PLC and NC (read/write NC variables, FB2, 3, 5, or PI general services, FB4) has been aborted by POWER OFF, the start jobs must be deleted in the first OB1 run after cold restart or reset (signal: Req = 0).

FB 4 can start PI services only if the basic program parameter NCKomm has been set to “1” (in OB100: FB 1, DB7).

© Siemens AG, 2003. All rights reserved

Declaration

```
FUNCTION_BLOCK FB 4
VAR_INPUT
  Req :          BOOL;
  PIService :    ANY;
  Unit :         INT;
  Addr1 :        ANY;
  Addr2 :        ANY;
  Addr3 :        ANY;
  Addr4 :        ANY;
  WVar1 :        WORD;
  WVar2 :        WORD;
  WVar3 :        WORD;
  WVar4 :        WORD;
  WVar5 :        WORD;
  WVar6 :        WORD;
  WVar7 :        WORD;
  WVar8 :        WORD;
  WVar9 :        WORD;
  WVar10 :       WORD;
  FM-NCNo:       INT;          //On FM-NC only
END_VAR
VAR_OUTPUT
  Error :        BOOL;
  Done :         BOOL;
  State :        WORD;
END_VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the function PI_SERV.

Signal	Type	Type	Value range	Notes
Req	I	Bool		Job request
PIService	I	Any	[DBName].[VarName] default is:"PI"[VarName]	PI service description[1)]
Unit	I	Int	1 ...	Area number
Addr1 to Addr4	I	Any	[DBName].[VarName]	Reference to strings specification according to selected PI service
WVar1 to WVar10	I	Word	1 ...	Integers or word variables. Specification according to selected PI service (WVar10 SW 4 and higher)
FM-NCNo[2)]	I	Int	0, 1, 2	0, 1=1 NCU, 2=2 NCUs
Error	A	Bool		Negative acknowledgment of job or execution of job impossible
Done	A	Bool		Job successfully executed
State	A	Word		See error identifiers

1) See README file on basic program diskette supplied
2) On FM-NC only

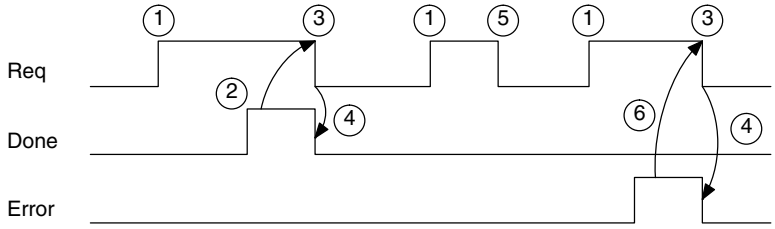
© Siemens AG, 2003. All rights reserved

Error identifiers

If it was not possible to execute a job, the failure is indicated by 'logic 1' on status parameter error. The error cause is coded at block output State. The error identifiers which may be encountered are as follows:

State	Meaning	Note
3	Negative acknowledgment, job not executable	Internal error, possible remedy: NC Reset
6	FIFO full	Job must be repeated since queue is full
7	Option not set	BP parameter "NCKomm" is not set
9	Transmission occupied	Job must be repeated

Pulse diagram



(1) Activation of function
(2) Positive acknowledgment: PI service has been executed
(3) Reset function activation after receipt of acknowledgment
(4) Signal change by means of FB
(5) If function activation is reset prior to receipt of acknowledgment, the output signals are not updated without the operational sequence of the activated function being affected
(6) Negative acknowledgment: Error has occurred. Error code in output parameter State

Overview PI services

The following Section shows an overview of the PI services that can be started from the PLC. The meaning and application of the general FB 4 input variables (Unit, Addr ..., WVar ...) depends on the individual PI service concerned.

PI service	Function	Available with FM-NC	Available with 840D/810D
SELECT	Select program for processing for one channel	x	x
ASUB	Assign interrupt	x	x
FINDBL	Activate block search	x	x
SETUFR	Activate user frames	x	x
CONFIG	Reconfiguration of marked machine data	x	x
CANCEL	Execute cancel	x	x
DELETO	Delete tool	x	x
CREATO	Generate tool	x	x

© Siemens AG, 2003. All rights reserved

PI service	Function	Available with FM-NC	Available with 840D/810D
CREACE	Create cutting edge	x	x
TMCRTO	Create tool		x
TMFDPL	Empty location search for loading		x
TMMVTL	Prepare magazine location for loading, unload tool		x
TMPOSM	Position magazine location or tool		x
LOGIN	Activate password	x	x
LOGOUT	Reset password	x	x
MMCSEM	Semaphores for various PI services		x (SW 4 and higher)
CRCEDN	Create new cutting edge		x (SW 4 and higher)
DELECE	Delete a cutting edge		x (SW 4 and higher)
TMFPBP	Empty location search		x (SW 4 and higher)
TSEARC	Complex search using search screen forms		x (SW 4 and higher)
TMPCIT	Set increment value for workpiece counter		x (SW 4 and higher)
DIGION	Digitizing on		x
DIGIOF	Digitizing off		x
NCRES	Initiate NC Reset		x (SW 5 and higher)
TMRASS	Reset active status		x (SW 5 and higher**)**
TRESMO	Reset monitoring values		x (SW 6.3 and higher**)**
ACTDEF	Activate a definition (GUD or macro)		x (SW 4 and higher)
x: PI service is available			

PI service: SELECT

Select a program for execution for a channel.

Function:

A program stored on the NCK is selected for processing for one channel. This is possible only if the file may be executed. The path name and program name must be entered as described in the Programming Guide (Chapter File and Program Management, Section Program memory). Please also refer to example of FB 4 for notation of path and program names.

Table 4-2 Possible block types

Block types	
Workpiece directory	WPD
Main program	MPF

© Siemens AG, 2003. All rights reserved

Table 4-2 Possible block types

Block types	
Subprogram	SPF
Cycles	CYC
Asynchronous subprograms	ASP
Binary files	BIN

Table 4-3 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.SELECT	Program selection
Unit	INT	1 to 10	Channel
Addr1	STRING		Path name
Addr2	STRING		Program name

PI service: ASUB Assign interrupt

Function: A program stored on the NCK is assigned an interrupt signal for a channel. This is possible only if the file may be executed. The path name and program name must be entered as described in the Programming Guide (Chapter File and Program Management, Section Program memory). Please also refer to example of FB 4 for notation of path and program names.

Table 4-4 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.ASUP	Assign interrupt
Unit	INT	1 to 10	Channel
WVar1	INT	1 to 8	Interrupt number
WVar2	INT	1 to 8	Priority
WVar3	INT	0/1	LIFTFAST
WVar4	INT	0/1	BLSYNC
Addr1	STRING		Path name
Addr2	STRING		Program name

Note

The SETINT instruction is also used to make the assignment.

PI service ASUB may only be executed when the channel to be activated is in the Reset state.
With an ASUB selection via FB 4 and subsequent ASUB start with FC 9, only PRIO 1 is possible.

References: /PA/, Programming Guide

© Siemens AG, 2003. All rights reserved

PI service: FINDBL Activate block search

Function: A channel is switched to block search mode and the appropriate acknowledgment then transmitted. The block search is then executed immediately by the NCK. The search pointer must already be in the NCK at this point in time. The search can be interrupted at any time by an NC RESET. Once the search is successfully completed, the normal processing mode is reactivated automatically. NC Start then takes effect from the located search target. The operator is responsible for providing a collisionfree approach path.

Table 4-5 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.FINDBL	Block search
Unit	INT	1 to 10	Channel
WVar1	WORD	x	Preprocessing mode
x: Defines preprocessing mode x = 1 Without calculation x = 2 With calculation x = 3 With consideration of main block			

PI service: SETUFR Activate user frames

Function: User frames are loaded to the NCK. All necessary frame values must be transferred to the NCK beforehand through writing variables with FB 3.

Table 4-6 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.SETUFR	User Frames spindle number converter
Unit	INT	1 to 10	Channel

PI service: CONFIG Reconfiguration

Function: The Reconfigure command permits the quasi-parallel activation of machine data that have been input sequentially by the user or the PLC.
The command can be activated only with the control in the RESET state or after a program interrupt (NC stop at block limit). An FB 4 error checkback message is output if these conditions are not fulfilled (state = 3).

Table 4-7 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.CONFIG	Reconfiguration
Unit	INT	1	
WVar1	INT	1	Classification

© Siemens AG, 2003. All rights reserved

PI service: CANCEL

Carry out Cancel

Function:

The CANCEL command activates the Cancel function (according to key on MMC).

Table 4-8 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.CANCEL	Cancel

PI service: DELETO

Delete tool

Function:

Deletes the tool assigned to the transferred T number with all cutting edges (in TO, in some cases TU, TUE and TG (type 400), TD and TS blocks).

Table 4-9 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.DELETO	Delete tool
Unit	INT	1, 2	TOA
WVar1	INT		T number

PI service: CREATO

Create tool

Function:

Generation of a tool with specification of a T number. The tool is entered as existing in the tool directory area (TV). The first “cutting edge” D1 (with zero contents) is created for tool offsets in the TO block. D1 (with zero contents) is also created for the OEM “cutting edge” data in the TUE block - if one is present. If a TU block exists, it contains the data block for the tool.

Table 4-10 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.CREATO	Create tool
Unit	INT	1, 2	TOA
WVar1	INT		T number

© Siemens AG, 2003. All rights reserved

PI service: CREACE

Create tool cutting edge

Function:

Creation of the cutting edge with the next higher/next unassigned D number for the tool with the transferred T number in TO, TS (if present). The cutting edge for the OEM cutting edge data is set up simultaneously in the TUE block - if one is present.

Table 4-11 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.CREACE	Create cutting edge
Unit	INT	1, 2	TOA
WVar1	INT		T number

PI service: TMCRTO

Create tool

Function:

Creation of a tool with specification of an identifier, a duplo number and a T number (optional). The tool is entered as existing in the tool directory area (TV). The first cutting edge “D1” (with zero contents) is created for tool offsets in the TO block. “D1” (with zero contents) is also set up for the monitoring data in the TS block, and simultaneously with zero contents for the OEM cutting edge data in the TUE block - if one is present. The TD block contains the identifier, duplo number and number of cutting edges (=1) for the T number that is entered optionally or allocated by the NCK. If a TU block exists, it contains the data block for the tool.
After execution of the PI, the T number for the created tool is stored in the TV block under TnumWZV.

Note

Before and after this PI service, the MMCSEM PI service must be called up with the associated parameter WVar1 for this PI service. See PI service MMCSEM for more information.

Table 4-12 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.TMCRTO	Create tool
Unit	INT	1, 2	TOA
WVar1	INT		T number
WVar2	INT		Duplo number
Addr1	STRING	max. 32 characters	Tool identifier
T number > 0 means that a T number must be input T number = –1 means that NCK must allocate a T number Example shows T number = –1 ==> T number allocated by NCK			

© Siemens AG, 2003. All rights reserved

PI service: TMFDPL

Searching for empty location for loading

Function (dependent on parameter assignment)

Location_number_where = -1, magazine_number_where = -1:
Searches through all the magazines in the specified area (= channel) for an empty location for the tool specified by a T number. After execution of the PI, the magazine and locations numbers found during the search are listed in the configuration block of the channel (component magCMCmdPar1 (magazine number) and magCMCmdPar2 (location number)). Location_number_ID and magazine_number_ID can be set as search criteria or not (= -1). The PI is acknowledged positively or negatively depending on the search result.

Location_number_where = -1, magazine_number_where = magazine number:
A search is executed in the specified magazine for an empty location to load the tool specified by the T number. Location_number_ID and magazine_number_ID can be set as search criteria or not (= -1). The PI is acknowledged positively or negatively depending on the search result.

Location_number_where = location number, magazine_number_where = magazine number:
The specified location is checked to establish whether it is free for loading with the specified tool. Location_number_ID and magazine_number_ID can be set as search criteria or not (= -1). The PI is acknowledged positively or negatively depending on the search result.

Command parameters 1 and 2 are located at source.
Loading: If source is an internal loading magazine, then the command parameters are located at the destination (a real magazine).
Unloading: Source is always a real magazine.

Note

Before and after this PI service, the MMCSEM PI service must be called up with the associated parameter WVar1 for this PI service. See PI service MMCSEM for more information.

Table 4-13 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.TMFDPL	Empty location for loading
Unit	INT	1, 2	TOA
WVar1	INT		T number
WVar2	INT		Location number _Where
WVar3	INT		Magazine_number _Where
WVar4	INT		Location_number _ID
WVar5	INT		Magazine number _ID

© Siemens AG, 2003. All rights reserved

PI service: TMMVTL

Provide magazine location for loading, unload tool

Function:

This PI service is used both to load and unload tools. Whether the PI initiates a loading or unloading operation depends on the assignment between the real locations and the from parameters and to parameters: Loading => from =loading point/station, unloading => to = loading point/station.

PI service TMMVTL is used for all movements:
1) Loading and unloading (loading point <-> magazine)
2) Loading and unloading (loading point <-> buffer, e.g. spindle)
3) Relocation within a magazine
4) Relocation between different magazines
5) Relocation between a magazine and a buffer
6) Relocation within the buffer

The following variables from block TM are used to monitor cases 1), 3), 4), 5):

magCmd (area no. = TO unit, line = magazine number)
magCmdState <- “acknowledgment”

The following variables from block TMC are used to monitor cases 2), 6):

magCBCmd (area no. = TO unit)
magCBCmdState <- “acknowledgment”

Loading function

Prepares the specified real magazine for the specified channel for loading, i.e. traverses the magazine to the selected location for loading at the specified loading point/station (location_number_from, magazine_number_from) and inserts the tool.
When location_number_to = -1, an empty location for the tool specified by a T number is first sought in the specified magazine and the magazine then traversed. After execution of the PI, the location number found during the search is listed in the TM area in component **magCMCmdPar2** for the **real** magazine of the channel.
With a location_number_to = -2 and a valid magazine number, the tool is loaded to the nearest empty position of the specified magazine. After execution of the PI, the number of the location for tool loading is listed in the TM area in component **magCMCmdPar2** for the real magazine of the channel.

Unloading function

The tool specified with a tool number is unloaded at the specified loading point/station (location_number_to, magazine_number_to), i.e. the magazine is traversed to the position for unloading and the tool then removed. The magazine location for the tool is marked as being free in the TP block. The tool can be specified either via a T number or by means of the location and magazine numbers. The value -1 is entered at unused specification points.

© Siemens AG, 2003. All rights reserved

Note

Before and after this PI service, the MMCSEM PI service must be called up with the associated parameter WVar1 for this PI service. See PI service MMCSEM for more information.

Table 4-14 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.TMMVTL	Make magazine location ready for loading, unload tool
Unit	INT	1, 2	TOA
WVar1	INT		T number
WVar2	INT		Location_number_from
WVar3	INT		Magazine_number_from
WVar4	INT		Location_number_to
WVar5	INT		Magazine_number_to

PI service: TMPOSM

Position magazine location or tool

Function: (dependent on parameter assignment)

A magazine location, which has either been specified directly or which has been qualified via a tool located on it, is traversed to a specified position (e.g. in front of a load location) via the PI service.

The PI service makes a magazine location, which can be qualified in various ways, traverse in front of a specified load location.

The load location must be specified in the PI parameters location number_from and magazine number_from.

The magazine location to be traversed can be qualified by the following:

either

- the T number of the tool: The location where the tool is positioned traverses; the parameters tool identifier, duplo number, location number_from and magazine number_from are irrelevant (i.e. values “”, “-0001”, “-0001”, “-0001”).

or

- tool identifier and duplo number: The location where the tool is positioned traverses; the parameters T number, location number_from and magazine number_from are irrelevant (i.e. value “-0001”each).

© Siemens AG, 2003. All rights reserved

or

- direct specification of the location in the parameters location number_from and magazine number_from; the tool-qualifying parameters T number, tool identifier and duplo number are irrelevant (i.e. values “-0001”, “” ,“-0001”).

Table 4-15 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.TMPOSM	Position magazine location or tool
Unit	INT	1, 2	TOA
Addr1	STRING	max. 32 characters	Tool identifier
WVar1	INT		T number
WVar2	INT		Duplo number
WVar3	INT		Location_number_from
WVar4	INT		Magazine_number_from
WVar5	INT		Location number_ref
WVar6	INT		Magazine number_ref

PI service: LOGIN

Create password

Function:

Transfers the parameterized password to the NCK. The passwords generally consists of 8 characters. If required, blanks must be added to the string of the password.

Example

Password STRING[8] := ‘SUNRISE␣‘;

Table 4-16 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.LOGIN	Create password
Unit	INT	1	NCK
Addr1	STRING	8 characters	Password

PI service: LOGOUT

Reset password

Function:

The password last transferred to the NCK is reset.

© Siemens AG, 2003. All rights reserved

Table 4-17 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.LOGOUT	Reset password
Unit	INT	1	NCK

Semaphores for PI services

MMCSEM

For use by MMC and PLC

10 semaphores are provided for each channel. These protect critical functions for the MMC/PLC. By setting the semaphore for the corresponding function number, several MMC/PLC units can be synchronized with it in cases where a function contains a critical section with respect to data to be fetched by the NCK. Semaphores are managed by the MMC/PLC. A semaphore value of 1stipulates a Test & Set operation for the semaphore of the specified function number. The return value of the PI service represents the result of this operation:
Return value OK:
Semaphore has been set, critical function can be called
Return value REJECTED:
Semaphore was already set, critical function cannot be called at the present time. The operation must be repeated later.

Note

On completion of the operation (reading data of this PI service) it is **essential** that the **semaphore** is **enabled again**.

Parameter:

WVar1=FunctionNumber
This function number represents a PI service:
1: TMCRTO (create tool)
2: TMFDPL (search for empty location for loading)
3: TMMVTL (prepare magazine location for loading, unload tool)
4: TMFPBP (search for location)
5: TMGETT (search for tool number)
6: TSEARC (search for tool)
7 ... 10: Reserved

WVar2=SemaphorValue

0: Reset semaphore
1: Test and set semaphore

© Siemens AG, 2003. All rights reserved

Parameterization:

Table 4-18 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.MMCSEM	Create new cutting edge
Unit	INT	1, 2 to 10	Channel
WVar1	INT	1 to 10	FunctionNumber
WVar2	Word	0, 1	SemaphoreValue

Create new cutting edge

CRCEDN

Function

Create tool edge by specifying the cutting edge number.

If the T number of an existing tool is specified in parameter “T number” in the PI service, then a cutting edge is set up for this particular tool (in this case, parameter “D number” (number of cutting edge to be created) has a value range of 00001–00009). If a positive T number is specified as a parameter and the tool for the T number entered does not exist, then the PI service is aborted. If a value of 00000 is entered for the T number (model of absolute D numbers), then the D number value range might extend from 00001 to 31999. The new cutting edge is set up with the specified D number. If the specified cutting edge already exists, then the PI service is aborted in both cases.

Parameterization:

Table 4-19 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.CRCEDN	Create new cutting edge
Unit	INT		TOA
WVar1	INT		T number of tool for which cutting edge must be created. A setting of 00000 states that the cutting edge should not refer to any particular tool (absolute D number).
WVar2	INT	1-9 or 01 - 31999	Edge number of tool cutting edge

© Siemens AG, 2003. All rights reserved

Delete a cutting edge

DELETE

Function:

If the T number of an existing tool is specified in parameter “T number” in the PI service, then a cutting edge is deleted for this particular tool (in this case, parameter “D number” (number of cutting edge to be created) has a value range of 00001–00009). If a positive T number is specified as a parameter and the tool for the T number entered does not exist, then the PI service is aborted. If a value of 00000 is entered for the T number (model of absolute D numbers), then the D number value range might extend from 00001 to 31999. If the specified cutting edge does not exist, then the PI service is aborted in both cases.

Parameterization:

Table 4-20 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.DELETE	Delete cutting edge
Unit	INT		TOA
WVar1	INT		T number of tool for which the cutting edge must be deleted. A setting of 00000 states that the cutting edge should not refer to any particular tool (absolute D number)
WVar2	INT	1-9 or 01 - 31999	Edge number of cutting edge that must be deleted

Empty location search

TMFPBP

Function: (dependent on parameter assignment)

See description in FB7

© Siemens AG, 2003. All rights reserved

Complex search using search screen forms

TSEARC

Function: (dependent on parameter assignment)

The PI service allows you to search for tools with specified properties within a search domain (in one or more magazines starting and ending at a specific location). The specified properties refer only to data of the tools and their cutting edges. The PI service is only available if tool management is activated. You can define a search direction and the number of hits for the PI service (e.g. one tool for the next tool with matching properties or all tools with the specified properties). The service returns a list of the internal T numbers for the tools found in the search.
Search criteria can only be ANDed. If an application needs to define an OR operation for the search criteria, it must first execute a series of queries with ANDed criteria and then combine/evaluate the results of the individual queries.

To assign the parameters of the PI service, the properties of the required tools are first defined via variable service in the TF block. This is achieved by selecting the relevant comparison criteria in the operand masks (parMaskT..) in the TF block (i.e. which tool data are to be compared?), entering the types of comparison logic (==, <, >, <=, >=, &&) in the comparison operator data (parDataT..), and entering the comparison values in the operand data. The PI service is then initiated and, after its successful return, the variable service from the TF block is used to read out the number of hits in the variable resultNrOfTools and the result list in the variable resultToolNr (i.e. the list of internal T numbers of the tools found in the search - resultNrOfTools quantity). The PI service must be encapsulated with a semaphore from its preparation until the successful return of the result. This is the only way to ensure exclusive access and the exclusive use of the TF block in conjunction with the TSEARC PI service. The function number provided for the semaphore feature (PI service MMCSEM) is the function number for TSEARC.

If the service is configured incorrectly, a malfunction occurs. In all other cases, it will return a result, even if no tools are found (resultNrOfTools = 0).

The search domain can be defined as follows in the parameters MagNrFrom, PlaceNrFrom, MagNrTo, PlaceNrTo:

MagNrFrom	PlaceNrFrom	MagNrTo	PlaceNrTo	Search area
WVar1	**WVar2**	**WVar3**	**WVar4**	
#M1	#P1	#M2	#P2	Locations starting at magazine #M1, location #P1 up to magazine #M2, location #P2 are searched
#M1	-1	#M1	-1	All locations in magazine #M1 - and no others - are searched
#M1	-1	-1	-1	All locations starting at magazine #M1 are searched
#M1	#P1	-1	-1	All locations starting at magazine #M1 and location #P1 are searched
#M1	#P1	#M1	-1	Locations in magazine #M1 starting at magazine #M1 and location #P1 in this magazine are searched

© Siemens AG, 2003. All rights reserved

MagNrFrom	PlaceNrFrom	MagNrTo	PlaceNrTo	Search area
#M1	#P1	#M2	-1	Locations starting at magazine #M1, location #P1 up to magazine #M2 are searched
#M1	-1	#M2	#P2	Locations starting at magazine #M1 up to magazine #M2, location #P2 are searched
#M1	-1	#M2	-1	Locations starting at magazine #M1 up to and including magazine #M2 are searched
-1	-1	-1	-1	All magazine locations are searched

For a symmetrical search (see parameter “SearchDirection”), the search domain must only include one magazine (cases 2 and 5 in the table above). If another search domain is specified, the service will malfunction. A reference location must be specified in the parameters MagNrRef and PlaceNrRef for a symmetrical search (see parameter “SearchDirection”). The reference location is specified in the parameters MagNrRef and PlaceNrRef. The reference location is a buffer location (a location from the magazine buffer, i.e. change position, gripper ...) or a load point (a location from the internal loading magazine). The search is executed symmetrically with reference to the magazine location in front of the specified reference location. A multiple assignment to the magazine being searched must be configured in the TPM block for the reference location. If this is not the case, a malfunction occurs. If the magazine location in front of the reference location is outside the search domain, the service responds as if it has not found a matching location.

Note

Before and after this PI service, the MMCSEM PI service must be called up with the associated parameter WVar1 for this PI service. See PI service MMCSEM for more information.

The PI service is valid from NCK SW 4.2 and higher.

Parameterization:

Table 4-21 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.TSEARC	Complex search using search screen forms
Unit	INT	1, 2	TOA
WVar1	INT		MagNrFrom Magazine number of magazine from which search must begin.
WVar2	INT		PlaceNrFrom Location number of location in magazine MagNrFrom, at which search must begin.

© Siemens AG, 2003. All rights reserved

Table 4-21 Parameterization

Signal	Type	Value range	Meaning
WVar3	INT		MagNrTo Magazine number of magazine at which search must end.
WVar4	INT		PlaceNrTo Location number of location in magazine MagNrTo, at which search must end.
WVar5	INT		MagNrRef Magazine number of (internal) magazine, with reference to which the symmetrical search is to be performed. (this parameter is only relevant with a “symmetrical” search direction)
WVar6	INT		PlaceNrRef Location number of location in magazine MagNrRef, with reference to which the symmetrical search is to be performed. This parameter is only relevant with a “symmetrical” search direction.

© Siemens AG, 2003. All rights reserved

Table 4-21 Parameterization

Signal	Type	Value range	Meaning
WVar7	INT	1, 2, 3	SearchDirection specifies the desired search direction. 1. Forwards from the first location of the search domain 2. Backwards from the last location of the search domain 3. Symmetrical with real magazine location positioned in front of the location specified by MagNrRef and PlaceNrRef.
WVar8	INT	0, 1, 2, 3	KindofSearch 0: Find all tool with this property cutting edge specifically 1: Search for the first tool found with this property (cutting edge specifically) 2: Browse all cutting edges to find all tool with this property 3: Browse all tools to search for the first tool found with this property

© Siemens AG, 2003. All rights reserved

Set increment value for workpiece counter TMPCIT

Function: Increment the workpiece counter of the spindle tool.

Parameterization:

Signal	Type	Value range	Meaning
PIService	ANY	PI.TMPCIT	Set increment value for workpiece counter
Unit	INT	1 to 10	TOA
WVar1	WORD	0 ... max.	Spindle number; corresponds to the type index in the location data with spindle location type of the buffer magazine in channel.000 = main spindle
WVar2	WORD	0 ... max.	Increment value; indicates the number of spindle revolutions after which the workpiece counter is incremented.

Reset active status TMRASS

Function: Reset active status for worn tools.
This PI service is used to search for all tools with the tool status active and disabled. The active status is then canceled for these tools. Potentially appropriate times for this PI service are the negative edge of VDI signal “tool disable ineffective” an end of program, or a channel reset. This PI service is intended mainly for the PLC, since it knows when the disabled tool is finally no longer to be used.

Parameterization:

Signal	Type	Value range	Meaning
PIService	ANY	PI. TMRASS	Reset active status
Unit	INT	1 to 10	TO area

© Siemens AG, 2003. All rights reserved

Reset monitoring values (TRESMO)

This PI service resets the monitoring values of the designated edges of the designated tools back to their setpoint (initial) values. The service applies only to tools for which monitoring is already active.
See NC command RESETMON.

Parameterization:

Signal	Type	Value range	Meaning
PIService	ANY	PI. TRESMO	Reset monitoring values
Unit	INT	1 to 10	TO area
WVar1	WORD	-max ..max	ToolNumber 0: Applies to all tools >0: Applies only to this tool <0: Applies to all sister tools of the specified T No
WVar2	WORD	0 .. max.	D number >0: Monitoring of specified edge of specified tools is reset. 0: Monitoring of all edges of specified tools is reset.
WVar3	WORD	0 ..15	Monitoring types Type of monitoring to be reset. This parameter is binary coded. 1: Tool life monitoring is reset 2: Workpiece count monitoring is reset 4: Wear monitoring is reset 8: Sum offset monitoring is reset The values above can be added to reset combinations of monitoring functions 0: All active monitoring functions of the tool ($TC_TP9) are reset

Digitizing on (DIGION)

Function: Select digitizing in the specified channel

Parameterization:

Signal	Type	Value range	Meaning
PIService	ANY	PI.DIGION	Digitizing on
Unit	INT	1 to 10	Channel

Digitizing OFF (DIGIOF)

© Siemens AG, 2003. All rights reserved

Function: Deactivate digitizing in the specified channel

Parameterization:

Signal	Type	Value range	Meaning
PIService	ANY	PI.DIGIOF	Digitizing off
Unit	INT	1 to 10	Channel

Initiate NC Reset NCRES

Function: Initiates an NCK Reset. The Unit and WVar1 parameters must be assigned 0.

Parameterization:

Signal	Type	Value range	Meaning
PIService	ANY	PI.NCRES	Initiate NC Reset
Unit	INT	0	0
WVar1	WORD	0	0

Defining GUD or macro activation ACTDEF

Function: A program stored on the NCK is selected for processing for one channel. This is possible only if the file may be executed. The path name and program name must be entered as described in the Programming Guide (Chapter File and Program Management, Section Program memory). Please also refer to example of FB 4 for notation of path and program names.

Parameterization:

Signal	Type	Value range	Meaning
PIService	ANY	PI.ACTDEF	Activate GUDs or macros
Unit	INT	1	
Addr1	STRING		Path name
Addr2	STRING		GUD or macro

© Siemens AG, 2003. All rights reserved

Call example

Program selection in channel 1 (main program and workpiece program)

Entry PI for DB 16 and STR for DB 124 with the S7 SYMBOL editor:

Parameterization:

Symbol	Operand	Data type
PI	DB16	DB16
STR	DB124	DB124

```
DATA_BLOCK DB 126                    //Unassigned user DB, as instance for FB 4
FB 4
BEGIN
END_DATA_BLOCK
DATA_BLOCK db 124
  struct
    PName:          string[32]:= '_N_TEST_MPF ';
    Path:           string[32]:= '/_N_MPF_DIR/';          //Main program

    PName_WST:      string[32]:= '_N_ABC_MPF';
    Path_WST:       string[32]:= '/_N_WKS_DIR/_N_ZYL_WPD';
                                                          //Workpiece program
  end_struct
BEGIN
END_DATA_BLOCK
Function FC "PICall" : VOID
  call fb4,db126(
  I 7.7;                        //Unassigned key on MCP
  S     M 0.0:                  //Activate Req
  A     M 1.1;                  //Done message
  R     M 0.0:                  //End job
  A     I 7.6;                  //Acknowledge error manually
  A     M 1.0;                  //Error is present
  R     M 0.0:                  //End job
        Req :=          F0.0,
        PIService :=    PI.SELECT,
        Unit :=         1,                                // CHAN 1
        Addr1 :=        STR.Path,
        Addr2 :=        STR.PName,      //Main program selection
                                        //Addr1:=STR.Path_WST,
                                        //Addr2:=STR.PName_WST,
                                        //Workpiece program selection
        FM-NCNo :=      1,              //On FM-NC only
        Error :=        F1.0,
        Done :=         F 1.1,
        State :=        MW2);
```

© Siemens AG, 2003. All rights reserved

4.5 FB 5: Read GUD variable GETGUD

Description of functions

The PLC user program can read a GUD variable (GUD = Global User Data) from the NCK or channel area using the FB GETGUD. Capital letters must be used for the names of GUD variables. Every FB 5 call must be assigned a separate instance DB from the user area. (multi-instance capability in SW 3.7 and higher). A job is started when FB 5 is called by means of a positive edge change at control input “Req”. This job includes the name of the GUD variable to be read in parameter “Addr” with data type “STRING”. The pointer to the name of the GUD variable is assigned symbolically with < DataBlockName >.<VariableName> Additional information about this variable is specified in parameters “Area” “Unit” “Index1” and “Index2” (see table of block parameters).

When parameter “CnvtToken” is activated, a variable pointer (token) can be generated for this GUD variable as an option. This pointer is generated via the VAR selector (see Section 3.2) for system variables of the NC. Only this method of generating pointers is available for GUD variables. Once a pointer has been generated for the GUD variable, then it is possible to read and write via FB 2 and FB 3 (GET, PUT) with reference to the pointer. This is the only method by which GUD variables can be read. When FB 2 or FB 3 is parameterized, only parameter Addr1 ... Addr8 need to be parameterized for the variable pointer. GUD variable fields are an exception. In these, Line1 .. Line8 must also be parameterized with the field index of this variable.

Successful completion of the read process is indicated by a logic “1” in status parameter Done.

The read process extends over several PLC cycles (generally 1 to 2). The block can be called up in cyclic mode only.

Any errors are indicated by Error and State.

Important

After communication between the PLC and NC (read/write NC variables, FB2, 3, 5, or PI general services, FB4) has been aborted by POWER OFF, the start jobs must be deleted in the first OB1 run after cold restart or reset (signal: Req = 0).

FB 5 can read GUD variables only if basic program parameter NCKomm has been set to “1” (FB 1, DB7 in OB100, see Section 4.1).

© Siemens AG, 2003. All rights reserved

Declaration

```
FUNCTION_BLOCK FB 5                 //Name server
  KNOW_HOW_PROTECT
  VERSION: 3.0
VAR_INPUT
  Req:          bool;
  Addr:         any;                //Name string of variable
  Area:         byte;               //Area: NCK = 0, channel = 1
  Unit:         byte;
  Index1:       INT;                //Field index 1
  Index2:       INT;                //Field index 2
  CnvtToken:    BOOL ;              //Conversion to 10 byte token
  VarToken:     ANY;                //Struct with 10 bytes for the variable token
  FM-NCNo:      INT;                / On FM-NC only
END_VAR

VAR_OUTPUT
  Error: bool;
  Done: bool;
  State: word;
END_VAR

VAR_IN_OUT
  RD:           any;
END_VAR

BEGIN
END_FUNCTION_BLOCK
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the function GETGUD.

Signal	Type	Type	Value range	Notes
Req	I	Bool		Job start with positive signal edge
Addr	I	Any	[DBName].[VarName]	GUD variable name in a variable of data type STRING
Area	I	Byte		Area address: 0: NCK variables 2: Channel variables
Unit	I	Byte		NCK area: Unit:=1 Channel area: Channel no.
Index1	I	Int		Field index 1 of variable Variable has the value 0 if no field index is used.
Index2	I	Int		Field index 2 of variable Variable has the value 0 if no field index is used.
CnvtToken	I	Bool		Activate generation of a variable token
VarToken	I	Any		Address to a 10 byte token (see example)
FM-NCNo[1)]	I	Int	0, 1, 2	0, 1=1 NCU, 2=2 NCUs
Error	O	Bool		Job negatively acknowledged or not executable
Done	O	Bool		Job successfully executed
State	O	Word		See error identifiers
RD	I/O	Any	P#Mm.n BYTE x... P#DBnr.dbxm.n BYTE x	Data to be written

1) On FM-NC only

Error identifiers

If it was not possible to execute a job, the failure is indicated by 'logic 1' on status parameter error. The error cause is coded at the block output State:

State		Meaning	Note
WORT H	WORT L		
0	1	Access error	
0	2	Error in job	Incorrect compilation of Var. in a job
0	3	Negative acknowledgment, job not executable	Internal error, possible remedy: NC Reset
0	4	Data areas or data types do not tally	Check data to be read in RD;
0	6	FIFO full	Job must be repeated since queue is full

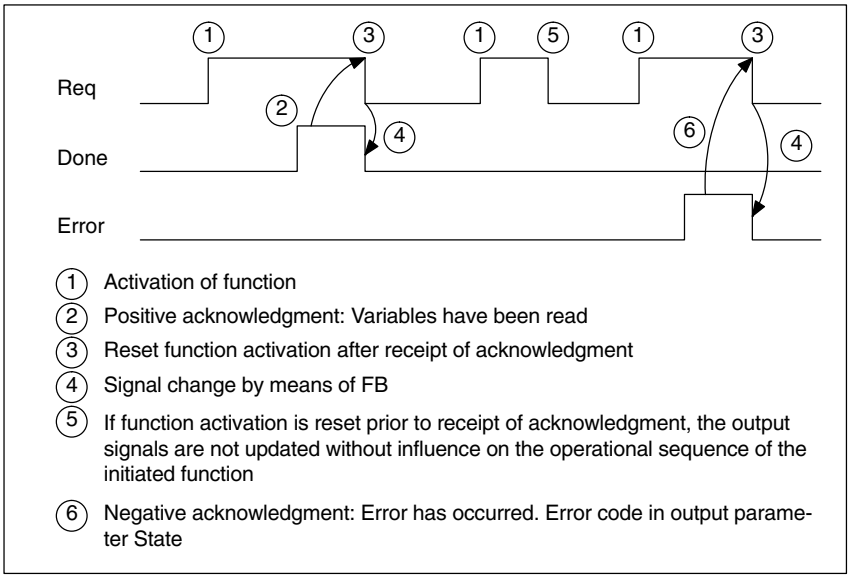
© Siemens AG, 2003. All rights reserved

State		Meaning	Note
WORT H	WORT L		
0	7	Option not set	BP parameter “NCKomm” is not set
0	8	Incorrect target area (RD)	RD may not be local data
0	9	Transmission occupied	Job must be repeated
0	10	Error in addressing	Unit contains value 0
0	11	Address of variable invalid	Check Addr (or variable name), area, unit

Configuration steps

To be able to read a GUD variable, its name must be stored in a string variable. The data block with this string variable must be defined in the symbol list so that parameter “Addr” can be assigned symbolically for FB GETGUD. A structure variable can be defined optionally in any data area of the PLC to receive the variable pointer (see specification in following example).

Pulse diagram



© Siemens AG, 2003. All rights reserved

Call example

Reading of a GUD variable with the name “GUDVAR1” as an integer variable. (See also table in FB 2: Assignment of NC data type in SIMATIC data type)

Call and parameterization of FB 5 with instance DB 111:

```
DATA_BLOCK DB GUDVAR                    //Make assignment in symbol list
STRUCT
  GUDVar1 : STRING[32] := 'GUDVAR1';  //Name is defined by user
  GUDVar1T :
  STRUCT
    SYNTAX_ID :      BYTE;
    area_&_unit :    byte;
    column :         word;
    line :           word;
    block type :     byte;
    NO. OF LINES :   BYTE;
    type :           byte;
    length :         byte;
  END_STRUCT ;
END_STRUCT ;
BEGIN
END_DATA_BLOCK
DATA_BLOCK DB 111                       //Unused user DB as instance for FB 5
FB 5
BEGIN
END_DATA_BLOCK
//A user-defined channel variable from channel 1 must be read
//with conversion into a variable pointer to allow subsequent
//writing of a variable.
Function FC “Variable Call” :           VOID
  I 7.7;                                //Unassigned key on MCP
  S      M 100.0:                       //Activate Req
  A      M 100.1;                       //Done message
  R      M 100.0:                       //End job
  A      I 7.6;                         //Acknowledge error manually
  A      M 102.0;                       //Error is present
  R      M 100.0:                       //End job
  Call fb 5, db 111(
       Req :=          F 100.0,        //Start edge for reading
       Addr :=         GUDVAR.GUDVar1,
       Area :=         B#16#2,         //Channel variable
       Unit :=         B#16#1,         //Channel 1
       Index1 :=       0,              //No field index
       Index2 :=       0,              //No field index
       CnvtToken :=    TRUE,           //Conversion to 10 byte token
       VarToken :=     GUDVAR.GUDVar1T,
       FM-NCNo :=      1,              //On FM-NC only
       Error :=        F102.0,
       Done :=         F100.1,
       State :=        FW104,
       RD :=           P#DB99.DBX0.0 DINT 1
       );
```

© Siemens AG, 2003. All rights reserved

4.6 FB 7: PI_SERV2 General PI services

Description of functions

A detailed description of the FB 7 is contained in the description of FB 4. The only difference to FB 4 is the number of WVar1 and subsequent parameters. In FB 7, WVar1 to WVar16 are defined in VAR_INPUT (FB4 has WVar1 to WVar10). All other parameters are identical to FB 4. This PI server can be used for all PI services previously implemented with FB 4. In addition, the PI services listed below can only be handled with FB 7.

Declaration

```
FUNCTION_BLOCK FB 7
Var_INPUT
   Req :          BOOL;
   PIService :    ANY;
   Unit :         INT;
   Addr1 :        ANY;
   Addr2 :        ANY;
   Addr3 :        ANY;
   Addr4 :        ANY;
   WVar1 :        WORD;
   WVar2 :        WORD;
   WVar3 :        WORD;
   WVar4 :        WORD;
   WVar5 :        WORD;
   WVar6 :        WORD;
   WVar7 :        WORD;
   WVar8 :        WORD;
   WVar9 :        WORD;
   WVar10 :       WORD ;
   WVar11 :       WORD;
   WVar12 :       WORD;
   WVar13 :       WORD;
   WVar14 :       WORD;
   WVar15 :       WORD;
   WVar16 :       WORD ;
   FM-NCNo:       INT;          //(On FM-NC only)
END_VAR

VAR_OUTPUT
   Error :        BOOL;
   Done :         BOOL;
   State :        WORD;
END_VAR
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the function PI_SERV.

Signal	Type	Type	Value range	Notes
Req	I	Bool		Job request
PIService	I	Any	[DBName].[VarName] default is:“PI”[VarName]	PI service description
Unit	I	Int	1 ...	Area number
Addr1 to Addr4	I	Any	[DBName].[VarName]	Reference to strings specification according to selected PI service
WVar1 to WVar16	I	Word	1 ...	Integers or word variables. Specification according to selected PI service,
FM-NCNo (on FM-NC only)	I	Int	0, 1, 2	0,1=1 NCU, 2=2 NCUs
Error	O	Bool		Negative acknowledgment of job or execution of job impossible
Done	O	Bool		Job has been executed successfully.
State	O	Word		See error identifiers

Overview of additional PI services

The following Section shows an overview of the PI services that can be started from the PLC. The meaning and application of the general FB 7 input variables (Unit, Addr..., WVar...) depends on the individual PI service concerned.

PI service	Function	Available in	
		FM-NC	**840D/810D**
TMFPBP	Empty location search		*(SW 4 and higher)

© Siemens AG, 2003. All rights reserved

Empty location search

TMFPBP

Function: (dependent on parameter assignment)

This service searches the magazine(s) named in the relevant parameters for an empty location which satisfies specified criteria (tool size and location type). The result of the empty location search can be fetched from variables magCMCmdPar1 (magazine number) and magCMCmdPar2 (location number) in block TMC when the service has functioned correctly. Since this PI services stores a result in variables magCMCmdPar1 and magCMCmdPar2, the service must be protected by the semaphore mechanism (PI service MMCSEM) with the function number for _N_TMFDPL in cases where several control units or PLCs are operating on one NC. The search area can be predefined in the following way by setting parameters MagazinNumber_From, LocNumber_From, MagazinNumber_To, LocNumber_To:

Magazin Number _From	Loc_Number_From	Magazine Number_To	Loc_Number_To	Search area
WVar1	**WVar2**	**WVar3**	**WVar4**	
#M1	#P1	#M1	#P1	Only location #P1 in magazine #M1 is checked
#M1	#P1	#M2	#P2	Locations starting at magazine #M1, location #P1 up to magazine #M2, location #P2 are searched
#M1	-1	#M1	-1	All locations in magazine #M1 - and no others - are searched
#M1	-1	-1	-1	All locations starting at magazine #M1 are searched
#M1	#P1	-1	-1	All locations starting at magazine #M1 and location #P1 are searched
#M1	#P1	#M1	-1	Locations in magazine #M1 starting at magazine #M1 and location #P1 in this magazine are searched
#M1	#P1	#M2	-1	Locations starting at magazine #M1, location #P1 up to magazine #M2, location #P2 are searched
#M1	-1	#M2	#P2	Locations starting at magazine #M1 up to magazine #M2, location #P2 are searched
#M1	-1	#M2	-1	Locations starting at magazine #M1 up to and including magazine #M2 are searched
-1	-1	-1	-1	All magazine locations are searched

Note

Before and after this PI service, the MMCSEM PI service must be called up with the associated parameter WVar1 for this PI service. See PI service MMCSEM for more information.

© Siemens AG, 2003. All rights reserved

Parameterization:

Table 4-22 Parameterization

Signal	Type	Value range	Meaning
PIService	ANY	PI.TMFPBP	Empty location search
Unit	INT	1 ... max. TOA	TOA
WVar1	INT		MagazinNumber _From: Magazine number of magazine from which search must begin (start of search area)
WVar2	INT		Loc_Number_From: Location number of location in magazine MagazinNumber_From at which search must begin
WVar3	INT		MagazinNumber_To: Magazine number of magazine at which search must end
WVar4	INT		Loc_Number_To: Location number of location in magazine MagazinNumber_To at which search must end
WVar5	INT		MagazineNumber_ref:
WVar6	INT		LocationNumber_ref:
WVar7	INT	0, 1 .. 7	Number of required half locations to left
WVar8	INT	0, 1 .. 7	Number of required half locations to right
WVar9	INT	0, 1 .. 7	Number of required half locations in upward direction
WVar10	INT	0, 1 .. 7	Number of required half locations in downward direction
WVar11	INT	0, 1 .. 7	Number of desired location type
WVar12	INT	0: Default 1: Forwards 2: Backwards 3: Symmetrical	Specifies the desired search direction 0: Empty location search strategy is set in $TC_MAMP2

© Siemens AG, 2003. All rights reserved

4.7 FB 9: M : N operating unit changeover

Description of functions

This block enables you to switch between several **control units** (MMC operator panels and/or machine control panels (MCP)) that are linked via a bus system to one or several NCU control modules.

For further details, please refer to publication:
SINUMERIK 840D/840Di/810D/FM-NC Description of Functions Extended Functions (FB2), Several Operator Panels on Several NCUs, Distributed Systems (B3).
The **interface** between the individual contorl units and the NCU (PLC) is the M : N interface in data block DB19 (see FB2 above, Chapter 5 Signal Description). FB 9 works with the signals from these interfaces.

Apart from initialization, signoflife monitoring and error routines, the following **basic functions** are also performed by the block for control unit switchover:

Table 4-23 Tabulated overview of functions:

Basic function	Meaning
MMC call waiting	MMC wants to go online to an NCU
MMC coming	MMC is connecting to an NCU
MMC going	MMC is disconnecting from an NCU
Forced break	MMC must break connection to an NCU
Operating focus changeover to server mode	Change operating focus from one NCU to the other
Active/passive operating mode	Operator control and monitoring/monitoring only
MCP switchover	As an option, MCP can be switched over with the MMC

Brief description of a few important functions

Active/passive operating mode:

An online MMC can operate in two different modes:

Active mode: Operator can control and monitor

Passive mode: Operator can monitor (MMC header only)

After switchover to an NCU, this initially requests active operating mode in the PLC of the online NCU. If two MMCs are simultaneously connected online to one NCU, one of the two is always in active and the other in passive operating mode. The operator can request active mode on the passive MMC at the press of a button.

© Siemens AG, 2003. All rights reserved

MCP switchover

As an option, an MCP assigned to the MMC can be switched over at the same time. This can be done on condition that the MCP address is entered in parameter **mstt_adress** of MMC configuration file NETNAMES.INI and **MCPEnable** is set to true. The MCP of the passive MMC is deactivated so that there is only ever one active MCP on an NCU at one time.

Boot condition

To prevent the previously selected MCP from being activated again when the NCU is restarted, input parameters **MCP1BusAdr = 255** (address of 1st MCP) and **MCP1STOP =TRUE** (deactivate 1st MCP) must be set when FB1 is called in OB100.

Enabling signals

When one MCP is switched over to another, any active feedrate or axis enabling signals may be transferred at the same time.

!

Important

Keys actuated at the moment of switchover remain operative until the new MCP is activated (by the MMC which is subsequently activated). The override settings for feedrate and spindle also remain valid. To deactivate actuated keys, the input image of the machine control signals must be switched to non-actuated signal level on a falling edge of DB10.DBX104.0. The override settings should remain unchanged.
Measures for deactivating keys must be implemented in the PLC user program. (See below: Example of override switchover)

Declaration of function

```
FUNKTION_BLOCK FB9

VAR_INPUT
  Quit          :BOOL;             // Acknowledge alarms
 OPMixedMode:BOOL:= FALSE; // Mixed mode deactivated for OP without
                                    M to N capability!
 ActivEnable   :BOOL:= TRUE; // Is not supported
  MCPEnable  :BOOL:= TRUE; // Activate MCP switchover
END_VAR
VAR_OUTPUT
  Alarm1        :BOOL ;            // Alarm: Error in MMC bus address, bus type!
  Alarm2        :BOOL ;            // Alarm: No confirmation MMC1 offline!
  Alarm2        :BOOL ;            // Alarm: MMC1 is not going offline!
  Alarm4        :BOOL ;            // Alarm: No confirmation MMC2 offline!
  Alarm5        :BOOL ;            // Alarm: MMC2 is not going offline!
  Alarm6       : BOOL ;            // Alarm: Queuing MMC is not going online!
  Report       : BOOL ;            // Alarm: Signoflife monitoring
  ErrorMMC    : BOOL ;            // Error detection MMC
END_VAR
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the MtoN function.

Table 4-24 Formal parameters of M to N function

Signal	Type	Type	Notes
Quit	I	BOOL	Acknowledge alarms
OPMixedMode	I	BOOL	Mixed mode deactivated for OP without M to N capability
ActivEnable	I	BOOL	Function is not supported. Control panel switch-over Interlocking via MMCx_SHIFT_LOCK in DB 19
MCPEnable	I	BOOL	Activate MCP switchover **TRUE =** MCP is switched over with operator panel. **FALSE: =** MCP is not switched over with operator panel. This can be used to permanently link an MCP. See also MMCx_MCP_SHIFT_LOCK in DB 19
Alarm1	O	BOOL	Alarm: Error in MMC bus address, bus type!
Alarm2	O	BOOL	Alarm: No confirmation MMC1 offline!
Alarm3	O	BOOL	Alarm: MMC1 is not going offline!
Alarm4	O	BOOL	Alarm: No confirmation MMC2 offline!
Alarm5	O	BOOL	Alarm: MMC2 is not going offline!
Alarm6	O	BOOL	Alarm: Queuing MMC is not going online!
Report	O	BOOL	Message: Sign-of-life monitoring MMC
ErrorMMC	O	BOOL	Error detection MMC

Note

The block must be called by the user program. The user must provide an instance DB with any number for this purpose. The call is not multi-instan-cecapable.

FB9 call

```
CALL FB 9, DB 109 (
  Quit          := Error_Ackn,      // e.g MCP reset
  OPMixedMode   := FALSE,
  ActivEnable   = TRUE,             //
  MCPEnable     := TRUE,            // Enable MCP switchover
  Alarm1        := DB2.dbx188.0,    // Error message 700.100
  Alarm2        := DB2.dbx188.1,    // Error message 700.101
  Alarm3        := DB2.dbx188.2,    // Error message 700.102
  Alarm4        := DB2.dbx188.3,    // Error message 700.103
  Alarm5        := DB2.dbx188.4,    // Error message 700.104
  Alarm6        := DB2.dbx188.5,    // Error message 700.105
  Report        :=DB2.dbx192.0);// Operating message 700.132
```

© Siemens AG, 2003. All rights reserved

Note

Input parameter “MCPEnable” must also be set to true to enable MCP switchover. The default value of these parameters is set in this way and need not be assigned specially when the function is called.

Alarm, error

The output parameters “Alarm1” to “Alarm6” and “Report” can be passed in the DB2 areas for MMC alarm and error messages.

If execution of an MMC function has failed (for which an appropriate error message cannot be displayed), status parameter ErrorMMC is set to ’logic 1’ (e.g. booting error when no connection is made).

Example of FB1 call (call in OB100)

```
CALL “RUN_UP” , “gp_par” (
  MCPNum                  := 1,
  MCP1In                  := P#E 0.0,
  MCP1Out                 := P#A 0.0,
  MCP1StatRec             := P#A 8.0,
  MCP1StatRec             := P#A 12.0,
  MCP1BusAdr              := 255,          // Address of 1st MCP
  MCP1Cycl                := S5T#200MS,
  MCP1Stop                := TRUE,         // MCP deactivated
  NCCyclTimeout           := S5T#200MS,
  NCRunupTimeout           := S5T#50S);
```

Example of override switchover

```
// Auxiliary flags used M100.0, M100.1, M100.2, M100.3
// Positive edge of MCP1Ready must check override and initiate measures for
// activation of MCP block
// This example applies to the feedrate override; interface and input bytes must
   be exchanged
// for spindle override.

U    DB10.DBX  104.0;      //MCP1Ready
     FN   M    100.0;      //Edge trigger flag 1
     SPBN  wei1;
     S    M    100.2;      //Set edge trigger flag 1
     R    M    100.3;      //Reset edge trigger flag 2


// Save override

      L DB21.DBB 4;        //Feedrate override interface
     T EB 28;              //Buffer memory (free input or flag byte)
```

© Siemens AG, 2003. All rights reserved

```
wei1:
    U    M   100.2;          //Switchover
    O    DB10.DBX  104.0;    //MCP1Ready
    SPBN  wei2;
    U    DB10.DBX  104.0;    //MCP1Ready
    FP   M   100.1;          //Edge trigger flag 2
    SPB wei2;
    U    M   100.2;          //Switchover
    R    M   100.2;          //Reset edge trigger flag 1
    SPB wei2;
    U   M    100.3;          //Comparison made
    SPB MCP;                 //Call MCP program

// Route saved override to interface of switched MCP
// until the override values match

    L EB28;                  //Route buffer memory
    T DB21.DBB 4;            //to override interface
    L EB 3;                  //Does it match override input byte
   <>i;                      //for feedrate ?
    SPB wei2;                //No, return
    S   M100.3;              //Yes, set auxiliary flag 2

 // When override values match, call the MCP program again

MCP:  CALL “MCP_IFM”(              //FC 19
        ModeGrNo      := B#16#1,
        ChanNo        := B#16#1,
        SpindleIFNo := B#16#0,
        FeedHold    := M    101.0,
        SpindleHold := M    101.1);

wei2: NOP 0;
```

© Siemens AG, 2003. All rights reserved

4.8 FB 10: Safety relay (SI relay)

Description of functions

The SPL module “Safety relay” for “Safety Integrated” is the PLC equivalent to the NC function of the same name. The standard SPL “Safety Relay” block is designed to support the implementation of an emergency stop function with safe programmable logic. However, it can also be used to implement other similar safety functions, e.g. control of a protective door. The function contains 3 input parameters (In1, In2, In3). On switchover of one of these parameters to the value 0, the output Out0 is deactivated without delay and outputs Out1, Out2 and Out3 deactivated via the parameterized timer values (parameters TimeValue1, TimeValue2, TimeValue3). The outputs are activated again without delay, if inputs In1 to In3 take the value 1 and a positive edge change is detected at one of the acknowledgement inputs Ack1, Ack2. To bring the outputs to their basic setting (values = 0) after booting, the parameter FirstRun must be configured as follows. Parameter FirstRun must be switched to the value TRUE via a retentive data (memory bit, bit in data block) on the first run after control booting. The data can be preset, e.g. in OB 100. The parameter is reset to FALSE when FB 10 is executed for the first time. A separate data must be used for each call with separate instance for parameter FirstRun.
The relevant NCK SPL block is described in:
References: /FBSI/Description of Functions Safety Integrated, Chapter 3

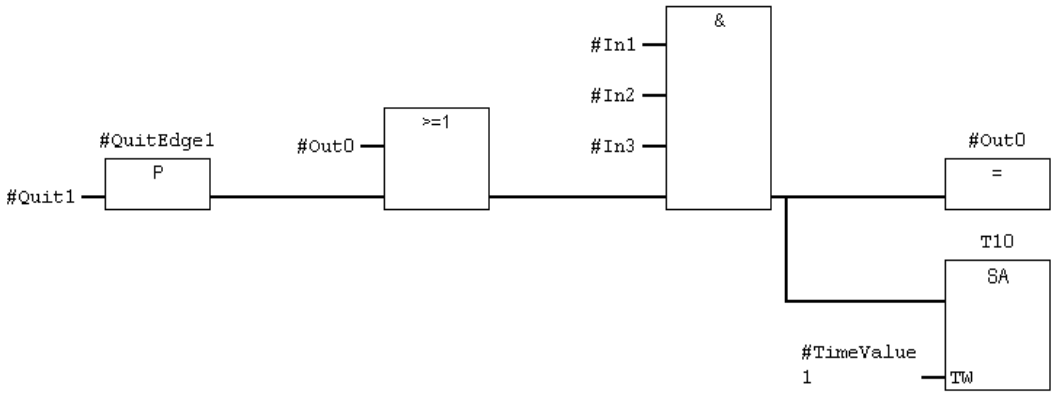
Simplified block diagram in CSF

The figure below shows only one acknowledgement input Ack1 and one delayed deactivation output Out1. The circuit for Ack2 and the other delayed outputs are identical. The parameter FirstRun is also missing in the function diagram. The mode of operation is desribed above.

© Siemens AG, 2003. All rights reserved

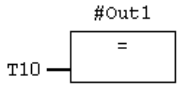
Network 2 : Title:

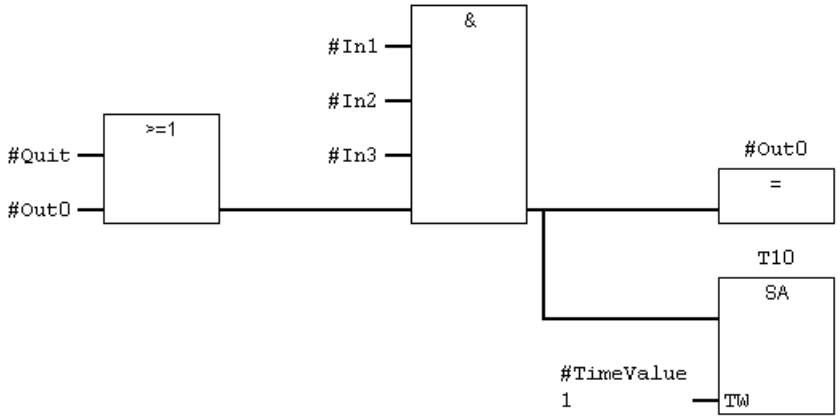
Simplified block diagram of saftey relay. In the diagram, timer T10 is used for the OFF delay.



Network 3: Title:

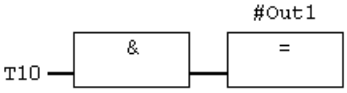
After TimeValue1 output OUT1 = FALSE.





Network 2: Title:

Comment:



© Siemens AG, 2003. All rights reserved

Declaration of the function

FUNCTION_BLOCK FB 10

```
VAR_INPUT
  In1 : BOOL := True ; //Input 1
  In2 : BOOL := True ; //Input 2
  In3 : BOOL := True ; //Input 3
  Quit1 : BOOL ; //Quit 1 Signal
  Quit2 : BOOL ; //Quit 2 Signal
  TimeValue1 : TIME := T#0ms ; //TimeValue for Output 1
  TimeValue2 : TIME := T#0ms ; //TimeValue for Output 2
  TimeValue3 : TIME := T#0ms ; //TimeValue for Output 3
END_VAR

VAR_OUTPUT
  Out0 : BOOL ; //Output without Delay
  Out1 : BOOL ; //Delayed Output to False by Timer 1
  Out2 : BOOL ; //Delayed Output to False by Timer 2
  Out3 : BOOL ; //Delayed Output to False by Timer 3
END_VAR

VAR_INOUT
   FirstRun BOOL: //True by User after 1st Start of SPL
END_VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the SI relay function.

Table 4-25 Formal parameters of SI relay function

Signal	Type	Type	Notes
In1	I	BOOL	Input 1
In2	I	BOOL	Input 2
In3	I	BOOL	Input 3
Quit1	I	BOOL	AcknowledgeInput 1
Quit2	I	BOOL	AcknowledgeInput 2
TimeValue1	I	TIME	Time value 1 for OFF delay
TimeValue2	I	TIME	Time value 2 for OFF delay
TimeValue3	I	TIME	Time value 3 for OFF delay
Out0	A	BOOL	Output undelayed
Out1	A	BOOL	Output delayed by TimeValue1
Out2	A	BOOL	Output delayed by TimeValue2
Out3	A	BOOL	Output delayed by TimeValue3
FirstRun	I/O	BOOL	Activation of initial setting

Note

The block must be called cyclically by the user program on starting the PLC program. The user must provide an instance DB with any number for this purpose. The call is multi-instance-capable.

© Siemens AG, 2003. All rights reserved

4.9 FB 11: Brake test

Description of functions

The braking operation check should be used for all axes which must be prevented from moving in an uncontrolled manner by a holding brake. This check function is primarily intended for the socalled “vertical axes”.
The machine manufacturer can use his PLC user program to close the brake at a suitable moment in time (guide value every 8h similar to the SI test stop) and allow the drive to produce an additional torque/additional force equivalent to the weight of the axis. In error-free operation, the brake can produce the necessary braking torque/braking force and keep the axis at a virtual standstill. When an error occurs, the actual position value exits the parameterizable monitoring window. In this instance, the position controller prevents the axis from sagging and negatively acknowledges the mechanical brake test. The required parameter settings for NC and drive are described in

References: /FBSI/ Description of Functions SINUMERIK Safety Integrated, Chapter 8

The brake test must always be started when the axis is at a standstill. For the entire duration of the brake test, the enable signals of the parameterized axis must be set to Enable (e.g. the Servo Disable, Feedrate Enable signals). The “PLC controlling axis” (DB “Axis”.DBX 28.7) must also be set to status 1 by the user program for the entire duration of the test. Before the “PLC controlling axis” signal is activated, the axis must be switched to “neutral axis”status (e.g. byte 8 must be set to channel 0 in the axis DB, activation signal must be set in the same byte, checkback about current status can be interrogated in byte 68). The block must not be started until the NC checkback via the appropriate bit (DB “Axis” DBX 63.1) has arrived. The direction in which the drive must produce its torque/force is specified by the PLC in the form of a “traversing motion” (e.g. via FC 18). The axis must be able to reach the destination of this movement without risk of collision if the brake is unable to produce the necessary torque/force.

The brake test is divided into the following steps

Table 4-26 Brake test sequence

Step	Expected checkback	Monitoring time value
Start brake test	DBX 71.0 = 1	TV_BTactiv
Close brake	Bclosed = 1	TV_Bclose
Issue travel command	DBX 64.6 Or DBX 64.7	TV_FeedCommand
Issue test travel command	DBX62.5 = 1	TV_FXSreached
Wait for hold time	DBX62.5 = 1	TV_FXShold
Deselect brake test/ open brake	DBX71.0 = 0	TV_BTactiv
Issue Test O.K.		

© Siemens AG, 2003. All rights reserved

Declaration of the function

FUNCTION_BLOCK FB 11

```
VAR_INPUT
  Start : BOOL ;  //Start for Braketest
  Quit : BOOL ;   //Quit Error
  Bclosed : BOOL ;    //brake closed input (single channel – PLC)
  Axis : INT ; //testing axisno.
  TimerNo : TIMER ;  //Timer from User
  TV_BTactiv : S5TIME ;      //TimeValue –> braketest activ
  TV_Bclose : S5TIME ;   //TimeValue –> close Brake
  TV_FeedCommand : S5TIME ; //TimeValue –> force FeedCommand
  TV_FXSreached : S5TIME ;  //TimeValue –> Fixed stop reached
  TV_FXShold : S5TIME ;    //TimeValue –> test brake
END_VAR
VAR_OUTPUT
  CloseBrake : BOOL ;  //Signal Close brake
  MoveAxis : BOOL ;    //do move axis
  Done : BOOL ;
  Error : BOOL ;
  State : BYTE ;   //Errorbyte
END_VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the brake test function

Table 4-27 Formal parameters of brake test function

Signal	Type	Type	Notes
Start	I	BOOL	Start brake test
Quit	I	BOOL	Acknowledge error
Bclosed	I	BOOL	Checkback input whether Close Brake is activated (singlechannel - PLC)
AXIS	I	INT	**Axis number of axis to be tested**
TimerNo	I	TIMER	Timer from user program
TV_BTactiv	I	S5TIME	Monitoring time value -> brake test active, check of axis signal DBX71.0
TV_Bclose	I	S5TIME	Monitoring time value -> close brake. Check of input signal Bclosed after output CloseBrake has been set.
TV_FeedCommand	I	S5TIME	Monitoring time value -> issue travel command. Check travel command after MoveAxis has been set.
TV_FXSreched	I	S5TIME	Monitoring time value -> fixed stop reached
TV_FXShold	I	S5TIME	Monitoring time value -> test brake.
CloseBrake	O	BOOL	Request for close brake.
MoveAxis	O	BOOL	Request for initiate travel movement
Done	O	BOOL	Test successfully completed
Error	O	BOOL	Error has occurred.
State	O	BYTE	ErrorStatus

© Siemens AG, 2003. All rights reserved

Error identifiers

Table 4-28 Error identifiers

State	Meaning
0	No error
1	Start conditions not fulfilled, e.g. axis not under closed-loop control/brake closed/axis disabled
2	No NC checkback in “Brake test active” signal on selection of brake test
3	No “Brake applied” checkback by input signal BClosed
4	No travel command output (e.g. axis motion has not been started)
5	Axis is not reaching fixed stop -> axis RESET has been activated.
6	Travel disable/approach too slow -> fixed stop cannot be reached. TV FXSreached monitoring timeout.
7	Brake is not holding at all (end position is reached)/approach velocity too high
8	Brake opens during the holding period
9	Error in brake test deselection
10	Internal error
11	“PLC controlling axis” signal not activated by the user program

Alarm no.	Meaning	Remedy
411101	Parameter axis not within permissible range	Use the legal axis number

Note

The block must be called by the user program. The user must provide an instance DB with any number for this purpose. The call is multi-instance-capable.

Example of FB11 call:

```
UN   M    111.1; //Request close Z axis brake from FB
=    A    85.0; //Control Z axis brake

AUF   “Axis3”; //Test on Z axis brake

O    E    73.0; //Initiate brake test on Z axis
O    M    110.7; //Brake test in progress
FP   M    110.0;
UN   M    111.4; //Error
S    M    110.7; //Brake test in progress
S    M    110.6; //Next step
S    DBX    8.4; //Request neutral axis

U    DBX    68.6; //Axis is neutral checkback
U    M    110.6;
FP   M    110.1
R    M    110.6
```

© Siemens AG, 2003. All rights reserved

```
S    M   110.5; //Next step
R    DBX  8.4;
S    DBX  28.7; //Request PLC-controlled axis

U    DBX  63.1; //PLC-controlled axis checkback
U    M   110.5;
FP   M   110.2;
R    M   110.5;
S    M   111.0; //Start brake test for FB

CALL FB  11 , DB  211 (//Brake test block
     Start          :=M 111.0, //Start brake test
     Quit           :=E 3.7, //Acknowledge error with Reset key
     Bclosed        := E 54.0, //Close brake initiated checkback
     Axis           := 3, //Axis number of axis to be tested, Z axis
     TimerNo        := T   110, //Timer number
     TV_BTactiv     := S5T#200MS, //Monitoring time value: Brake test active
                       DBX71.0
     TV_Bclose      := S5T#1S, //Monitoring time value: Brake closed
     TV_FeedCommand := S5T#1S, //Monitoring time value: Travel command
                               OFF ssued
     TV_FXSreache   := S5T#1S, //Monitoring time value: Fixed stop
                               reached
     TV_FXShold     := S5T#2S, //Monitoring time value: Test time
                               Brake
     CloseBrake     := M   111.1, //Request close brake
     MoveAxis       := M   111.2, //Request initiate travel
     Done           := M   111.3, //Test ended successfully
     Error          := M   111.4, //Error
     State          := MB  112); //Error status

OPEN                “Axis3”; //Brake test on Z axis

O    M   111.3; //Test ended successfully
O    M   111.4; //Error
FP   M   110.3;
R    DBX  28.7; //Request PLC-controlled axis

UN   DBX  63.1; //PLC-controlled axis checkback
U    M   111.0; //Start brake test for FB
U    M   110.7; //Brake test in progress
FP   M   110.4;
R    M   111.0; //Start brake test for FB
R    M   110.7; //Brake test in progress

CALL “SpinCtrl” (//Move Z axis
     Start          := M   111.2, //Start travel
     Stop           := FALSE,
     Funct          := B#16#5, //Mode: Axis mode
     Mode           := B#16#1, //Traverse: Incremental
     AxisNo         := 3, //Axis number of travel axis, Z axis E
     Pos            := –5.000000e+000, //Distance: Minus 5 mm
     FRate          := 1.000000e+003, //Feedrate: 1000 mm/min
     InPos          := M   113.0, //Position reached
     Error          := M   113.1, //Error
     State          := MB  114); //Error status
```

© Siemens AG, 2003. All rights reserved

4.10 FB 29: Signal recorder and data trigger diagnostics

Signal recorder

The “Diagnostics” FB allows various diagnostic routines to be performed on the PLC user program. A diagnostic routine logs signal states and signal changes. In this diagnostic routine, function number 1 is assigned to the Func parameter. Up to 8 Boolean signals (parameters Signal_1 to Signal_8) are recorded in a circular buffer each time one of the signals changes. The current information of parameters Var1 (byte value), and Var2 and Var3 (integer values) is also stored in the circular buffer. The number of past OB 1 cycles is also stored in the buffer as additional information. This information enables the graphical evaluation of signals and values in OB 1 cycle grid. The first time the “diagnostics” FB is called in a new PLC cycle, the NewCycle parameter must be set to TRUE. If the “diagnostics” FB is called several times in the same OB 1 cycle, the NewCycle parameter must be set to FALSE for the second and subsequent calls. This prevents a new number of OB 1 cycles from being calculated. The circular buffer is set up by the user. The DB of the circular buffer must be passed to the diagnostics FB in the BufDB parameter. The circular buffer must use an array structure, as specified in the source code. The array can have any number of elements. A size of 250 elements is recommended. The ClearBuf parameter is used to clear the circular buffer and set the BufAddr pointer (I/O parameter) to the start. The associated instance DB for the FB is a DB from the user area.

Data trigger

The data trigger function is intended to allow triggering on specific values (or bits) at any permissible memory cell. The cell to be triggered is “rounded” with a bit mask (AndMask parameter) before the TestVal parameter is compared in the diagnostic block.

Note

The source code for the function is available in the source container of the basic program library under the name Diagnose.awl. The instance DB and the circular buffer DB are also defined in this source block. The function call is also described in the function. The DB numbers and the call must be modified.

© Siemens AG, 2003. All rights reserved

```
FUNCTION_BLOCK FB 29
VAR_INPUT
 Func : INT ;          //Function number
//0 = No Function, 1 = Signal recorder, 2 = Data trigger

   Signal_1 : BOOL ;
    Signal_2 : BOOL ;
    Signal_3 : BOOL ;
    Signal_4 : BOOL ;
    Signal_5 : BOOL ;
    Signal_6 : BOOL ;
    Signal_7 : BOOL ;
    Signal_8 : BOOL ;
    NewCycle : BOOL ;
    Var1 : BYTE ;
    Var2 : INT ;
    Var3 : INT ;
    BufDB : INT ;
    ClearBuf : BOOL ;
    DataAdr : POINTER ;                    //Area pointer to testing word
    TestVal : WORD ;   //Value for triggering
    AndMask : WORD ;                       //AND- Mask to the testing word
END_VAR
VAR_OUTPUT
   TestIsTrue : BOOL ;
END_VAR
VAR_IN_OUT
   BufAddr : INT ;
END_VAR
```

Structure for circular buffer

```
DATA_BLOCK DB 81
TITLE =
//Ring buffer DB for FB 29
VERSION : 1.0

   STRUCT
    Field : ARRAY [0 .. 249 ] OF STRUCT   //Can be any size of this struct

   Cycle : INT ;          //Delta cycle to last storage in buffer
     Signal_1 : BOOL ;                    //Signal names same as FB 29
     Signal_2 : BOOL ;
     Signal_3 : BOOL ;
     Signal_4 : BOOL ;
     Signal_5 : BOOL ;
     Signal_6 : BOOL ;
     Signal_7 : BOOL ;
     Signal_8 : BOOL ;
     Var1 : BYTE ;
     Var2 : WORD ;
     Var3 : WORD ;
   END_STRUCT ;
    END_STRUCT ;
BEGIN
END_DATA_BLOCK
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the function Diagnostics

Signal	Type	Type	Value range	Notes
Func	I	Int	0, 1, 2	Function 0: Deactivate 1: Signal recorder 2: Data trigger
Parameters for function 1				
Signal_1 to Signal_8	I	Bool		Bit signals checked for change
NewCycle	I	Bool		See the “Signal recorder” description above
Var1	I	Byte		Additional value
Var2	I	Int		Additional value
VAR	I	Int		Additional value
BufDB	I	Int		Circular buffer DB no.
ClearBuf	I	Bool		Delete circular buffer DB and reset pointer BufAddr
BufAddr	I/O	Int		Target area for read data
Parameters for function 2				
DataAdr	I	Pointer		Pointer to word to be tested
TestVal	I	Word		Comparison value
AndMask	I	Word		See description
TestIsTrue	O	Bool		Result of comparison

Configuration steps

Select function of diagnostics block. Define suitable data for the recording as signal recorder or data triggering. Find a suitable point or points in the user program for calling the diagnostics FB. Create a data block for the circular buffer (see call example). Call the diagnostics FB with parameters in the user program. In function 1, it is advisable to clear the circular buffer with the ClearBuf parameter. When the recording phase is complete (function 1), read out the circular buffer DB in STEP7 by opening the data block in the data view. You can now analyze the contents of the circular buffer DB (e.g. in a signal timing chart).

© Siemens AG, 2003. All rights reserved

Call example

```
FUNCTION FC 99: VOID
TITLE =
VERSION : 0.0

BEGIN
NETWORK
TITLE =   NETWORK

CALL FB    29, DB    80 (

  Func               := 1,
       Signal_1            := M    100.0,
       Signal_2            := M    100.1,
       Signal_3            := M    100.2,
       Signal_4            := M    100.3,
       Signal_5            := M    100.4,
       Signal_6            := M    100.5,
       Signal_7            := M    100.6,
       Signal_8            := M    100.7,
       NewCycle              := TRUE,
       Var1                := MB   100,
       BufDB               := 81,
       ClearBuf            := M     50.0);
END_FUNCTION
```

© Siemens AG, 2003. All rights reserved

4.11 FC 2: GP_HP Basic program, cyclical section

Description of functions

The complete processing of the NCKPLC interface is carried out in cyclic operation. In order to minimize the execution time of the basic program, only the control/status signals are transmitted cyclically; transfer of the auxiliary functions and G functions only takes place when requested by the NCK.

Declaration

```
FUNCTION FC 2: VOID
//No parameters
```

Call example

As far as the time is concerned, the basic program must be passed **prior to the** execution of the user program. It is therefore called first in OB 1.

The following example contains the standard declarations for OB 1 and the calls for the basic program (FC2), the transfer of the machine control panel signals (FC19) and the acquisition of error and operating messages (FC10).

```
ORGANIZATION_BLOCK OB 1
VAR_TEMP
  OB1_EV_CLASS :                BYTE;
  OB1_SCAN_1 :                  BYTE;
  OB1_PRIORITY :                BYTE;
  OB1_OB_NUMBR :                BYTE;
  OB1_RESERVED_1 :              BYTE;
  OB1_RESERVED_2 :              BYTE;
  OB1_PREV_CYCLE :              INT;
  OB1_MIN_CYCLE :               INT;
  OB1_MAX_CYCLE :               INT;
  OB1_DATE_TIME :               DATE_AND_TIME;
END_VAR
BEGIN
  CALL FC 2;                            //Call basic program as 1st FC
  //INSERT USER PROGRAM HERE
  CALL FC 19(                           //MCP signals to interface
       ModeGroupNo :=   B#16#1,         //Mode group No. 1
       ChanNo :=        B#16#1,         //Channel no. 1
       SpindleIFNo :=   B#16#4,         //Spindle interface number = 4
       FeedHold :=      f22.0,          //Feed stop signal
                                        //modal
       SpindleHold :=   db2.dbx151.0);  //Spindle stop modal in message
                                        //data block
  CALL FC 10(                           //Error and operational messages
       ToUserIF :=      TRUE,           //Transfer signals from DB2
                                        //to interface
       Quit :=          I6.1);          //Acknowledgment of error
                                        //messages via I 6.1
END_ORGANIZATION_BLOCK
```

© Siemens AG, 2003. All rights reserved

4.12 FC 3: GP_PRAL Basic program, interrupt-controlled section

Description of functions

In the interrupt-driven part of the basic program, the block-synchronized transfers from the NCK to the PLC (auxiliary and G functions) are carried out. **Auxiliary functions** are subdivided into normal and high-speed auxiliary functions. The high-speed functions of an NC block are buffered and the transfer acknowledged to the NC. These are passed to the application interface at the start of the next OB1 cycle. Normal auxiliary functions are only acknowledged when they have existed for the duration of one cycle. This allows the application to issue a read disable to the NC.

High-speed auxiliary functions programmed immediately one after the other, are not lost for the user program. This is ensured by a mechanism in the basic program.

The G functions are evaluated immediately and passed to the application interface.

NC process alarms

If the interrupt is triggered by the NC (possible in each IPO cycle), a bit in the local data of OB 40 (“GP_IRFromNCK“) is set by the basic program. (only if FB 1 parameter UserIR := TRUE). This data is not set on other events (process interrupts through I/Os). This information makes it possible to branch into the associated interrupt routine in the user program in order to initiate the necessary action. To be able to implement high-speed, job-controlled processing of the user program for the machine, the following NC functions are available in the interrupt processing routine (OB 40 program section) for the PLC user program with SW 3.2 and higher:

- Selected **auxiliary functions**,
- **Tool change function** for tool management option,
- **Position reached** for positioning axes, indexing axes and spindles with activation via PLC,
- Block transfer to FM (function available soon).

The functions listed above can or must be evaluated by the user program in OB 40 in order to initiate reactions on the machine. As an example, the revolver switching mechanism can be activated when a T command is programmed on a turning machine.

For further details on programming hardware interrupts (time delay, interruptibility, etc.) refer to the corresponding SIMATIC documentation.

© Siemens AG, 2003. All rights reserved

Auxiliary functions

Generally, fast or acknowledging auxiliary functions are processed with or without interrupt control independently of any assignment.
Basic program parameters in FB 1 can be set to define which auxiliary functions (T, H, DL) must be processed solely on an interrupt-driven basis by the user program.
Functions which are not assigned via interrupts are only made available by the cyclic basic program as in earlier versions. The change signals of these functions are available in a PLC cycle.
Even if the selection for the auxiliary function groups (T, H, DL) is made using interrupt control, only one interrupt can be processed by the user program for the selected functions.
A bit is set channel-specifically in the local data “GP_AuxFunction” for the user program (if “GP_AuxFunction[1]” is set, then an auxiliary function is available for the 1st channel).
The change signals and function value are available to the user in the associated channel DB. The change signal for this interrupt-controlled function is reset to zero in the cyclical basic program section after the execution of at least one full OB1 cycle (max. approx. two OB1 cycles).

Tool change

With the Tool management option, the tool change command for revolver and the tool change in the spindle is supported by an interrupt. The local data bit “GP_TM” in OB 40 is set for this purpose. The PLC user program can thus check the tool management DB (DB 72 or DB 73) for the tool change function and initiate the tool change operation.

Position reached

In the bit structure, “GP_InPosition” of the local data of OB 40 is specific to the machine axis (each bit corresponds to an axis/spindle, e.g. GP_InPosition[5] corresponds to axis 5).
If a function has been activated via FC 15 (positioning axis), FC 16 (indexing axis) or FC 18 (spindle control) for an axis or spindle, the associated “GP_InPosition” bit makes it possible to implement instantaneous evaluation of the “InPos” signal of the FCs listed above. This capability can be used, for example, to obtain immediate activation of clamps for an indexing axis.

Declaration

```
FUNCTION FC 3: VOID
//No parameters
```

© Siemens AG, 2003. All rights reserved

Call example

As far as the time is concerned, the basic program must be passed **prior to the** execution of further interrupt-controlled user programs. It is therefore called first in OB 40.

The following example contains the standard declarations for OB 40 and the call for the basic program.

```
ORGANIZATION_BLOCK OB 40
VAR_TEMP
  OB40_EV_CLASS :                BYTE;
  OB40_STRT_INF :                BYTE;
  OB40_PRIORITY :                BYTE;
  OB40_OB_NUMBR :                BYTE;
  OB40_RESERVED_1 :              BYTE;
  OB40_MDL_ID :                  BYTE;
  OB40_MDL_ADDR :                INT;
  OB40_POINT_ADDR :              DWORD;
  OB40_DATE_TIME :               DATE_AND_TIME;
//Assigned to basic program
GP_IRFromNCK : BOOL;                  //Interrupt durch NCK fuer Anwender
GP_TM : BOOL;                         //Tool management
GP_InPosition : ARRAY [1..3] OF BOOL; //Axis-oriented for positioning axes,
                                       //Indexing axes, spindles
GP_AuxFunction : ARRAY [1..10] OF BOOL; //Channel-oriented for //auxiliary functions
GP_FMBlock : ARRAY [1..10] OF BOOL; //Channel-oriented for
                                       //Transfer to FM (available soon)

  //Further local user data may be defined from this point onwards
END_VAR
BEGIN
  CALL FC 3;

  //INSERT USER PROGRAM HERE
END_ORGANIZATION_BLOCK
```

© Siemens AG, 2003. All rights reserved

4.13 FC 5: GP_DIAG basic program, diagnostic alarm (only FM-NC)

Description of functions

Module faults are acquired in the diagnostic alarm controlled part of the basic program.

In the FM-NC, NCK Reset is recorded by the PLC via the diagnostic alarm as a module failure.

Once the response program has been initiated, another diagnostic alarm signals that the module fault has been eliminated.

Declaration

FUNCTION FC 5: VOID

//No parameters

Call example

As far as the time is concerned, the basic program must be passed prior to the execution of further interrupt-controlled user programs. It must therefore be called first in OB82.

The following example contains the standard declarations for OB 82 and the call for the basic program.

```
ORGANIZATION_BLOCK OB 82
VAR_TEMP
  OB82_EV_CLASS :          BYTE ;
  OB82_PRIORITY :          BYTE ;
  OB82_RESERVED_1 :        BYTE ;
  OB82_MDL_ADDR :          INT ;
  OB82_INT_FAULT :         BOOL ;
  OB82_EXT_FAULT :         BOOL ;
  OB82_EXT_VOLTAGE :       BOOL ;
  OB82_NO_CONFIG :         BOOL ;
  OB82_MDL_TYPE :          BYTE ;
  OB82_COMM_FAULT :        BOOL ;
  OB82_WTCH_DOG_FLT :      BOOL ;
  OB82_PRIM_BATT_FLT :     BOOL ;
  OB82_RESERVED_2 :        BOOL ;
  OB82_PROC_FLT :          BOOL ;
  OB82_RAM_FLT :           BOOL ;
  OB82_FUSE_FLT :          BOOL ;
  OB82_RESERVED_3 :        BOOL ;
END_VAR
BEGIN
  CALL FC 5;
  //INSERT USER PROGRAM HERE
END_ORGANIZATION_BLOCK
```

© Siemens AG, 2003. All rights reserved

4.14 FC 7: TM_REV Transfer block for tool change with revolver

Description of functions

After a revolver has been changed, the user will call this block. The revolver number (corresponding to interface number in DB 73) must be specified in parameter “ChgdRevNo” for this purpose. As this block is called, the associated “Interface active” bit in data block DB 73, word 0 of FC 7 is reset after parameter “Ready” := TRUE is returned.

Block FC TM_REV may be started (with “Start” parameter = “TRUE”) only if an activation signal for the appropriate interface (DB 73, word 0) for this transfer has been supplied by the tool management function.

Output parameter “Ready” is set to the value TRUE when the job has been executed correctly.
The user must then set
Parameter “Start” = “FALSE”
or not call the block again. If
parmeter “Ready” = FALSE,
the error code in **parameter “Error”** must be interpreted.
If the error code = 0, then this job must be repeated in the next PLC cycle (e.g. “Start” remains set to “TRUE”). This means that the transfer job has not yet been completed (see example FC 7 call and pulse diagram). The “Start” parameter does not need a signal edge for a subsequent job.



Warning

It is not permissible to abort the transfer (e.g. by an external signal reset). The *Start* parameter must always retain the 1 signal until the *Ready* and/or *Error* parameters are <> 0.

An error code of <> 0 indicates incorrect parameterization.

Note

For further details on tool management (also with regard to PLC) refer to the Description of Functions Tool Management. Furthermore, PI services for tool management are provided via the FB 4, FC 8 and FC 22 (see also corresponding Sections in this documentation). Machine data 20310 bit 12 must not be set to 1 for circular magazines.

© Siemens AG, 2003. All rights reserved

Manual revolver switching

If a manual action is used to rotate the revolver, it is necessary to pass this information on to the tool management system. The asynchronous transfer function of FC 8 must be used to transfer the modified positions of the revolver. This must only occur once on the first manual rotation in the sequence. In this case, the following parameterization of the asynchronous transfer is needed via FC 8:

TaskIdent = 4
TaskIdentNo = channel
NewToolMag = magazine number of revolver
NewToolLoc = original location of tool
OldToolMag = magazine number of buffer (spindle) = 9998
OldToolLoc = buffer number of spindle
Status = 1

This measure also causes the same T command to be repeated at the tool management interface if the previous T is programmed again.

Declaration of the function

STL representation

```
FUNCTION FC 7 :    void
//NAME :TM_REV
VAR_INPUT
   Start:          BOOL;
   ChgdRevNo:      BYTE;
END_VAR
VAR_OUTPUT
   Ready:          BOOL;
   Error:          INT;
END_VAR
BEGIN
END_FUNCTION
```

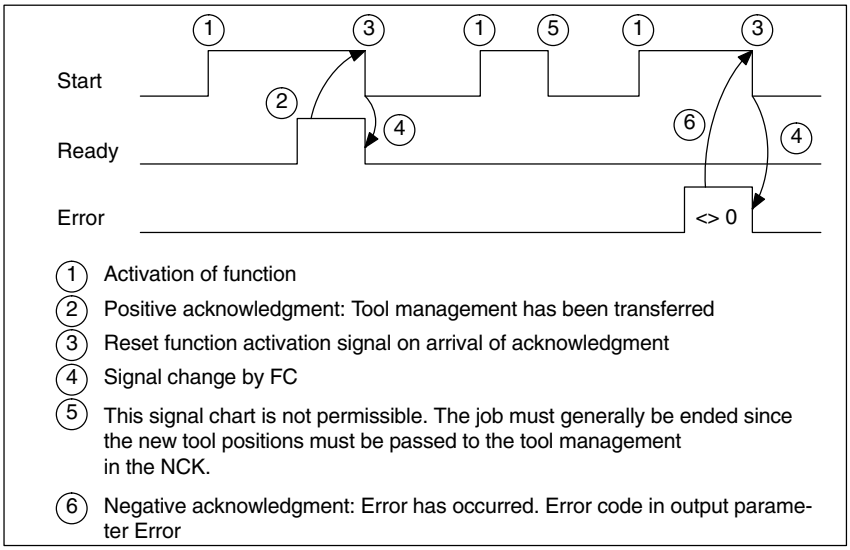
Explanation of the formal parameters

The following table shows all formal parameters of the function TM_REV.

Signal	Type	Type	Value range	Notes
Start	I	Bool		1 = Start of transfer.
ChgdRevNo	I	Byte	1..	Number of revolver interface
Ready	A	Bool		1 = Transfer complete
Error	A	Int	0..3	Error checkback 0: No error has occurred 1: No revolver present 2: Illegal revolver number in parameter “ChgdRevNo” 3: Illegal task (“signal “Interface active” of selected revolver = “FALSE”)

© Siemens AG, 2003. All rights reserved

Pulse diagram



(1) Activation of function
(2) Positive acknowledgment: Tool management has been transferred
(3) Reset function activation signal on arrival of acknowledgment
(4) Signal change by FC
(5) This signal chart is not permissible. The job must generally be ended since the new tool positions must be passed to the tool management in the NCK.
(6) Negative acknowledgment: Error has occurred. Error code in output parameter Error

Call example

```
CALL FC 7(          //Tool management transfer block for revolver
          Start :=          m 20.5, //Start := "1 " => Initiation of transfer
          ChgdRevNo :=      DB61.DBB 1,
          Ready :=          f 20.6,
          Error :=          DB61.DBW 12);

a f 20.6;                   //Interrogate ready
r f 20.5;                   //Reset start
jc m001;                    //Jump if everything is OK
l db61.dbw 12;              //Error information
ow w#16#0;                  //Evaluate error
jn err;                     //Jump to error treatment if <> 0
f001:                       //Commence run of remaining program

err:
r f 20.5                    //Reset start if error is present
```

© Siemens AG, 2003. All rights reserved

4.15 FC 8: TM_TRANS Transfer block for tool management

Description of functions

In the case of changed tool positions or status changes, the user will call FC TM_TRANS. This FC is parameterized with parameter “TaskIdent”:

1. For loading/unloading positions,
2. For spindle change positions,
3. For revolver change positions as transfer identifier.
4. Asynchronous transfer
5. Asynchronous transfer with location reservation

The interface number is indicated in parameter “TaskIdentNo”.

Example for loading point 5:
Parameter “TaskIdent”:= 1 and “TaskIdentNo”= 5.

Furthermore, the current tool positions and status data (list of “Status” parameter in the following text) are also transmitted for this transfer function.

Note

FC8 informs the NCK of the current positions of the old tool.

The NCK knows where the old and the new tool have been located until the position change.

In the case of a transfer without a socalled “old tool” (e.g. on loading), the value 0 is assigned to parameters “OldToolMag”, “OldToolLoc”.

Block FC TM_TRANS may be started (with “Start” parameter = “TRUE”) only if an activation signal for the appropriate interface (DB 71, DB 72, DB 73 in word 0) for this transfer has been supplied by the tool management function.

If this job has been executed correctly, the
output parameter “Ready” contains the value TRUE.
The user must then set **parameter “Start”** = “FALSE”
or not call the block again. If
parmeter “Ready” = FALSE,
the error code in **parameter “Error”** must be interpreted (see Call example FC 8 and pulse diagram).

If the error code = 0, then this job must be repeated in the next PLC cycle (e.g. “Start” remains set to “TRUE”). This means that the transfer operation has not yet been completed.

If the user assigns a value of less than 100 to the parameter “Status” then the associated interface in data block DB71 or DB 72 or DB 73, word 0 is deactivated (process complete). The appropriate bit for the interface is set to 0 by FC 8.

The “Start” parameter does not need a signal edge for a subsequent job. This means that new parameters can be assigned with “Start = TRUE” immediately when “Ready = TRUE” is received.

© Siemens AG, 2003. All rights reserved

Asynchronous transfer

To ensure that changes in the position of a tool are automatically signaled from PLC to the tool management (e.g. power failure in the case of an active command or accidental changes in the position by the PLC), the block FC TM_TRANS with “TaskIdent”:= 4 or 5 is called. This call does not require any interface activation by the tool management function.

If parameter “TaskIdent”= 5, the tool management system reserves the location in addition to changing the position. The location is only reserved, however, if the tool has been transported from a real magazine into a buffer.
An associated NC channel must be configured in the “TaskIdentNo” parameter. The previous position of the tool is specified in parameters “OldToolMag”, “OldToolLoc”; the current position of the tool is specified in parameters “NewToolMag”, “NewToolLoc” Status = 1 must be specified.

With status 5, the specified tool remains at location “OldToolMag”, “OldToolLoc”. This location must be a buffer (e.g. spindle). The real magazine and location must be specified in the parameters “NewToolMag”, “NewToolLoc”; the location is at the position of the buffer. This procedure must always be used if the tool management system is to be informed of the position of a specific magazine location. This procedure is used for alignment in search strategies.



Warning

It is not permissible to abort the transfer (e.g. by an external signal reset). The *Start* parameter must always retain the 1 signal until the *Ready* and/or *Error* parameters are <> 0.

An error code of <> 0 indicates incorrect parameterization.

Note

For further details on tool management (also with regard to PLC) refer to the Description of Functions Tool Management. Furthermore, PI services are provided for tool management via the FB 4, FC 7 and FC 22 (see also the corresponding Section in this documentation).

© Siemens AG, 2003. All rights reserved

Declaration of the function

STL representation

```
FUNCTION FC 8 : void
//NAME :TM_TRANS
VAR_INPUT
  Start:          BOOL;
  TaskIdent:      BYTE;
  TaskIdentNo:    BYTE;
  NewToolMag:     INT;
  NewToolLoc:     INT;
  OldToolMag:     INT;
  OldToolLoc:     INT;
  Status:         INT;
END_VAR
VAR_OUTPUT
  Ready:          BOOL;
  Error:          INT;
END_VAR
BEGIN
END_FUNCTION
```

© Siemens AG, 2003. All rights reserved

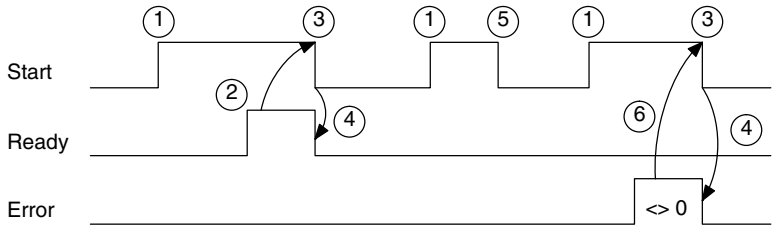
Explanation of the formal parameters

The following table shows all formal parameters of the function TM_TRANS.

Signal	Type	Type	Value range	Notes
Start	I	Bool		1 = Start of transfer
TaskIdent	I	Byte	1..5	Interface or tank identifier 1: Loading/unloading point 2: Spindle/change point 3: Revolver change point 4: Asynchronous transfer 5: Asynchronous transfer with reservation of
TaskIdentNo	I	Byte	1..	Number of associated interface or channel number. The upper nibble can specify the interface number for asynchronous transfer (e.g. B#16#12, 1st interface, 2nd channel).
NewToolMag	I	Int	-1, 0..	Current magazine number of new tool -1: Tool remains at its location. NewToolLoc = any. Only with TaskIdent = 2 (SW 5.3)
NewToolLoc	I	Int	0..max. location number	Current location number of new tool
OldToolMag	I	Int	-1, 0..	Current magazine number of new tool -1: Tool remains at its location. OldToolLoc = any. Only with TaskIdent = 2 (SW 5.3)
OldToolLoc	I	Int	0..max. location number	Current location number of new tool
Status	I	Int	1..7, 103..105	Status information about transfer operation
Ready	O	Bool		1 = Transfer complete
Error	O	Int	0..65535	Error checkback 0: No error has occurred 1: Unknown “TaskIdent” 2: Unknown “TaskIdentNo” 3: Illegal task (“signal “Interface active” of selected revolver = FALSE) Other values: The number corresponds to the error message of the tool management function in the NCK caused by this transfer.

© Siemens AG, 2003. All rights reserved

Pulse diagram



(1) Activation of function
(2) Positive acknowledgment: Tool management has been transferred
(3) Reset function activation signal on arrival of acknowledgment
(4) Signal change by FC
(5) This signal chart is not permissible. The job must generally be ended since the new tool positions must be passed to the tool management in the NCK.
(6) Negative acknowledgment: Error has occurred. Error code in output parameter Error

Status list

Status = 1:
The operation is complete (loading/unloading/reloading, prepare change, change).

The parameters of FC 8 (FC TM_TRANS) “NewToolMag”, “NewToolLoc”, “OldToolMag”and “OldToolLoc” must be parameterized to the tool target positions specified in the interface (except for Prepare change). For further information, please refer to description of parameters of FC TM_TRANS or general Section of tool management in the PLC.

1. In the case of loading/unloading/reloading, the tool has arrived at the required target address. If the bit in the interface in DB 71.DBX (n+0).3 “position at loading point” is enabled, status 1 cannot be used for the function termination. Status 5 must be used for correct termination.
2. In the case of “Prepare change” the new tool is now available. The tool may, for example, be positioned in a buffer (gripper). In some cases, the target (magazine, location) of the old tool has been moved to the toolchange position after placement of the new tool in a buffer. However, the old tool still remains in the spindle. The preparations for a tool change are thus complete. After this acknowledgment, the “Change” command can be received. The positions in parameters “NewToolMag”, “NewToolLoc” ,“OldToolMag” and “OldToolLoc” correspond to the current tool positions.
3. In the case of “Change” (spindle or revolver), the tools addressed in the interface have now reached the required target addresses. The tool change operation is thus complete.

© Siemens AG, 2003. All rights reserved

Status = 2:
The "new" tool cannot be made available. This status is only admissible in conjunction with the "Change tool" command. When this status is applied, the PLC must be prevented from making a change with the proposed tool. The proposed (new) tool is disabled by the tool management function in the NCK. A new command is then output by the tool management with a duplo tool. The positions in parameters "NewToolMag", "NewToolLoc", "OldToolMag" and "OldToolLoc" correspond to the original tool positions.

Status = 3:
An error has occurred. The tool positions must not have been changed. Any changes to the magazine positions which have taken place in the meantime must be notified beforehand, for example, with status = 105 via FC TM_TRANS. Only then will the tool positions be taken into account by the tool management function.

Status = 4:
It would be better to position the "old" tool in the magazine position specified in parameters "OldToolMag" and "OldToolLoc". This status is permissible only in conjunction with preparation for tool change (change into spindle). After this status has been transferred to the tool management function in the NCK, the tool management tries to take the specified magazine position into account in the subsequent command. However, it can only do so if the position is free. Parameters "NewToolMag" and "NewToolLoc" are not taken into account.

Status = 5:
The operation is complete. The "new" tool is **in** the position specified in parameters "NewToolMag", "NewToolLoc". In this case, the specified tool is not really in this position, but is still in the same magazine location. However, this magazine location has been moved to the position set in the parameters (e.g. tool change position). This status may be used only for revolvers, chain-type magazines and disk magazines. Status 5 enables the tool management function to adjust the current position of a magazine and to improve the search strategy for subsequent commands. This status is permissible only in conjunction with loading, unloading, and reloading operations and with preparations for a tool change.

The "OldToolMag" and "OldToolLoc" parameters must be parameterized with the data of a buffer.

- **Loading, reloading**:
  On loading or reloading, a location for the tool is already reserved in the NCK. The machine operator must then insert the tool at the target location. Caution: The location reservation is canceled when the control system is switched on again.
- **Preparations for change**:
  Tool motions still to be executed are not carried out until after the tool has been changed.
- **Positioning at loading point:**
  If the bit in the interface in DB 71.DBX (n+0)0.3 "position at loading point" is enabled, only status 5 (not status 1) may be used for the function termination.

© Siemens AG, 2003. All rights reserved

Status = 6:
The tool management is completed. This status has the same function as status 1, but, in addition, a reservation of the source location is carried out. This status is only permitted when reloading. The command is ended and the source location of the tool is reserved if the target location is in a buffer magazine.

Status = 7: (ab SW 6)
Repetition of the "Prepare Tool" command initiated. This status is only admissible in conjunction with the Prepare Tool Change command. This status is intended for use when the "new" tool has changed its position (e.g. via an asynchronous command of the "new" tool). After Ready = 1 has been provided from FC 8, the Prepare Change command is repeated automatically with the same tool.
After "Ready = 1" has been provided from FC 8, the "Prepare Change" command is repeated automatically with the same tool. For the automatic repetition, a new tool search is carried out. The positions in parameters "NewToolMag", "NewToolLoc", "OldToolMag", "OldToolLoc" must correspond to the original positions of the tools.

Status = 103:
The "new" tool can be inserted. This status is only permissible in conjunction with the preparations for a tool change if the PLC may reject the new tool (MD: MC_TOOL_MANAGEMENT_MASK, bit 4). The tool positions have remained unchanged. This status is required to ensure that preprocessing continues in the NCK (otherwise processing is stopped).

References: /FBW/, Description of Functions, Tool Management

Status = 104:
The "new" tool is in the position specified in parameters "NewToolMag", "NewToolLoc". This status is permissible only if the tool is still in the magazine in the same location. The "old" tool is in the position (buffer) specified in parameters "OldToolMag", "OldToolLoc". In this case, however, the new tool is not really in this position, but is still in the same magazine location. However, this magazine location has been moved to the position set in the parameters (e.g. tool change position). This status may be used only in conjunction with revolvers, chain-type magazines and disk magazines for the "Tool change preparation" phase. Status 104 enables the tool management function to adjust the current position of a magazine and to improve the search strategy for subsequent commands.

Status = 105:
The specified buffer has been reached by all tools involved (standard case if the operation has not yet been completed). The tools are in the specified tool positions (parameters "NewToolMag", "NewToolLoc", "OldToolMag" ,"OldToolLoc").

Status definition

A general rule for the acknowledgment status is that the state information 1 to 7 result in ending of the command. If FC 8 receives one of the statuses, the "Interface active bit" of the interface specified in FC 8 is reset to "0" (see also interface lists DB71 to DB 73), thus completing the operation. The response if different in the case of status information 103 to 105. When the FC 8 receives one of these items of status information, the "Interface active bit" of this interface remains at "1". Further processing is required by the user program in the PLC (e.g. continuation of magazine positioning). This item of status information is generally used to transfer changes in position of one or both tools while the operation is still in progress.

© Siemens AG, 2003. All rights reserved

Call example

```
CALL FC 8(          //Tool management transfer block
     Start :=       m 20.5,        //Start := "1 "=> Initiation of transfer
     TaskIdent :=   DB61.DBB 0,
     TaskIdentNo := DB61.DBB 1,
     NewToolMag :=  DB61.DBW 2,    //Current position of new tool
     NewToolLoc :=  DB61.DBW 4,
     OldToolMag :=  DB61.DBW 6,          //Current position of old tool
     OldToolLoc :=  DB61.DBW 8,
     Status :=      DB61.DBW 10,         //Status
     Ready :=       f 20.6,
     Error :=       DB61.DBW 12);
a f 20.6;                          //Interrogate ready
r f 20.5;                          //Reset start
jc m001;                           //Jump if everything is OK
lDB61.dbw 12;                      //Error information
ow w#16#0;                         //Evaluate error
jn err;                            //Jump to error treatment

m001:                              //Normal branch

err:                               //Error treatment
r f 20.5:                          //Reset start
```

© Siemens AG, 2003. All rights reserved

4.16 FC 9: ASUB Start-up of asynchronous subprograms

Description of functions

The FC ASUB can be used to trigger any functions in the NC. Before an ASUB can be started from the PLC, it must have been selected and parameterized by an NC program or by FB 4 (PI service ASUB). Once prepared in this way, it can be started at any time from the PLC. The NC program running on the channel in question is interrupted by the asynchronous subprogram. Only one ASUB can be started in the same channel at a time. If the start parameter is set to logical 1 for two FC 9s within **one** PLC cycle, the ASUBs are started in the sequence in which they are called.

The start parameter must be set to logical 0 by the user when the ASUB has been terminated (Done) or if an error has occurred.

For the purpose of job processing, every FC ASUB required its own WORD parameter (Ref) from the global user memory area. This parameter is for internal use only and must not be changed by the user. Parameter **Ref** is initialized with the value 0 in the first OB1 cycle and for this reason, **every FC 9 must be called absolutely**. Alternatively, the user can initialize parameter *Ref* with a value of 0 during start-up. This option makes conditional calls possible. When a conditional call is activated by parameter *Start* = 1, it must remain active until parameter *Done* has made the transition from 1 to 0.

Note

The function is initiated only with the Low-High edge.

Alarms

400902	Parameter ChanNo in FC 9 is not permitted
Explanation	The parameterized channel does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

Declaration

```
FUNCTION FC 9: VOID                    //ASUP
VAR_INPUT
  Start:      BOOL;
  ChanNo:     INT;
  IntNo:      INT;
END_VAR
```

© Siemens AG, 2003. All rights reserved

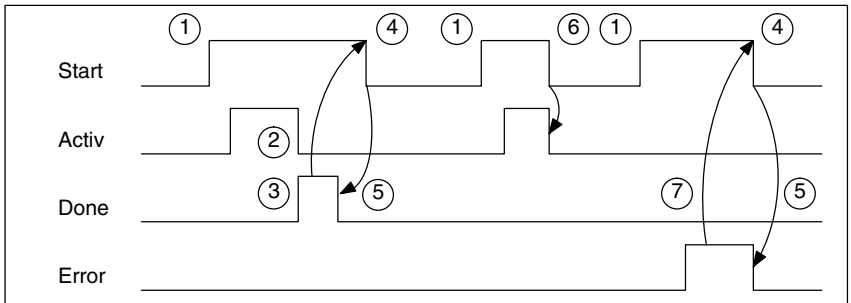
```
VAR_OUTPUT
   Active:      BOOL;
   Done:        BOOL;
   Error:       BOOL;
   StartErr:    BOOL;
END_VAR
VAR_IN_OUT
   Ref:         WORD;
END_VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the function ASUB.

Signal	Type	Type	Value range	Notes
Start	I	Bool		
ChanNo	I	Int	1 to 10	No. of NC channel
IntNo	I	Int	1 - 8	Interrupt no.
Activ	A	Bool		1 = active
Done	A	Bool		1 = ASUB terminated
Error	A	Bool		
StartErr	A	Bool		1 = Interrupt number not allocated
Ref	I/O	Word	Global variable (MW, DBW,..)	1 word per FC 9 (for internal use)

Pulse diagram



(1) Activation of function
(2) ASUB active
(3) Positive acknowledgment: ASUB ended
(4) Resetting of function activation after receipt of acknowledgment
(5) Signal change by FC
(6) If function initiation is reset before the acknowledgment is received, the output signals are not updated, without influence on the sequence of the initiation function
(7) Negative acknowledgment: Error has occurred

© Siemens AG, 2003. All rights reserved

Call example

```
CALL FC 9(          //Starting an asynchronous subprogram
                    //in channel 1 interrupt number 1
     Start :=       I 45.7,
     ChanNo :=      1,
     IntNo :=       1,
     Active :=      F 204.0,
     Done :=        F204.1,
     Error :=       F 204.4,
     StartErr :=    F 204.5,
     Ref :=         FW 200);
```

© Siemens AG, 2003. All rights reserved

4.17 FC 10: AL_MSG Error and operating messages

Description of functions

The FC AL_MSG evaluates the signals entered in DB 2 and displays them as incoming and going error and operational messages on the MMC. The incoming signals (positive edge) are displayed immediately in the case of both error and operating messages. Outgoing signals (negative edge) are only canceled immediately in the case of operating messages. Error messages remain stored on the MMC - even if the signals no longer exist - until the “acknowledge” parameter is issued, i.e. until the user acknowledges the messages. The “ToUserIF” parameter can be used to transfer the group signals for the feed, read and NC start disabling signals and feed stop signal to the existing axis, spindle and channel interfaces. The group signals are transferred to the user interface directly from the status information in DB 2 irrespective of an alarm acknowledgment.

1. If parameter “ToUserIF” is set to FALSE, signals are not transferred to the user interface. In this case, the user must take measures in his PLC program to ensure that these signals are influenced in the interface.
2. If parameter “ToUserIF” is set to TRUE, all signals listed above are sent to the user interface as a group signal in each case. The user PLC program can therefore influence these signals only via DB 2 in connection with a message or alarm output. The appropriate information is overwritten in the user interface.

As an alternative to the procedure described under paragraph 2, the user can influence the disable and stop signals without a message output by applying a disable or stop signal state to the interface signals after FC AL_MSG has been called.

The following program illustrates this method:

```
CALL FC 10 (
                ToUserIF:=          TRUE,
                Quit:=              I 6.1);
a f 50.0;           //Feed disable for channel 1
to db 21;
s DBX 6.0           //Set disable condition, reset via
                    //FC AL_MSG if F50.0 signal = “0”.
```

The error and operating messages are supplied by the user in the data block DB 2 (see description of DB2 Section 5.4).

Note

In DB 2, a “1” signal must be present for several OB1 cycles to ensure that a message can also be displayed on the MMC.

© Siemens AG, 2003. All rights reserved

Declaration of the function

STL representation

```
FUNCTION FC 10:  Void
//NAME :         AL_MSG
VAR_INPUT
   ToUserIF :    BOOL;
   Quit :        BOOL;
END_VAR
END_FUNCTION
```

Explanation of the formal parameters

The following table shows all formal parameters of the function AL_MSG.

Signal	Type	Type	Value range	Notes
ToUserIF	I	Bool		1 = Transfer signals to user interface every cycle.
Quit	I	Bool		1 = Acknowledge error messages.

Call example

```
CALL FC 10(                    //Error and operational messages
     ToUserIF:=    TRUE,       //Signals from DB2 are
                               //transferred to interface
     Quit :=       I6.1);      //Error message
                               //is acknowledged via
                               //input I6.1
```

© Siemens AG, 2003. All rights reserved

4.18 FC 12: AUXFU Call interface for user with auxiliary functions

Description of functions

FC AUXFU is generally called on an event-driven basis in the basic program if the channel transferred in the input parameter contains new auxiliary functions. The PLC user can extend FC AUXFU with program instructions for processing his auxiliary function to avoid cyclic polling of the channel DBs. This mechanism permits auxiliary functions to be processed on a job-controlled basis. FC AUXFU is supplied as a compiled empty block in the basic program. In this case, the basic program supplies parameter “Chan” with the channel number. The PLC user knows which channel has new auxiliary functions available. The new auxiliary functions can be determined by the auxiliary function change signals in the channel concerned.

Declaration

```
FUNCTION FC 12: VOID                //Event control of auxiliary functions
VAR_INPUT
  Chan:             BYTE;
END_VAR
BEGIN
  BE;
END_FUNCTION
```

Explanation of formal parameters

The following table shows all formal parameters of the AUXFU function.

Signal	Type	Type	Value range	Notes
Chan	I	Byte	0 to 9	No. of NC channel -1

Example

```
FUNCTION FC 12: VOID                //Event control of auxiliary functions

VAR_INPUT
  Chan:             BYTE; //Parameter is supplied by basic program
END_VAR
VAR_TEMP
  ChanDB:           INT;
END_VAR
BEGIN
L Chan;             //Channel no. (0,1,2,..)
+ 21;               //Channel DB offset
T ChanDB;           //Store channel DB no.
AUF DB[ChanDB]; //Channel DB is opened indirectly
// The change signals of the auxiliary functions are now scanned, etc.
  BE;
END_FUNCTION
```

© Siemens AG, 2003. All rights reserved

4.19 FC 13: BHGDisp Display control for handheld unit

Description of functions

This module carries out the display control of the handheld unit (HHU). The information to be output on the display is stored as 32 characters in string data ChrArray. A fixed text assignment of 32 characters is therefore required for this string when the data block is created. Variable components within this string can be inserted by means of the optional numerical converter, for which parameter Convert must be set to TRUE. The variable to be displayed is referenced via the pointer Addr. Parameter DataType contains the format description of this parameter (see parameter table). The number of bytes of the variable is linked to the format description. The address justified to the right within the string is specified by parameter StringAddr. The number of written characters is shown in the parameter table. By setting parameter Row to 0, it is possible to suppress the display (e.g. if several variables in one or several PLC cycles must be entered in the string without any display output).

Signals

Byte 1 is supplied by the output signals of the HHU and the character specifications are supplied by the module. These may not be written by the PLC user program.

Additional parameters

The pointer parameters for the input and output data of the handheld unit must be parameterized in the start OB 100 in FB 1, DB 7. Parameter BHGIn corresponds to the input data of the PLC from the handheld unit (data received by PLC). Parameter BHGOut corresponds to the output data of the PLC to the handheld unit (data transmitted by PLC). These two pointers must be set to the starting point of the relevant data area which is also parameterized in SDB 210 with an MPI link.

Note

If the numerical converter is used to display information, then it is better to avoid performing a conversion in every PLC cycle for the sake of reducing the PLC cycle time. In this case, it is advisable to use the input signal from the HHU to the PLC “Acknowledgment digital display” (DB m+5.7) for parameter “Convert”. In this way it can be ensured that the most recent numerical information is displayed.

© Siemens AG, 2003. All rights reserved

Declaration of the function

STL representation

```
DATA_BLOCK “strdat”                  //The data block number is
                                     //defined in the symbol file
  STRUCT
  disp:          STRING [32]:= ’Line 1 Line 2 ’; //32 characters are defined
  END_STRUCT;
BEGIN
END_DATA_BLOCK
FUNCTION FC 13: VOID
VAR INPUT
  Row :          Byte;          //Display line (see table)
  ChrArray :     STRING ;       //Transfer at least string[32]
  Convert :      BOOL ;         //Activate numerical conversion
  Addr :         POINTER;       //Points to variable to be converted
  DataType :     Byte ;         //Variable data type
  StringAddr :   INT ;          //Rightaligned string address (1...32)
  Digits :       BYTE ;         //Number of decimal places (1...3)
END VAR
VAR OUTPUT
  Error :        BOOL ;         //Conversion or string error
END VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the function BHGDisp.

Signal	Type	Type	Value range	Notes
Row	I	Byte	0 - 3	Display line 0: No display output 1: Line 1 2: Line 2 3; Line 1 and line 2
ChrArray	I	String	>= string[32]	This string contains the entire display contents
Convert	I	Bool		Activation of numerical conversion
Addr	I	Pointer		Points to variable to be converted
DataType	I	Byte	1 - 8	Data type of variable 1: Bool, 1 character 2: Byte, 3 characters 3: Char, 1 character 4: Word, 5 characters 5: Int, 6 characters 6: Dword, 7 characters 7: Dint, 8 characters 8: Real, 9 characters (see parameter Digits)
StringAddr	I	Int	1 - 32	Address within variable ChrArray
Digits	I	Byte	1 - 4	Relevant only for data type Real with sign 1: 6.1 digits without sign 2: 5.2 digits without sign 3: 4.3 digits without sign 4: 3.4 digits without sign
Error	O	Bool		Conversion error, numerical overflow or error in StringAddr

© Siemens AG, 2003. All rights reserved

Ranges of values

Table 4-29 Value ranges of data types

Data type	Representable numerical range
BOOL	0, 1
BYTE	0 to 255
WORD	0 to 65535
INT	-32768 to +32767
DWORD	0 to 9999999
DINT	-9999999 to +9999999
REAL (Digits := 1)	-999999.9 to +999999.9
REAL (Digits := 2)	-99999.99 to +99999.99
REAL (Digits := 3)	-9999.999 to +9999.999
REAL (Digits := 4)	-999.9999 to +999.9999

Call example

```
CALL FC 13 (
     Row :=          MB 26,
     ChrArray :=     “strdat”.disp,    //DB named strdat in the
                                       //symbol table, data element
                                       //disp is declared as string[32]
                                       //and assigned completely
                                       //with characters
     Convert :=      F 90.1,
     Addr :=         P#M 20.0,         //Number to be converted
     DataType :=     FB 28,            //Data type of variable
     StringAddr :=   FW 30,
     Digits :=       B#16#3,           //3 places after decimal point
     Error :=        F 90.2);
```

© Siemens AG, 2003. All rights reserved

4.20 FC 15: POS_AX Positioning of linear and rotary axes

(Do not use for new applications, function is integrated in FC 18 with SW 3.6 and higher).

Description of functions

The FC POS_AX can be used to traverse axes in any operating mode, also from the PLC. (See also Description of Functions **Positioning Axes SW 2**).

In order to traverse the NC axes via the PLC, the traversing check must be activated for the PLC.

This can be achieved, for example, by calling FC “POS_AX” with activation of the “Start” parameter.

FC “POS_AX” then requests an axis check by the NC.

The NC feeds back the status of this axis in byte 68 in the associated axis interface (DB 31, ...) (see interface lists).

Once the axis check has been completed (“InPos” is True, “Start” changes to zero), it is switched to a neutral state by FC POS AX.

Alternatively, the PLC user program can also request the check for the PLC prior to calling FC “POS_AX”.

By calling this function several times in succession, a better response reaction by the axes can be obtained as the switchover process in the FC can be omitted.

Activation through the PLC user program is executed in the corresponding axis interface in byte 8.

After return of the check, the axis can again be programmed by the NC program.

© Siemens AG, 2003. All rights reserved

Note

- Rotary axes can be positioned by the shortest possible route through the programming of a negative feed value in absolute programming mode. In incremental mode (parameter “IC” := TRUE), the traversing direction can be determined by the sign of parameter “Pos”:
  A positive sign causes axis to traverse in the + direction.
  A negative sign causes the axis to traverse in the - direction.
- After the FC has been called, ACCU1 contains an error statement by the NCK (but not if the output parameters are assigned to a data block). This is generally the value 0 (signifies: No error has occurred). The interpretation of other numerical values is shown in the following table.
- FC 15 must be called cyclically until the “Active” signal produces an edge transition from 1 to 0. Only when the “Active” signal has a 0 state can the axis concerned be started again (the next start must be delayed by at least one PLC cycle). This also applies when the assignment in data byte 8 has changed.
- The function cannot be aborted by means of parameter “Start” but only by means of the axial interface signals (e.g. delete distancetogo). The axial interface also returns status signals of the axis that may need to be evaluated (e.g. exact stop, traverse command).



Warning

If several block calls (FC 15, FC 16, FC 18) are programmed for the same axis/spindle in the PLC user program, then the functions concerned must be interlocked by conditional calls in the user program. The conditional call of a started block (parameter Start or Stop = TRUE) must be called cyclically until the signal state of output parameter Active or InPos changes from 1 to 0.

Alarms

401502	Parameter AxisNo in FC 15 is not permitted
Explanation	The parameterized axis does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

© Siemens AG, 2003. All rights reserved

Error identifiers

Table 4-30 Error identifiers

State	Meaning
Errors caused by PLC handling	
1	Several axis/spindle functions have been activated simultaneously
20	A function has been started without the position being reached
30	The axis/spindle has been transferred to the NC while still in motion
40	The axis is programmed in the NC program
Errors caused by NCK handling. The alarm numbers are described in the Diagnostic Guide for the 840D.	
100	Corresponds to alarm no.: 16830
105	Corresponds to alarm no.: 16770
106	Corresponds to alarm no.: 22052
107	Corresponds to alarm no.: 22051
108	Corresponds to alarm no.: 22050
109	Corresponds to alarm no.: 22055
110	Velocity/speed is negative
111	Setpoint speed is zero
112	Invalid gear stage
115	Programmed position has not been reached
117	G96/G961 is not active in the NC
118	G96/G961 is still active in the NC
120	Not an indexing axis
121	Indexing position error
125	DC (shortest path) not possible
126	Absolute minus value impossible
127	Absolute plus value impossible
130	Software limit switch plus
131	Software limit switch minus
132	Working area limitation plus
133	Working area limitation minus
System or other serious alarms	
200	Corresponds to system alarm no.: 450007

© Siemens AG, 2003. All rights reserved

Declaration

```
FUNCTION FC 15: VOID                    //POS_AX
VAR_INPUT
   Start :          BOOL;
   AxisNo:          INT;
   IC:              BOOL;
   Inch:            BOOL;
   HWheelOv:        BOOL;
   Pos:             REAL;
   FRate:           REAL;
END_VAR
VAR_OUTPUT
   InPos:           BOOL;
   Activ:           BOOL;
   StartErr:        BOOL;
   Error:           BOOL;
END_VAR
```

Explanation of the formal parameters

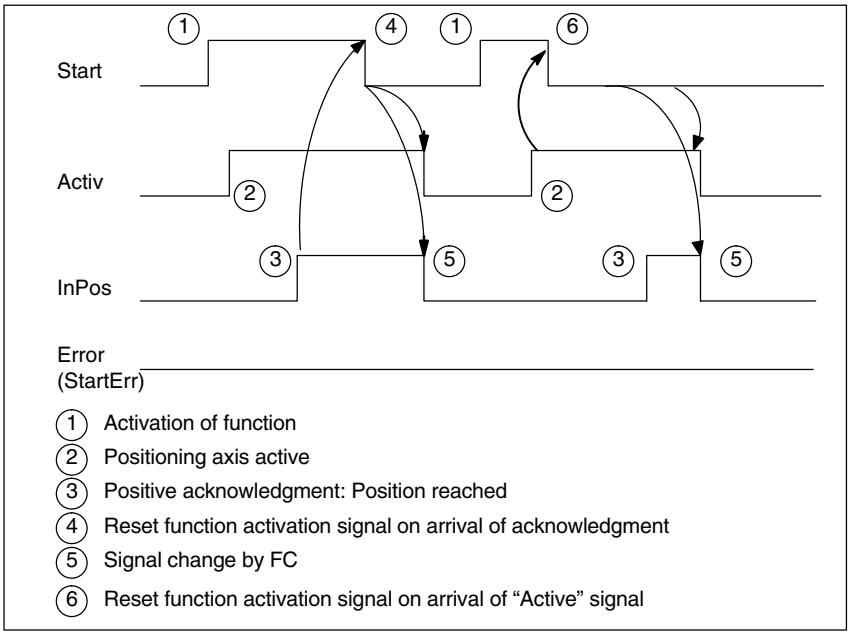
The following table shows all formal parameters of the function POS_AX.

Signal	Type	Type	Value range	Notes
Start	I	Bool		
AxisNo	I	Byte	1 - 31	No. of axis to be traversed
IC	I	Bool		0 = Absolute 1 = Incremental
Inch	I	Bool		0 = mm 1 = inch
HWheelOv	I	Bool		1 = Handwheel override
Pos	I	Real	± 0.1469368 E -38 bis ± 0.1701412 E +39	Position of Linear axis: mm Rotary axis: Degrees
FRate	I	Real	± 0.1469368 E -38 to ± 0.1701412 E +39	Feedrate of Linear axis: mm/min Rotary axis: rev/min
InPos	O	Bool		1 = In position
Activ	O	Bool		1 = Active
StartErr	O	Bool		Axis cannot be started, see Table 4-30 Error codes
Error	O	Bool		Traversing error [1] see Table 4-30 Error codes

[1] Error evaluation by user in the PLC

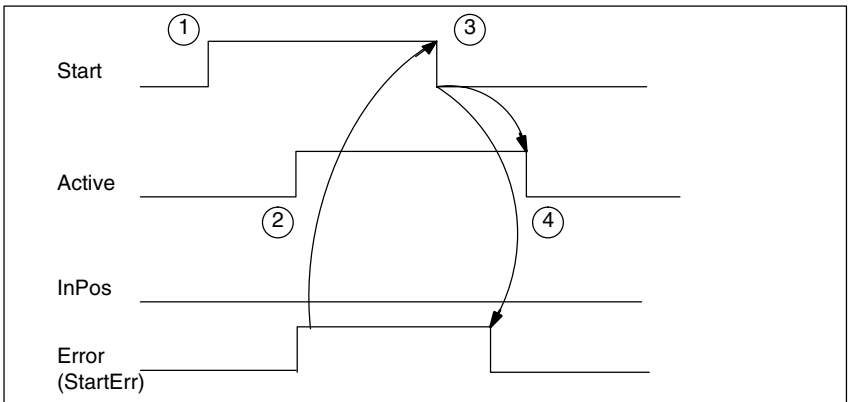
© Siemens AG, 2003. All rights reserved

Pulse diagram



(1) Activation of function
(2) Positioning axis active
(3) Positive acknowledgment: Position reached
(4) Reset function activation signal on arrival of acknowledgment
(5) Signal change by FC
(6) Reset function activation signal on arrival of “Active” signal

Pulse diagram (fault scenario)



(1) Activation of function by positive edge
(2) Negative acknowledgment: Error has occurred
(3) Reset function activation signal on arrival of acknowledgment
(4) Signal change by FC

Call example

```
CALL FC 15 (
     Start :=          F 100.0,
     AxisNo :=         5,
     IC :=             #incr,          //e.g. local variable
     Inch :=           FALSE,
     HWheelOv :=       FALSE,
     Pos :=            MD 160,
     FRate :=          MD 164,
     InPos :=          Q 36.0,
     Activ :=          Q 36.1,
     StartErr :=       Q 36.2,
     Error :=          Q 36.3);
```

© Siemens AG, 2003. All rights reserved

4.21 FC 16: PART_AX Positioning of indexing axes

(Do not use for new applications, function is integrated in FC 18 with SW 3.6 and higher).

Description of functions

The FC PART_AX can be used to traverse NC axes defined via machine data as “indexing axes” also from the PLC. (See also Description of Functions **Indexing Axes T1**.)

In order to traverse the indexing axes via the PLC, the traversing check for the PLC must be activated.

This can be achieved, for example, by calling FC “PART_AX” with activation of the “Start” parameter.

The FC “PART_AX” then requests checking of the axes from the NC.

The NC feeds back the status of this axis in byte 68 in the associated axis interface (DB 31, ...) (see interface lists).

On completion (“InPos” is True, “Start” changes to zero), the axis check function is switched to a neutral status by FC PART_AX.

Alternatively, the PLC user program can also request the check for the PLC prior to calling FC “PART_AX”.

By calling this function several times in succession, a better response reaction by the axes can be obtained as the switchover process in the FC can be omitted.

Activation through the PLC user program is executed in the corresponding axis interface in byte 8.

After return of the check, the axis can again be programmed by the NC program.

Note

After the FC has been called, ACCU1 contains an error statement by the NCK (but not if the output parameters are assigned to a data block). This is generally the value 0 (signifies: No error has occurred). The interpretation of other numerical values is shown in the table 4-30 of error identifiers in FC 15.

FC 16 must be called cyclically until the “Active” signal produces an edge transition from 1 to 0. Only when the “Active” signal has a 0 state can the axis concerned be started again (the next start must be delayed by at least one PLC cycle). This also applies when the assignment in data byte 8 has changed.

The function cannot be aborted by means of parameter “Start” but only by means of the axial interface signals (e.g. delete distance-to-go). The axial interface also returns status signals of the axis that may need to be evaluated (e.g. exact stop, traverse command).

© Siemens AG, 2003. All rights reserved



Warning

If several block calls (FC 15, FC 16, FC 18) are programmed for the same axis/spindle in the PLC user program, then the functions concerned must be interlocked by conditional calls in the user program. The conditional call of a started block (parameter Start or Stop = TRUE) must be called cyclically until the signal state of output parameter Active or InPos changes from 1 to 0.

Alarms

401602	Parameter AxisNo in FC 16 is not permitted
Explanation	The parameterized axis does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

Declaration

```
FUNCTION FC 16: VOID                    //PART_AX
VAR_INPUT
   Start :          BOOL;
   AxisNo:          INT;
   IC:              BOOL;
   DC:              BOOL;
   Minus:           BOOL;
   Plus:            BOOL;
   Pos:             INT;
   FRate:           REAL;
END_VAR
VAR_OUTPUT
   InPos:           BOOL;
   Activ:           BOOL;
   StartErr:        BOOL;
   Error:           BOOL;
END_VAR
```

Explanation of the formal parameters

The following table shows all formal parameters of the function PART_AX.

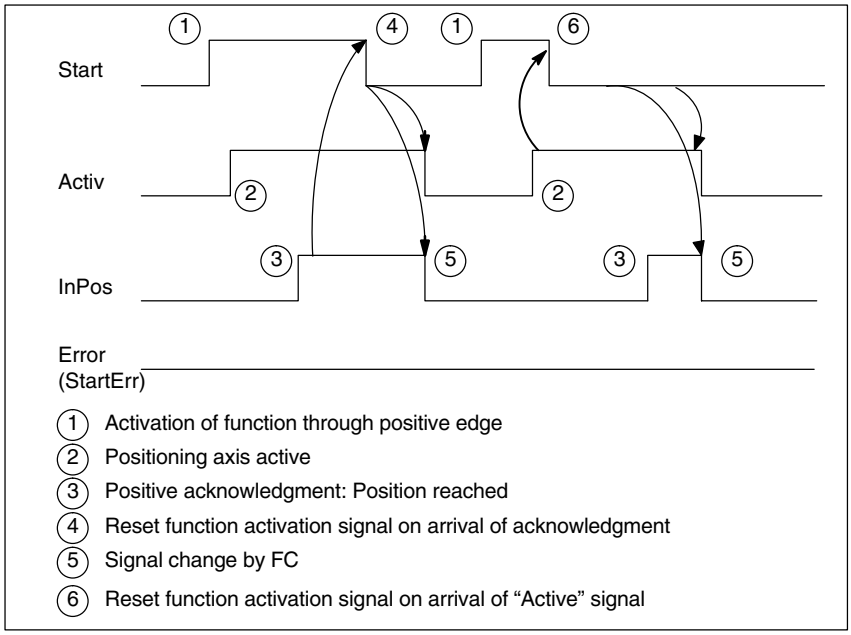
Signal	Type	Type	Value range	Notes
Start	I	Bool		
AxisNo	I	Int	1 - 31	No. of axis to be traversed
IC	I	Bool		0 = absolute direction input 1 = incremental direction input
DC	I	Bool		0 = defined direction 1 = shortest path

© Siemens AG, 2003. All rights reserved

Signal	Type	Type	Value range	Notes
Minus	I	Bool		0: Rotary axis motion as for linear axis 1: Motion in negative direction with rotary axes
Plus	I	Bool		0: Rotary axis motion as for linear axis 1: Motion in positive direction with rotary axes
Pos	I	Int	1 to 60	No. of indexing position
FRate	I	Real	± 0.1469368 E -38 to ± 0.1701412 E +39	Feedrate of Linear axis: mm/min Rotary axis: rev/min
InPos	A	Bool		1 = in position
Activ	A	Bool		1 = active
StartErr	A	Bool		Axis cannot be started, see Table 4-30 Error codes
Error	A	Bool		Traversing error [1] see Table 4-30 Error codes

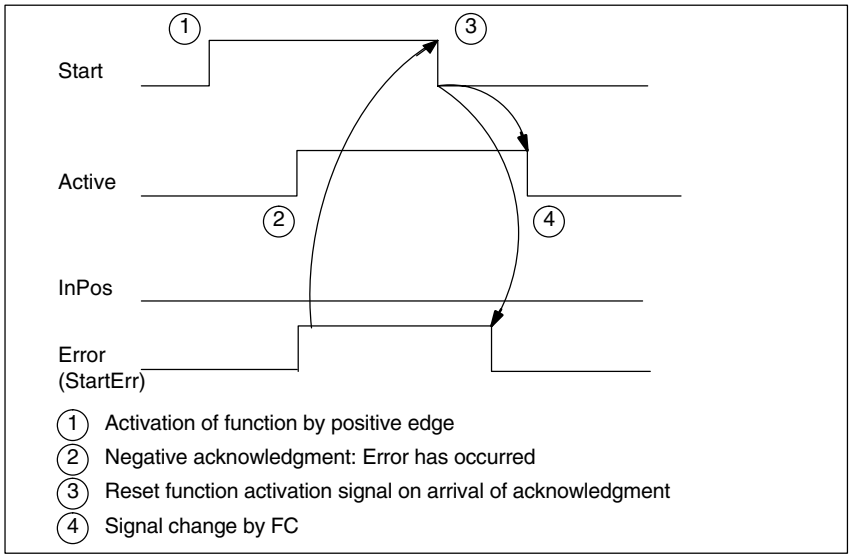
[1] Error evaluation by user in the PLC

Pulse diagram



© Siemens AG, 2003. All rights reserved

Pulse diagram (fault scenario)



(1) Activation of function by positive edge
(2) Negative acknowledgment: Error has occurred
(3) Reset function activation signal on arrival of acknowledgment
(4) Signal change by FC

Call example

```
CALL FC 16 (                    //Positioning of an indexing axis
     Start :=       I72.4,
     AxisNo :=      6,
     IC :=          FALSE,
     DC :=          #short,          //e.g. local variable
     Minus :=       FALSE,
     Plus :=        FALSE,
     Pos :=         FW 168,
     FRate :=       MD 164,
     InPos :=       Q 36.4,
     Activ :=       Q 36.5,
     StartErr :=    Q 36.6,
     Error :=       Q 36.7);
```

© Siemens AG, 2003. All rights reserved

4.22 FC 17: Y Delta Star/delta switchover

Description of functions

The data block for the star/delta switchover controls a defined switchover logic allowing switchover in both directions even with the spindle running. This block may be used only for digital main spindle drives and must be called separately for each spindle.

The switchover operation is implemented via 2 separate contactors in a sequence involving 4 steps:

Step 1: Deletion of interface signal “Motor selection in progress” in the associated axis DB (DB31, ... DBX21.5) and signaling of switchover process via “Motor Selection” A (DB31, ... DBX21.3).

Step 2: As soon as the checkback signal “Pulses enabled” = 0 (DB31, ... DBX93.7) and the acknowledgment of the announced motor selection from the drive have appeared, the currently energized contactor drops out.

Step 3: The other contactor is energized after the time period set by the user in parameter “TimeVal” has elapsed.

Step 4: After a further delay, the switchover is signaled to the drive by the message “Motor selection executed” (DB31, ... DBX21.5).

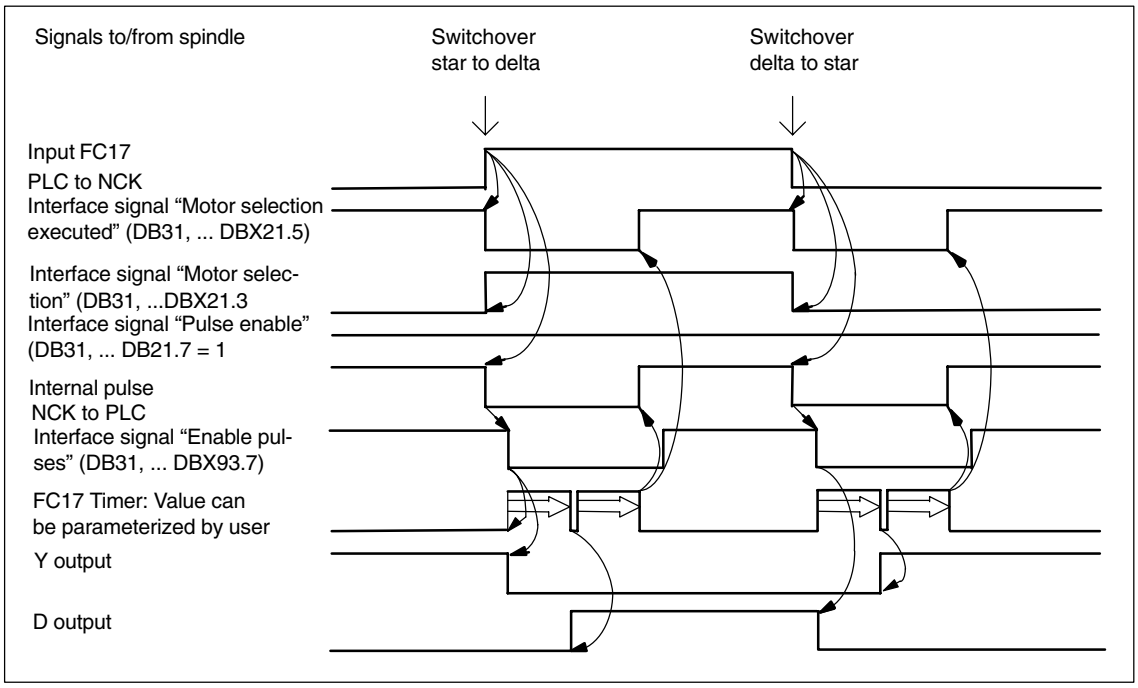


Fig. 4-1 Time sequence of interface signals with set delay in FC17 of 500 ms

For further explanations regarding motor speed adjustments, please see:

References: /FB1/, S1, “Spindles” Configurable Gear Adaption
/FB1/, G2, “Velocities, Setpoint/Actual-Value Systems, Closed-Loop Control

© Siemens AG, 2003. All rights reserved

Alarms

401702	Parameter ChanNo in FC 17 is not permitted
Explanation	The parameterized spindle does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

Error message

If the parameter “SpindleIFNo” is not in the permissible range, the PLC is stopped with output of alarm message number 401702.

Special points to be noted

When parameterizing the “TimeVal” with the value 0, a default value of 100 ms is used. With a value of less than 50 ms, the minimum setting of 50 ms is set.

The block must be called unconditionally.

Supplementary conditions

Using the star/delta switchover for main spindle drives, a process is triggered which also includes control engineering sequences. Since the control supports the automatic star/delta switchover, some supplementary conditions will have to be taken into account.

- Due to the automatic switch-off of pulses in the drive, interface signals “Enable pulses” (DB31, ... DBX93.7), “Current controller active” (DB31 ... DBX61.6) are simultaneously switched off.
- If interface signal “Position controller active” (DB31, ... DBX61.5) is switched over from star to delta while the spindle is rotating and the position controller of the spindle is active, alarm message 25050 “Contour monitoring” is displayed.
- A triggered star/delta switchover with FC17 cannot be delayed by the user, e.g. through operational waiting for a successful switchover of the star/delta contactors. This signal interplay can be implemented by the user via a PLC logic.

Declaration of the function

STL representation

```
VAR_INPUT
  YDelta:       BOOL;    //Star = 0, delta = 1
  SpindleIFNo:  INT;     //Machine axis number
  TimeVal:      S5TIME;  //Time value
  TimerNo:      INT;     //User timer for switchover time
END_VAR
VAR_OUTPUT
  Y:            BOOL;    //Star contactor
  Delta:        BOOL;    //Delta contactor
END_VAR
VAR_IN_OUT
  Ref:          WORD;    //Block status word (instance)
END_VAR
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the function YDelta.

Signal	Type	Type	Value range	Notes
YDelta	I	Bool		0 = Star 1 = Delta The switchover edge of the signal initiates the switchover operation.
SpindleIFNo	I	Int	1..	Number of spindle interface
TimeVal	I	S5time	0..	Switchover time
TimerNo	I	Int	10..	Timer for programming the wait time.
Y	A	Bool		Energizing of star contactor
Delta	A	Bool		Energizing of delta contactor
Ref	I/O	Word		Instance for status information. Internal use

Call example

```
CALL FC 17 (
     YDelta :=          i 45.7,          //Star delta
     SpindleIFNo :=     4,
     TimeVal :=         S5T#150ms,
     TimerNo :=         10,              //Timer 10
     Y :=               q 52.3,          //Star contactor
     Delta :=           q 52.4,          //Delta contactor
     Ref :=             mw 50);          //Instance
```

© Siemens AG, 2003. All rights reserved

4.23 FC 18: SpinCtrl Spindle control

Description of functions

The FC SpinCtrl can be used to control spindles and axes from the PLC.

References: /FB1/, S1, “Spindles”
/FB2/, P2, “Positioning axes”
/FB2/, T1, “Indexing Axes”

This block supports the following functions:

- Position spindle
- Rotate spindle
- Oscillate spindle
- Indexing axes
- Positioning axes

Each function is activated by the positive edge of the appropriate initiation signal (start, stop). This signal must remain in the logic “1” state until the function has been acknowledged positively or negatively by InPos=“1” or Error = “1”. The output parameters are deleted when the relevant trigger signal is reset and the function terminated.

To be able to control an axis or spindle via the PLC, it must be activated for the PLC. This can, for example, be achieved by calling the FC “SpinCtrl” with activation of the “Start” or “Stop” parameter. In this case, the FC “SpinCtrl” requests control over the spindle/axis from the NC.

The NC feeds back the status of this spindle/axis in byte 68 in the associated spindle/axis interface (DB 31, ...) (see interface lists). Once the axis/spindle is operating under PLC control, the travel command for the active state can be evaluated via the relevant axis interface.

On completion (“InPos” is True, “Start” changes to zero), the axis/spindle check function is switched to a neutral status by FC “SpinCtrl”.

Alternatively, the PLC user program can also request the check for the PLC prior to calling FC “SpinCtrl”.

By calling this function several times in succession, a better response reaction by the spindle/axis can be obtained as the switchover process in the FC can be omitted.

Activation through the PLC user program is executed in the corresponding spindle interface in byte 8.

After return of the check, the spindle can again be programmed by the NC program.

© Siemens AG, 2003. All rights reserved

Note

Note about call:
FC 18 must be called cyclically until signal “InPos” or, in the case of an error “Error” produces an edge transition of 1 to 0. Only when the “InPos“/“Error” signal has a 0 state can the axis/spindle concerned be “started” or “stopped” again (the next “start” must be delayed by at least one PLC cycle). This also applies when the assignment in data byte 8 on the axial interface has been changed.

Abort:
The function cannot be aborted by means of parameter “Start” or “Stop” but only by means of the axial interface signals (e.g. delete distance-to-go). The axial interface also returns status signals of the axis that may need to be evaluated (e.g. exact stop, traverse command).

InPos for rotate/oscillate spindle:
The meaning of parameter “InPos” is defined as follows for the functions “Rotate spindle” and “Oscillate spindle”
Set speed is output --> function has started without an error.
Spindle acceleration up to the required spindle speed must be evaluated via the spindle interface.

Simultaneity:
Several axes can be traversed simultaneously or in a staggered time sequence by blocks FC 15, 16 and 18. The upper limit is defined by the maximum number of axes. The NCK handles the PLC function request (FC 15, 16, 18) via independent interfaces for each axis/spindle.

Axis disable:
With the axis disabled, an axis controlled via FC18 will not move. Only a simulated actual value is generated. (Behavior as with NC programming).



Warning

If several block calls (FC 15, FC 16, FC 18) are programmed for the same axis/spindle in the PLC user program, then the functions concerned must be interlocked by conditional calls in the user program. The conditional call of a started block (parameter Start or Stop = TRUE) must be called cyclically until the signal state of output parameter Active or InPos changes from 1 to 0.

Alarms

401802	Parameter AxisNo in FC 18 is not permitted
Explanation	The parameterized axis does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

© Siemens AG, 2003. All rights reserved

Functions

1. Position spindle
The following signals are relevant:

Start:	Initiation signal
Funct:	"1" = Position spindle
Mode:	Positioning mode 1, 2, 3, 4
AxisNo:	Number of machine axis
Pos:	Position
FRate:	Positioning velocity. When FRate = 0, value from MD 35300: SPIND_POSCTRL_VELO (position control activation speed) is used
InPos:	Is set to "1" when position is reached with "Exact stop fine"
Error:	With positioning error = "1"
State:	Error code

2. Rotate spindle:
The following signals are relevant:

Start:	Initiation signal for start rotation
Stop:	Initiation signal for stop rotation
Funct:	"2" = Rotate spindle
Mode:	Positioning mode 5 (rotational direction M4) Positioning mode <> 5 (rotational direction M3)
AxisNo:	Number of machine axis
FRate:	Spindle speed
InPos:	Function has started without an error
Error:	With positioning error = "1"
State:	Error code

3.Oscillate spindle:
The following signals are relevant:

Start:	Initiation signal for start oscillation
Stop:	Initiation signal for stop oscillation
Funct:	"3" = Oscillate spindle
AxisNo:	Number of machine axis
Pos:	Set gear step
InPos:	Error: With positioning error = "1"
State:	Error code

The oscillation speed is taken from machine data SPIND_OSCILL_DES_VELO.

MD 35010: GEAR_STEP_CHANGE_ ENABLE = 0	Function	MD 35010: GEAR_STEP_CHANGE_ ENABLE = 1	Function
Pos = 0	Oscillation	Pos = 0	
Pos = 1	Oscillation	Pos = 1	Oscillation with gear stage change M41
Pos = 2	Oscillation	Pos = 2	Oscillation with gear stage change M42
Pos = 3	Oscillation	Pos = 3	Oscillation with gear stage change M43
Pos = 4	Oscillation	Pos = 4	Oscillation with gear stage change M44
Pos = 5	Oscillation	Pos = 5	Oscillation with gear stage change M45

© Siemens AG, 2003. All rights reserved

4. Traverse indexing axes:

The following signals are relevant:

Start:	Initiation signal
funct:	“4” = Indexing axis

Note

When
Funct: “4” = Indexing axis
The modulo conversion can be compared with approaching the indexing position via POS[AX] = CIC (value) in the parts program.

Mode:	Positioning mode 0, 1, 2, 3, 4
AxisNo:	Number of machine axis
Pos:	Indexing position
FRate:	Positioning speed; if FRate = 0, the value is taken from machine data POS_AX_VELO (unit as set in machine data)
InPos:	Is set to “1” when position is reached with “Exact stop fine”
Error:	With positioning error = “1”
State:	Error code

5. to 8. Position axes:

The following signals are relevant:

Start:	Initiation signal
Funct:	“5 to 8” = Position axes
Mode:	Positioning mode 0, 1, 2, 3, 4
AxisNo:	Number of machine axis
Pos:	Position
FRate:	Positioning speed; if FRate = 0, the value is taken from machine data POS_AX_VELO (unit as set in machine data)
InPos:	Is set to “1” when position is reached with “Exact stop fine”
Error:	With positioning error = “1”
State:	Error code

© Siemens AG, 2003. All rights reserved

9. Rotate spindle with automatic gear stage selection:

The following signals are relevant:

Start:	Initiation signal for start rotation
Stop:	Initiation signal for stop rotation
Funct:	“9” = Rotate spindle with gear stage selection
Mode:	Positioning mode 5 (rotational direction M4) Positioning mode < >5 (rotational direction M3)
AxisNo:	Number of machine axis
FRate:	Spindle speed
InPos:	Setpoint speed is output
Error:	With positioning error = “1”
State:	Error code

10./11. Rotate spindle with constant cutting rate:
The “constant cutting rate” function must first be activated by the NC program

The following signals are relevant:

Start:	Initiation signal for start rotation
Stop:	Initiation signal for stop rotation
Funct:	”B#16#0A = Rotate spindle with constant tabcutting rate (m/min)
Funct:	”B#16#0B = Rotate spindle with constant cutting rate (feet/min)
Mode:	Positioning mode 5 (rotational direction M4) Positioning mode <>5 (rotational direction M3)
AxisNo::	Number of machine axis
FRate:	Cutting rate
InPos:	Setpoint speed is output
Error:	With position error = “1”
State:	Error code

```
FUNCTION FC 18: VOID                    //SpinCtrl
VAR_INPUT
   Start:      BOOL;
   Stop:       BOOL;
   Funct:      BYTE;
   Mode:       BYTE;
   AxisNo:     INT;
   Pos:        REAL;
   FRate:      REAL;
END_VAR
VAR_OUTPUT
   InPos:      BOOL;
   Error:      BOOL;
   State:      BYTE;
END_VAR
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the function SpinCtrl.

Signal	Type	Type	Value range	Notes
Start	I	Bool		Start spindle control from PLC
Stop	I	Bool		Stop spindle control from PLC
Funct	I	Byte	1 to B#16#0B	1: Position spindle 2: Rotate spindle 3: Oscillate spindle 4: Indexing axis 5: PosAxis metric 6: PosAxis inch 7: PosAxis metric with handwheel override 8: PosAxis inch with handwheel override 9: Rotate spindle with gear stage selection A: Rotate spindle with constant cutting rate (m/min) B: Rotate spindle with constant cutting rate (feet/min)
Mode	I	Byte	0 to 5	0: Pos to absolute pos 1: Pos incrementally 2: Pos shortest path 3: Pos absolute, positive approach direction 4: Pos absolute, negative approach direction 5: Rotational direction as for M4
AxisNo	I	Int	1 - 31	No. of axis/spindle to be traversed
Pos	I	Real	± 0.1469368 E -38 to ± 0.1701412 E +39	Rotary axis: Degrees Indexing axis: Indexing position Linear axis: mm or inches
FRate	I	Real	± 0.1469368 E -38 bis ± 0.1701412 E +39	Rotary axis and spindle: rev/min See under table containing info about FRate
InPos	O	Bool		1 = position reached or function executed
Error	O	Bool		1 = error
State	O	Byte	0 to 255	Error code

FRate

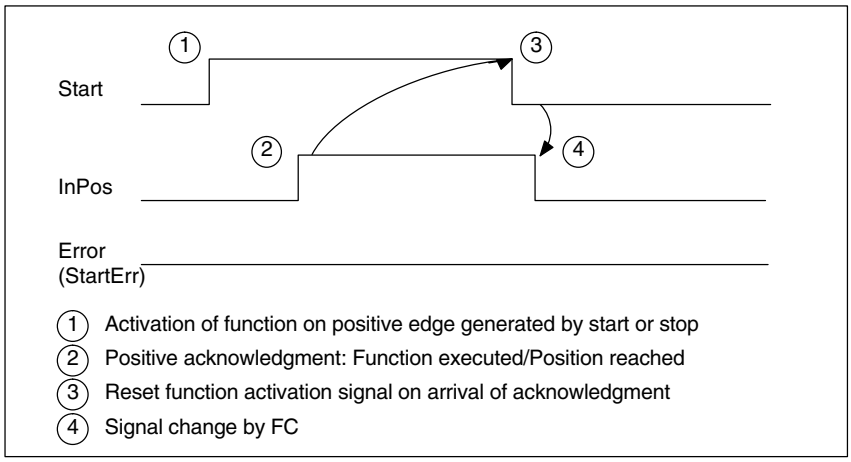
With SW 4 and higher, the feed velocity can also be specified in FC 18 as
1. Cutting velocity in the unit m/min or ft/min.
2. Constant grinding wheel surface speed in m/s or ft/s. These alternative velocity settings can be used only if this function is activated by the NC program. Checkback signals for successful activation can be found in byte 84 on the axis interface.

Error identifiers

If a function could not be carried out, this is indicated by the Error status parameter with ‘logically 1’. The error cause is coded at block output State. For a list of the error identifiers, see Table 4-30 in the description of FC 15 (see above).

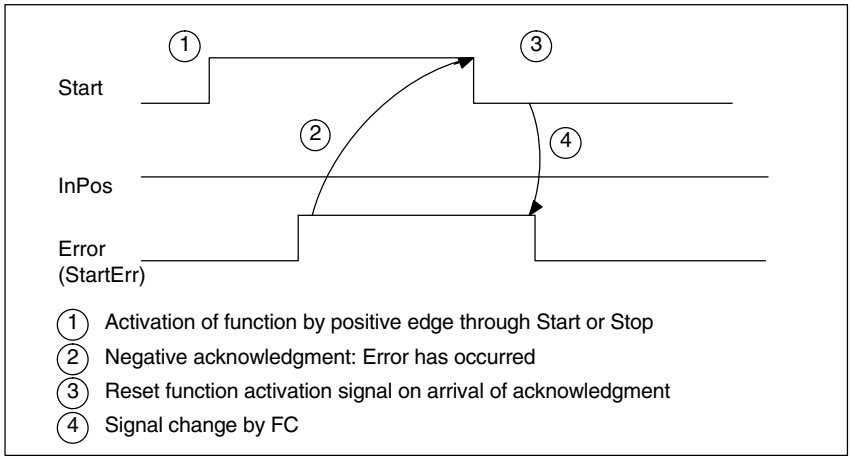
© Siemens AG, 2003. All rights reserved

Pulse diagram



1 Activation of function on positive edge generated by start or stop
2 Positive acknowledgment: Function executed/Position reached
3 Reset function activation signal on arrival of acknowledgment
4 Signal change by FC

Pulse diagram (fault scenario)



1 Activation of function by positive edge through Start or Stop
2 Negative acknowledgment: Error has occurred
3 Reset function activation signal on arrival of acknowledgment
4 Signal change by FC

© Siemens AG, 2003. All rights reserved

Call examples

1. Position spindle

```
//Positive acknowledgment resets Start:
A M112.0;          //InPos
R M 100.0;         //Start
//Negative acknowledgment, after error evaluation (state: MB114) reset with T12 start
A M113.0;          //Error
A I 6.4;           //Key T12
R M 100.0;         //Start
//Start with T13
A I 6.3;           //Key T13
AN F 112.0;        //New start only when InPos or Error = 0
AN F 113.0;
S M 100..0;
  CALL FC 18 (
              Start :=    F100.0,
              Stop :=     FALSE,
              Funct :=    B#16#1,     //Position spindle
              Mode :=     B#16#2,     //Shortest path
              AxisNo :=   5,
              Pos :=      MD104,
              FRate :=    MD108,
              InPos :=    F112.0,
              Error :=    F113.0,
              State :=    FY114);
```

2. Start spindle rotation:

```
  CALL FC 18 (
              Start :=    F100.0,
              Stop :=     FALSE,
              Funct :=    B#16#2,     //Rotate spindle
              Mode :=     B#16#5,     //Rotational direction as for M4
              AxisNo :=   5,
              Pos :=      0.0,
              FRate :=    MD108,
              InPos :=    F112.0,
              Error :=    F113.0,
              State :=    FY114);
```

© Siemens AG, 2003. All rights reserved

3. Start spindle oscillation

```
CALL FC 18 (
     Start :=    F100.0,
     Stop :=     FALSE,
     Funct :=    B#16#3,     //Oscillate spindle
     Mode :=     B#16#0,
     AxisNo :=   5,
     Pos :=      0.0,
     FRate :=    MD108,
     InPos :=    F112.0,
     Error :=    F113.0,
     State :=    MB114);
```

4. Traverse indexing axis

```
CALL FC 18 (
     Start :=    M100.0,
     Stop :=     FALSE,      //Not used
     Funct :=    B#16#4,     //Traverse indexing axis
     Mode :=     B#16#0,     //Position absolutely
     AxisNo :=   4,
     Pos :=      MD104,      //Specification in REAL: 1.0;2.0;..
     FRate :=    MD108,
     InPos :=    F112.0,
     Error :=    F113.0,
     State :=    FY114);
```

5. Position axes

```
CALL FC 18 (
     Start :=    M100.0,
     Stop :=     FALSE,      //Not used
     Funct :=    B#16#5,     //Position axes
     Mode :=     B#16#1,     //Position incrementally
     AxisNo :=   6,
     Pos :=      MD104,
     FRate :=    MD108,
     InPos :=    F112.0,
     Error :=    F113.0,
     State :=    FY114);
```

© Siemens AG, 2003. All rights reserved

4.24 FC 19: MCP_IFM Transmission of MCP signals to interface

Description of functions

The FC MCP_IFM (M variant) is used to transfer

- Mode groups
- Axis selections
- WCS/MCS switchover commands
- Traversing keys
- Overrides
- Keyswitch

from the machine control panel (MCP) to the corresponding signals of the NCK/PLC interface. In the basic program (FC 2), the FC 27 transmits handwheel selections, modes and other operation signals from the operator panel or MMC to the NCK/PLC interface in such a way as to allow the modes to be selected from the machine control panel or the operator panel.

Transfer of MMC signals to the interface can be deactivated by setting the value of the parameter “MMCToIF” to “FALSE” in FB 1 (DB 7).

The following specifications apply to the **feed override, axis travel keys** and **INC keys** depending on the active mode or on the coordinate system selected:

- **Feed override***:*
  - The feed override is transferred to the interface of the selected channel and to the interface of the axes.
  - The feed override signals are transferred in addition to interface byte “Rapid traverse” override (DBB 5) to the NC channel if MMC signal “Feed override for rapid traverse effective” is set (Exception: Switch setting “Zero”). “Rapid traverse override effective” is also set with this MMC signal.
- **Machine functions for INC and axis travel keys:**
  - When the MCS is selected, the signals are transferred to the interface of the selected machine axis.
  - When the WCS is selected, the signals are transferred to the geoaxis interface of the parameterized channel.
  - When the system is switched between MCS and WCS, the active axes are generally deselected.

  SW version-dependent response: See next page.

The **handwheel selection signals from the MMC** are decoded and activated in the machine axis or the Geo axis interface of the handwheel selected (only if parameter “HWheelMMC := TRUE” in FB1).

The LEDs on the machine control panel derived from the selections in the acknowledgment.

Feedrate and spindle Start/Stop are not transferred to the interface, but output modally as a “FeedHold” or “SpindleHold” signal. The user can link these signals to other signals leading to a feed or spindle stop (this can be implemented, e.g., using the appropriate input signals in FC 10: AL_MSG). The associated LEDs are activated at the same time.

© Siemens AG, 2003. All rights reserved

If the MCP fails, the signals which it outputs are preset to zero; this also applies to “FeedHold” and “SpindleHold” output signals.

With **SW 4 and higher**, multiple calls of FC 25 or FC 25 are permitted within the same PLC cycle. In this case, the first call in the cycle drives the LED displays. Furthermore, all actions of the parameterized block are carried out in the first call. In the following calls, only a reduced level of processing of the channel and mode group interface takes place. The geometry axes are supplied with directional data only in the first block call in the cycle.

Single block processing can be selected/deselected only in the first call in the cycle.

The second machine control panel can be processed if parameter ModeGroupNo has been increased by B#16#10. The mode group number is contained in the lower nibble (lower 4 bits) when the parameter is set.
ModeGroupNo = 0 or B#16#10 means that mode group signals are not processed.
ChanNo = 0 means that the channel signals are not processed.

With **SW 5 and higher**, the **INC selections** are only transferred to the mode group interface. This results in runtime improvements. The activation for this command is performed by this block once after powerup via DB10.DBX 57.0. Machine control panels can still be handled in parallel by this module. The module 2 call for the 2nd machine control panel in OB1 cycle must come after the call of the 1st MCP. A support for two MCPs is still provided within certain restrictions in the control panel blocks (support is not provided for standard axis numbers 10 to 31, mutual interlocking of axis selections with two MCPs).

Flexible axis configuration

SW 6 and higher provide flexibility as regards the assignment of axis selections or direction keys to machine- axis numbers.
The use of 2 machine control panels in simultaneous operation is now better supported by the MCP blocks, in particular the application 2 channels, 2 mode groups. Note that the axis numbers are also specified in the parameterized mode group number of the MCP-block in the axis tables of the relevant MCP.

To afford this flexibility, tables for axis numbers are stored in DB 10. The table starts from byte 8 (symbolic name: MCP1AxisTbl[1..22]) for the first machine control panel (MCP) and from byte 32 (symbolic name: MCP2AxisTbl[1..22]) for the second MCP. The machine axis numbers must be entered here byte by byte. It is permissible to enter 0 in the axis table. No check is made for illegal axis numbers. Incorrect entries can cause the PLC to stop.
A limit on the maximum possible number of selected axes can be set for FC 19. This limit is set for the appropriate MCP in DB10.DBW30 (symbolic name: MCP1MaxAxis) or DB10.DBW54 (symbolic name: MCP2MaxAxis). The default setting is 0, corresponding to the maximum number of configured axes. The axis numbers and the limit can also be adapted dynamically. This must be followed, however, by another axis selected in FC 19.
The axis numbers must not be switched over whiles axes are being moved via the direction keys.
Set as a default is the compatibility mode with axis numbers 1 to 9 for both MCPs and the limit at the configured number of axes.

© Siemens AG, 2003. All rights reserved

Alarms

401901, 401902	Parameter ModeGroupNo in FC 19, parameter ChanNo in FC 19 not permitted
Explanation	The parameterized BAG group, channel does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

Declaration of the function

```
FUNCTION FC 19: void
//NAME              : MCP_IFM
VAR_INPUT
   ModeGroupNo :    BYTE;
   ChanNo :         BYTE;
   SpindleIFNo :    BYTE;
END_VAR
VAR_OUTPUT
   FeedHold :       BOOL;
   SpindleHold :    BOOL;
END_VAR
BEGIN
END_FUNCTION
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the function MCP_IFM.

Signal	Type	Type	Value range	Notes
BAGNo	I	Byte	0 - b#16# and b#16#10 -b#16#1A	Mode group no. to which operating mode signals must be transferred ModeGroupNo >= b#16#10 means access to the 2nd machine control panel.
ChanNo	I	Byte	0 - B#16#0A	Channel no. for the channel signals
SpindleIFNo	I	Byte	0 - 31 (B#16#1F)	Number of the axis interface declared as a spindle.
FeedHold	O	Bool		Feed stop from machine control panel, modal
SpindleHold	O	Bool		Spindle stop from machine control panel, modal

MCP selection signals to the user interface

Keyswitch

Source: MCP switch	Target: Interface DB
Position 0	DB10.DBX56.4
Position 1	DB10.DBX56.5
Position 2	DB10.DBX56.6
Position 3	DB10.DBX56.7

Operating modes and machine functions

Source: MCP key	Target: **Interface DB** (Parameter ChanNo)
AUTOMATIC	DB11, ... DBX0.0
MDA (MDI)	DB11, ... DBX0.1
JOG	DB11, ... DBX0.2
REPOS	DB11, ... DBX1.1
REF	DB11, ... DBX1.2
TEACH IN	DB11, ... DBX1.0
INC 1 ... 10 000, INC Var. **(SW 5 and higher)**	DB11.DBB2, Bits 0 to 5

SW 4 and lower

Source: MCP key	Target: **Interface DB** (Parameter ChanNo)
INC 1 ... 10 000, INC Var.	DB21, ... DBB13, Bit 0 to 5
INC 1 ... 10 000, INC Var.	DB21, ... DBB17, Bit 0 to 5
INC 1 ... 10 000, INC Var.	DB21, ... DBB21, Bit 0 to 5

© Siemens AG, 2003. All rights reserved

Source: MCP key	Target: **Interface DB** (All axis DBs)
INC 1 ... 10 000, INC Var. **(SW 4 and lower)**	DB 31, ... DBB5, Bit 0 to 5

Direction keys rapid traverse override

The transfer is dependent upon the selected axis. The associated interface bits are deleted for axes which are not selected.

Source: MCP key	Target: **Interface DB** (Parameter ChanNo)
Direction key +	DB21, ... DBX12.7
Direction key -	DB21, ... DBX12.6
Rapid traverse override	DB21, ... DBX12.5
Direction key +	DB21, ... DBX16.7
Direction key -	DB21, ... DBX16.6
Rapid traverse override	DB21, ... DBX16.5
Direction key +	DB21, ... DBX20.7
Direction key -	DB21, ... DBX20.6
Rapid traverse override	DB21, ... DBX20.5

Source: MCP key	Target: **Interface DB** (All axis DBs)
Direction key +	DB 31, ... DBX4.7
Direction key -	DB 31, ... DBX4.6
Rapid traverse override	DB 31, ... DBX4.5

Override

Source: MCP switch	Target: **Interface DB** (Parameter ChanNo)
Feedrate override	DB21, ... DBB4

Source: MCP switch	Target: **Interface DB** (All axis DBs)
Feedrate override	DB 31, ... DBB0 (selected axis number). The feed override of the 1st MCP applies to all axes.
Spindle override	DB 31, ... DBB19 (Parameter SpindleIFNo)

© Siemens AG, 2003. All rights reserved

Channel signals

Source: MCP key	Target: Interface DB (Parameter ChanNo)
NC Start	DB21, ... DBX7.1
NC Stop	DB21, ... DBX7.3
Reset	DB21, ... DBX7.7
Block	DB21, ... DBX0.4

Feedrate, spindle signals

Source: MCP key	Target: FC output parameters
Feed stop Feed enable	Parameter: “FeedHold” latched, LEDs are driven
Spindle stop Spindle enable	Parameter: “SpindleHold” latched, LEDs are driven

Checkback messages of the user interface for controlling displays

Operating modes and machine functions

Target: MCP LED	Source: Interface DB (Parameter ModeGroupNo)
AUTOMATIC	DB11, ... DBX6.0
MDA (MDI)	DB11, ... DBX6.1
JOG	DB11, ... DBX6.2
REPOS	DB11, ... DBX7.1
REF	DB11, ... DBX7.2
TEACH IN	DB11, ... DBX7.0

All signals are ORed to obtain the LED checkback signals of the INC selections.

Target: MCP LED	Source: Interface DB (Parameter ChanNo)
INC 1 ... 10 000, INC Var.	DB21, ... DBB41, Bit 0 to 5
INC 1 ... 10 000, INC Var.	DB21, ... DBB47, Bit 0 to 5
INC 1 ... 10 000, INC Var.	DB21, ... DBB53, Bit 0 to 5

© Siemens AG, 2003. All rights reserved

Target: MCP LED	Source: **Interface DB** (All axis DBs)
INC 1 ... 10 000, INC Var.	DB 31, ... DBB65, Bit 0 to 5

Channel signals

Target: MCP LED	Source: **Interface DB** (Parameter ChanNo)
NC Start	DB21, ... DBX35.0
NC Stop	DB21, ... DBX35.2 or DB21, ... DBX35.3
Block	DB21, ... DBX0.4

Note

LEDs on the direction keys are driven through actuation of the direction keys. LEDs for axis selection and WCS/MCS are driven through actuation of the relevant key.

Call example

```
CALL FC 19(       //Machine control panel M variant
                  //Signals to interface
   ModeGroupNo :=  B#16#1,        //Mode group no. 1
   ChanNo :=       B#16#1,        //Channel no. 1
   SpindleIFNo :=  B#16#4,        //Spindle interface
                                  //Number = 4
   FeedHold :=     f22.0,         //Feed stop signal
                                  //Modal
   SpindleHold :=  db2.dbx151.0); //Spindle stop modal in
                                  //Message data block
```

With these parameter settings, the signals are sent to the 1st mode group, the 1st channel and all axes. In addition the spindle override is transferred to the 4th axis/spindle interface. The feed hold signal is passed to bit memory 22.0 and the spindle stop signal to data block DB2, data bit 151.0.

Reconnecting the axis selections

To ensure a flexible assignment of the axis selection keys to the appropriate axis or spindle, FC 19 **need not be modified or reprogrammed**. The required flexibility can be obtained by applying the solution described below.

1. Before FC 19 is called, the information (RLO) relating to the newly defined axis selection key is transferred to the key selection identified by an axis number.
2. After FC 19 has been called, the information (RLO) relating to the LED identified by an axis number is transferred to the LED of the new axis selection key and the RLO of the previous axis LED then deleted.

© Siemens AG, 2003. All rights reserved

Example:
The spindle is defined as the 4th axis and must be selected via axis key 9.

STL extract:

```
a i 5.2;              //Selection 9th axis
= i 4.2;              //to selection 4th axis

   call fc 19(
                      ModeGroupNo :=   b#16#1,
                      ChanNo :=        b#16#1,
                      SpindleIFNo :=   b#16#4,

                      FeedHold :=      m 30.0,
                      SpindleHold :=   m 30.1);

a q 2.5;              //LED 4th axis
= q 3.3;              //LED 9th axis
clr;
= q 2.5;              //Switch off LED 4th axis
```

© Siemens AG, 2003. All rights reserved

4.25 FC 21: Transfer PLC NCK data exchange

Description of functions

When the Transfer block is called, data are exchanged between the PLC and NCK according to the selected function code. The data are transferred immediately that FC 21 is called, not just at the beginning of the cycle.

The block is activated by means of the “Enable” signal. FC 21 is processed only when “Enable” = “1”.

Declaration of function, STL representation

```
VAR_INPUT
  Enable :    BOOL ;
  Funct :     BYTE ;
  S7Var :     ANY ;
  IVar1 :     INT ;
  IVar2 :     INT ;
END_VAR
VAR_OUTPUT
  Error :     BOOL ;
  ErrCode :   INT
END_VAR
```

Explanation of formal parameters

The following table shows all formal parameters of the Transfer function.

Signal	Type	Type	Value range	Notes
Enable	I	Bool		1 = FC 21 active
Funct	I	Byte	1 .. 7	1: Synchronized actions to channel 2: Synchronized actions from channel 3: Read data 4: Write data 5: Control signals to channel 6, 7: Control signals to axis
S7Var	I	Any	S7 data storage area	Depends on “Funct”
IVAR1	I	Int	0 ..	Depends on “Funct”
IVAR2	I	Int	1 ..	Depends on “Funct”
Error	O	Bool		
ErrCode	O	Int		Depends on “Funct”

Functions

1: Signals synchronized actions to channel:
2: Signals synchronized actions from channel:

Synchronized actions can be disabled or enabled by the PLC.
The data area is stored on the user interface in DB21 to DB30.DBB 300..307 (to channel) and DBB 308..315 (from channel). Parameter “S7Var” is not evaluated for this function, but must be assigned an actual parameter (see call example).
The data are transferred to/from the NC as soon as FC 21 is processed.

© Siemens AG, 2003. All rights reserved

The following signals are relevant:

Signal	Type	Type	Value range	Notes
Enable	I	Bool		1 = FC 21 active
Funct	I	Byte	1, 2	1: Synchronized actions to channel 2: Synchronized actions from channel
S7Var	I	Any	S7 data storage area	Not used
IVAR1	I	Int	1..MaxChannel	Channel number
Error	O	Bool		
ErrCode	O	Int		1: “Funct” invalid 10: Channel no. invalid

Call example

```
FUNCTION FC 100: VOID
VAR_TEMP
  myAny:          ANY;
END_VAR
BEGIN
NETWORK
...
// Deactivate synchronized actions with ID3, ID10 and ID31 in NC channel 1 :
SYAK:           AUF DB21
  SET;
  S DBX300.2;   //ID3
  S DBX301.1;   //ID10
  S DBX303.6;   //ID31
  L B#16#1;
  T MB11;
  SPA TRAN;

// Synchronized actions from NCK channel:
SYVK:           L B#16#2;
                T MB11;
TRAN: CALL FC 21(
                Enable :=       F 10.0,         // If True, FC21 active
                Funct :=        MB 11,
                S7Var :=        #myAny,         //Not used
                IVAR1 :=        1,              //Channel no.
                IVAR2 :=        0,
                Error :=        F10.1,
                ErrCode :=      FW 12);
...
END_FUNCTION
```

© Siemens AG, 2003. All rights reserved

Functions

3, 4: Rapid PLC NCK data exchange

General

A separate, internal data area is provided to allow the high-speed exchange of data between the PLC and NCK. The size of the internal data field is preset to 1024 bytes. The NCK is accessed (read/written) from the PLC via FC21. The assignment of this area (structure) must declared identically in both the NC parts program and the PLC user program. These data can be accessed from the NC parts program by commands $A_DBB[x], $A_DBW[x], $A_DBD[x], $A_DBR[x] (see Programming Guide). The concrete address is the data field is specified by a byte offset (0 to max. size - 1) in parameter IVAR1. In this case, the alignment must be selected according to the data format, i.e. a Dword starts at a 4-byte limit and a word at a 2-byte limit. Bytes can be positioned on any chosen offset within the data field, single-bit access operations are not supported and converted to a byte access operation by FC 21. The data type information and quantity of data are taken from the ANY parameter transferred via S7Var.
Data consistency is guaranteed only for 1-byte and 2-byte access operations. In the case of longer data types or transfer of fields, a semaphore byte must be programmed in parameter IVAR2 that can be used by FC 21 to determine the validity or consistency of a block. This action must be supported in the NC, i.e. in the parts program, by writing or deletion of the semaphore byte.
The semaphore byte is identified by a value of between 0 and 1023 in IVAR2.

The PLC uses FC 21 to read and write the semaphore byte. The PLC programmer only needs to set up a semaphore variable. For access from the NC via the parts program, the semaphore feature must be programmed using individual instructions according to the flow chart shown below. The sequence is different for reading and writing variables.
Only individual variables or arrays can be supported directly by the semaphore technique. Structure transfers must be subdivided into individual jobs. The programmer must ensure data integrity when programming a dedicated semaphore system.

© Siemens AG, 2003. All rights reserved

Data exchange using semaphore

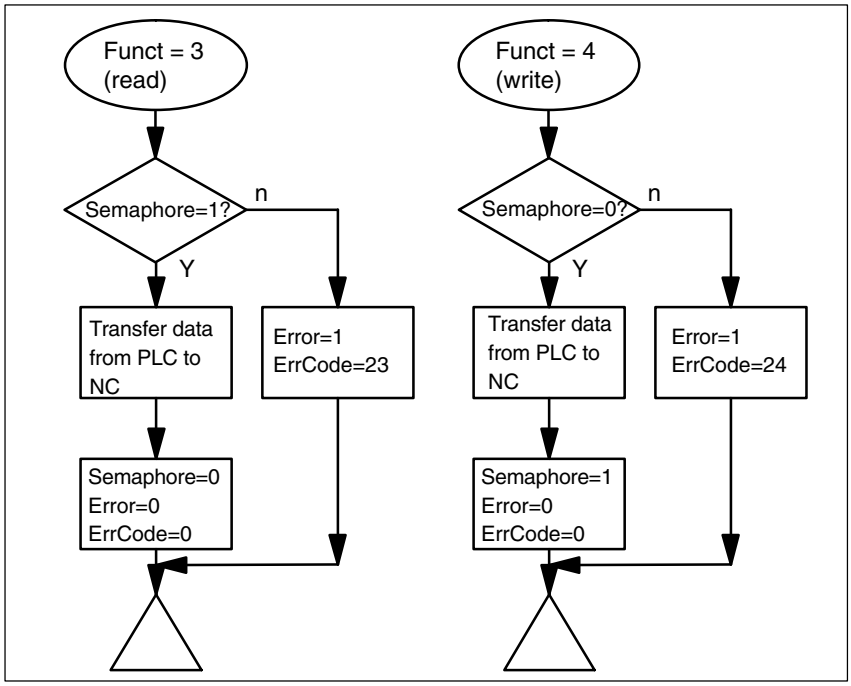


Fig. 4-2 Data exchange using semaphore

A semaphore is not required to transfer data of 1 or 2 bytes in length.
If IVAR2 is set to -1, data are transferred without a semaphore.

Variable value ranges

$A_DLB[n] -128 <= x <= 255
$A_DLW[n] -32768 <= x <= 65535
$A_DLD[n] -2147483648 <= x <= 2147483647

$A_DBB[n] -128 <= x <= 255
$A_DBW[n] -32768 <= x <= 65535
$A_DBD[n] -2147483648 <= x <= 2147483647

© Siemens AG, 2003. All rights reserved

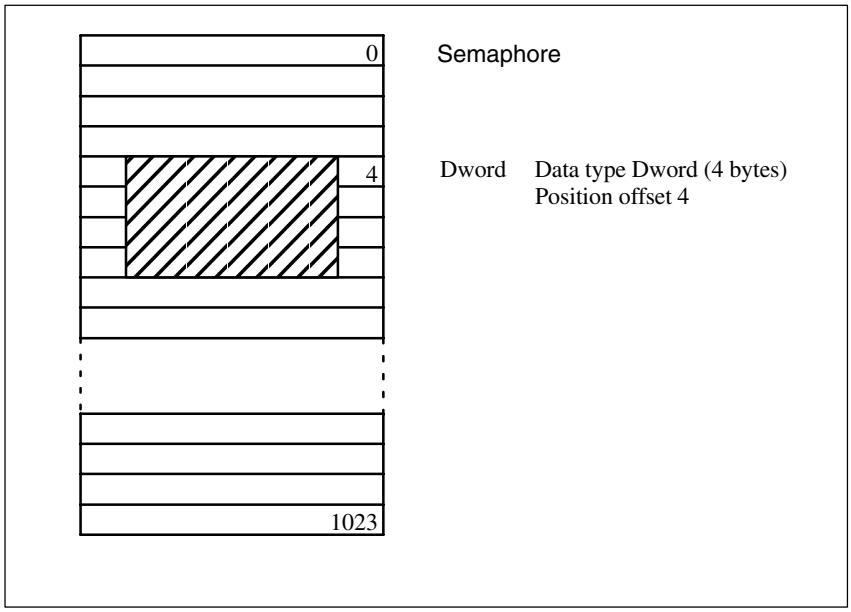
The following signals are relevant:

Signal	Type	Type	Value range	Notes
Enable	I	Bool		1 = FC 21 active
Funct	I	Byte	3, 4	3: Read data 4: Write data
S7Var	I	Any	S7 data area, except local data	Source/destination data storage area
IVAR1	I	Int	0..1023	Position offset
IVAR2	I	Int	-1 .. 1023	Semaphore byte Transfer without semaphore: -1
Error	O	Bool		
ErrCode	O	Int		20: Alignment error 21: Illegal position offset 22: Illegal semaphore byte 23: No new data to be read 24: Cannot write data 25: Local data parameterized for S7Var

Call example

1. Read double word of position offset 4 with semaphore in byte 0 and store in MD 100:

Data type Dword (4 bytes)
Position offset 4




```
CALL FC 21 (
     Enable :=      F 10.0,          // If True, FC21 active
     Funct :=       B#16#3,          //Read data
     S7Var :=       P#M 100.0 DWORD 1,
```

© Siemens AG, 2003. All rights reserved

```
                IVAR1 :=        4,
                IVAR2 :=        0,
                Error :=        F10.1,
                ErrCode :=      FW 12);
AN F10.1;       //Enable while 1, until value is read
R F10.0;
```

2. Read word of position offset 8 without semaphore and store in MW 104:

```
CALL FC 21 (
                Enable :=       F 10.0,         // If True, FC21 active
                Funct :=        B#16#3,         //Read data
                S7Var :=        P#M 104.0 WORD 1,
                IVAR1 :=        8,
                IVAR2 :=        -1,
                Error :=        F10.1,
                ErrCode :=      FW 12);
```

Function

5: Update control signals to channel:

The purpose of function 5 is to transmit important control signals at high speed in between cyclic data transfers. Data bytes 6 and 7 of user interfaces DB21 to DB30 are transferred to the NC. The channel is specified in parameter “IVAR1”. This allows the feed disable or read-in disable to be transferred outside the PLC cycle, for example.

The following signals are relevant:

Signal	Type	Type	Value range	Notes
Enable	I	Bool		1 = FC 21 active
Funct	I	Byte	5	5: Control signals to channel
S7Var	I	Any	S7 data storage area	Not used
IVAR1	I	Int	1..MaxChannel	Channel number
Error	O	Bool		
ErrCode	O	Int		1: “Funct” invalid 10: Channel no. invalid

6: Update control signals to axes: (ab SW 5)

The purpose of function 6 is to transmit important control signals at high speed in between cyclic data transfers. Data byte 2 of application interface DB31 to DB61 is transferred to the NC. The transfer is performed for all activated axes. This allows the servo enable to be transferred outside the PLC cycle, for example.

The following signals are relevant:

Signal	Type	Type	Value range	Notes
Enable	I	Bool		1 = FC 21 active
Funct	I	Byte	6	6: Control signals to axes

© Siemens AG, 2003. All rights reserved

Signal	Type	Type	Value range	Notes
S7Var	I	Any	S7 data storage area	Not used
IVAR1	I	Int	0	
Error	O	Bool		
ErrCode	O	Int		1: “Funct” invalid

7: Update control signals to axes: (ab SW 6)

The purpose of function 7 is to transmit important control signals at high speed in between cyclic data transfers. Data byte 4 of application interface DB31 to DB61 is transferred to the NC. The transfer is performed for all activated axes. This function can be used, for example, to transfer a feed stop outside the PLC cycle.

The following signals are relevant:

Signal	Type	Type	Value range	Notes
Enable	I	Bool		1 = FC 21 active
Funct	I	Byte	7	7: Control signals to axes
S7Var	I	Any	S7 data storage area	Not used
IVAR1	I	Int	0	
Error	O	Bool		
ErrCode	O	Int		1: “Funct” invalid

© Siemens AG, 2003. All rights reserved

4.26 FC 22: TM_DIR Direction selection for tool management

Description of functions

The block TM_DIR provides the shortest path for positioning a magazine or a revolver based on the actual and setpoint position.

As long as a 1 signal is applied to the **Start** input, all output parameters are updated cyclically. Changes can be made to input parameters (e.g. position values) in subsequent PLC cycles. The output signals are undefined when the start signal is at 0 level.

In the case of direction selection with special positioning input “Offset” > 0, a new setpoint position is calculated from the setpoint and special positions and the number of magazine locations according to the following formula:

New setpoint position = (setpoint pos. - (special pos. -1)) neg. modulo # locations

The new setpoint position corresponds to the location number at which the magazine must be positioned so that the setpoint position requested by the user corresponds to the location number of the special position. The directional optimization is active both with and without special positioning.

The block must be called once with the appropriate parameter settings for each magazine.



Warning

The block may be called only in conjunction with the tool management function or after a DB 74 data block has been set up as described below in the example. In this example there are two magazines. The first magazine has 10 locations and the second has 12 locations. When adapting to the real machine, you only need to change the data AnzMag, MagNo[.].

```
DATA_BLOCK DB 74
STRUCT
  P: ARRAY [1..9] OF DINT;
  w1: WORD;
  AnzMag: BYTE;
  res:BYTE;
    MagNo: array [1..16] of struct //Byte 40
    AnzPlatz:INT;
    res1:BYTE;
    res2:BYTE;
  end_struct;
end_struct
BEGIN
  P[4]:=L#320;                    // Absolutely necessary !!!
  AnzMag:=b#16#2;                 //Total number of magazines = 2
  MagNo[1].AnzPlatz:=10;          //Total locations for magazine 1 = 10
  MagNo[2].AnzPlatz:=12;          //Total locations for magazine 2 = 12
END_DATA_BLOCK
```

Note

For further details on tool management (also with regard to PLC) refer to the Description of Functions Tool Management. Furthermore, PI services are provided for tool management via the FB 4, FC 7 and FC 8 (see also the corresponding Sections in this documentation).

© Siemens AG, 2003. All rights reserved

Declaration of the function

STL representation

```
FUNCTION FC 22 : void
// NAME:          TM_DIR
VAR_INPUT
   MagNo:         INT;
   ReqPos:        INT;
   ActPos:        INT;
   Offset:        BYTE;
   Start:         BOOL;
END_VAR
VAR_OUTPUT
   Cw:            BOOL;
   Ccw:           BOOL;
   InPos:         BOOL;
   Diff:          INT;
   Error:         BOOL;
END_VAR
BEGIN
END_FUNCTION
```

Explanation of the formal parameters

The following table shows all formal parameters of the function TM_DIR.

Signal	Type	Type	Value range	Notes
MagNo	I	Int	1..	Magazine number
ReqPos	I	Int	1..	Setpoint location
ActPos	I	Int	1..	Actual location
Offset	I	Byte	0..	Offset for special positioning
Start	I	Bool		Start of calculation
Cw	O	Bool		1 = Move magazine clockwise
Ccw	O	Bool		1 = Move magazine counterclockwise
InPos	O	Bool		1 = In position
Diff	O	Int	0..	Differential path (shortest path)
Error	O	Bool		1 = Error

Call example

```
CALL FC 22(                    //Tool management direction selection
      MagNo :=      2,         //Magazine number
      ReqPos :=     fw 20,     //Destination position
      ActPos :=     fw 22,     //Current position
      Offset :=     b#16#0,    //Offset for special positioning
      Start :=      m 30.4,    //Activate start
                               //Return parameter
      Cw :=         f 30.0,    //Move magazine in clockwise
                               //direction
      Ccw :=        f 30.1,    //Move magazine in counterclock
                               //wise direction
      InPos :=      f 30.2,    //Magazine in position
      Diff :=       fw 32,     //Differential path
      Error :=      f 30.3     //Error has occurred
      );
```

© Siemens AG, 2003. All rights reserved

4.27 FC 24: MCP_IFM2 Transmission of MCP signals to interface PLC/NCK

Description of functions

The FC MCP_IFM2 (M variant, slimline operator panel) is used to transfer

- Modes
- Axis selections
- WCS/MCS switchover commands
- Traversing keys
- Overrides or override simulation signals

from the machine control panel (MCP) to the corresponding signals of the NCK/PLC interface. In the basic program (FC 2), the FC 27 transmits handwheel selections, modes and other operation signals from the operator panel or MMC to the NCK/PLC interface in such a way as to allow the modes to be selected from the machine control panel or the operator panel.

Transfer of MMC signals to the interface can be deactivated by setting the value of the parameter “MMCToIF” to “FALSE” in FB 1 (DB 7). “MMCToIF” can also be activated/deactivated in the cyclical program by setting and resetting (e.g. R gp_par.MMCToIF).

The following specifications apply to the **feed override, axis travel keys** and **INC keys** depending on the active mode or on the selected coordinate system:

- **Feed override:**
  - The feed override is transferred to the interface of the selected channel and to the interface of the axes.
  - The feed override signals are transferred in addition to interface byte “Rapid traverse” override (DBB 5) to the NC channel if MMC signal “Feed override for rapid traverse effective” is set (exception: Switch setting “Zero”). “Rapid traverse override effective” is also set with this MMC signal.
- **Machine functions for INC and axis travel keys:**
  - When the MCS is selected, the signals are transferred to the interface of the selected machine axis.
  - When the WCS is selected, the signals are transferred to the geoaxis interface of the parameterized channel.
  - When the system is switched between MCS and WCS, the active axes are generally deselected.

  SW version-dependent response: See next page.

The **handwheel selection signals of the MMC** are decoded and activated in the associated (machine) axis interface or Geo interface of the relevant handwheel (only if FB 1 parameter “HWheelMMC := TRUE”).

The associated LEDs of the machine control panel are derived from the acknowledgments from the relevant selections.

© Siemens AG, 2003. All rights reserved

Feedrate and spindle Start/Stop are not transferred to the interface, but output modally as a “FeedHold” or “SpindleHold” signal. The user can link these signals to other signals leading to a feed or spindle stop (this can be implemented, e.g., using the appropriate input signals in FC 10: AL_MSG). The associated LEDs are activated at the same time.

The spindle direction (+, -) is not switched directly either, but made available as output parameter “SpindleDir” permitting, for example, FC 18 to be parameterized. A spindle enable signal is also switched via parameter “SpindleHold”. One possible method of moving a spindle directly is to preselect it as an axis so that it can be traversed via (axis) direction keys.

If the machine control panel fails, the signals which it outputs are preset to zero; this also applies to “FeedHold” and “SpindleHold” output signals.

Multiple calls of FC 24 or FC 19, FC 25 are permitted within the PLC cycle from **SW 4 and higher**. In this case, the first call in the cycle drives the LED displays. Moreover, all actions of the parameterized block are performed in the first call. In the following calls, only a reduced level of processing of the channel and mode group interface takes place.
The geometry axes are supplied with directional data only in the first block call in the cycle. Single block processing can be selected/deselected only in the first call in the cycle.

The second machine control panel can be processed if parameter ModeGroupNo has been increased by B#16#10. The mode group number is contained in the lower nibble (lower 4 bits) when the parameter is set. ModeGroupNo = 0 or B#16#10 means that mode group signals are not processed.
ChanNo = 0 means that the channel signals are not processed.

With **SW 5 and higher**, the **INC selections** of the override switch are only transferred to the BAG group. This results in runtime improvements. The activation for this command is performed by this block once after power-up via DB10.DBX 57.0. Machine control panels can still be handled in parallel by this module. The module call for the 2nd machine control panel in OB1 cycle must come after the call of the 1st MCP. Support for of two MCPs is still provided within certain restrictions in the control panel blocks (the standard software does not support axis numbers 10 to 31, mutual interlocking of axis with two MCPs).

Flexible axis configuration

SW 6 and higher provide flexibility as regards the assignment of axis selections or direction keys to machine axis numbers.
The use of 2 machine control panels in simultaneous operation is now better supported by the MCP blocks, in particular the application 2 channels, 2 mode groups. Note that the axis numbers are also specified in the parameterized mode group number of the MCP block in the axis tables of the relevant MCP.

© Siemens AG, 2003. All rights reserved

To afford this flexibility, tables for axis numbers are stored in DB 10. The table starts from byte 8 (symbolic name: MCP1AxisTbl[1..22]) for the first machine control panel (MCP) and from byte 32 (symbolic name: MCP2AxisTbl[1..22]) for the second MCP. The machine axis numbers must be entered here byte by byte.
It is permissible to enter 0 in the axis table. No check is made for illegal axis numbers. Incorrect entries can cause the PLC to stop.
A limit on the maximum possible number of selected axes can be set for FC 19. This limit is set for the appropriate MCP in DB10.DBW30 (symbolic name: MCP1MaxAxis) or DB10.DBW54 (symbolic name: MCP2MaxAxis). The default setting is 0, corresponding to the maximum number of configured axes. The axis numbers and the limit can also be adapted dynamically. This must be followed, however, by another axis selected in FC 19.
The axis numbers must not be switched over whiles axes are being moved via the direction keys.
Set as a default is the compatibility mode with axis numbers 1 to 9 for both MCPs and the limit at the configured number of axes.

Alarms

402401, 402402	**Parameter ModeGroupNo in FC 24, parameter ChanNo in FC 24 not permitted**
Explanation	The parameterized mode group/channel does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

Declaration of the function

```
FUNCTION FC 24: void
//NAME :            MCP_IFM2
VAR_INPUT
  ModeGroupNo :     BYTE;
  ChanNo :          BYTE;
  SpindleIFNo :     BYTE;
END_VAR
VAR_OUTPUT
  FeedHold :        BOOL;
  SpindleHold :     BOOL;
  SpindleDir:       BOOL;
END_VAR
BEGIN
END_FUNCTION
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters

The following table shows all formal parameters of the function MCP_IFM2.

Signal	Type	Type	Value range	Notes
BAGNo	I	Byte	0 - b#16#0A and b#16#10 - b#16#1A	No. of mode group to which the mode signals are transferred. ModeGroupNo >= b#16#10 means access to the 2nd machine control panel.
ChanNo	I	Byte	0 - B#16#0A	Channel no. for the channel signals
SpindleIFNo	I	Byte	0 - 31 (B#16#1F)	Number of the axis interface declared as a spindle.
FeedHold	O	Bool		Feed stop from machine control panel, modal
SpindleHold	O	Bool		Spindle stop from machine control panel, modal
SpindleDir	O	Bool		Spindle direction 0 equals + (counterclockwise) 1 equals - (clockwise)

Call example

```
CALL FC 24(         //Slim-line machine control panel, M variant
                    //Signals to interface
     ModeGroupNo :=  B#16#1,          //Mode group No. 1
     ChanNo :=       B#16#1,          //Channel no. 1
     SpindleIFNo :=  B#16#4,          //Spindle interface
                                      //number = 4
     FeedHold :=     f22.0,           //Feed stop signal
                                      //modal
     SpindleHold :=  db2.dbx151.0);   //Spindle stop modal in
                                      //message data block
     SpindleDir:=    f22.1);          //Spindle direction feedback
```

With these parameter settings, the signals are sent to the 1st mode group, the 1st channel and all axes. In addition, the spindle override is transferred to the 4th axis/spindle interface. The feed hold signal is passed to bit memory 22.0 and the spindle stop signal to data block DB2, data bit 151.0. The spindle direction feedback signal supplied via parameter SpindleDir can be used as a direction input for an additional FC 18 call.

© Siemens AG, 2003. All rights reserved

4.28 FC 25: MCP_IFT Transfer of MCP/OP signals to interface

Description of functions

The FC MCP_IFT (T variant) is used to transfer

- Mode groups
- Direction keys of four axes
- WCS/MCS switchover commands
- Overrides
- Keyswitch

from the machine control panel (MCP) to the corresponding signals of the NCK/PLC interface. In the basic program (FC 2), the FC 27 transmits handwheel selections, modes and other operation signals from the operator panel or MMC to the NCK/PLC interface in such a way as to allow the modes to be selected from the machine control panel or the operator panel. Transmission of MMC signals to the interface can be deactivated by setting the value “FALSE” in the parameter in FB 1 (DB 7) “MMCToIF”.

The following specifications apply to the **feed override, axis travel keys** and **INC keys** depending on the active mode or on the coordinate system selected:

- **Feed override:**
  - The feed override is transferred to the interface of the selected channel and to the interface of the axes.
  - The feed override signals are transferred in addition to interface byte “Rapid traverse” override (DBB 5) to the NC channel if MMC signal “Feed override for rapid traverse effective” is set (exception: Switch setting “Zero”). “Rapid traverse override effective” is also set with this MMC signal.
- **Machine functions for INC and axis travel keys:**
  - When the MCS is selected, the signals are transferred to the interface of the selected machine axis.
  - When the WCS is selected, the signals are transferred to the geoaxis interface of the parameterized channel.

  SW version-dependent response: See next page.

The **handwheel selection signals from the MMC** are decoded and activated in the machine axis or the Geo axis interface of the handwheel selected (only if parameter “HWheelMMC := TRUE” in FB1).

The associated LEDs of the machine control panel are derived from the acknowledgments from the relevant selections.

Feedrate and spindle Start/Stop are not transferred to the interface, but output modally as a “FeedHold” or “SpindleHold” signal. The user can link these signals to other signals leading to a feed or spindle stop (this can be implemented, e.g., using the appropriate input signals in FC 10: AL_MSG). The associated LEDs are activated at the same time.

In the case of machine control panel failure, the signals it supplies are set to zero.

© Siemens AG, 2003. All rights reserved

With **SW 4 and higher**, multiple calls of FC 25 or FC 19 are permitted within the same PLC cycle. In this case, the first call in the cycle drives the LED displays. Furthermore, all actions of the parameterized block are carried out in the first call. In the following calls, only a reduced level of processing of the channel and mode group interface takes place. The geometry axes are supplied with directional data only in the first block call in the cycle.
Selection and deselection of single block processing is affected only in the first cycle.

The second machine control panel can be processed if parameter ModeGroupNo has been increased by B#16#10. The mode group number is contained in the lower nibble (lower 4 bits) when the parameter is set.
ModeGroupNo = 0 or B#16#10 means that mode group signals are not processed.
ChanNo = 0 means that the channel signals are not processed.

With **SW 5 and higher**, the **INC selections** of the override switch are only transferred to the BAG group. This results in runtime improvements. The activation for this command is performed by this block once after powerup via DB10.DBX 57.0. Machine control panels can still be handled in parallel by this module. Support for of two MCPs is still provided within certain restrictions in the control panel blocks (the standard software does not support axis numbers 10 to 31, mutual interlocking of axis with two MCPs).

Flexible axis configuration

SW 6 and higher provide flexibility as regards the assignment of axis selections or direction keys to machine axis numbers.
The use of 2 machine control panels in simultaneous operation is now better supported by the MCP blocks, in particular the application 2 channels, 2 mode groups. The module 2 call for the 2nd machine control panel in OB1 cycle must come after the call of the 1st MCP. Note that the axis numbers are also specified in the parameterized mode group number of the MCP block in the axis tables of the relevant MCP.

To afford this flexibility, tables for axis numbers are stored in DB 10. The table starts from byte 8 (symbolic name: MCP1AxisTbl[1..22]) for the first machine control panel (MCP) and from byte 32 (symbolic name: MCP2AxisTbl[1..22]) for the second MCP. The machine axis numbers must be entered here byte by byte.
It is permissible to enter 0 in the axis table. No check is made for illegal axis numbers. Incorrect entries can cause the PLC to stop.
A limit on the maximum possible number of selected axes can be set for FC 19. This limit is set for the appropriate MCP in DB10.DBW30 (symbolic name: MCP1MaxAxis) or DB10.DBW54 (symbolic name: MCP2MaxAxis). The default setting is 0, corresponding to the maximum number of configured axes. The axis numbers and the limit can also be adapted dynamically. This must be followed, however, by another axis selected in FC 19.
The axis numbers must not be switched over whiles axes are being moved via the direction keys.
Set as a default is the compatibility mode with axis numbers 1 to 9 for both MCPs and the limit at the configured number of axes.

© Siemens AG, 2003. All rights reserved

Alarms

402501, 402502	Parameter ModeGroupNo in FC 25, parameter ChanNo in FC 25 not permitted
Explanation	The parameterized mode group, channel does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

Declaration of the function

```
FUNCTION FC 25: void
//NAME :            MCP_IFT
VAR_INPUT
   ModeGroupNo :    BYTE;
   ChanNo :         BYTE;
   SpindleIFNo :    BYTE;
END_VAR
VAR_OUTPUT
   FeedHold :       BOOL;
   SpindleHold :    BOOL;
END_VAR
BEGIN
END_FUNCTION
```

Explanation of the formal parameters

The following table shows all formal parameters of the function MCP_IFT.

Signal	Type	Type	Value range	Notes
BAGNo	I	Byte	0 - b#16#0A and b#16#10 - b#16#1A	No. of mode group to which the mode signals are transferred. ModeGroupNo >= b#16#10 means access to the 2nd machine control panel.
ChanNo	I	Byte	0 - B#16#0A	Channel no. for the channel signals.
SpindleIFNo	I	Byte	0 - 31 (B#16#1F)	Number of the axis interface declared as a spindle.
FeedHold	O	Bool		Feed stop from machine control panel, modal
SpindleHold	O	Bool		Spindle stop from machine control panel, modal

Call example

```
CALL FC 25(          //Machine control panel T variant
                     //Signals to interface
```

© Siemens AG, 2003. All rights reserved

```
ModeGroupNo :=   B#16#1,          //Mode group No. 1
ChanNo :=        B#16#1,          //Channel no. 1
SpindleIFNo :=   B#16#4,          //Spindle interface
                                  //number = 4
FeedHold :=      f22.0,           //Feed stop signal
                                  //modal
SpindleHold :=   db2.dbx151.0);   //Spindle stop modal in
                                  //message data block
```

With these parameter settings, the signals are sent to the 1st mode group, the 1st channel and all axes. In addition, the spindle override is transferred to the 4th axis/spindle interface. The feed hold signal is passed to bit memory 22.0 and the spindle stop signal to data block DB2, data bit 151.0.

© Siemens AG, 2003. All rights reserved

4.29 FC 26: HPU_MCP Transfer of HPU/HT6 signals to the interface

Description of functions

FC HPU_MCP (machine control panel signals of handheld programming unit) is used to transfer:

- Mode groups
- WCS/MCS switchover commands
- Travel keys and
- Override

from the machine control panel (MCP) to the corresponding signals of the NCK/PLC interface. In the basic program (FC 2), the FC 27 transmits handwheel selections, modes and other operation signals from the operator panel or MMC to the NCK/PLC interface in such a way as to allow the modes to be selected from the machine control panel or the operator panel.

Transfer of MMC signals to the interface can be deactivated by setting the value of the parameter “MMCToIF” to “FALSE” in FB 1 (DB 7).

For **feedrate override, axis travel** and **INC keys,** the following statements apply according to the active mode or coordinate system:

Feed override:

– The feed override is transferred to the interface of the selected channel and to the interface of the axes.

– The feed override signals are transferred in addition to interface byte “Rapid traverse” override (DBB 5) to the NC channel if MMC signal “Feed override for rapid traverse effective” is set (exception: Switch setting “Zero”). “Rapid traverse override effective” is also set with this MMC signal.

Machine functions for INC and axis travel keys:

– When the MCS is selected, the signals are transferred to the interface of the selected machine axis (for 6 axes).

– When the WCS is selected, the signals of the first three axes are transferred to the Geo axis interface of the parameterized channel. The remaining three axes are transferred to the interface of the selected machine axis

SW version-dependent response: See next page.

The **handwheel selection signals of the MMC** are decoded and activated in the associated (machine) axis interface or geo axis interface of the relevant handwheel (only if FB 1 parameter “HWheelMMC := TRUE”).

The LEDs on the machine control panel derived from the selections in the acknowledgment.

© Siemens AG, 2003. All rights reserved

Feedrate and spindle Start/Stop are not transferred to the interface, but output modally as a “FeedHold” or “SpindleHold” signal. The user can link these signals to other signals leading to a feed or spindle stop (this can be implemented, e.g., using the appropriate input signals in FC 10: AL_MSG). The associated LEDs are activated at the same time.

In the case of a failure of the MCP, the signals it supplies are set to zero.

With **SW 4 and higher**, multiple calls of FC 25 or FC 25 are permitted within the same PLC cycle. In this case, the first call in the cycle drives the LED displays. Moreover, all actions of the parameterized block are performed in the first call. In the following calls, only a reduced level of processing of the channel and mode group interface takes place. The geometry axes are supplied with directional data only in the first block call in the cycle.

The second machine control panel can be processed if parameter ModeGroupNo has been increased by B#16#10. The mode group number is contained in the lower nibble (lower 4 bits) when the parameter is set.
ModeGroupNo = 0 or B#16#10 means that mode group signals are not processed.
ChanNo = 0 means that the channel signals are not processed.

With **SW 5 and higher**, the **INC selections** of the override switch are only transferred to the BAG group. This results in runtime improvements. The activation for this command is performed by this block once after powe-rup via DB10.DBX 57.0. 2 MCPs can still be handled in parallel by this module. The module 2 call for the 2nd machine control panel in OB1 cycle must come after the call of the 1st MCP. Support for of two MCPs is still provided within certain restrictions in the control panel blocks (the standard software does not support axis numbers 10 to 31, mutual interlocking of axis with two MCPs).

Flexible axis configuration

SW 6 and higher provide flexibility as regards the assignment of axis selections or direction keys to machine axis numbers.
The use of 2 machine control panels in simultaneous operation is now better supported by the MCP blocks, in particular the application 2 channels, 2 mode groups. Note that the axis numbers are also specified in the parameterized mode group number of the MCP block in the axis tables of the relevant MCP.

© Siemens AG, 2003. All rights reserved

To afford this flexibility, tables for axis numbers are stored in DB 10. The table starts from byte 8 (symbolic name: MCP1AxisTbl[1..22]) for the first machine control panel (MCP) and from byte 32 (symbolic name: MCP2AxisTbl[1..22]) for the second MCP. The machine axis numbers must be entered here byte by byte.
It is permissible to enter 0 in the axis table. No check is made for illegal axis numbers. Incorrect entries can cause the PLC to stop.
A limit on the maximum possible number of selected axes can be set for FC 19. This limit is set for the appropriate MCP in DB10.DBW30 (symbolic name: MCP1MaxAxis) or DB10.DBW54 (symbolic name: MCP2MaxAxis). The default setting is 0, corresponding to the maximum number of configured axes. The axis numbers and the limit can also be adapted dynamically. This must be followed, however, by another axis selected in FC 19.
The axis numbers must not be switched over whiles axes are being moved via the direction keys.
Set as a default is the compatibility mode with axis numbers 1 to 9 for both MCPs and the limit at the configured number of axes.

Alarms

402601, 402602	**Parameter ModeGroupNo in FC 26, parameter ChanNo in FC 26 not permitted**
Explanation	The parameterized mode group, parameterized channel does not exist
Reaction	Alarm display and PLC STOP
Remedy	Set parameter correctly
Continuation	After cold restart

Declaration of the function

```
FUNCTION FC 26: void
//NAME :            HPU_MCP
VAR_INPUT
  ModeGroupNo :     BYTE;
  ChanNo :          BYTE;
END_VAR

BEGIN
END_FUNCTION
```

Explanation of the formal parameters

The following table shows all formal parameters of the function HPU_MCP.

Signal	Type	Type	Value range	Notes
BAGNo	I	Byte	0 - b#16#0A and b#16#10 - b#16#1A	No. of mode group to which the mode signals are transferred. ModeGroupNo >= b#16#10 means access to the 2nd machine control panel.
ChanNo	I	Byte	0 - B#16#0A	Channel no. for the channel signals.

© Siemens AG, 2003. All rights reserved

MCP selection signals to the user interface

Operating modes and machine functions

Source: MCP key	Target: Interface DB (parameter ModeGroupNo) Representation for mode group 1
AUTOMATIC	DB11.DBX0.0
MDA (MDI)	DB11.DBX0.1
JOG	DB11.DBX0.2
REPOS	DB11.DBX1.1
REF	DB11.DBX1.2
TEACH_IN	DB11. DBX1.0
INC 1 ... 10 000, INC Var. **(SW 5 and higher)**	DB11.DBB2, Bit 0 to 5

SW 4 and lower

Source: MCP key	Target: Interface DB (parameter ChanNo)
INC 1 ... 10 000, INC Var.	DB21, ... DBB13, Bit 0 to 5
INC 1 ... 10 000, INC Var.	DB21, ... DBB17, Bit 0 to 5
INC 1 ... 10 000, INC Var.	DB21, ... DBB21, Bit 0 to 5

Source: MCP key	Target: **Interface DB** (6 axis DBs)
INC 1 ... 10 000, INC Var. **(up to and including SW 4)**	DB 31, ... DBB5, Bit 0 to Bit 5

© Siemens AG, 2003. All rights reserved

Direction keys Rapid traverse override

The transfer is dependent upon the selected axis. The corresponding interface bits are deleted for axes that are not selected.

Source: MCP key	Target: Interface DB (parameter ChanNo)
Direction key +	DB21, ... DBX12.7
Direction key -	DB21, ... DBX12.6
Rapid traverse override	DB21, ... DBX12.5
Direction key +	DB21, ... DBX16.7
Direction key -	DB21, ... DBX16.6
Rapid traverse override	DB21, ... DBX16.5
Direction key +	DB21, ... DBX20.7
Direction key -	DB21, ... DBX20.6
Rapid traverse override	DB21, ... DBX20.5

Source: MCP key	Target: Interface DB (6 assigned axis DBs)
Direction key +	DB 31, ... DBX4.7
Direction key -	DB 31, ... DBX4.6
Rapid traverse override	DB 31, ... DBX4.5

Override

Source: MCP setting	Target: Interface DB (parameter ChanNo)
Feedrate override	DB21, ... DBB4

Source: MCP setting	Target: Interface DB (6 assigned axis DBs)
Feedrate override	DB 31, ... DBB0
Spindle override	DB 31, ... DBB19 (Parameter SpindleIFNo)

Channel signals

Source: MCP keys	Target: Interface DB (parameter ChanNo)
NC Start	DB21, ... DBX7.1
NC Stop	DB21, ... DBX7.3
Reset	DB21, ... DBX7.7
Block	DB21, ... DBX0.4

© Siemens AG, 2003. All rights reserved

Checkback signals from user interface for controlling displays

Operating modes and machine functions

Table 4-31 Operating modes and machine functions

Target: MCP output	Source: Interface DB (parameter ModeGroupNo) Representation for mode group 1
AUTOMATIC	DB11, ... DBX6.0
MDA (MDI)	DB11, ... DBX6.1
JOG	DB11, ... DBX6.2
REPOS	DB11, ... DBX7.1
REF	DB11, ... DBX7.2
TEACH IN	DB11, ... DBX7.0

WCS/MCS output is are operated through key actuation.

Call example

```
CALL FC 26(         //Machine control panel of HPU/HT6
                    //Signals to interface
     ModeGroupNo :=  B#16#1,          //Mode group No. 1
     ChanNo :=       B#16#1);         //Channel no. 1
```

With these parameter settings, the signals from the first parameterized machine control panel are sent to the 1st mode group, the 1st channel and all axes.

© Siemens AG, 2003. All rights reserved

4.30 FC 19, FC 24, FC 25, FC 26 Source code description

Task

Machine control panel to application interface
(FC 19 M version, FC 24 slim-line version, FC 25 T version, FC 26 HPU/HT6 variant).

Associated blocks

DB 7 (this was DB 5 up to SW 3), number of mode groups, channels, axes
DB 7, Pointer to machine control panels
DB 8, Storage for the next cycle
FC 20, Output of error messages

Resources used

None

General

Blocks FC 19 (M version), FC 24 (slim-line version), FC 25 (T version) and FC 26 (HPU/HT6 version) transfer the machine control panel to and from the application interface. In the input parameters, “ModeGroupNo” selects the mode group to be processed by the block. The “ModeGroupNo” parameter also selects the number of the machine control panel. “ChanNo” selects the channel to be processed. The “SpindleIFNo” parameter defines the axis interface of the spindle. The spindle override is transferred to this spindle interface. The parameters are checked for errors. Output parameters “FeedHold” and “SpindleHold” are generated from the 4 feed/spindle disable and feed/spindle enable keys and are returned with “logical 1” for disable. Information for the next cycle is stored in DB8, bytes 0 to 3 or bytes 62 to 65, depending on the machine control panel number. This information is the edge trigger flag, feed value and selected axis number. The blocks are provided with user data via the pointer parameters in DB 7 MCP1In and MCP1Out (MCP2In and MCP2Out). The pointers are addressed indirectly via a further pointer from the VAR section of DB7 in order to avoid absolute addressing. This additional pointer is determined symbolically in FB1.

Block functions

All four blocks have a similar structure. The blocks are organized into separate sections for individual subtasks.
In the input network, various parameters are copied into local variables. The machine control signals (user data for input/output area) are also copied between locations using the various pointers in DB 7 (gp_par). These local variables are handled in the block for reasons of efficiency. Some values are initialized for the start-up.

The MCS/WCS switchover with edge evaluation, axis selections, direction keys and rapid traverse overlay are determined in Network Global_in for further processing in the block. User-specific modifications should be made to this Section of the program. The user-specific modifications will usually mainly involve axis selections.

© Siemens AG, 2003. All rights reserved

Only the keyswitch information is copied in Network NC.

The mode group network transfers the modes of the keys as dynamic signals to the NCK. If the mode group number is 0, this network is not processed. If the number is too large, message 401901 or 402501 is output and the control switches to Stop mode.

Selections for incrementing the geometry axes are passed dynamically from the keys to the interface for all geometry axes (up to SW 4) in the network channel. The NC Start, Stop, Reset and Single Block functions are activated by corresponding checkback signals. The direction keys of the geometry axes are supplied if a corresponding preselection is made, otherwise they are cleared. If the channel number is 0, this network is not processed. If the number is too large, message 401902 or 402502 is output and the control switches to Stop mode.

The spindle network transfers the spindle override into the interface configured via SpindleIFNo.

The axis network transfers the feedrate override to the selected axis interface. The direction keys are assigned to the selected axis/spindle. If an axis was selected previously, the direction information is set to 0. The INC checkback of the selected axis is displayed.

The output parameters are prepared and the LED signals of the INC machine function are generated in the global_out network.

The output network transfers the output signals of the machine control panel from the VAR_TEMP image to the logical address. The data for the next cycle are also saved.

Axis selection extension

Network Global_in must be modified for more than 9 axes. If further keys and machine control panel LEDs are to be used, you should perform the following measures:
1. Command UD DW#16#Value deletes all defined LEDs for axis selections. At the present time, all 9 axis selection LEDs are processed with the bit mask.
2. Command UW W#16# with the comment “mask all axis selection keys” checks whether a new direction selection has been made. The bit string must be modified here.
3. The jump string should be extended with new jump labels. The new jump labels should be inserted in descending order before label m009. The selection information should be extended for the new jump labels, as described for labels m009 and m008.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Signal/Data Descriptions 5

5.1 Interface signals NCK/PLC, MMC/PLC, MCP/PLC

General

The NCK/PLC, MMC/PLC and MCP/PLC interface signals are contained in the Lists document

References: /LIS/, Lists

for SINUMERIK 840D, with references to the appropriate Description of Functions in which the signals are described.
The NCK signals that are evaluated by the basic program and passed in a conditioned state to the user interface are listed in the following sections.

5.2 Decoded M signals

General

The decoded M signals are entered in the data bits listed in the following table in DB 21 for NC channel 1 and in DB 22 for NC channel 2. The signal length is one PLC cycle.

The user can easily derive the active signal for each M group by evaluating the signals.

Note

No M functions are decoded for spindle (M3, M4, M5, M70).

Address in DB21/22	Variable	Type	Initial value	Comment
DBX 194.0 ... 7	M_Fkt_M0 ... M7	Bool		M signals M0 ... M7
DBX 195.0 ... 7	M_Fkt_M8 ... M15	Bool		M signals M8 ... M15
DBX 196.0 ... 7	M_Fkt_M16 ... M23	Bool		M signals M16 ... M23
DBX 197.0 ... 7	M_Fkt_M24 ... M31	Bool		M signals M24 ... M31
DBX 198.0 ... 7	M_Fkt_M32 ... M39	Bool		M signals M32 ... M39
DBX 199.0 ... 7	M_Fkt_M40 ... M47	Bool		M signals M40 ... M47
DBX 200.0 ... 7	M_Fkt_M48 ... M55	Bool		M signals M48 ... M55
DBX 201.0 ... 7	M_Fkt_M56 ... M63	Bool		M signals M56 ... M63
DBX 202.0 ... 7	M_Fkt_M64 ... M71	Bool		M signals M64 ... M71

© Siemens AG, 2003. All rights reserved

Address in DB21/22	Variable	Type	Initial value	Comment
DBX 203.0 ... 7	M_Fkt_M72 ... M79	Bool		M signals M72 ... M79
DBX 204.0 ... 7	M_Fkt_M80 ... M87	Bool		M signals M80 ... M87
DBX 205.0 ... 7	M_Fkt_M88 ... M95	Bool		M signals M88 ... M95
DBX 206.0 ... 3	M_Fkt_M96 ... M99	Bool		M signals M96 ... M99

Note

If M02/M30 is received as an auxiliary function output it does not mean that the parts program is ended. To determine this status reliably, the signal DB 21... DBB33.5: M02/M30 must be polled for active status. The channel status must be RESET. The auxiliary function output could arise from an asynchronous subroutine (ASUB) or a synchronized action and has nothing to do with the real end of the parts program in this case.

5.3 G functions

General

The G functions are entered in DB 21 for NC channel 1, in DB 22 for NC channel 2 and so on. The significance of the value entered in the bytes (8-bit integer) is explained in Section 12.3 of the Programming Guide.

When the power is switched on, the value zero is defined in the user interface for all G groups (i.e. active G group undefined).

When the NC program terminates or is aborted, the last status of the groups is retained.

SW 3.1 and lower: Following an NC program start, the values of the first 8 G groups are overwritten in accordance with the default setting defined by machine data as are the values listed in the NC program.

SW 3.2 and higher: Following an NC program start, the values of the 8 G groups specified in MD 22510 are overwritten in accordance with the default setting defined by machine data as are the values listed in the NC program.

Address in DB21/22	Variable	Type	Initial value	Comment
DBB 208	G_FKT_GR_1	Byte	0	Active G function of group 1
DBB 209	G_FKT_GR_2	Byte	0	Active G function of group 2
DBB 210	G_FKT_GR_3	Byte	0	Active G function of group 3
DBB 211	G_FKT_GR_4	Byte	0	Active G function of group 4
DBB 212	G_FKT_GR_5	Byte	0	Active G function of group 5
DBB 213	G_FKT_GR_6	Byte	0	Active G function of group 6
DBB 214	G_FKT_GR_7	Byte	0	Active G function of group 7
DBB 215	G_FKT_GR_8	Byte	0	Active G function of group 8
DBB 216	G_FKT_GR_9	Byte	0	Active G function of group 9
DBB 217	G_FKT_GR_10	Byte	0	Active G function of group 10

© Siemens AG, 2003. All rights reserved

Address in DB21/22	Variable	Type	Initial value	Comment
DBB 218	G_FKT_GR_11	Byte	0	Active G function of group 11
DBB 219	G_FKT_GR_12	Byte	0	Active G function of group 12
DBB 220	G_FKT_GR_13	Byte	0	Active G function of group 13
DBB 221	G_FKT_GR_14	Byte	0	Active G function of group 14
DBB 222	G_FKT_GR_15	Byte	0	Active G function of group 15
DBB 223	G_FKT_GR_16	Byte	0	Active G function of group 16
DBB 224	G_FKT_GR_17	Byte	0	Active G function of group 17
DBB 225	G_FKT_GR_18	Byte	0	Active G function of group 18
DBB 226	G_FKT_GR_19	Byte	0	Active G function of group 19
DBB 227	G_FKT_GR_20	Byte	0	Active G function of group 20
DBB 228	G_FKT_GR_21	Byte	0	Active G function of group 21
DBB 229	G_FKT_GR_22	Byte	0	Active G function of group 22
DBB 230	G_FKT_GR_23	Byte	0	Active G function of group 23
DBB 231	G_FKT_GR_24	Byte	0	Active G function of group 24
DBB 232	G_FKT_GR_25	Byte	0	Active G function of group 25
DBB 233	G_FKT_GR_26	Byte	0	Active G function of group 26
DBB 234	G_FKT_GR_27	Byte	0	Active G function of group 27
DBB 235	G_FKT_GR_28	Byte	0	Active G function of group 28
DBB 236	G_FKT_GR_29	Byte	0	Active G function of group 29
DBB 237	G_FKT_GR_30	Byte	0	Active G function of group 30
DBB 238	G_FKT_GR_31	Byte	0	Active G function of group 31
DBB 239	G_FKT_GR_32	Byte	0	Active G function of group 32
DBB 240	G_FKT_GR_33	Byte	0	Active G function of group 33
DBB 241	G_FKT_GR_34	Byte	0	Active G function of group 34
DBB 242	G_FKT_GR_35	Byte	0	Active G function of group 35
DBB 243	G_FKT_GR_36	Byte	0	Active G function of group 36
DBB 244	G_FKT_GR_37	Byte	0	Active G function of group 37
DBB 245	G_FKT_GR_38	Byte	0	Active G function of group 38
DBB 246	G_FKT_GR_39	Byte	0	Active G function of group 39
DBB 247	G_FKT_GR_40	Byte	0	Active G function of group 40
DBB 248	G_FKT_GR_41	Byte	0	Active G function of group 41
DBB 249	G_FKT_GR_42	Byte	0	Active G function of group 42
DBB 250	G_FKT_GR_43	Byte	0	Active G function of group 43
DBB 251	G_FKT_GR_44	Byte	0	Active G function of group 44
DBB 252	G_FKT_GR_45	Byte	0	Active G function of group 45
DBB 253	G_FKT_GR_46	Byte	0	Active G function of group 46
DBB 254	G_FKT_GR_47	Byte	0	Active G function of group 47
DBB 255	G_FKT_GR_48	Byte	0	Active G function of group 48
DBB 256	G_FKT_GR_49	Byte	0	Active G function of group 49
DBB 257	G_FKT_GR_50	Byte	0	Active G function of group 50
DBB 258	G_FKT_GR_51	Byte	0	Active G function of group 51

© Siemens AG, 2003. All rights reserved

Address in DB21/22	Variable	Type	Initial value	Comment
DBB 259	G_FKT_GR_52	Byte	0	Active G function of group 52
DBB 260	G_FKT_GR_53	Byte	0	Active G function of group 53
DBB 261	G_FKT_GR_54	Byte	0	Active G function of group 54
DBB 262	G_FKT_GR_55	Byte	0	Active G function of group 55
DBB 263	G_FKT_GR_56	Byte	0	Active G function of group 56
DBB 264	G_FKT_GR_57	Byte	0	Active G function of group 57

G functions (values)

You will find the list of G functions in:

References: /PG/, Programming Guide: Fundamentals

© Siemens AG, 2003. All rights reserved

5.4 Message signals in DB2

General

DB2 allows the user to display the messages for individual signals on the operator panel. As the lists of interface signals show, signals are divided into predefined groups. When a message occurs, disappears or is acknowledged, the number entered in the message number column is transferred to the MMC. A text can be stored on the MMC for each message number (see Chapter "Message numbers" of the Installation and Startup Guide for SINUMERIK 840D).

Note

The number of user areas can be parameterized via FB 1**.**

After the configuration has been modified (FB 1: MsgUser), DB 2/3 must be deleted.

Channel areas in DB 2

Range	Address	Message number
Channel 1	DBX0.0 – DBX11.7	510.000 – 510.231
Channel 1, geo axes	DBX12.0 – DBX17.7	511.100 –511.315
Channel 2	DBX18.0 – DBX29.7	520.000 –520.231
Channel 2, geo axes	DBX30.0 – DBX35.7	521.100 –521.315
Channel 3	DBX36.0 – DBX47.7	530.000 –530.231
Channel 3, geo axes	DBX48.0 – DBX53.7	531.100 –531.315
Channel 4	DBX54.0 – DBX65.7	540.000 –540.231
Channel 4, geo axes	DBX66.0 – DBX71.7	541.100 –541.315
Channel 5	DBX72.0 – DBX83.7	550.000 –550.231
Channel 5, geo axes	DBX84.0 – DBX89.7	551.100 –551.315
Channel 6	DBX90.0 – DBX101.7	560.000 –560.231
Channel 6, geo axes	DBX102.0 – DBX107.7	561.100 –561.315
Channel 7	DBX108.0 – DBX119.7	570.000 –570.231
Channel 7, geo axes	DBX120.0 – DBX125.7	571.100 –571.315
Channel 8	DBX126.0 – DBX137.7	580.000 –580.231
Channel 8, geo axes	DBX138.0 – DBX143.7	581.100 –581.315
Channels 9 and 10 are not currently implemented		

© Siemens AG, 2003. All rights reserved

User areas in DB 2

Range	Address	Message number
Axis/spindle 1	DBX144.0 – DBX145.7	600.100 – 600.115
Axis/spindle 2	DBX146.0 – DBX147.7	600.200 – 600.215
Axis/spindle 3	DBX148.0 – DBX149.7	600.300 – 600.315
Axis/spindle 4	DBX150.0 – DBX151.7	600.400 – 600.415
Axis/spindle 5	DBX152.0 – DBX153.7	600.500 – 600.515
Axis/spindle 6	DBX154.0 – DBX155.7	600.600 – 600.615
Axis/spindle 7	DBX156.0 – DBX157.7	600.700 – 600.715
Axis/spindle 8	DBX158.0 – DBX159.7	600.800 – 600.815
Axis/spindle 9	DBX160.0 – DBX161.7	600.900 – 600.915
Axis/spindle 10	DBX162.0 – DBX163.7	601.000 – 601.015
Axis/spindle 11	DBX164.0 – DBX165.7	601.100 – 601.115
Axis/spindle 12	DBX166.0 – DBX167.7	601.200 – 601.215
Axis/spindle 13	DBX168.0 – DBX169.7	601.300 – 601.315
Axis/spindle 14	DBX170.0 – DBX171.7	601.400 – 601.415
Axis/spindle 15	DBX172.0 – DBX173.7	601.500 – 601.515
Axis/spindle 16	DBX174.0 – DBX175.7	601.600 – 601.615
Axis/spindle 17	DBX176.0 – DBX177.7	601.700 – 601.715
Axis/spindle 18	DBX178.0 – DBX179.7	601.800 – 601.815
Axes 19 to 31 are not currently implemented		

© Siemens AG, 2003. All rights reserved

User areas in DB 2

Range	Address	Message number
User area 0	DBX180.0 – DBX187.7	700.000 – 700.063
User area 1	DBX188.0 – DBX195.7	700.100 – 700.163
User area 2	DBX196.0 – DBX203.7	700.200 – 700.263
User area 3	DBX204.0 – DBX211.7	700.300 – 700.363
User area 4	DBX212.0 – DBX219.7	700.400 – 700.463
User area 5	DBX220.0 – DBX227.7	700.500 – 700.563
User area 6	DBX228.0 – DBX235.7	700.600 – 700.663
User area 7	DBX236.0 – DBX243.7	700.700 – 700.763
User area 8	DBX244.0 – DBX251.7	700.800 – 700.863
User area 9	DBX252.0 – DBX259.7	700.900 – 700.963
User area 10	DBX260.0 – DBX267.7	701.000 – 701.063
User area 11	DBX268.0 – DBX275.7	701.100 – 701.163
User area 12	DBX276.0 – DBX283.7	701.200 – 701.263
User area 13	DBX284.0 – DBX291.7	701.300 – 701.363
User area 14	DBX292.0 – DBX299.7	701.400 – 701.463
User area 15	DBX300.0 – DBX307.7	701.500 – 701.563
User area 16	DBX308.0 – DBX315.7	701.600 – 701.663
User area 17	DBX316.0 – DBX323.7	701.700 – 701.763
User area 18	DBX324.0 – DBX331.7	701.800 – 701.863
User area 19	DBX332.0 – DBX339.7	701.900 – 701.963
User area 20	DBX340.0 – DBX347.7	702.000 – 702.063
User area 21	DBX348.0 – DBX355.7	702.100 – 702.163
User area 22	DBX356.0 – DBX363.7	702.200 – 702.263
User area 23	DBX364.0 – DBX371.7	702.300 – 702.363
User area 24	DBX372.0 – DBX379.7	702.400 – 702.463

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

6

Note

The assignment of FBs, FCs, DBs and an overview of the interface signals are given in
References: /LIS/, Lists.

6.1 FB/FC assignment

FB number	FC number	Meaning
1		Basic program
2 – 29		Reserved for Siemens
	1	Initialization of basic program
	2 – 29	Reserved for Siemens
	30 – 35	See Note referring to ShopMill, Manual-Turn
	30 – 127	Free for user assignment
30 – 127		Free for user assignment

Note

* The true upper limit of the block number (FB/FC) depends on the the PLC CPU contained in the selected NCU.
See Section 2.1 ”Key data of PLC CPUs for FM-NC, 810D, 840D”.

For assignment of FCs and FBs, see Section 6.4 “Memory requirements of basic PLC program for FMNC, 810D, 840D”.

6.2 DB assignment

Note

Only the required number of DBs according to NCMD is set up.

© Siemens AG, 2003. All rights reserved

Table 6-1 Overview of data blocks

DB No.	Designation	Name	Package
1		Reserved for Siemens	BP
2 – 4	PLC–MELD	PLC messages	BP
5 – 8		Reserved for Siemens	
9	NC–COMPILE	Interface for NC compile cycles	BP
10	NC INTER-FACE	Central NC interface	BP
11	Mode group 1	Interface mode group	BP
12		Computer link and transport system interface	
13 – 14		Reserved for Siemens	
15		Program	
16		PI Service definition	
17		Version identifier	
18		Reserved for basic program	
19		MMC interface	
20		PLC machine data	
21 to 30	CHANNEL 1	Interface NC channels	BP
31–61	AXIS 1	Interfaces for axes/spindles or free for user assignment	BP
62 – 70		Free for user assignment	
71 – 74		Tool management	BP
75 – 76		M group decoding	
77 – 80		Reserved for Siemens	
81 – 89		See Note referring to ShopMill, ManualTurn	
81 – 127		**Free for user assignment**	

Note

* The true upper limit of the block number (DB) depends on the PLC CPU contained in the selected NCU.
See Section 2.1 “Key data of PLC CPUs for FM-NC, 810D, 840D”.

The data blocks of channels, axes/spindles and tool management functions that are not activated may be assigned as desired by the user.

ManualTurn

ManualTurn uses FC 30 to 35 and DB 81 to 89.

ManualTurn is an operating system for conventional, cycle-controlled turning machines. The above FCs and DBs can be used without hesitation provided that the machine to be configured is not a turning machine with a maximum of two axes and one spindle. If your machine is of this type and if you might require conventional operating methods in addition to CNC functions, then you should not use the FCs and DBs above.

© Siemens AG, 2003. All rights reserved

ShopMill

ShopMill uses FC30 to 35 and DB 81 to 89.

ShopMill is an operating system for 2 1/2D milling machines in the workshop environment. The above FCs and DBs can be used without hesitation provided that the machine to be configured is not a milling machine for 2 1/2D machining operations. However, if you intend to use your machine for applications of this type, then the FCs and DBs should not be used.

6.3 Assigned timers

Timer No.	Meaning
0 – 9	Reserved for Siemens
10 – 127	Free for user assignment

Note

* The true upper limit of the block number (timer) depends on the PLC CPU contained in the selected NCU.
See Section 2.1 “Key data of PLC CPUs for FMNC, 810D, 840D”.

© Siemens AG, 2003. All rights reserved

6.4 Memory requirements of basic PLC program for FM-NC, 810D, 840D

General

The basic program consists of basic and optional functions. The **basic functions** include the cyclical signal exchange between the NC and PLC.
The **options** include, for example, the FCs which can be used if required.

The following table lists the memory requirements for the basic functions and the options. The data quoted represent guide values, the actual values depend on the current software version.

Table 6-2 Memory requirements of blocks with SINUMERIK 840D

Block Type, No.	Function	Remark	Block size (bytes) Load memory	Working memory
Basic functions in basic program				
FB 1, 15, 16, 17, 18		Must be loaded	3616	3052
FC 1, 2, 3, 4, 11, 20		Must be loaded	7208	6608
DB 4, 5, 7, 8, 17, 19		Must be loaded	2490	966
DB 2, 3, 6		Are generated by the BP	992	812
OB 1, 40, 100		Must be loaded	490	282
		Total	14796	11720
PLC/NCK, PLC/MMC interface				
DB 10	PLC/NCK signals	Must be loaded	318	262
DB 11	Signals PLC/	Is generated by BP	80	44
DB 21, 30	PLC/channel signals	Are generated by BP as a function of NC-MD	352 each	316 each
DB 31, ...61	PLC/axis signals Spindel	Are generated by BP as a function of NC-MD	180 each	144 each
Basic program options				
Machine control panel				
FC 19	Transfer of MCP signals, M variant	Must be loaded when M variant of MCP is installed	1498	1258
FC 25	Transfer of MCP signals, T variant	Must be loaded when T variant of MCP is installed	1358	1160
FC 24	Transfer of MCP signals, slim variant	Must be loaded when slim variant of MCP is installed	1358	1160
FC 26	Transfer of MCP signals, HPU variant	Must be loaded for HPUs	1358	1160
FC 14	MPI/OPI transfer	Must be loaded with MCPNum > 0	942	802
Handheld unit				
FC 13	Display control HHU	Can be loaded for handheld units	1264	1044

© Siemens AG, 2003. All rights reserved

Table 6-2 Memory requirements of blocks with SINUMERIK 840D

Block Type, No.	Function	Remark	Block size (bytes) Load memory	Working memory
Error/status messages				
FC 10	Acquisition FM/BM	Load when FM/BM is used	1572	1350
ASUB				
FC 9	ASUB start	Load when PLC ASUBs are used	656	538
Basic program options				
Concurrent axes				
FC 15	Positioning of linear/rotary axes	Load for axis positioning by PLC	656	546
FC 16	Positioning of indexing axes	Load for axis positioning by PLC	674	560
Star/delta switchover				
FC 17	Star/delta switchover of MSD	Load for star/delta switchover	612	494
Spindle control				
FC 18	Spindle control	Load for spindle control from PLC	826	676
PLC/NC communication				
FB 2	Read NC variable	Load for Read NC variable	396	224
DB n[1)]	Read NC variable	One instance DB per FB2 call	426 each	270 each
FB 3	Write NC variable	Load for Write NC variable	396	224
DB m[1)]	Write NC variable	One instance DB per FB3 call	426 each	270 each
FB 4	PI services	Load for PI services	334	214
DB o[1)]	PI services	One instance DB per FB4 call	234 each	130 each
DB 16	PI services description	Load for PI services	1190	408
FB5	Read GUD variables	Load for PI services	532	365
DB p	Read GUD variables	One instance DB per FB5 call	308 each	166 each
FB 6	Allgemeine Communication	Load for Read/write NC variables and PI services	5986	5228
DB 15	Allgemeine Communication	Instance DB for FB6	440	172
Tool management				
FC 6	Basic function	Load for tool management option	1382	1182
FC 7	Transfer function Revolver	Load for tool management option	530	430
FC 8	Transfer function	Load for tool management option	1002	834
FC 22	Direction selection	Load if direction selection is required	404	300
DB 71	Loading locations	Is generated by BP as a function of NC-MD	76+30*B	40+30*B
DB 72	Spindles	Is generated by BP as a function of NC-MD	76+48*Sp	40+48*Sp
DB 73	Revolver	Is generated by BP as a function of NC-MD	76+44*R	40+44*R
DB 74	Basic function	Is generated by BP as a function of NC-MD	136+(B+Sp+R)*20	100+(B+Sp+R)*20

© Siemens AG, 2003. All rights reserved

Table 6-2 Memory requirements of blocks with SINUMERIK 840D

Block Type, No.	Function	Remark	Block size (bytes) Load memory	Working memory
Compile cycles				
DB 9	PLC/NCK PLC compile cycles	Is generated by BP as a function of NC option	472	436
1): DB number must be specified by PLC user				

Example: Based on the memory requirements in the table above, the memory requirements have been determined for two sample configurations (see table below).

Block Type, No.	Function	Remark	Block size (bytes) Load memory	Working memory
Minimum configuration (1 spindle, 2 axes and TMCP)				
See above	Basic program, base		14796	11720
	Interface DBs		1290	1054
	MCP		2300	1962
		Total	**18386**	**14736**

Block Type, No.	Function	Remark	Block size (bytes) Load memory	Working memory
Maximum configuration (2 channels, 4 spindles, 4 axes, TMCP)				
See above	Basic program, base		14796	11720
See above	Interface DBs		2542	2090
See above	MCP		2300	1962
See above	Error/status messages		1572	1350
See above	ASUBs	1 ASUB initiation	656	538
See above	Concurrent axis	For 2 turrets	674	560
See above	PLC/NC Communication	1 x read variable and 1 x write variable	8070	6388
See above	Tool management	2 turrets with one loading point each	3430	2854
See above	Compile cycles		472	436
		Total	**34512**	**27898**

■

© Siemens AG, 2003. All rights reserved

Useful Tips on Programming with STEP7 7

General

Some useful tips on programming complex machining sequences in STEP7 are given in the following. This information concentrates mainly on the handling of data type POINTER and ANY. Detailed information about the structure of data types POINTER and ANY can be found in Chapter “CPU register and storage of data” in STEP7 manual “Designing user programs”.

© Siemens AG, 2003. All rights reserved

7.1 Copying data

The following is an example of how to copy data at high speed from one DB into another.

```
// DB xx.[AR1] is the source
// DI yy.[AR2] is the destination

   AUF     DB 100;                          //Source DB
   LAR1    P#20.0;     //Source start address at data byte 20
   AUF     DI 101;                          //Destination DB
   LAR2    P#50.0;     //Destination start address at data byte 50

//AR1, AR2, DB, DI loaded beforehand

   L    42;              //Transfer 84 bytes
F001:
   L    DBW  [AR1,P#0.0];                   //Copy word by word
   T    DIW  [AR2,P#0.0];
   +AR1    P#2.0;
   +AR2    P#2.0;
   TAK;
   LOOP   F001;
```

7.2 ANY and POINTER

General

The following programming examples illustrate different programming mechanisms. They demonstrate how input/output and transit variables (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT) are accessed by data types “POINTER” or “ANY” within an FC or FB. The access operations are described in such a way that a part symbolic method of programming can be used.

© Siemens AG, 2003. All rights reserved

7.2.1 Use of POINTER and ANY in FC if POINTER or ANY is available as parameter

Description of functions

FC 99 has inputs parameters that are defined as POINTER or ANY. The example shows a body program via which the subcomponents of the POINTER or ANY can be accessed. In this case, the DB parameterized with POINTER or ANY is opened and the address offset stored as a cross-area pointer in address register AR1, thus allowing access to data elements of variables (generally structures and arrays) that are addressed via the POINTER, ANY. This access operation is described at the end of the relevant program sequence in the example. With data type ANY, it is also possible to execute a check or branch when the variable is accessed based on the data type and the number of elements involved.

```
FUNCTION FC 99: VOID
VAR_INPUT
   Row : BYTE ;
   Convert : BOOL ;   //Activate numerical conversion
   Addr : POINTER ;   //Points to variable
   Addr1 : ANY;
END_VAR

VAR_TEMP
   dbchr : WORD ;
   Number : WORD ;
   type : BYTE ;
END_VAR
BEGIN
NETWORK
TITLE =

//POINTER
   L    P##Addr;
   LAR1  ;                                  //Fetch pointer
   L    W [AR1,P#0.0]; //Fetch DB number
   T    #dbchr;
   L    D [AR1,P#2.0];  //Offset part of pointer
   LAR1  ;
   AUF   DB [#dbchr];  //Open DB of variable
   L    B [AR1,P#40.0];                     //Fetch byte value via pointer with
                                            //address offset 40
//ANY
   L    P##Addr1;
   LAR1  ;                                  //Fetch ANY
   L    B [AR1,P#1.0];                      //Fetch type
   T    #typ;
   L    W [AR1,P#2.0];                      //Fetch number
   T    #Number;
   L    W [AR1,P#4.0];                      //Fetch DB number
   T    #dbchr;
   L    D [AR1,P#6.0];                      //Offset part of pointer
   LAR1  ;
   AUF   DB [#dbchr];                       //Open DB of variable
   L    B [AR1,P#0.0];                      //Fetch byte value via ANY
```

© Siemens AG, 2003. All rights reserved

7.2.2 Use of POINTER and ANY in FB if POINTER or ANY is available as parameter

Description of functions

FB 99 has inputs parameters that are defined as POINTER or ANY. The example shows a body program via which the subcomponents of the POINTER or ANY can be accessed. In this case, the DB parameterized with POINTER or ANY is opened and the address offset stored as a crossarea pointer in address register AR1, thus allowing access to data elements of variables (generally structures and arrays) that are addressed via the POINTER, ANY. This access operation is described at the end of the relevant program sequence in the example. With data type ANY, it is also possible to execute a check or branch when the variable is accessed based on the data type and the number of elements involved.

```
FUNCTIONBLOCK FB 99
  VAR_INPUT
    Row : BYTE ;
    Convert : BOOL ;                  //Activate numerical conversion
    Addr : POINTER ;                  //Points to variable
    Addr1 : ANY;
  END_VAR
  VAR_TEMP
    dbchr : WORD ;
    Number : WORD ;
    type : BYTE ;
  END_VAR
BEGIN
NETWORK
TITLE =

//POINTER
  L    P##Addr;
  LAR1  ;                             //Fetch pointer from instance DB
  L    DIW [AR1,P#0.0];               //Fetch DB number
  T    #dbchr;
  L    DID [AR1,P#2.0];               //Offset part of pointer
  LAR1  ;
  AUF   DB [#dbchr];                  //Open DB of variable
  L    B [AR1,P#40.0];                //Fetch byte value via pointer with
                                      //address offset 40

//ANY
  L    P##Addr1;
  LAR1  ;                             //Fetch ANY from instance DB
  L    DIB [AR1,P#1.0];               //Fetch type
  T    #typ;
  L    DIW [AR1,P#2.0];               //Fetch number
  T    #Number;
  L    DIW [AR1,P#4.0];               //Fetch DB number
  T    #dbchr;
  L    DID [AR1,P#6.0];               //Offset part of pointer
  LAR1  ;
  AUF   DB [#dbchr];                  //Open DB of variable
  L    B [AR1,P#0.0];                 //Fetch byte value via ANY
```

© Siemens AG, 2003. All rights reserved

7.2.3 POINTER or ANY variable for transfer to FC or FB

With version 1 or later of STEP7 it is possible to define a POINTER or an ANY in VAR_TEMP. The following two examples show how an ANY can be supplied.

1. Several ANY parameters are defined in an FB (FC). A specific ANY parameter must now be chosen from a selection list for transfer to another FB (FC). This can only be done by means of an ANY in VAR_TEMP. 1 to 4 can be set in parameter “WhichAny” in order to select Addr1 to Addr4.

Note

Address register AR2 is used in the block. However, this register is also used for multi-instance DBs. For this reason, the relevant FB must not be declared as a multi-instance DB.

```
FUNCTIONBLOCK FB 100
  CODE_VERSION1                    //To deactivate multi-instance DB with
                                   //STEP7 version 2 and higher
  VAR_INPUT
    WhichAny : INT;
     Addr1 : ANY;        //Observe predetermined order
     Addr2 : ANY;
     Addr3 : ANY;
     Addr4 : ANY;
  END_VAR
  VAR_TEMP
    dbchr : WORD ;
      Number : WORD ;
      type : BYTE ;
      Temp_addr : ANY;
  END_VAR
BEGIN
NETWORK
TITLE =

  L    WhichAny;
  DEC 1;
  L    P#10.0;                     //10 bytes per ANY
   *I;
  LAR2;
  L    P##Addr1;
  +AR2;                            //Add start address of ANYs
  L    P##Temp_addr;
  LAR1  ;                          //Fetch pointer from VAR_TEMP
  L    DID [AR2,P#0.0];            //Transfer pointer values to VAR_TEM
  T    LD [AR1,P#0.0];
  L    DID [AR2,P#4.0];
  T    LD [AR1,P#4.0];
  L    DIW [AR2,P#8.0];
  T    LW [AR1,P#8.0];

CALL FB 101, DB 100
                (ANYPAR := #Temp_addr);        //ANYPAR is
                                               //data type ANY
```

© Siemens AG, 2003. All rights reserved

2. An ANY parameter that has already been compiled must be transferred to another FB (FC). This can be done only by means of an ANY stored in VAR_TEMP

```
FUNCTIONBLOCK FB 100
  VAR_INPUT
    DBNumber: INT;
     DBOffset : INT;
     Data type: INT;
     Number: INT;
  END_VAR

  VAR_TEMP
    dbchr : WORD ;
     Temp_addr : ANY;
  END_VAR
BEGIN
NETWORK
TITLE =
  L    P##Temp_addr;
  LAR1  ;                          //Fetch pointer from VAR_TEMP
  L    B#16#10;                    //ID ANY
  T    LB [AR1,P#0.0];
  L    Data type;
  T    LB [AR1,P#1.0];
  L    Number;
  T    LW [AR1,P#2.0];
  L    DBNumber;
  T    LW [AR1,P#4.0];
  L    DBOffset;
  SLD 3;                           //Offset is a bit offset
  T    LD [AR1,P#6.0];
CALL FB 101, DB 100
                (ANYPAR := #Temp_addr);        //ANYPAR is
                                               //data type ANY
```

© Siemens AG, 2003. All rights reserved

7.3 Multi-instance DB

With version 2 and higher of STEP7, FBs might have a multi-instance capability, i.e. they might incorporate multi-instance DBs. The primary characteristic of multi-instance DBs is that they can be used for various instances of FBs (see STEP7 documentation), thus allowing the DB data quantity to be increased. However, multi-instance DBs should be activated only when they are actually going to be used since the runtime and code size for FBs of this type are greater as compared to normal instance DBs.

Note

When complex programs are implemented in multi-instance FBs that use a pointer and address register, it is important for the programmer to observe certain rules. These rules must be observed by the programmers.
With multi-instance DBs, the start address of the variable (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR) is transferred with the DI data block register and address register AR2. When variables are accessed within the multi-instance FB, the compiler independently controls the access operation via address register AR2. However, when complex program sections also have to work with address registers in the same FB (e.g. to copy data), then the old contents of AR2 must be saved before the register is changed. The contents of AR2 must be restored to their original state before an instance variable (VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT, VAR) is accessed. The contents of AR2 of the instance FB can best be saved in a local variable (VAR_TEMP). Command 'Load pointer to an instance variable' supplies a pointer value starting at the beginning of the instance data. To be able to access this variable via a pointer, the offset stored in AR2 must be added.

Example

```
FUNCTION_BLOCK FB 99
VAR_INPUT
  varin: INT;
END_VAR
VAR
  variable1: ARRAY[0..9] OF INT;
  variable2: INT;
END_VAR
```

© Siemens AG, 2003. All rights reserved

```
BEGIN
  L  P##variable1;      //Pointer to beginning of ARRAY.
                        //8500 0010 is now stored in the accumulator

  //AR2 also contains a cross-area pointer. If cross-area accessing is to be used, then an
  area must be masked out when these two pointers are added.

  UD   DW#16#00FF_FFFF;             //Mask out area
  LAR1                              //Load to AR1
  TAR2;
  +AR1   AR2;                       //Add AR2 instance offset
  //The ARRAY of variable 1 can now be indirectly accessed via AR1.
  L   DIW [AR1, P#0.0];             //e.g. access to first element
END_FUNCTION_BLOCK
```

7.4 Strings

General

The STRING data type is required by certain services of the basic program. For this reason, some additional facts about the string structure and general handling procedures for parameter assignments are given below.

Structure of STRING

A data of type STRING is generally stored (defined) in a data block. There are two methods of defining a string:

1. Only the data type STRING is assigned to a variable. The STEP7 compiler automatically generates a length of 254 characters.
2. Data type STRING is assigned to a variable together with a string length in square parenthesis (e.g. [32]). With this method, the STEP7 compiler generates a string length corresponding to the input.

Two bytes more than prescribed by the definition are always stored for variables of the STRING data type. The STEP7 compiler stores the maximum possible number of characters in the 1st byte. The 2nd byte contains the number of characters actually used. Normally, the useful length of the assigned string is stored in byte 2 by the compiler. The characters (1 byte per character) are then stored from the 3rd byte onwards.

String parameters are generally assigned to blocks of the basic program by means of a POINTER or ANY. Such assignments must generally by made using symbolic programming methods. The data block which contains the parameterizing string must be stored in the symbol list. The assignment to the basic program block is then made by means of the symbolic data block name followed by a full stop and the symbolic name of the string variable.

© Siemens AG, 2003. All rights reserved

7.5 Determining offset addresses for data block structures

General

A problem arises with symbolic determination of an offset address within a structured DB, e.g. an ARRAY or STRUCTURE is stored somewhere within the DB. After loading the address register symbolically with the start address, you might like to access the individual elements of the ARRAY or STRUCTURE via an address register. One way of loading the address register symbolically is to use an FC whose input parameter is a pointer. The address of the ARRAY or STRUCTURE is then assigned symbolically to the input parameter of this FC in the program. The program code in the FC now determines the offset address from the input parameter, and passes the offset address in the address register (AR1) to the calling function. Symbolic addressing is thus possible even with indirect access.

```
FUNCTION FC 99: VOID
VAR_INPUT
Addr : POINTER ;    //Points to variable
END_VAR

BEGIN
NETWORK
TITLE =

  L    P##Addr;
  LAR1  ;                          //Fetch pointer from Addr
  L    D [AR1,P#2.0];              //Offset part of variable pointer
  LAR1  ;

END_FUNCTION
```

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

8 Coupling two FM-NC Modules to one PLC

General

As the complexity of machine tools increases, more axes have to be configured.

By coupling two FM-NC modules to the machine, it is possible to increase the number of available axes to eight. By connecting a local axis module (FM354) to each FM-NC, the number of axes can be further increased to ten.

Through the interlinking of two FM-NC modules, it has been possible to produce a lowcost variant for controlling more than five axes. FM modules can be coupled with one PLC and thus coordinated by the basic PLC program.

There are no major differences in basic PLC program operation between single FM-NC and dual FM-NC operation. Functions or sequences that are identical for both configurations are not mentioned in this Section, but described in detail in the description of the basic program. This description of the two FM-NC module configuration should therefore be treated as a type of “differential comparison”.

© Siemens AG, 2003. All rights reserved

8.1 Coupling two FM-NCs with one PLC

General

When two FM-NC modules are connected to a PLC, a second NC is created that is capable of fully autonomous operation. However, it can also be synchronized with the first NC via the PLC interface. The FM-NC behaves like an AS-300 FM (Function Module) with Cbus connection. Data can therefore be exchanged between the AS300CPU and FM-NCs via a P bus or a C bus.

The second FM-NC can be installed in any position in the rack.

Data are exchanged with the operator panel and machine control panel via the MPI interface on the AS-300-CPU. From an external viewpoint, the AS-CPU and FM-CPU are each addressed directly.

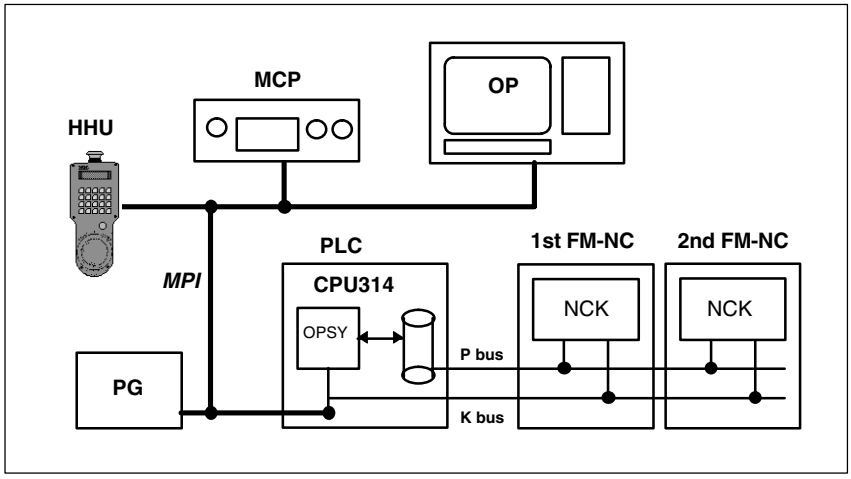


Fig. 8-1 General configuration

Suggested component configuration

Minimum configuration (2 FM-NCs, 1 MCP, 1 MMC or OP):

The MMC is always linked to one of the FM-NCs via soft key menu “Connect”. The FM-NC can be operated and monitored after the link has been made.

Coupling the PLC CPU to the FM-NCs

The basic program organizes the data block transfer between the PLC and NCK via the P bus using SFCs. Since each NCK has its own logical address, the basic program ensures that data are correctly distributed to the relevant FM-CPU.

© Siemens AG, 2003. All rights reserved

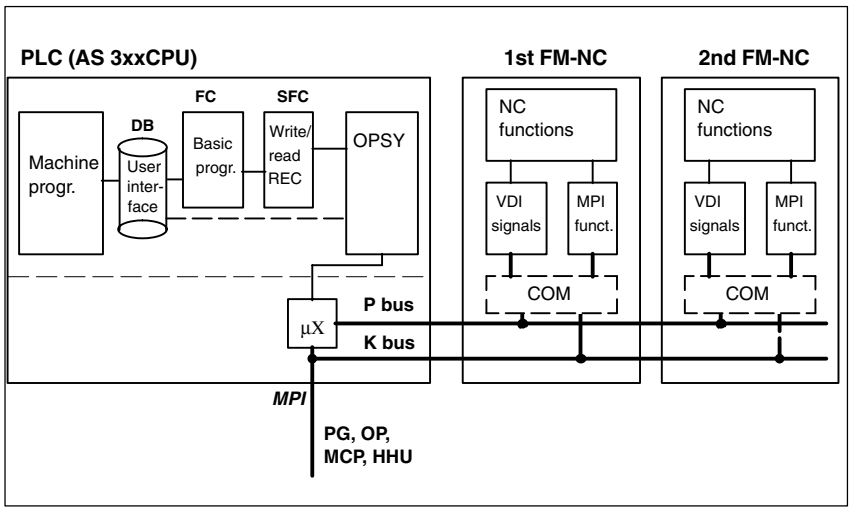


Fig. 8-2 Coupling PLC/FM-NCs

NCK/PLC interface (user interface)

The NCK/PLC interface has the same structure elements irrespective of whether one or several FM-NCs are connected. If the configuration includes several FM-NCs, then only the number of elements changes, but not the structure.

The basic program organizes the data transfer between the PLC and NCKs via the P bus using SFCs.

© Siemens AG, 2003. All rights reserved

8.2 Coupling with the components MMC/MSTT/OP

General

The following diagram shows one possible method of coupling two FM-NC modules to one PLC. The MMC can either be connected to the two FM-NCs alternately or replaced by an OP after complete startup of the installation.

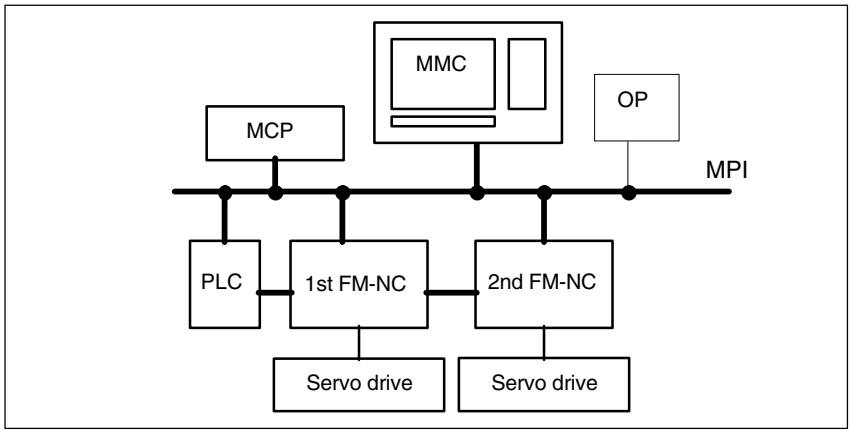


Fig. 8-3 Example of configuration (2 FM-NCs, 1MMC/1 OP, 1 MCP)

In this configuration, the MMC is connected to the two FM-NCs alternately. In this case, the assignment is always 1:1, i.e. the existing operating tree remains unchanged. The FM-NC can be operated and monitored after the link has been made.

As an option, a programming device can also be connected to the MPI link. The maximum permissible number of bus nodes (16) must not, however, be exceeded.

© Siemens AG, 2003. All rights reserved

8.3 Interface structure

General

The PLC user interface (modified PLC/NCK interface specially for user) for two FM-NCs is identical to that for one FM-NC in every respect except for data quantity. Mapping in interface data blocks is required due to the large number of signals involved. These are global data blocks from the viewpoint of the PLC program. The basic program generates these DBs during system start-up using NC machine data that are transferred by the NCK to the basic PLC program in the start-up routine (number of channels, axes, etc.).

Signals and data can be distributed and assigned by means of unambiguous segmentation of the PLC/NCK interface and the associated assignment between data blocks and FM-NCs in the basic PLC program.

8.3.1 PLC/NCK interface

Fundamental principle

The user interface is structured such that it is ideally suited for the purpose of distributing signals for several FM-NC modules. A separate user interface is provided for each FM-NC with the result that each FM-NC has a fixed, block-based assignment.

In order to ensure that two function modules connected to a PLC can function and interact properly, two interface signal blocks, two mode group blocks and two channel data blocks must be provided so that signals can be assigned correctly.

Table 8-1 Suggested structure for PLC/NCK interface structure

Signals	1st FM-NC	2nd FM-NC
NC	DB 10	[DB user]
Mode group	DB 11	DB 12
CHANNEL	DB 21	DB 22
AXIS 1	DB 31	DB 36
AXIS 2	DB 32	DB 37
AXIS 3	DB 33	DB 38
AXIS 4	DB 34	DB 39
AXIS 5	DB 35	DB 40

By applying a consistent NCK/PLC interface structure, it is possible to ensure an unambiguous assignment between data blocks and the relevant FM-NCs. As a result, the source and destination of the appropriate data for the exchange of basic program data between the PLC and the NCK are unique. The basic program is executed cyclically and coordinates the assignment of signals to the respective data blocks. This system guarantees that each FM-NC can be supplied cyclically with current data.

© Siemens AG, 2003. All rights reserved

Since two NCs are created in the system when two FM-NCs are interlinked, two NCK signal DBs are also required. Since the next block number of DB10 is already assigned elsewhere, the user must assign a free data block number when parameterizing FB1 in OB100. This data block can then be accessed by the basic program and used to store the signals of the second FM-NC.

Each FM-NC consists of a mode group and a CHANNEL. Data blocks are assigned to these elements on the user interface.

To be able to operate the axes of channels of two FM-NCs in one operating mode, the user program must supply the mode group interface DBs in parallel. It is not possible to combine the axes of two FM-NCs to form one mode group.

The following overviews show the structures of the first PLC/NCK interface (1st FM-NC)

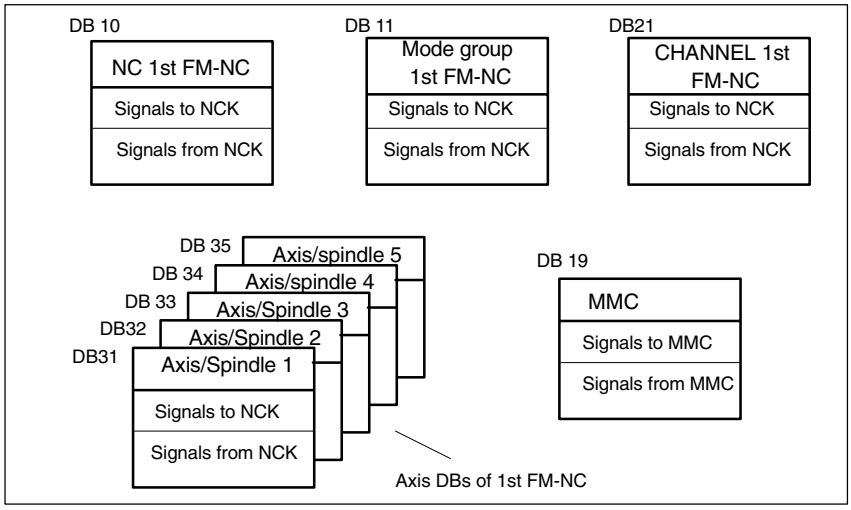


Fig. 8-4 PLC/NCK interface (1st FM-NC)

and the structure of the second PLC/NCK interface (2nd FM-NC).

© Siemens AG, 2003. All rights reserved

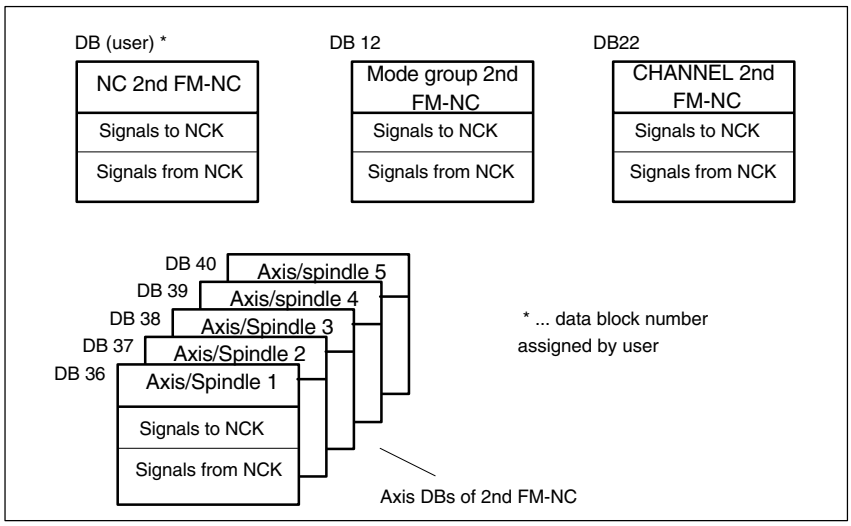


Fig. 8-5 PLC/NCK interface (2nd FM-NC)

As the first diagram shows, the data block (DB19) containing the MMC signals is assigned to only one FM-NC. Although the MMC can be switched over between the two FM-NCs, allowing both to be operated and monitored, it addresses only the data blocks of the first FM-NC (DB10, DB11, DB21).

This special feature must be noted with respect to signal exchange between PLC/mode group, PLC/NC channel and PLC/axes, spindle, drives.

It is not possible at present to assign MMC signals to the second data area (2nd FM-NC).

8.3.2 PLC/MMC interface

The system configuration to date provided for one FM-NC to be coupled with the PLC. The MMC is therefore always assigned to one specific FM-NC and linked to it via the PLC.

To allow two FM-NC modules to be coupled with one PLC, it was necessary to depart from the old basic concept. While the MMC can be switched over to the second FM-NC, it is permanently assigned to the first FM-NC in the system, i.e. it addresses only the interface DBs of the first NC (see also previous topic).

8.3.3 PLC/MCP/HHU interface

General

The PLC CPU registers the function modules with an MPI connection during start-up and assigns default MPI addresses to them. These are subsequently sent to the relevant modules via the P bus.

When two FM-NC modules are installed, the PLC also assigns an MPI address to the second module so that it can be addressed in the system by the PLC, MMC and MCP. The diagram shows default bus addresses and the new address of the second FM-NC.

© Siemens AG, 2003. All rights reserved

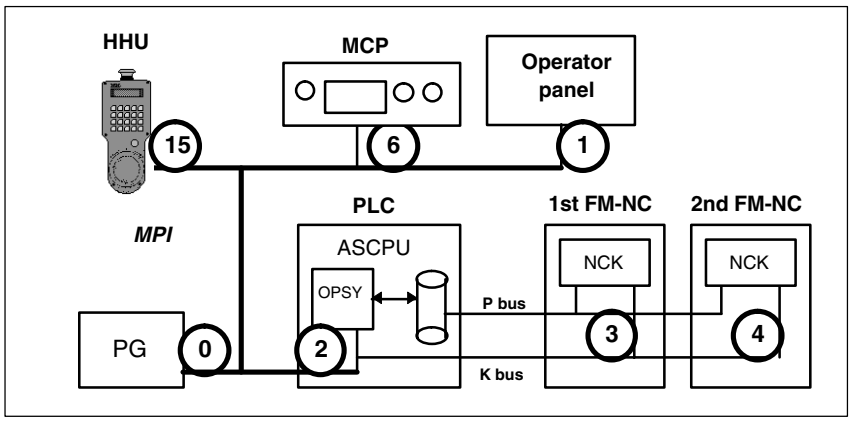


Fig. 8-6 Default bus addresses for configuration with two FM-NCs

Table 8-2 Permissible bus address for FM-NC

Node	Perm. setting range	Standard
Operator panel	MMC 100–103: 1–15	1
PLC	1–15	2
1st FM-NC	PLC addr+1	3
2nd FM-NC	PLC addr+2	4
MCP/keyboard int.	0–15	6
PG/PC	Permanent	0

As with MMC to PLC signals, FC19 uses only the interface data blocks of the first FM-NC. The user program must ensure correct distribution of signals to the second FM-NC in accordance with the overall system application.

© Siemens AG, 2003. All rights reserved

8.4 Function of the basic program for two FM-NCs

General

The PLC and the first and second FM-NCs are all synchronized during startup.

The system and user data blocks are checked for completeness and the most important basic program parameters are verified. In cases of error, the basic program outputs an error identifier to the diagnostics buffer and switches the PLC to STOP.

To ensure correct control system start-up, it is vital to synchronize the NCK and PLC, as these systems have their own types of power-up procedure. The start-up procedure is devised such that the PLC is synchronized alternately with the first NCK and then with the second NCK. If the first or second FM-NC has not run up correctly, then the affected module outputs an alarm on power-up timeout. All other nodes start to operate together after a successful power-up and exchange “signs of life” with one another.

Cyclic operation

The cyclical exchange of data is performed by the basic program at the start of the PLC cycle (OB1). Each FM-NC has its own logical address and the basic PLC program ensures that data are distributed correctly among the relevant FM-NCs on the basis of unambiguous interface segmentation. Data is exchanged between the FM modules and the PLC in every cycle.

NCK reset and synchronization

MMC soft key function “NCK reset” is effective only in the FM-NC that is currently linked to the MMC in control systems with more than one FM-NC. The function is activated by an MMC key. Since the MMC can only operate and monitor one FM-NC at a time, the effect of the Reset command is restricted to only one FM-NC. The other FM-NC continues to operate cyclically, unaffected by the Reset on the other module.

Status/control signals for mode group, channel, axes and spindles

A common feature of control and status signals is that they are all bit arrays. From the PLC viewpoint, these are updated at the beginning of OB1.

Since both modules operate in mutual independence, they both receive cyclically updated values from the PLC.

Response to NC failure

General: During cyclical operation, the PLC continuously monitors the NC availability by querying the sign-of-life character. If one of the FM-NCs fails to respond, then the corresponding NCK/PLC interface is neutralized and the relevant *NCK-CPU Ready signal* in the area for *signals from NC (DB 10.DBX 104.7 and/or DB 10[1].DBX 104.7)* is reset. Furthermore, the signals exchanged between the NCK and PLC are set to their initial state

(for a detailed description see topic “Response in case of a failure of the NC”).
The PLC itself remains active so that it can continue to control machine functions.

The user’s programming engineer must evaluate the “NCK-CPU Ready” signal since the control system will be disabled and thus NC block processing interrupted if the signal has been set to zero as a result of a PLC or FM-NC failure.

---

1) DB number assigned by user for second NCK signal DB

© Siemens AG, 2003. All rights reserved

8.5 Special features of blocks when using two FM-NCs

For a detailed description of the parameters and the declaration of the functions and function blocks, please refer to the basic program description on the topic “Block Descriptions” (see Chapter 4).

8.5.1 FB1: RUN_UP

Call example FM-NC

A call example for the FB 1 in OB 100 is given below. The purpose of this example is to explain the new parameters required for coupling of a second FM-NC.

```
ORGANIZATION_BLOCK OB 100

TITLE = “Complete Restart”

VERSION : 3.0

VAR_TEMP

        OB100_EV_CLASS      : BYTE ;
        OB100_STRTUP        : BYTE ;
        OB100_PRIORITY      : BYTE ;
        OB100_OB_NUMBR      : BYTE ;
        OB100_RESERVED_1    : BYTE ;
        OB100_RESERVED_2    : BYTE ;
        OB100_STOP          ; WORD ;
        OB100_STRT_INFO     : DWORD ;
        OB100_DATE_TIME     : DATE_AND_TIME ;
        OB100_RESERVED_3    : WORD;
        OB100_RESERVED_4    : WORD;

END_VAR

BEGIN

        Call FB 1, DB 7(    //First Statement in OB 100
        MCPNum              :=1,
        MCP1In              :=P#E118.0,
        MCP1Out             :=P#A120.0,
        MCP1StatRec         :=P#A108.0,
        MCP1Timeout         :=S5T#700MS,
        NCCyclTimeout       :=S5T#200MS,
        NCRunupTimeout      :=S5T#50S,
        NCLaddr1            :=320,   //I/O address of 1st FM-NC (default)
        NCLaddr2            :=336,   //I/O address of 2nd FM-NC (default)
        NCKSigDB2           :=102,   //DB no. of NCKSignalDB of 2nd FM-NC
        UserVersion         :=0,     //V. NO of user program
        UserDate            :=0,     //V. date of user program
        UserTime            :=0);    //V. time of user program

//USER program

END_ORGANIZATION_BLOCK
```

© Siemens AG, 2003. All rights reserved

Explanation of the formal parameters The following table shows all formal parameters of the RUN_UP function for FM-NC.

Signal	Type	Type	Value range	Notes
MCPNum	I	Int	0 to 2	Number of active MCPs 0: No MCP installed
MCP1In MCP2In	I	Pointer	I0.0 to I120.0 or M0.0 to M248.0 or DBn.DBX0.0 to DBXm.0	Start address for input signals of relevant machine control panel
MCP1Out MCP2Out	I	Pointer	Q0.0 to Q120.0 or M0.0 to M248.0 or DBn.DBX0.0 to DBXm.0	Start address for output signals of relevant machine control panel *1
MCP1StatRecMCP2 StatRec	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for receiving from machine control panel: DW#16#00040000:Time monitoring Timeout, otherwise 0
MCP1TimeoutMCP2T imeout	I	S5time	Recommendation: 700 ms	Cyclic sign-of-life monitoring for machine control panel
HHU	I	Int		Handheld unit interface: 0 – No HHU 1 – HHU on MPI
BHGIn	I	Pointer	I0.0 to I124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address for PLC receive data from handheld unit
BHGOut	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address PLC for send data to handheld unit
BHGStatRec	I	Pointer	Q0.0 to Q124.0, M0.0 to M252.0 or DBn.DBX0.0 to DBXm.0	Start address for status double word for receiving from handheld unit: DW#16#00040000:Timeout, otherwise 0
BHGTimeout	I	S5time	Recommendation: 700 ms	Cyclical sign-of-life monitoring for handheld unit
NCLaddr1	**I**	**INT**	**320 (default)**	**I/O address of 1st FM-NC**
NCLaddr2	**I**	**INT**	**336 (default)**	**I/O address of 2nd FM-NC**
NCKSigDB2	**DB**	**INT**	**DB 81 to DB 127**	**NCK signal DB of 2nd FM_NC**
NCCyclTimeout	I	S5time	Recommendation: 200 ms	Cyclical sign-of-life monitoring NCK
NCRunupTimeout	I	S5time	Recommendation: 5 min	NCK power-up monitoring
ListMDecGrp	I	INT	0	
NCKomm	I	Bool		PLC NC communications services (FB 2/3/4/5/7: Put/Get/PI/GETGUD) 1: Active
MMCToIF	I	Bool		Transfer of MMC signals to interface (operating modes, program control, etc.) True: Active
HWheelMMC	I	Bool		True: Handwheel selection via MMC False: Handwheel selection by user prog.
MsgUser	I	Int	0..25	Number of user areas for messages (DB 2)
UserVersion	I	Dword		User version
UserDate	I	Dword		User date
UserTime	I	Dword		User time

© Siemens AG, 2003. All rights reserved

8.5.2 FB2...5: Read/write NC variables, PI services, GUD variables

General

The “Read/write NC variables”, “General PI services” and “Read GUD variables” functions operate for two FM-NC modules in the same way as described in the relevant block description. Signals and data are assigned to the relevant FM-NCs by means of parameter “FMNCNo”. The user conveys the ID of the FM-NC to be processed to the basic program in this parameter.

For example, a parameter setting of “FMNCNo:= 2” when the function is called indicates that the second channel in the system must be processed, thereby automatically assigning the job to the second FM-NC.

Call example

Example of “FM-NCNo” parameter on call of Read NC variable

```
CALL FB 2, DB 110 (
     Req          := M 0.0,
     NumVar       := 2,
     Addr1        := ”NCVAR”.C1_RP_rpa0_0,
     Line1        := W#16#1,
     Addr2        := ”NCVAR”.C1_RP_rpa0_0,
     Line2        := W#16#2,
     FM-NCNo      := 1,          //No. of FM-NC to be processed
     Error:       := M 1.0,
     NDR          := M 1.1,
     State        := MW 2,
     RD1          := P#M 4.0 REAL 1,
     RD2          := P#M 24.0 REAL 1);
```

Explanation of the formal parameters

The following table shows all formal parameters of the GET function

Signal	Type	Type	Value range	Notes
Req	I	Bool		Job start with positive signal edge
NumVar	I	Int	1 to 8 (corresponds to use of Addr1 to Addr8)	Number of variables to be read
Addr1 to Addr8	I	Any	[DBName].[VarName]	Variable identifiers from NC Var selector
Unit1 to Unit8	I	Byte		Area address, optional for variable addressing
Column1 to Column8	I	Word		Column address, optional for variable addressing
Line1 to Line8	I	Word		Line address, optional for variable addressing
FM-NCNo	**I**	**Int**	**0, 1, 2**	**0.1 = 1st FM-NC 2 = 2nd FM-NC**
Error	O	Bool		Negative acknowledgment of job or execution of job impossible
NDR	O	Bool		Job has been executed successfully. Data are available
State	I	Word		See error identifiers
RD1 to RD	I/O	Any	P#Mm.n BYTE x... P#DBnr.dbxm.n BYTE x	Target area for read data

© Siemens AG, 2003. All rights reserved

8.5.3 FC 15, 16, 18: POS_AX, PART_AX, SpinCtrl

General

The user can employ functions FCs 15, 16 and 18 to start various positioning tasks. When the function is called, a parameter must be supplied to convey the axis number to be processed. This parameter combined with the segmentation of the PLC/NCK interface ensures that the relevant axis in the correct FM-NC receives its signals and can start the positioning task.

For example, if parameter “AxisNo := 6” when the function is called, the first axis of the second FM-NC is addressed and DB36 contains the corresponding signals of the axis to be processed.

Call example

```
CALL FC 15 (
     Start       := TRUE,
     AxisNo      := 7,        //Axis number (2nd axis of 2nd FM-NC)
     IC          := #inkr,
     Inch        := FALSE,
     HWheelOv    := FALSE,
     Pos         := MD 160,
     FRate       := MD 164,
     InPos       := Q 36.0,
     Activ       := Q 36.1,
     StartErr    := Q 36.2,
     Error       := Q 36.3);
```

Explanation of the formal parameters

The following table shows all formal parameters of the POS_AX function

Signal	Type	Type	Value range	Notes
Start	I	Bool		
AxisNo	**I**	**Byte**	**1–5** **6–10**	**Axis no. of 1st FM-NC** **Axis no. of 2nd FM-NC**
IC	I	Bool		0 = absolute 1 = incremental
Inch	I	Bool		0 = mm 1 = inch
HWheelOv	I	Bool		1 = Handwheel override
Pos	I	Real	± 0.1469368 E –38 to ± 0.1701412 E +39	Position of Linear axis: mm Rotary axis: Grad
FRate	I	Real	± 0.1469368 E –38 to ± 0.1701412 E +39	Feedrate of Linear axis: mm/min Rotary axis: deg./min
InPos	O	Bool		1 = in position
Activ	O	Bool		1 = active
StartErr	O	Bool		Axis cannot be started
Error	O	Bool		Traversing error

© Siemens AG, 2003. All rights reserved

8.5.4 FC 9: ASUB

General

The FC ASUB can be used to trigger any functions in the NC.

In operation with two FM-NC modules, the channel to be processed is entered by means of parameter “ChanNo”. Every FM-NC has its own channel and “ChanNo” is therefore used to address the FM-NC to be processed.

Call example

Start an asynchronous subprogram in channel 1 interrupt no. 1:

```
CALL FC 9 (
     Start      := I 45.7,
     ChanNo     := 1,          // Channel no.=1, i.e. 1st FM-NC
     IntNo      := 1,
     Activ      := M 204.0,
     Done       := M204.1,
     Error      := M 204.4
     StartErr   := M 204.5
     Ref        := FW 200);
```

Explanation of the formal parameters

The following table shows all formal parameters of the ASUB function

Signal	Type	Type	Value range	Notes
Start	I	Bool		
ChanNo	**I**	**Int**	**1, 2**	**1 = 1st FM-NC 2 = 2nd FM-NC**
IntNo	I	Int	1 – 8	Interrupt no.
Activ	O	Bool		1 = active
Done	O	Bool		1 = ASUB terminated
Error	O	Bool		
StartErr	O	Bool		1 = Interrupt number not assigned
Ref	I/O	Word	Global variable (MW, DBW,..)	1 word per FC 9 (for internal use)

© Siemens AG, 2003. All rights reserved

8.6 Basic program error messages

There are three separate error texts and numbers for each FM-NC. The FM-NC-specific error numbers (see table) are displayed on the OP, allowing the user to assign them easily to modules.

Error message with one FM-NC	Error message of second FM-NC
400250 Sign-of-life monitoring NCK	400255 Sign-of-life monitoring NCK**2**
400251 NCK has not powered up	400256 NCK**2** has not powered up
400252 Sign-of-life monitoring	400257 Sign-of-life monitoring NCK**2**

8.7 Memory space

When two FM-NC modules are connected, the main memory increases (depending on number of configured axes) by only a maximum of 1.4KB. This extra space is used by data blocks required to manage data for the second FM-NC i.e. the NC, mode group, CHANNEL and AXIS signal data blocks of the user interface.

Looking at the total basic program, the main memory increases by a maximum of 10% of the total main memory requirement when two FM-NC modules are connected to one PLC with 10 configured axes.

FM-NC system ...	Content of FM-NC-specific DB for data management	Increase in main memory required by specific FM-NC data blocks for data management
With 5 configured axes	DB10, DB11, DB21, DB31–DB35	Total: 1362 bytes
With 10 configured axes	DB10, DB NCKSigDB2 DB11, DB12 DB21, DB22 DB31–DB35, DB36–DB40	Total: 2724 bytes

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Machine Data, Alarms

9

9.1 Machine data

9.1.1 General machine data

10100 MD number	**PLC_CYCLIC_TIMEOUT** Cyclic PLC monitoring time		
Default setting: 0.1	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: s	
Data type: DOUBLE	Applies from SW SW 1		
Significance:	Cyclic PLC monitoring time This machine data defines the maximum monitoring time period after which the PLC must have incremented its sign of life. The graduation scale is implemented internally in interpolation cycles.		

14504 MD number	**MAXNUM_USER_DATA_INT** Number of user data (INT)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 256	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	Maximum quantity of user machine data (displayed in integer format)		
Related to	MD 14506, 14508		

14506 MD number	**MAXNUM_USER_DATA_HEX** Number of user data (HEX)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 256	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	Maximum quantity of user machine data (displayed in HEX format)		
Related to	MD 14504, 14508		

© Siemens AG, 2003. All rights reserved

14508 MD number	MAXNUM_USER_DATA_FLOAT Number of user data (FLOAT)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 32	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	Maximum quantity of user machine data in IEEE format (displayed in floating point format)		
Related to	MD 14504, 14506		

14510 MD number	USER_DATA_INT[n] User data (INT)	
Default setting: 0	Minimum input limit: –32768	Maximum input limit: 32767
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: 1)	Applies from SW 4.1	
Significance:	User machine data (displayed in integer format)	
Related to	MD 14512, 14514	

14512 MD number	USER_DATA_HEX[n] User data (HEX)	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0x0FF
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: 1)	Applies from SW 4.1	
Significance:	User machine data (display in HEX format) 1)	
Related to	MD 14510, 14514	

14514 MD number	USER_DATA_FLOAT[n] User data (FLOAT)	
Default setting: 0	Minimum input limit: –3.40e38	Maximum input limit: 3.40e38
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: 2)	Applies from SW 4.1	
Significance:	User machine data in IEEE format (floating point format)	
Related to	MD 14510, 14512	

1) Machine data in integer/HEX format are treated like a DWORD in the NCK. The data is stored in the NCKPLC interface and can be read from DB20 by the PLC user while the PLC is powering up.

2) Machine data in floating point format are handled as FLOAT (8 byte IEEE) in the NCK. The data is stored in the NCK-PLC interface and can be read from DB 20 by the PLC user while the PLC is powering up.

© Siemens AG, 2003. All rights reserved

9.1.2 Channel-specific machine data

28150 MD number	**MM_NUM_VDIVAR_ELEMENTS** Number of elements for writing PLC variables		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	This MD defines the number of elements that are available to the user for programming PLC variables ($A_OUT_DBx). This number also applies during block searches, but not to synchronized actions. The memory space required by each element is approximately 24 bytes. One element is required to perform each write operation for highspeed sequential writing of PLC variables.		

9.2 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Reference Point Approach (R1)

1	**Brief Description**		**1/R1/1-3**
2	**Detailed Description**		**1/R1/2-5**
	2.1	Referencing with incremental measurement systems	1/R1/2-8
	2.1.1	Phase 1: Travel to reference cam	1/R1/2-9
	2.1.2	Phase 2: Synchronization with the zero mark	1/R1/2-11
	2.1.3	Phase 3: Travel to reference point	1/R1/2-16
	2.1.4	Actual value buffer after power off for incremental encoders	1/R1/2-18
	2.2	Measuring systems with distance-coded reference marks	1/R1/2-20
	2.2.1	Phase 1: Crossing of two reference markers with Synchronization	1/R1/2-21
	2.2.2	Phase 2: Traveling to a fixed destination point	1/R1/2-23
	2.3	Referencing with absolute value encoders	1/R1/2-25
	2.3.1	Adjustment by entering the zero offset in the machine data	1/R1/2-26
	2.3.2	Manual adjustment	1/R1/2-27
	2.3.3	Automatic calibration with probe	1/R1/2-28
	2.3.4	Calibration with BERO	1/R1/2-29
	2.3.5	Validate reference points	1/R1/2-30
	2.3.6	Additional information	1/R1/2-31
	2.3.7	Automatic detection of encoder serial numbers	1/R1/2-32
	2.4	Referencing by means of actual value adjustment	1/R1/2-33
	2.4.1	Actual value compensation to referencing measurement system	1/R1/2-33
	2.4.2	Actual value adjustment to the referenced measuring system (only for 611 D measuring systems)	1/R1/2-34
	2.5	Referencing in follow-up mode (SW 6 and higher)	1/R1/2-35
	2.6	Referencing passive measuring systems (SW 5 and later)	1/R1/2-37
	2.6.1	Function	1/R1/2-37
	2.6.2	Supplementary conditions	1/R1/2-37
	2.6.3	Activation	1/R1/2-38
	2.7	Referencing with cam switch at the drive (SW 5 and later)	1/R1/2-39
	2.7.1	Function	1/R1/2-39

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions 1/R1/3-41

3.1 General supplementary conditions for absolute encoders 1/R1/3-41
3.1.1 Scope of application in 840D/810D 1/R1/3-41
3.1.2 Scope of application in FM-NC 1/R1/3-41

3.2 Calibration 1/R1/3-42

3.3 Special features for large traversing ranges 1/R1/3-43

3.4 Special features of SINUMERIK 840Di 1/R1/3-45

4 Data Descriptions (MD, SD) 1/R1/4-47

4.1 Channel-specific machine data 1/R1/4-47

4.2 Axis/spindle-specific machine data 1/R1/4-48

5 Signal Descriptions 1/R1/5-65

5.1 Channel-specific signals 1/R1/5-65
5.1.1 Signals to channel 1/R1/5-65
5.1.2 Signals from channel 1/R1/5-66

5.2 Axis/spindle-specific signals 1/R1/5-66
5.2.1 Signals to axis/spindle 1/R1/5-66
5.2.2 Signals from axis/spindle 1/R1/5-67

6 Example 1/R1/7-69

7 Data Fields, Lists 1/R1/7-69

7.1 Interface signals 1/R1/7-69

7.2 Machine data 1/R1/7-70

7.3 Alarms 1/R1/7-71

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

Machine zero

The machine zero of a machine tool is defined as the point at which the actual position of the machine axes is “zero”.

Point approach delay

The machine zero is communicated axis-specifically to the NC control via the connected position measurement systems and the “reference point approach (or referencing)” function.

Position measurement systems

The following position measurement systems can be connected to the control:

- Incremental rotary measuring system with at least one zero mark
- Incremental linear measurement system
- Rotary measuring system with distance-coded reference markers (supplied by Heidenhain)
- Linear measurement system with distance-coded reference marks (supplied by Heidenhain)
- Absolute rotary measurement system
- Absolute linear measurement system.

Expansions (SW 5 and higher)

SW 5 has the following extensions:

- Referencing of the passive measuring systems with the option for automatic zero mark search outside the “REF” mode.

  Advantage: Reduction in processing time through omission of a referencing process.

  This function is available only for passive distance-coded measuring systems that have not been selected via the PLC interface signals. The function requires two measuring systems per axis.
  Please refer to Section 2.6 for a detailed description.

- Referencing with cam switch at the drive through extension of the referencing mode MD 34200: ENC_REFP_MODE = 5.
  Please refer to Section 2.7 for a detailed description.

Extensions in SW6 and higher

For distance-coded measuring systems (MD 34200: ENC_REFP_MODE = 3) in conjunction with rotary axes with bit 1 enabled 1 in MD 30455: MISC_FUNCTION_MASK, the reference position is determined between MD 30340: MODULO_RANGE_START and MD 30330: MODULORANGE.

© Siemens AG, 2003. All rights reserved

With the setting
MD 34200: ENC_REFP_MODE = 8
the reference point can be calculated more reliably thanks to redundant evaluation of four reference markers.

■

© Siemens AG, 2003. All rights reserved

Detailed Description 2

Why reference?

The control must be synchronized with the position measurement system of each machine axis so that the control can detect the exact machine zero when it is switched on. This process is known as referencing.

Position measurement systems

The following position measurement systems can be connected to the control:

- Incremental rotary measuring system with at least one zero mark
- Incremental linear measurement system
- Linear measurement system with distance-coded reference marks (supplied by Heidenhain)
- Rotary measuring system with distance-coded reference marks (supplied by Heidenhain)
- Absolute rotary measurement system
- Absolute linear measurement system.

The connected position measurement systems can be subdivided into nine referencing groups MD 34200: ENC_REFP_MODE (referencing mode):

- Referencing where an absolute encoder is available (accept MD 34100: REFP_SET_POS) (SW 2.2 and lower)
- Referencing with incremental measurement systems
- Referencing with measuring systems with distance-coded reference markers through traversal of 2 zero marks
- Referencing with BERO and single signal edge detection
- Referencing with BERO and double signal edge detection (only for FM – NC)
- Referencing where BERO replaces the reference cam
- Referencing where the measurement system is calibrated to an encoder that has already been referenced (SW 3.2 and lower, not NCU570)
- Referencing, BERO with configured approach velocity for spindle applications (from SW 3.6)
- Referencing with measuring systems with distance-coded reference markers through traversal of four zero marks (SW 6.4 and higher).

Cams

A cam has the following tasks:

- Selection of the direction of travel when approaching the zero mark
- Selection of the zero mark, where required.

© Siemens AG, 2003. All rights reserved

Active machine function REF DB11, ...DBX5.2

The reference point approach is performed with the REF machine function activated (IS “active machine function REF” interface signal (DB11, ... DBX5.2)). The REF machine function can be selected in the JOG and MDA modes (“REF machine function” interface signal (DB11, ... DBX1.2)).

Axis-specific referencing

Axis-specific referencing is started separately for each machine axis with the “plus/minus travel keys” interface signal (DB31, ... DBX4.7 and 4.6). Up to 8 axes on the SINUMERIK 840D and up to 5 axes on the SINUMERIK FMNC/810D can be referenced simultaneously. If the machine axes are to be referenced in a particular sequence, the following options are available:

- The operator observes the correct sequence on start-up
- The PLC user program checks the sequence on start-up or defines the sequence itself
- The order is defined in MD 34110: REFP_CYCLE_NR (see channel-specific referencing)

Channel-specific referencing

Channel-specific referencing is started with the “activate referencing” interface signal (DB21, ... DBX1.0). The control acknowledges a successful start with the “referencing active” interface signal (DB21, ... DBX33.0). Each machine axis assigned to the channel can be referenced with channel-specific referencing (this is achieved internally on the control by simulating the plus/minus travel keys). The axis-specific MD 34110: REFP_CYCLE_NR (Axis sequence for channel-specific Referencing) can be used to define the sequence in When all the axes entered in REFP_CYCLE_NR have reached their end points, the “all axes referenced” interface signal (DB21, ... DBX36.2) is enabled.

Referencing by parts program

A machine axis sometimes loses the reference point during execution of a parts program and needs to be referenced in the parts program.

When should a parts program be used for referencing?

Examples of cases when a machine axis could lose the reference point are:

- The axis was parked and is to be reprogrammed
- A rotary axis has exceeded the encoder limit frequency and lost the reference point
- The actual value was set with PRESETON.

Programming

Referencing per parts program is carried out with the command G74.
You can reference several axes simultaneously.
The sequence of the individual phases is identical to axis-specific referencing, except that the process is started with the G74 command instead of the plus/minus travel keys.

© Siemens AG, 2003. All rights reserved

Note

Before using the referencing with parts program function (G74), please read:
References: /PA1/, Programming Guide

Special points to be noted

- Axis-specific referencing and channel-specific referencing are not mutually exclusive.
- IS “Reset” (DB21, ... DBX35.7 / DB11, ... DBX0.7) aborts referencing. All axes which have not reached their reference point at this time are not referenced.
  The “referencing active” interface signal is reset and alarm 20005 is output.
- The following limit monitors are not active for referenced machine axes:
  - Working area limits
  - Software limit switches
  - Protection zones
- The defined axis-specific accelerations are observed at all times during referencing (except when alarms occur).
- The reference point approach can be started only with the direction key for the direction stored in MD 34010: REFP_CAM_DIR_IS_MINUS (exception: The approach can be started with both direction keys in the case of a distance-coded measurement system).

© Siemens AG, 2003. All rights reserved

2.1 Referencing with incremental measurement systems

Referencing sequence

The referencing sequence for incremental measurement systems can be subdivided into three phases:

1. Phase : Travel to reference cam
2. Phase : Synchronization with the zero mark
3. Phase : Travel to reference point

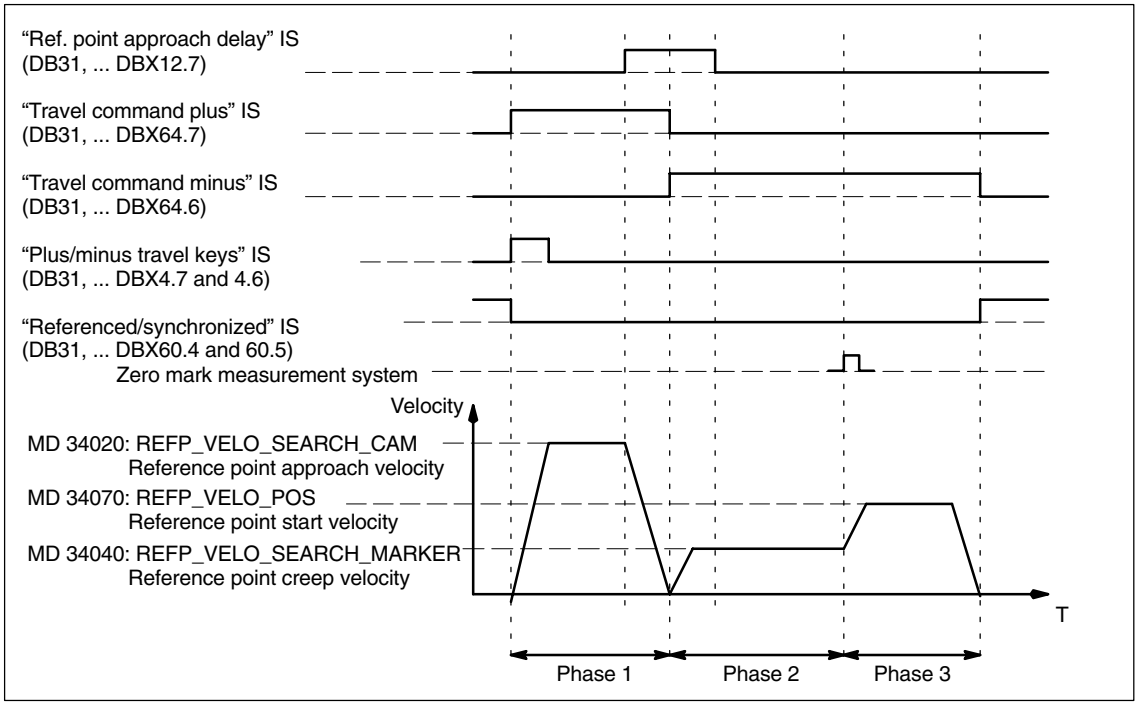


Fig. 2-1 Procedure for referencing an incremental measurement system (example)

© Siemens AG, 2003. All rights reserved

2.1.1 Phase 1: Travel to reference cam

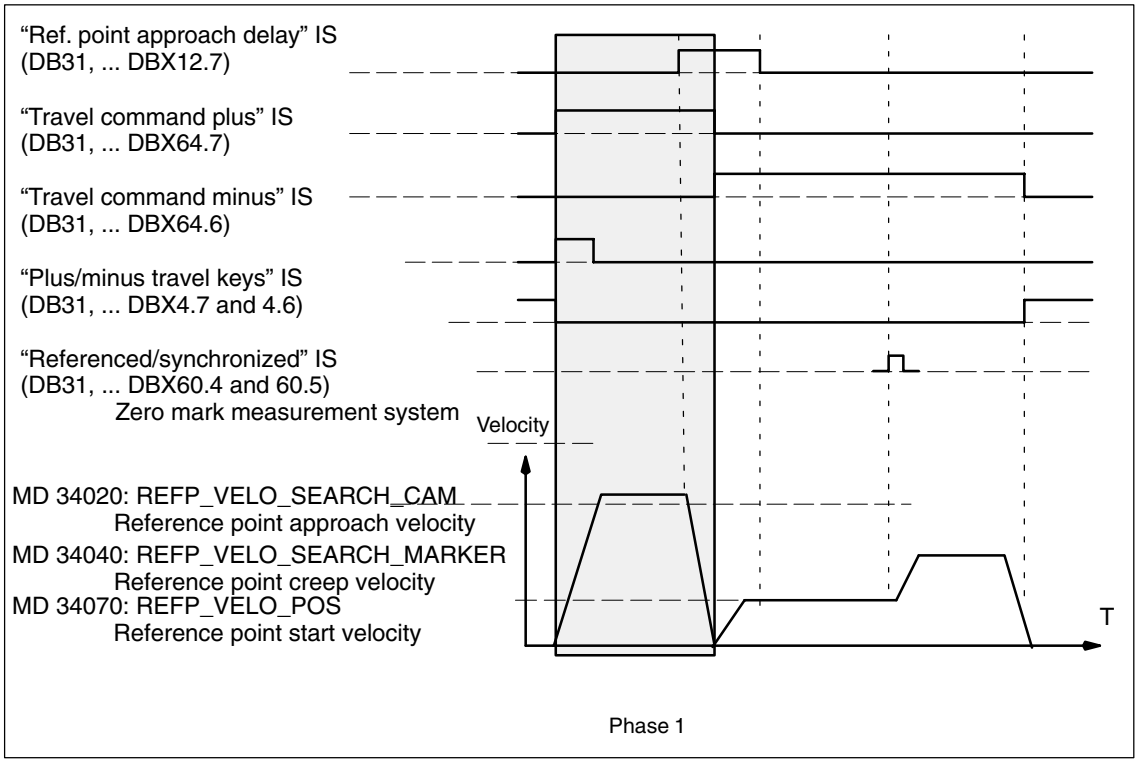


Fig. 2-2 Phase 1: Traversing to the reference cam

Start of phase 1

Phase 1 commences with the Plus/minus travel keys (DB31 to 48, DBX4.7 and 4.6) (only applies to axis-specific referencing).
MD 11300: JOG_INC_MODE_LEVELTRIGGRD (INC/REF in jog/continuous mode) defines whether the plus/minus travel key must remain depressed for phases 1 to 3, or whether a single keystroke is sufficient to terminate phases 1 to 3.
IS “Referenced/synchronized 1 or 2” (DB31, ... DBX60.4 and 60.5) and IS “All reference point axes referenced” (DB 21–30, DBX36.2) are reset.

Phase 1 can commence from three situations:

1. The machine axis is positioned before the reference cam
2. The machine axis is positioned on the reference cam
3. Machine axis has no reference cam (e.g. rotary axis).

Machine axis positioned before the reference cam

The machine axis accelerates to the velocity defined in MD 34020: REFP_VELO_SEARCH_CAM (reference point approach velocity before the reference cam) in the direction specified in MD 34010: REFP_CAM_DIR_IS_MINUS (approach reference point in minus direction).

© Siemens AG, 2003. All rights reserved

The “Reference point approach delay” interface signal (DB31, ... DBX12.7) is used to inform the control when the reference cam has been reached, at which point the machine axis is braked to a stop. During this time, it travels through a further distance calculated by the formula:

$$\text{Min. distance} = \frac{(\text{velocity from MD: REFP_VELO_SEARCH_MARKER})^2}{2 * \text{acceleration from MD: MAX_AX_ACCEL}}$$

This min. distance is required in order to ensure that the machine axis exits the reference cam in phase 2 at the exact reference point creep velocity.
Phase 1 is complete and the process continues with phase 2.

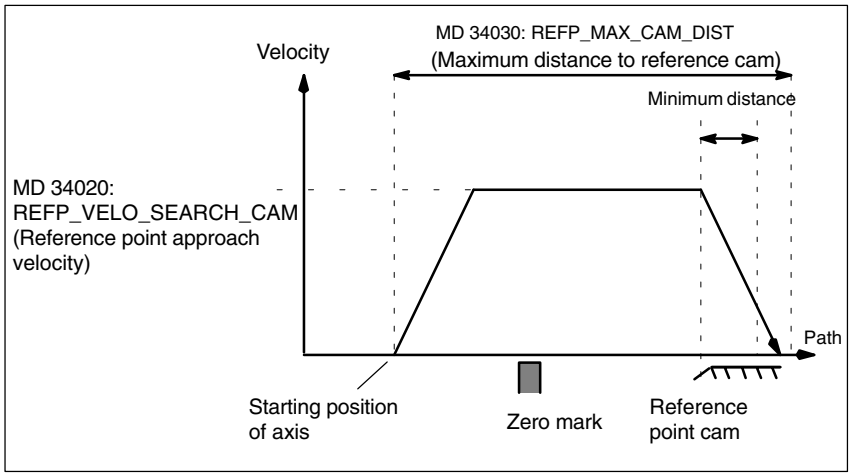


Fig. 2-3 Minimum distance for delay

Machine axis on reference cam

The machine axis remains at the starting position.
Phase 1 is complete and the process continues with phase 2.

Machine axis has no reference cam

Machine axes without reference point cams are identified by entering a “0” in MD 34000: REFP_CAM_IS_ACTIVE (axis with reference cam). This is the case for machine axes with only one zero mark across their entire traversing range or rotary axes with only one zero mark per revolution.
These machine axes remain at their starting position.
Phase 1 is complete and the process continues with phase 2.

© Siemens AG, 2003. All rights reserved

Properties phase 1

- The feedrate override (feedrate switch) is active.
- The feed stop (channel-specific and axis-specific) is active.
- The machine axis can be stopped and restarted with NC stop/NC start.
- If the machine axis travels a maximum distance (defined in MD 34030: REFP_MAX_CAM_DIST from the starting position in the direction of the reference cam without reaching the reference cam ("Reference point approach delay" interface signal (DB31, ... DBX12.7) is reset) and Alarm 20000 "Reference cam not reached" is output.

2.1.2 Phase 2: Synchronization with the zero mark

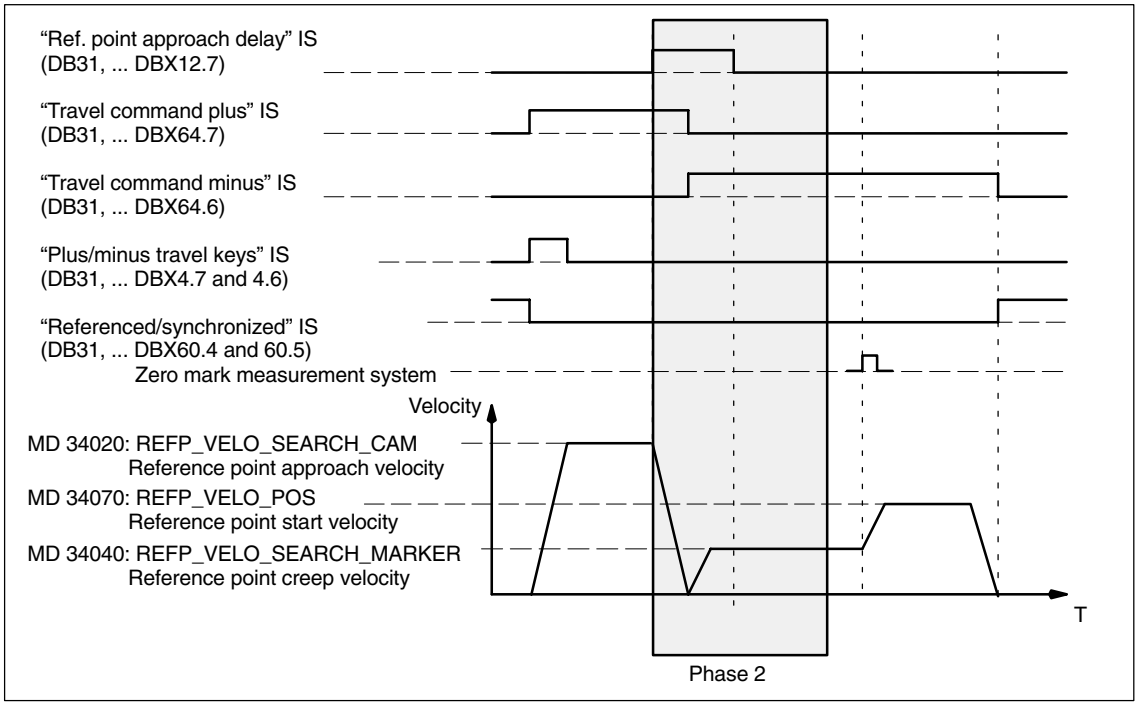


Fig. 2-4 Phase 2: Synchronization with the zero mark

Start of phase 2

Phase 2 is started automatically if phase 1 has been successfully completed (no alarm) and the machine axis is positioned on the reference cam. If the machine axis is not positioned on the reference cam (IS "Reference point approach delay" (DB31, ... DBX12.7) is reset) and Alarm 20001 "No cam signal present" issued. Alarm 20001 can occur if the reference cam is too short and the machine axis travels over it when decelerating in phase 1.

© Siemens AG, 2003. All rights reserved

Types of synchronization

The MD 34050: REFP_SEARCH_MARKER_REVERSE can be used to define the direction of the search for the zero mark:

- Synchronization with falling reference cam signal edge
- Synchronization with rising reference cam signal edge

If the actual speed has not reached the setpoint for phase 2 when a cam is approached (± the tolerance from MD 35150: SPIND_DES_VELO_TOL), phase 1 is started again. This can occur if the axis was precisely over the cam at the start.

Synchronization with falling reference cam signal edge

The machine axis accelerates to the velocity defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity) specified in MD 34010: REFP_CAM_DIR_IS_MINUS (approach reference point in minus direction). If the reference cam is existed (IS “Reference point approach delay” (DB31, ... DBX12.7) is reset), the control synchronizes with the first zero mark. The machine axis continues to traverse at constant speed. Phase 2 is considered completed and the movement is continued with phase 3.

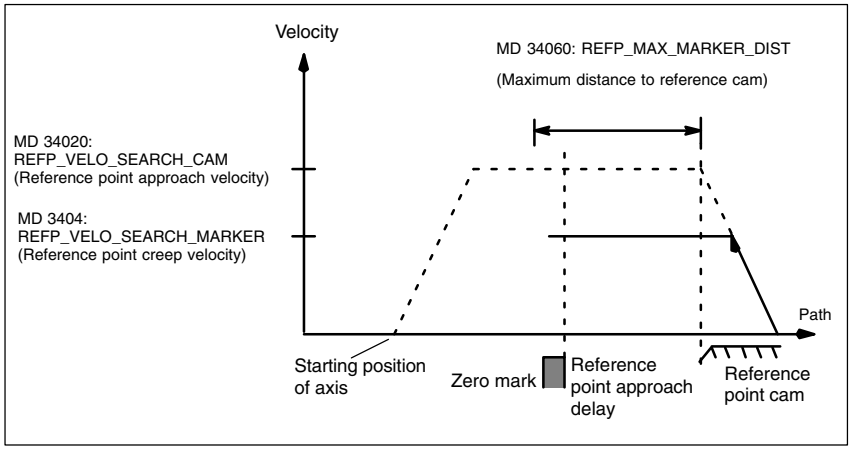


Fig. 2-5 Synchronization with falling reference cam signal edge

Synchronization with rising reference cam signal edge

The machine axis accelerates to the velocity defined in MD 34020: REFP_VELO_SEARCH_CAM (reference point approach velocity) in the opposite direction to that specified in MD 34010: REFP_CAM_DIR_IS_MINUS (approach reference point in minus direction). When the axis leaves the reference cam (“Reference point approach delay” interface signal is reset), the machine axis decelerates to a halt and accelerates in the opposite direction towards the reference cam at the velocity defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity). When the reference cam is reached (“Reference point approach delay” interface signal is enabled), the control is synchronized with the first zero mark. The machine axis continues traveling at a constant velocity.

© Siemens AG, 2003. All rights reserved

If MD 34050: REFP_SEARCH_MARKER_REVERSE is enabled, the cam is also approached at the velocity defined in REFP_VELO_SEARCH_CAM. This setting is useful for longer cams.
Phase 2 is complete and the process continues with phase 3.

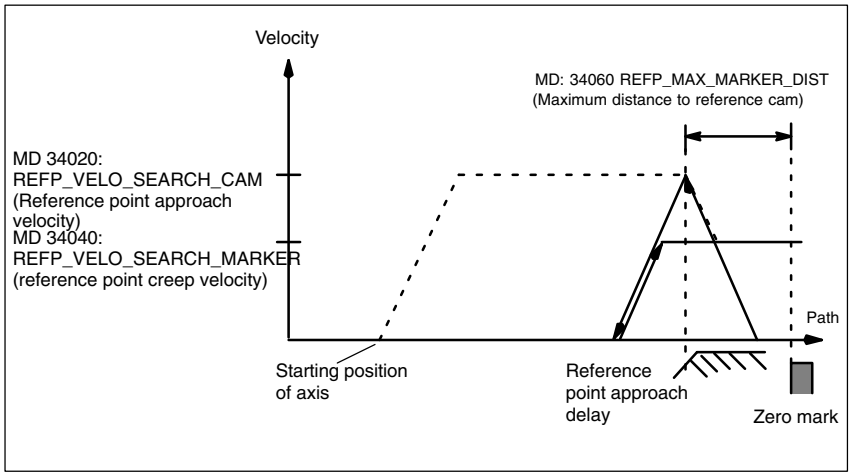


Fig. 2-6 Synchronization with rising reference cam signal edge

Electronic reference cam offset

If (depending on the type of synchronization “rising” or “falling reference cam edge”) the reference cam signal occurs, the zero mark search is not started immediately; rather, after the distance in MD 34092: REFP_CAM_SHIFT. The reproducibility of the zero mark search can thus be ensured even in the case of temperature-dependent expansion of the reference cam through defined selection of a zero mark.

As the reference cam offset is calculated by the control in the interpolation cycle, the actual reference cam offset is
– at least REFP_CAM_SHIFT and
– at most REFP_CAM_SHIFT+
(MD 34040: REFP_VELO_SEARCH_MARKER*interpolation cycle)

The reference cam offset acts in the direction of zero mark search.
The reference cam offset is active only if the cam MD 34000: REFP_CAM_IS_ACTIVE=1 is present.

© Siemens AG, 2003. All rights reserved

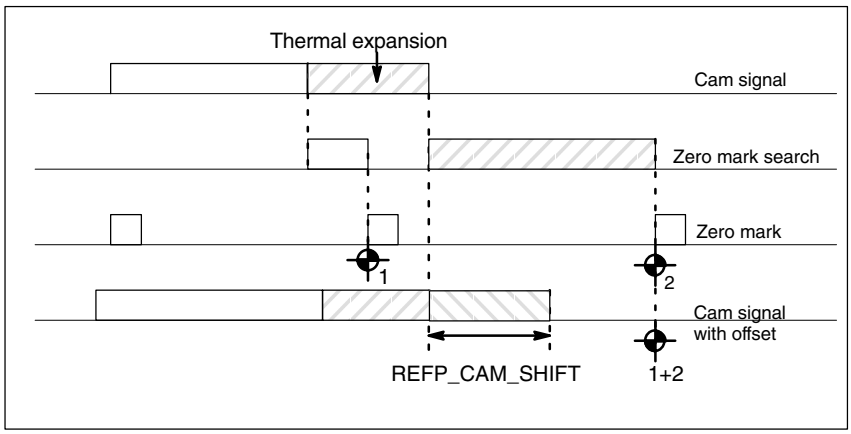


Fig. 2-7 Electronic reference cam offset

Reference cam adjustment

Where the position measurement system has several zero marks repeated in cycles (such as in an incremental rotary indirect position measurement system connected to the motor), the reference cam must be aligned precisely. If the reference cam is not precisely aligned, an incorrect zero mark may be evaluated, or alarm 20002 “Zero mark missing” output.

The reference cam is reported from the PLC to the NCK by the “Reference point approach delay” interface signal. The following factors influence the response time of the control (NCK) when detecting the reference cam:

- Switching accuracy of the reference cam switch
- Delay of the reference cam switch (NC contact)
- Delay at the PLC input
- PLC cycle time
- Cycle time for updating the NCK/PLC interface
- Interpolation cycle
- Position control cycle.

Practice has shown that the signal edge of the reference cam, which is required for synchronization, is aligned between two zero marks.

MD 34094: REFP_CAM_MARKER_DIST (SW 6.3) returns the distance between the zero mark and the cam signal. The readonly value can be used as a guide for setting the electronic reference cam offset.

© Siemens AG, 2003. All rights reserved



Warning

If the reference cam is not aligned precisely, an incorrect zero mark can be evaluated. In this case, the control assumes an incorrect machine origin and moves the axes to incorrect positions. Software limit switches, protected areas and working area limitations act on incorrect positions and are therefore incapable of protecting the machine. The difference is equivalent to ± one encoder revolution in each case.

Properties phase 2

- The feed override (of the feed override switch) is not active.
  The setting is defined as 100%, at 0% the procedure is canceled.
- The feed stop (channel-specific and axis-specific) is active, the axis stops and alarm 20005 is output.
- The machine axis cannot be stopped and restarted with NC stop/NC start.
- If, after leaving the reference cam (IS: “Reference point approach delay” is reset) the machine axis travels a path defined in MD 34060: REFP_MAX_MARKER_DIST (maximum distance to the reference mark) without detecting the zero mark, the axis stops and alarm 20002 “Zero mark missing” is output.

© Siemens AG, 2003. All rights reserved

2.1.3 Phase 3: Travel to reference point

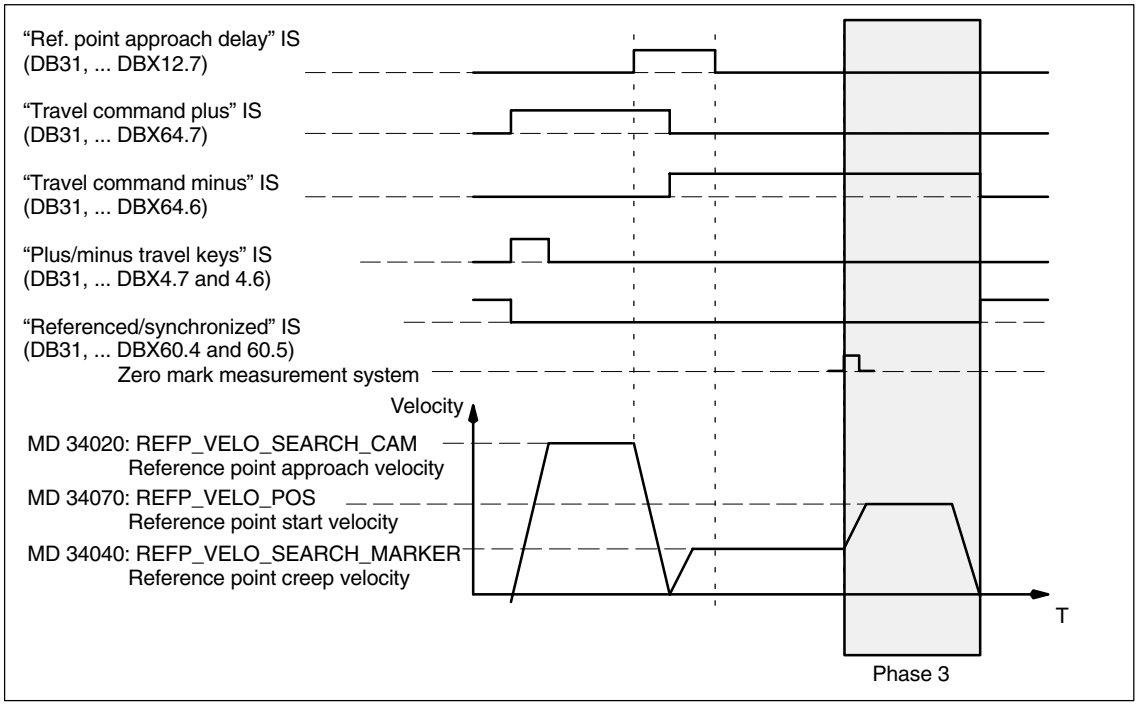


Fig. 2-8 Phase 3: Travel to reference point

Start of phase 3

Phase 3 is automatically started when phase 2 has been successfully completed (without alarm). Since the machine axis travels at the reference creep velocity at the end of phase 2, the transition to phase 3 is performed on the fly.

Depending on the leading sign of the distance, the machine axis accelerates to the velocity defined in MD 34070: REFP_VELO_POS (reference point start velocity) and travels a distance calculated from the addition of MD 34080: REFP_MOVE_DIST (reference point distance/destination for distance-coded system) and MD 34090: REFP_MOVE_DIST_CORR (reference point offset). This distance precisely matches the distance between the zero mark detected in phase 2 and the reference point.

© Siemens AG, 2003. All rights reserved

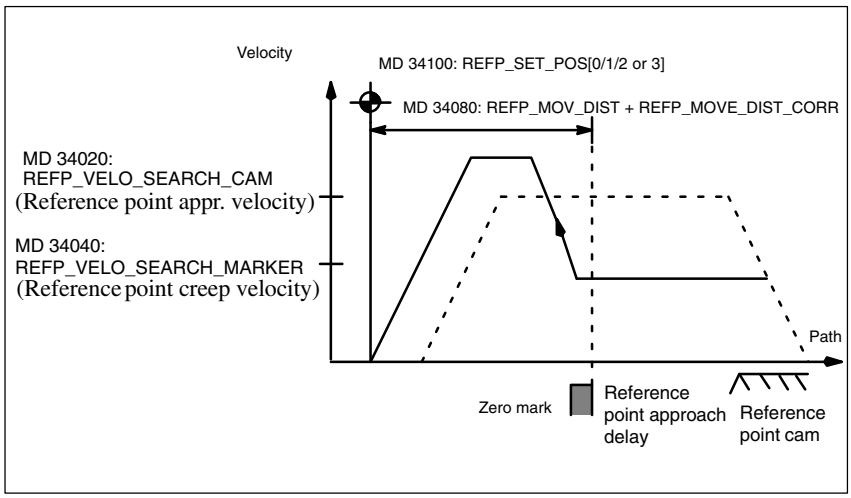


Fig. 2-9 Reference point position

Reference point reached

If the machine axis has arrived at the referenced point (the axis is stationary), the reference point value selected by the “Reference point value 1 to 4” (DB31, ... DBX2.4 –2.7) interface signal from MD 34100: REFP_SET_POS (reference value) is accepted by the control as the new reference position (the value which is selected at the “time of the Rising signal edge”). The machine axis is now referenced and the “Referenced/synchronized 1 or 2” interface signal (DB31, ... DBX60.4 or 60.5, depending on which position measurement system is active) is enabled.

Properties phase 3

- The feedrate override (feedrate switch) is active.
- The feed stop (channel-specific and axis-specific) is active.
- The machine axis can be stopped and restarted with NC stop/NC start.

Special feature phase 3

If the total distance from MD 34080: REFP_MOVE_DIST (reference point distance/destination for distance-coded system) and MD 34090: Reference point_MOVE_DIST_CORR (reference point offset point) smaller than the braking distance of the machine axis from the reference point creep velocity to stop, the reference point is approached from the opposite direction.

© Siemens AG, 2003. All rights reserved

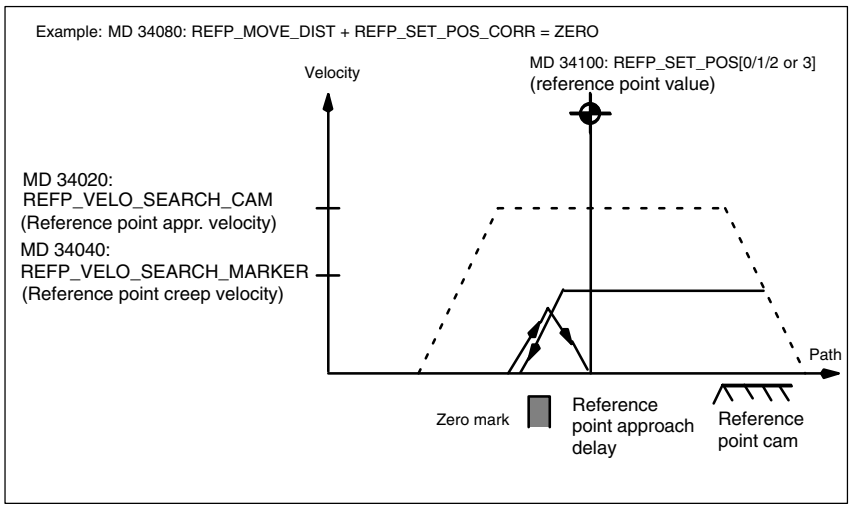


Fig. 2-10 Reference point distance+reference point offset < braking distance

2.1.4 Actual value buffer after power off for incremental encoders

Purpose

Up to now, axes with incremental encoders had to be referenced again after each POWER ON or each time the control was parked.

With SW 4.1 and higher, it is possible to continue to operate a conventional machine tool with the original position information without having to explicitly reference it again after POWER OFF/POWER ON.

Conditions

The condition for proper continued operation of the referenced axes after POWER OFF/POWER ON is that the axes in question have not been moved in the meantime.

Appropriate holding brakes can be used to achieve this.

Note

- When using this function, it should be remembered that possible position errors can accumulate after several POWER OFF/POWER ON operations if no machine design precautions are taken to prevent this.
- The functionality is permanently linked to the “exact stop fine” axis signal. Axes or spindles which do not support this signal cannot use this functionality.

© Siemens AG, 2003. All rights reserved

Automatic referencing

Preconditions for “automatic” referencing after POWER OFF/ON:

1. The encoder used to control the axis is (mechanically) calibrated.
2. “Automatic” referencing has been entered in MD 34210 ENC_REFP_STATE=1 or 2.
3. The axis has been referenced with the encoder. Here, the internal value in MD 34210: ENC_REFP_STATE is changed from 1 to 2.
4. After POWER OFF/POWER ON MD 34210: ENC_REFP_STATE=2. This means that the axis was referenced before POWER OFF/POWER ON and was stationary, i.e. a valid old position is available internally on the control.

Note

It is the responsibility of the user to ensure that there is no non-calibrated motion when the encoder is deactivated.

When the encoder is switched on, the NC (controller, interpolator and block processing) is synchronized with the old absolute value stored. Axis motion is disabled internally until synchronization is complete, spindles can continue to operate.

Powerup response

- The axis has been referenced and is at a standstill (MD 34210: ENC_REFP_STATE=2)
  - In contrast to the conventional power-up response of axes with incremental encoder, the “referenced” interface signal appears automatically. The axis position is not initialized with “0”, but starts with the absolute value stored.
  - An explicit reference point approach is not necessary, but presents no problem if required.
- The axis has not been referenced and/or is not at a standstill (MD 34210: ENC_REFP_STATE=1)
  - An explicit reference point approach is necessary. The axis is initialized with start position “0”. The “axis is referenced” status does **not** appear.
  - The MD 34210: ENC_REFP_STATE=2 is set with the next reference point approach internally in the control (for standstill/exact stop). During traversing (no standstill/exact stop), MD 34210: ENC_REFP_STATE returns briefly to “1”.

Note

Spindles that do not supply exact stop signals cannot reach the status MD 34210: ENC_REFP_STATE=2, i.e. the “stored actual value” is not effective here.

© Siemens AG, 2003. All rights reserved

2.2 Measuring systems with distance-coded reference marks

Basic concept

In the case of measurement systems with distance-coded reference marks, it is not necessary to evaluate a reference cam or approach a defined reference point in order to reference the machine axis. Measurement systems of this type consist of a line grid and a reference mark track running parallel to this. The distance between two consecutive reference marks is defined variably, so that the absolute position of the machine axis can be determined from the distance.
Requirement: MD 34200: ENC_REFP_MODE = 3 or

MD 34200: ENC_REFP_MODE = 8 (see Note below)

In the case of linear measuring systems with distance-coded reference markers (Heidenhain), the absolute position value is available only after a maximum distance of 20 mm when MD 34200: ENC_REFP_MODE = 3 (i.e. once two reference markers have been traversed) regardless of whether the machine axis travels in the positive or minus direction.

For rotary measuring systems with distance-coded reference marks, the absolute position value is available after a distance of only max. 20 º (i.e. when two reference marks have been crossed).

Note

The effect of MD 34200: ENC_REFP_MODE = 8 is equivalent to MD 34200: ENC_REFP_MODE = 3, but 4 reference markers instead of 2 are evaluated. As a result, the position value is available only after a maximum distance of 40 mm. For further details, see the end of this subsection.

Referencing sequence

The referencing sequence can be subdivided into 2 phases:

Phase 1: Crossing of two reference markers with synchronization
Phase 2: Traveling to a fixed destination point

© Siemens AG, 2003. All rights reserved

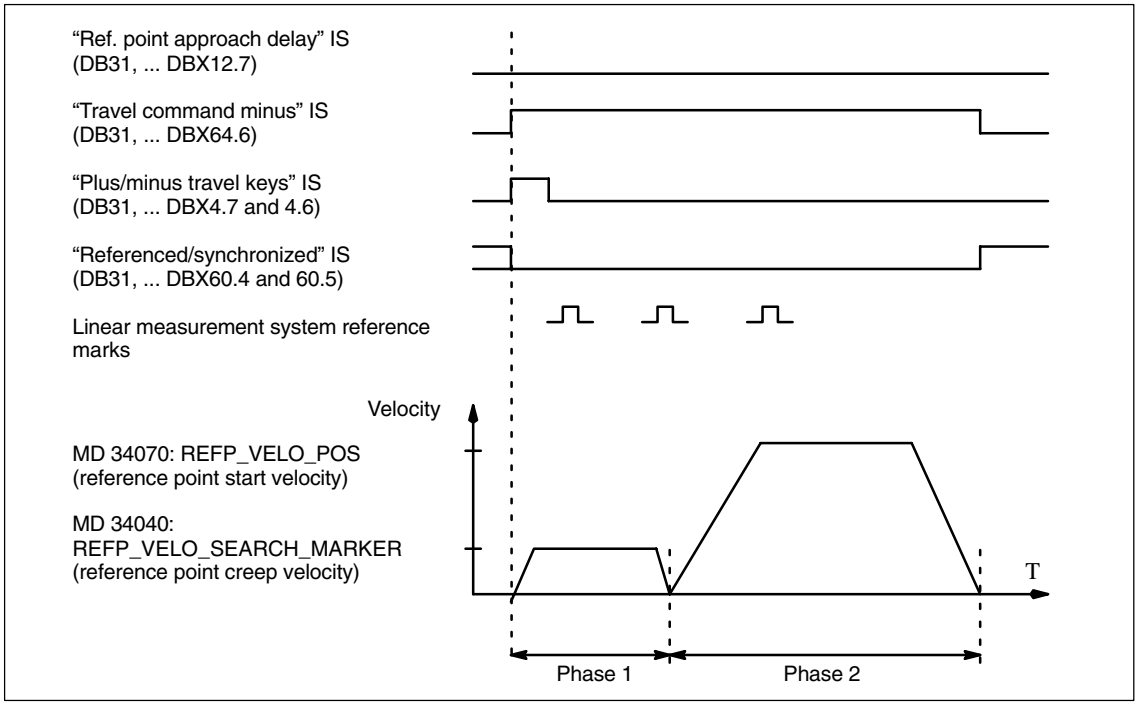


Fig. 2-11 distance-coded reference marks

2.2.1 Phase 1: Crossing of two reference markers with Synchronization

Start of phase 1

Phase 1 commences with the "Plus/minus travel keys" (DB31, ... DBX4.7 and 4.6) (only applies to axis-specific referencing). The MD 11300: JOG_INC_MODE_LEVELTRIGGRD (INC/REF in jog/continuous mode) defines whether the plus/minus travel key must remain depressed for phases 1 to 2, or whether a single keystroke of the travel key is sufficient to terminate phases 1 to 2.
IS "Referenced/synchronized 1 or 2" 1 (DB31, ... DBX60.4 and 60.5) and IS "All axes referenced" (DB21, ... DBX36.2) are reset.

© Siemens AG, 2003. All rights reserved

Reference cam

The linear measurement system with distance-coded reference marks does not require a reference cam for referencing. However, it is necessary to install a reference cam before the end of the traversing range of a machine axis, in order to provide a safeguard in case two consecutive reference marks are not crossed during channel-specific referencing and referencing in the parts program (G74). The purpose of the reference cam is to select the opposite direction of travel to that specified in MD 34010: REFP_CAM_DIR_IS_MINUS (approach reference point in minus direction) when the machine is located on the reference cam at the start of phase 1 (independent of the traversing key plus/minus pressed). The axis must come to a standstill from the reference point cam. The result is to force the axis to travel off the cam, a safeguard required to prevent the machine axis from reaching the traversing range limit (hardware limit switch or EMERGENCY STOP) before it detects two reference marks.

If the axis travels onto the cam before detecting the second zero mark, it reverses immediately and starts searching in the opposite direction.

Without reference cam

The machine axis accelerates to the velocity defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the direction of the travel key (plus or minus), crosses two reference marks (synchronization) and stops.

With reference cam

If the machine axis is not positioned on the reference cam, it accelerated to the velocity specified in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the direction of the travel key (plus or minus), crosses two reference marks (synchronization) and stops (phase 2).
When the machine axis is positioned on the reference cam, it accelerates to the velocity defined in MD: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the opposite direction to that defined in MD 34010: REFP_CAM_DIR_IS_MINUS (approach reference point in minus direction), regardless of whether the plus or minus travel key is depressed, crosses two reference marks (synchronization) and stops (phase 2).

Reference mark distance

If the control detects that the distance between two reference marks exceeds 2 * MD 34300: ENC_REFP_MARKER_DIST (reference mark distance), and is therefore incorrect, the machine axis accelerates to half the velocity defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the opposite direction to the depressed travel key (plus or minus) and crosses two reference marks. If the control detects that the distance between two reference marks exceeds 2 * MD 34300: ENC_REFP_MARKER_DIST (reference mark distance), the axis stops and alarm 20003 “Error in measurement system” is output.
Phase 1 is complete and the process continues with phase 2.

Properties phase 1

If the machine axis travels a distance defined in MD 34060: REFP_MAX_MARKER_DIST (maximum path to reference mark) from the starting position without crossing two reference marks, the axis stops and alarm 20004 “Reference mark missing” is output.

© Siemens AG, 2003. All rights reserved

2.2.2 Phase 2: Traveling to a fixed destination point

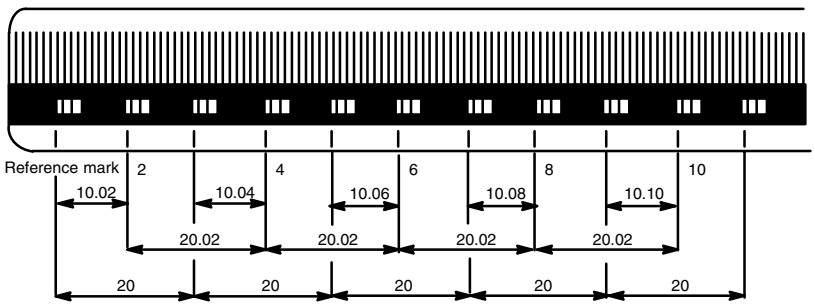
Start of phase 2

Phase 2 is automatically started when phase 1 has been successfully completed (without alarm).

Set reference point setting

The distance between two consecutive reference marks is defined variably, so that it is possible to accurately identify the reference mark and the actual position on the linear measurement system (in relation to the first reference mark on the linear measurement system).

The actual position on the linear measurement system is assigned to a precise machine axis position (in relation to the machine origin) in order to set the reference point. This is achieved by entering the absolute offset between the machine zero and the position of the 1st reference mark on the linear measurement system in MD 34090: MD 34320: ENC_INVERS (linear measurement system inverse to machine system) must also be set to define whether or not the linear measurement system is installed inversely to the machine system.



DIADUR graduated glass scale with distance-coded reference marks
Dimensions in mm for 20mm graduation cycle

Fig. 2-12 Linear graduated glass scale

Rotary system

The graduations shown in the top diagram must be interpreted as being on the circumference of a measuring disk with divisions in degrees instead of length measurements. Otherwise, the principle of measurement is the same as for linear measurement systems.

Determining the absolute offset

The absolute offset between the machine origin and the position of the first reference mark on the linear measurement system can be entered directly in (MD 34090: reference mark on the linear measurement system in MD 34090: REFP_MOVE_DIST_CORR (this can be problematic, as the offset is often difficult to measure), or the absolute offset can be calculated from the following formula with the machine axis at any position (the position should allow accurate measurement of the actual machine position in relation to the machine origin using a laser interferometer):

© Siemens AG, 2003. All rights reserved

Linear measurement system noninverse to machine system:

Absolute offset = actual machine position + actual position of linear measurement system

Linear measurement system inverse to machine system:

Absolute offset = actual machine position – actual position of linear measurement system

The actual position of the linear measurement system can be read on the NC screen under actual position when two consecutive reference marks have been crossed (synchronized). In this case, the MD: REFP_MOVE_DIST_CORR must contain the value zero.



Warning

When the absolute offset has been calculated and the value entered in REFP_MOVE_DIST_CORR, the position measurement system must be referenced again.

Shortening phase 2

Depends on MD 34330: STOP_AT_ABS_MARKER (linear measurement system with no destination point). The following settings are possible:

- Approach with no destination point with distance-coded linear measurement systems (shortened)
- Approach with destination point with distance-coded linear measurement systems (normal)

Approach with no destination point

The machine axis remains stationary. The machine axis is now referenced and the “Referenced/synchronized 1 or 2” interface signal (DB31, ... DBX60.4 or 60.5, depending on which position measurement system is active) is enabled.

Approach with destination point

The machine axis accelerates to the velocity defined in MD 34070: REFP_VELO_POS (reference point start velocity) and approaches the destination point defined in MD 34100: REFP_SET_POS (reference point value). The machine axis is now referenced and the “Referenced/synchronized 1 or 2” interface signal (depending on which position measurement system is active) is enabled.

Properties phase 2

- The feedrate override (feedrate switch) is active.
- The feed stop (channel-specific and axis-specific) is active.
- The machine axis can be stopped and restarted with NC stop/NC start.

© Siemens AG, 2003. All rights reserved

Properties

The position of rotary, distance-coded measuring systems can be determined uniquely (absolutely) only in relation to a rotation. Depending on the mechanical mounting of the encoder, the overtravel of the absolute position in the hardware does not always coincide with the traversing range of the rotary axis.

Automatic adaptation to modulo range

For modulo rotary axes, the position calculated during referencing is automatically adapted to the range defined via

MD 30330: MODULO_RANGE and

MD 30340: MODULO_RANGE_START

for modulo rotary axes.

For rotary axes that are not defined as modulo axes (MD 30310: ROT_IS_MODULO = 0) up to SW 6.2, certain encoder mounting requirements result (encoder overflow outside the traversing range of the rotary axis). In SW 6.3 and higher, the automatic adaptation function is also available above the specified theoretical modulo range if bit 1 is enabled

- in MD 30455: MISC_FUNCTION_MASK (definition of reference point position of rotary, distance-coded encoder within the configured modulo range) and a (fictitious) modulo range is defined with
- MD 30330: MODULO_RANGE and
- MD 30340: MODULO_RANGE_START

SW 6.4 and higher

For distance-coded measuring systems, the reference point can be calculated on the basis of four instead of two (as in earlier versions) reference markers with the setting MD 34200: ENC_REFP_MODE = 8ENC_REFP_MODE = 8.

A plausibility check can be performed through the evaluation of redundant zero mark information of a second zero mark pair, thereby increasing the reliability of the referencing result.

The traversing path for the referencing operation is twice as long as the path used for MD 34200: ENC_REFP_MODE = 3. The time needed to reference the axis increases accordingly.

2.3 Referencing with absolute value encoders

Preconditions

An axis with absolute value encoder is referenced automatically when the control is switched on if the system detects that the relevant axis is already calibrated. This transfer of the absolute value takes place without any axis motion, e.g. on POWER ON. Two conditions must be fulfilled before an axis can be automatically referenced:

- The axis has an absolute value encoder with which the position control operates
- The absolute value encoder is calibrated (MD 34210: ENC_REFP_STATE[n] = 2)

© Siemens AG, 2003. All rights reserved

Calibration

In the case of axes with absolute value encoders, the measurement system is not synchronized by approaching a reference cam. It is calibrated instead. For this purpose, the actual value of the absolute value encoder is set once during start-up and transferred to the control. The absolute encoder can be adjusted as follows:

- Adjustment by entering the zero offset in the machine data
- Manual adjustment
- Automatic calibration with probe
- Calibration with BERO.

2.3.1 Adjustment by entering the zero offset in the machine data

General procedure

The zero offset between displayed and real machine position is determined and entered in the machine data MD 34090: REFP_MOVE_DIST_CORR.

Chronological procedure

1. Measure the reference point offset, e.g. traverse the machine to a known position and read out the displayed position.
2. MD 34090: REFP_MOVE_DIST_CORR to the offset value determined.
3. MD 34210: ENC_REFP_STATE = 2. This adjustment status for automatic referencing will come into effect when the encoder is activated next time.
4. POWER ON of the control system and check the machine position or carry out POWER ON as a preventive action. The new value entered in MD 34090: REFP_MOVE_DIST_CORR will come into effect for ever.

Note

If operation of the axis is continued after such a calibration without a POWER OFF/POWER ON, the previous value, which is still in use internally, may be stored in the MD again (in the course of automatic overflow correction), thus overwriting the new entry.
If a backlash compensation is parameterized for this encoder, make sure that axes with a backlash are at the correct edge when adjusting.

© Siemens AG, 2003. All rights reserved

2.3.2 Manual adjustment

General procedure

Move the axis to be adjusted to a defined position to set the resulting actual value.

Chronological procedure

1. MD 34200: ENC_REFP_MODE to 0 and activate by means of a POWER ON. (ENC_REFP_MODE = 0 means that the actual value of the axis is set once)
2. Traverse axis manually in JOG mode to a known position. The direction in which the position is approached must match the direction stored in MD 34010: REFP_CAM_DIR_IS_MINUS (0 = positive direction, 1 = negative direction).

Note

This known position must always be approached at a low velocity and from a defined direction so that it is not falsified by any backlash present in the drive train.

---

3. Enter the actual value corresponding to the approached position in MD 34100: REFP_SET_POS. This value may be a specified design value (e.g. fixed stop) or can now be determined with a measuring instrument.
4. MD 34210: Sset ENC_REFP_STATE to 1 to enable the “calibration” function.
5. The modified machine data become effective after a RESET.
6. Switch to JOG-REF mode.
7. When you press the traversing key used in step 2, the current offset is entered in MD 34090: REFP_MOVE_DIST_CORR and ENC_REFP_STATE changes to “2”, i.e. the axis is effectively calibrated.

Note

The axis does not move when the correct traversing key is actuated! The value entered in REFP_SET_POS appears in the actual value display of the axis position.

---

8. Exit JOG-REF operating mode. The calibration operation for this axis is now complete.

Calibration of multiple axes

To save time, it possible to calibrate multiple axes by moving all or a number axes (providing the machine design permits) to their calibration point (points 15) and then changing to JOG-REF (point 6). Each axis is then calibrated individually (point 7).

© Siemens AG, 2003. All rights reserved

2.3.3 Automatic calibration with probe

General procedure

The axis to be calibrated is caused to traverse by the NC program. The axis is stopped in response to a signal from the probe and the appropriate actual value then set.

Note

When calibrating absolute value encoders automatically with a probe, particular care must be taken to avoid any risk of collisions on the machine.

NC program

The user must generate the NC program for calibrating absolute value encoders, taking the predefined machine specifications into account.

The NC program for adjusting the absolute encoders with sensor must perform the points listed below for each axis in this order:

1. Approach calibration position of axis which is detected from the probe response. The position must be approached several times from the same direction, but at a velocity which is gradually reduced on each approach, to ensure that the measured value $AA_IM obtained is as accurate as possible.
2. Calculation of the difference between MD 34100: REFP_SET_POS and $AA_IM by the NC program.
3. Entry of the difference in MD 34090: REFP_MOVE_DIST_CORR by the NC program.
4. MD 34210: ENC_REFP_STATE=2, set by the NC program.

Chronological procedure

1. MD 20700: REFP_NC_START_LOCK=0 (enable program start for nonreferenced axes as well).
2. MD 34100: REFP_SET_POS to the correct actual positions for the calibration points on the machine for all axes to be calibrated.
3. Changed MD are activated by means of a RESET operation.
4. Start NC program.
5. MD 20700: REFP_NC_START_LOCK=1 (disable program start for nonreferenced axes).
6. Perform a POWER ON. The new values are entered in MD 34090: REFP_MOVE_DIST_CORR will come into effect for ever.

Note

If operation of the axis is continued after such a calibration without a POWER OFF/POWER ON, the previous value, which is still in use internally, may be stored in the MD again (in the course of automatic overflow correction), thus overwriting the new entry.

---

7. Check/verify the result of calibration.

© Siemens AG, 2003. All rights reserved

2.3.4 Calibration with BERO

Preconditions

In order to adjust the absolute value encoder with a BERO, the BERO can be mounted either permanently or temporarily for the purpose of calibration.

After the encoder has been calibrated, it remains referenced provided the appropriate “Position measurement system 1/2” interface signal is active. After the measurement system has been connected, the “Axis referenced” interface signal is reenabled.

Chronological procedure

1. MD 34200: ENC_REFP_MODE=2 (“Referencing with BERO” selected).
2. MD 34100: REFP_SET_POS according to the mechanically defined position of the BERO.
3. Initiate reference point approach. Reference point approach can be started either by the user in JOGREF or via G74 in program mode.
4. The BERO can now be removed again if it is not required permanently. In this case, ENC_REFP_MODE must be set to 0 (“Referencing with absolute value encoder”).

Connecting the measurement system

The synchronism between the axis and the measurement system is checked when switched on. If the axis is already calibrated and synchronized, no further calibration is required.

Restriction: If the spindle is operated above the limit frequency, the limit MD 36302: ENC_FREQ_LIMIT_LOW must be set according to the lowest value of the absolute track.

© Siemens AG, 2003. All rights reserved

2.3.5 Validate reference points

Preconditions

When using the function Buffer “actual value after POWER OFF” (see Subsection 2.1.4), axes which have no absolute value encoder remain referenced while the power in off, provided they were in the exact stop state when the machine was switched off.

Since this condition cannot always be fulfilled, the axis must be referenced again on POWER ON if the position is lost. For axes with no reference cam, it is sufficient to set the reference position to any point within the traversing area. This is achieved by traversing the axis to a known position, setting the reference point, and indicating that the axis is referenced.

Note

A zero mark of the measuring system is not used to set the reference point!

Chronological procedure

1. MD 34200: ENC_REFP_MODE to 0 and activate by means of a POWER ON. You can now set the reference point value for this axis.
2. Traverse axis manually in JOG mode to a known position. The direction in which the position is approached must match the direction stored in MD 34010: REFP_CAM_DIR_IS_MINUS.
   (0 = positive direction, 1 = negative direction).

Note

This known position must always be approached at a low velocity and from a defined direction so that it is not falsified by any backlash present in the drive train.

---

3. Enter the actual value corresponding to the approached position in MD 34100: REFP_SET_POS. This value may be a specified design value (e.g. fixed stop) or can now be determined with a measuring instrument.
4. Switch to JOG-REF mode.
5. When you press the traversing key used in step 2, the current offset is entered in MD 34090: REFP_MOVE_DIST_CORR.

Note

The axis does not move when the correct traversing key is actuated! The value entered in MD 34100: REFP_SET_POS appears in the actual value display of the axis position.

---

6. The axis is referenced.

© Siemens AG, 2003. All rights reserved

2.3.6 Additional information

Other reference variants

Referencing and calibration to zero marks, distance-coded marks and BERO with doubleedge evaluation is **no**t supported in conjunction with the EQN 1325, LC181 and encoders of similar design. Incorrect parameterization of MD 34200: ENC_REFP_MODE, alarm 26015 is output in SW 3.6 and higher.

Axes with two encoders

If the axis has two encoders (MD 30200: NUM_ENCS=2) and at least one of these is an absolute encoder, MD 34102: REFP_SYNC_ENCS=1 must be enabled (see Subsection 2.4.1).

Special features when synchronizing with BERO

The position falsification caused by the signal delay with BERO can be corrected internally in the NC by entering a signal runtime compensation.

The MD 31122: BERO_DELAY_TIME_PLUS or MD 31123: BERO_DELAY_TIME_MINUS together with the setting MD 34200: ENC_REFP_MODE = 2 or 7.

- Setting MD 34200: ENC_REFP_MODE = 7 means that the position synchronization is carried out only for a fixed velocity/speed defined in MD 34040: REFP_VELO_SEARCH_MARKER.

  The velocity set in MD 34040 is also effective during referencing in JOG–REF mode and for the parts program with G74.

- Setting MD 34200: ENC_REFP_MODE = 2 carries out position synchronization without specifying a particular velocity/speed.

Note

Compensation of the signal runtime by the NC requires the use of type 611D drives.

The signal run times are preset on delivery so that the contents generally do not have to be changed.

© Siemens AG, 2003. All rights reserved

2.3.7 Automatic detection of encoder serial numbers

Detection of serial numbers SW 5.3 and higher

On start-up, the control system will detect all encoders known to it. In SW 5.3 and higher, the encoder serial numbers of all detected encoder can be read from machine data MD 34230: ENC_SERIAL_NUMBER. This is the case for example with EnDat encoders.

A “0” is displayed for encoders that do not have an encoder serial number.

Note

Manipulating machine data MD 34230: ENC_SERIAL_NUMBER will deadjust the absolute value encoder automatically. MD 34200: ENC_REFP_MODE will fall back to “0”.

Configurable expansion SW 6.3 and higher

In SW 6.3 and higher, MD 34232: EVERY_ENC_SERIAL_NUMBER can be used on the SIMODRIVE 611 to set the expansion of the encoder serial numbers in MD 34230: ENC_SERIAL_NUMBER digitally.

Please note that for MD 34232: EVERY_ENC_SERIAL_NUMBER:

- If the SIMODRIVE 611 displays digital 0, the encoder has not been detected or is invalid. The last valid serial number is retained in MD 34232: EVERY_ENC_SERIAL_NUMBER.
- If the SIMODRIVE 611 displays digital 1, the encoder serial number is being MD 34232: EVERY_ENC_SERIAL_NUMBER each time the controller starts up.

Note

This functionality is permanently assigned the code “0” on PROFIBUS DP drives and thus has no effect on PROFIBUS DP drives.

© Siemens AG, 2003. All rights reserved

2.4 Referencing by means of actual value adjustment

2.4.1 Actual value compensation to referencing measurement system

Function

When several measurement systems are in use, referencing to the zero mark, BERO or absolute position of the encoder used for referencing allows **all** other measurement systems of the axis to be adjusted.
In MD 34102: REFP_SYNC_ENCS, the value is set to 1.

A referencing process references all measurement systems. The reference position is derived solely from the referencepointrelated MDs of the measurement system used for referencing.

Note

If one of the two measurement systems is an absolute encoder, REFP_SYNC_ENCS=1 should always be set to ensure that both measurement systems always have the same value.

Example

Two incremental encoders, referencing the one encoder also affects the other one.

CHANDATA (1)
$MA_NUM_ENCS[AX1]=2
$MA_REFP_SYNC_ENCS[AX1]=1
$MA_REFP_MODE[0,AX1]=1
$MA_REFP_MODE[1,AX1]=1

© Siemens AG, 2003. All rights reserved

2.4.2 Actual value adjustment to the referenced measuring system (only for 611 D measuring systems)

Function

To accept a reference value from a measurement system that is not selected for the control, use the setting MD 34200: ENC_REFP_MODE=6.

The travel path for the referencing process is entered in MD 34080: REFP_MOVE_DIST. This value should be higher than the measurement system backlash, from which the reference value is derived.

This setting is intended, for example, for measurement system configurations with an indirect absolute value encoder at the motor and a direct incremental encoder at the table. After elimination of the backlash, further actual value changes in the two measurement systems are identical except for leadscrew errors. At this point in time, the coarse position of the absolute value encoder is adopted by the direct incremental encoder and then corrected to the full accuracy of the direct encoder by the NC.

Direct incremental measurement systems are thus referenced after traversal of the path stored in MD: REFP_MOVE_DIST. If this MD: REFP_MOVE_DIST is set to zero, then the measurement system adjustment is executed immediately after the travel key is actuated. For this purpose, the backlash must be less than half the pitch of the indirect measurement system.

Example

```
CHANDATA(1)
$MA_NUM_ENCS[AX1]=2
$MA_ENC_TYPE[0,AX1]=4
$MA_ENC_TYPE[1,AX1]=1
$MA_REFP_MODE[0,AX1]=0
$MA_REFP_MODE[1,AX1]=6
$MA_REFP_MODE_DIST[1,AX1]=2.0
```

© Siemens AG, 2003. All rights reserved

2.5 Referencing in follow-up mode (SW 6 and higher)

Use

Measuring systems can also be referenced to an encoder zero mark in follow-up mode. The axis motion required for this application must be generated outside the NC.

Preconditions

When axis motion is being generated externally, the reference point can only be calculated definitively for a single zero mark in the traversing range using distance-coded linear scales and modulo rotary axes.

If there is more than one zero mark in the traversing range of the axis, an external reference cam should be used. As a reference cam configured in MD 34000: REFP_CAM_IS_ACTIVE will be ignored in follow-up mode, reference mode MD 34200: ENC_REFP_MODE = 5 should be used in its place. This MD supports a cam connected to the BERO input of the 611D drive module for zero mark selection.

Function

In follow-up mode, referencing is enabled using machine data MD 34104: REFP_PERMITTED_IN_FOLLOWUP.

In follow-up mode, reference can be implemented both in reference point approach mode and in the parts program via G74.
The block change on G74 only takes place once the zero mark has been detected.

The referencing sequence is set implicitly via the status of the axial IS “Follow-up mode active” DB31. ... DBX61.3 active at the start of a referencing operation. In follow-up mode referencing is external or, if closed-loop control is being applied, it is executed on the NC side.

The point in time at which the zero mark search begins in follow-up mode is derived from the axis motion (DB31, ... DBX61.4=0 “axis/spindle at standstill” in reference point approach mode. On G74, it is the time at which block execution takes place.

If the follow-up request is canceled during a referencing operation, the zero mark search will be aborted. An active referencing operation can be aborted by resetting the MCP.

The axial status signal NCK → PLC “Referenced/synchronized 1” DB31, ... DBX60.4 can be used to check the success of a referencing process.

In follow-up mode, the zero mark search is suppressed in reference point approach mode for axes that have already been referenced. It can only be restarted via G74.

© Siemens AG, 2003. All rights reserved

Activation

Function with MD 34104: REFP_PERMITTED_IN_FOLLOWUP = 1 activate.

Switch the axis to follow-up mode with
IS “Follow-up mode” DB31, ... DBX1.4=0 and
IS “Servo enable” (DB31, ... DBX2.1=0.

Wait for the checkback signal in status signal
IS “Follow-up mode active” DB31, ... DBX61.3.

Switch to reference point approach mode.

Move the axis via the encoder zero mark.

© Siemens AG, 2003. All rights reserved

2.6 Referencing passive measuring systems (SW 5 and later)

2.6.1 Function

The axis has two measuring systems. A rotary absolute encoder is used as the motor measuring system, whereas the direct measuring system is a distance-coded linear scale.

SW 4 and lower

Up to SW 4, when the control starts up with MD 34102: REFP_SYNC_ENCS = 1, the passive, direct measuring system is automatically compared to the referenced, absolute motor measuring system. After the actual value comparison, the direct measuring system has the same actual value as the motor measuring system and is also referenced. Since machining is to be performed using the direct measuring system, the PLC activates both measuring systems. Due to the higher loadside accuracy, the direct measuring system is referenced again in “REF” mode.

SW 5 and higher

With SW 5 and higher, the second referencing process may be omitted:
The zero mark search is carried out prior to switchover of the measuring system without additional operating steps at the same time as the next axis movement automatically in the passive measuring system. The PLC application program then switches to the passive measuring system referenced in this way.

IS “Synchronized 1” DB31, ... DBX60.4 or IS “Synchronized 2” DB31, ... DBX60.5 can be used to evaluate the current referencing status of the measuring systems (n = index of the machine axis).

2.6.2 Supplementary conditions

In contrast to referencing in “REF” mode, the following restrictions apply to the referencing of passive measuring systems:

- Supported referencing type ENC_REFP_MODE = 3: distance-coded reference markers.
- No separate traversing movements
- No support from reference cam REFP_CAM_IS_ACTIVE = 1
- Reference monitoring is inactive.

If the reference is missing, referencing is started automatically using the passive measuring systems. The referencing is aborted by a reset or a measuring system switchover and may be restarted if necessary.

© Siemens AG, 2003. All rights reserved

2.6.3 Activation

The time for referencing switchover in the PLC application program is derived form the referencing status of the passive measuring system. Therefore, an actual value comparison to the position of the active measuring system (IS “Position measurement system 1” DB31, ... DBX1.5 or IS “Synchronized 2” DB31, ... DBX1.6) without setting IS “Synchronized 2” DB31, ... DBX60.5 must be configurable (n = index of the machine axis).

IS “Synchronized 2” DB31, ... DBX60.5 must not be set until the passive measuring system has traversed its own zero mark.

This behavior can be configured separately for each encoder via MD 30242: ENC_IS_INDEPENDENT = 2. The value range of the machine data has been expanded to incorporate this feature:

Value = 0: The passive encoder is dependent. The encoder actual value is modified by the active encoder. In conjunction with MD 35102: REFP_SYNC_ENCS =1, the passive encoder is matched with the active encoder when approaching the reference point and then marked as referenced in IS “Synchronized 1” DB31, ... DBX60.4 or IS “Synchronized 2” DB31, ... DBX60.5.

Value = 1: The passive encoder is dependent. The encoder value is not modified by the active encoder.

Value = 2: The passive encoder is dependent. The encoder actual value is modified by the active encoder. In conjunction with MD 35102: REFP_SYNC_ENCS = 1, the passive encoder is adjusted to the active encoder during reference point approach, but not marked as referenced. To reference, first the passive encoder must have crossed its own zero mark.

© Siemens AG, 2003. All rights reserved

2.7 Referencing with cam switch at the drive (SW 5 and later)

2.7.1 Function

Referencing is made to the zero mark (6) of the measuring system (3), after a cam switch (7) connected to the BERO terminal of the drive has tripped.

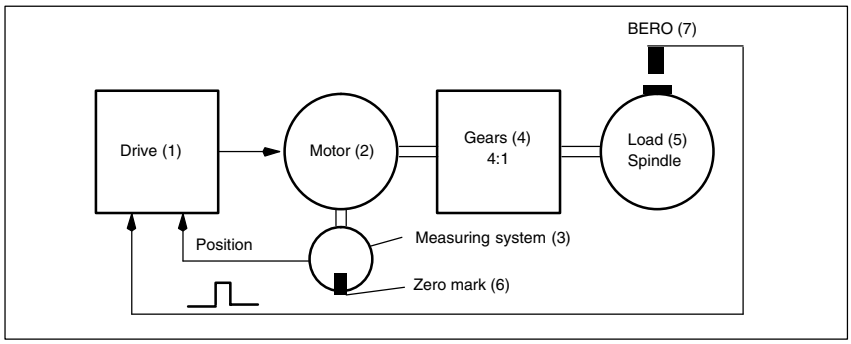


Fig. 2-13 Referencing with cam switch at the drive

In the application described in Fig. 2-13, four measuring system zero marks are found on each load side. A BERO is connected to the 611D drive module as a cam for zero mark selection.

In reference mode MD 34200: ENC_REFP_MODE = 5 is synchronized with the first zero mark after the falling cam edge.
In this configuration, the cam should always be traversed from the same direction.

SW 5 and higher

The MD 34120: REFP_BERO_LOW_ACTIVE=1 can be used to reconfigure the evaluation of the cam edge. The zero mark search will then begin on the rising cam edge. As long as the cam is not exited again before the zero mark is detected, a zero mark overlapped by the cam can be detected reliably in both directions of travel. MD 34120 may only be used in conjunction with MD 34200=5.

Note

The 'Referencing with cam switch on drive' function can only be ensured in conjunction with the SIMODRIVE 611 closed-loop control modules:

Performance 1 closed-loop control (1 axis) 6SN11180DG2*0AA1
Performance 1 closed-loop control (2 axes) 6SN11180DH2*0AA1
Performance 2 closed-loop control (2 axes) 6SN11180DK230AA0

On older closed-loop control modules, the function may differ from that described.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Supplementary Conditions 3

3.1 General supplementary conditions for absolute encoders

3.1.1 Scope of application in 840D/810D

Encoder selection

The 840D supports absolute encoders supplied by Heidenhain, type EQN 1325, and encoders of similar design with the following features:

- 2 incremental tracks, offset by 90 degrees, with raw signal output
- one absolute track (singleturn or multiturn) of which the value can be read serially via the EnDat protocol.

For FDD or MSD closedloop control modules with SIDAASIC (new “Standard” and “Performance” 611D modules) and drive SW 02.60.xx must be available on the drive side.

Availability of NC – software

- The NC software for operating the absolute encoder is an integral component of the 840D package SW 2.2 and is available for NCU571, NCU572 and NCU573.
- Extended NC software for operating the absolute encoder is available in SW 3.6 and 4.1 for NCU571, NCU572 and NCU573, as well as SINUMERIK 810D, and is marked specially in each case.

  The essential difference in the support offered for any whole number gear transmission ratios on infinite rotary axes.

  Furthermore, the travel distance now be longer than the range that can be displayed specifically by the absolute encoder, without losing the absolute position.

3.1.2 Scope of application in FM-NC

The FM-NC supports absolute encoders with SSI protocol, but **not** type EQN 1325 devices since these can only be connected to digital 611D drives. All statements relating to the EQN 1325 or to drive machine data cannot therefore be applied to the FM-NC. There are also differences in the service display, machine data and alarms (separate documentation).

© Siemens AG, 2003. All rights reserved

3.2 Calibration

When to calibrate the system

The calibration process determines the offset between the machine zero and the encoder zero and stores it in a non-volatile memory. Normally, calibration need only be performed once, i.e. during initial start-up. The control can calculate the absolute machine position from the encoder absolute value at any time. This status is identified by MD 34210: ENC_REFP_STATE=2.

The offset is stored in MD 34090: REFP_MOVE_DIST_CORR (SRAM).

The calibration process must be repeated in the following situations:

- After loss of the offset value (SRAM erased, battery voltage failure, PRESET)
- After mounting/removal or replacement of encoder or of motor with builtin encoder.
- After change of an existing gear unit between motor (with absolute encoder) and load.
- Generally speaking, every time the mechanical connection between the encoder and load is separated and not reconnected in exactly the same way.

Caution: The control may not always recognize the need for recalibration. If it detects such a need, it sets machine data MD 34210: ENC_REFP_STATE to 0 or 1.

It can detect the following conditions:

- Loss of the offset value (SRAM erased, battery voltage failure, PRESET)
- Changeover to another gear speed with a different gear ratio between the encoder and load.

In all other cases, the user must overwrite MD 34210: ENC_REFP_STATE (through PLC intervention, cycle, operator input).

Data backup

When the machine data are backed up, the status of MD 34210: ENC_REFP_STATE is also stored.

By loading this block, the axis is automatically deemed calibrated!

If the block has been taken from another machine (e.g. series installation), calibration must still be carried out after loading and activating the data.

Please also observe the notes on MD 30250, see Section 4.2 (SW 3.6 and higher).

© Siemens AG, 2003. All rights reserved

3.3 Special features for large traversing ranges

Unique traversing range

The position reported by a linear scale (e.g. LC181) is always unique for the scales lengths available.

The absolute value of the rotary encoder EQN 1325, on the other hand, overflows after 4096 revolutions.

The absolute position is unique only for the following maximum values:

- Rotary axis
  - Encoder at load ⇒ 4096 load revolutions
  - Encoder at motor ⇒ 4096 motor revolutions
- Linear axis
  - Encoder at motor ⇒ 4096 · rms spindle lead

Example

A linear axis with effective spindle rise of 10mm thus covers a distance of –20.48 to 20.48m.

If the full range is to be utilized, the encoder value overflows within the useful traversing range, since the encoder zero and machine zero cannot be matched accurately. The correct machine actual value is calculated nevertheless using an internal offset in the NC. This allows the full traversing range to be utilized, without having to take account of the rotation with which the encoder was installed.

Selection of the machine zero (for 10mm spindle lead):

- Traversing range of lengths up to 20.48m

  ⇒ machine zero can be freely selected
- Traversing ranges from 20.48m to 40.96m

  ⇒ Distance between the machine zero from both ends ≤ 20.48m (at length 40.96m, the machine zero is then in the middle).

Special points to be noted

- For machines with linear axis traversing range > 4096 encoder revolutions with EQN 1325 (for input of MD 34220: ENC_ABS_TURNS_MODULO = 4096):
  - From SW 3.6, the traversing range that can be displayed has been extended in the software to the full number format in the servomotor, i.e. it is identical to that for incremental encoders.
  - When the encoder is switched off (POWER OFF/ON, parking), it must be ensured that the axis for rotary encoders (e.g. EQN 1325) is moved less than half of the traversing range defined in MD 34220: ENC_ABS_TURNS_MODULO (less than 2048 revolutions for EQN 1325). In this case, the software is able to reconstruct the new position using shortest path detection.

© Siemens AG, 2003. All rights reserved

- For rotary axes with absolute encoders operated infinitely:
  - From SW 3.6, any number of transmission ratios are permitted; the overflow correction required for this is carried out internally by the software.
  - The actual value of rotary axes with infinite rotation is generally reduced to the “basic range” (e.g. 0 to 360 degrees) by means of modulo offset after POWER ON. Otherwise, the axis could have to be traversed quite a distance the first time it is positioned at zero. The range used for reduction can be set using MD 34220: ENC_ABS_TURNS_MODULO (important for EQN 1325!).
  - It must be ensured that the axis does not move more than half the uniquely representable number range of the absolute encoder when the encoder is switched off, i.e. the encoder motion must not exceed ENC_ABS_TURNS_MODULO/2 encoder revolutions. Recommendation: Use handbrakes.
  - The control cannot detect all case that require recalibration and/or cases in which the adjustable absolute encoder position may be invalid (e.g. disconnection of the encoder from the load, possibly even with the control deactivated, inactive encoder, parking, etc.)! If the control detects the need for calibration, it reports it by setting MD 34210: ENC_REFP_STATE to the value 0 or 1 (Section 3.2).

    In all cases that are **not** detected, it is the responsibility of the user to set MD 34210: ENC_REFP_STATE=0.
- Also for finite rotary axes:
  - From SW 3.6/4.1, the internal software extension means that the traversing range that can be represented included positive and negative values (e.g. –180 ...180 degrees).

© Siemens AG, 2003. All rights reserved

3.4 Special features of SINUMERIK 840Di

Background

With the SINUMERIK 840Di, the overflow of encoders is saved cyclically in an SRAM. If a fault scenario (e.g. power failure) occurs, it is possible to use these data when the system is restarted. If the fault scenario and writing the data to the SRAM are performed at the same time, it may happen that the current value gets lost. In this case, the previous value is stored as the current value in the SRAM. After the system has been restarted, the value of the overflow is read from the SRAM and assumed as the real overflow.

Referencing after a fault scenario

When using absolute encoders with axes turning endlessly and incremental encoders, the overflow is saved with the backup in the SRAM. When the system is restarted without referencing after a fault scenario, this value is defined as the current overflow. The correct value of the overflow is thus lost. In order to be able to rule out this case, it is recommended to use an UPS (uninterruptible power supply). When doing so, the current value will also be saved reliably after a power failure.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

4.1 Channel-specific machine data

20700 MD number	**REFP_NC_START_LOCK** NC start disable without reference point		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after Reset	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	0: IS “NC Start” DB21, ... DBX7.1) for starting parts programs or part program blocks (MDA and Overstore) is operative even if one or all the axes in the channel have still to be referenced. To ensure that the axes move to the correct position after NC Start, the workpiece coordinate system (WCS) must be set to a correct value by other methods (scratching method, automatic zero offset calculation, etc.). 1: NC start only if all axes have been referenced.		
Further information	see Subsection 2.3.3.		

© Siemens AG, 2003. All rights reserved

4.2 Axis/spindle-specific machine data

Number	Identifier	Name	Refer-ence
30200	NUM_ENCS	Number of encoders	G2

30240 MD number	**ENC_TYPE** Actual value encoder type		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 5	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Encoder type: 0: Simulation 1: Raw signal encoder (high resolution) 2: Rectangular signal encoder (standard, no. of encoder marks multiplied) 3: Encoder for stepper motor (FMNC only) 4: Absolute encoder with EnDat interface 5: Absolute encoder with SSI interface (only FMNC)		
Further information	See Subsection 2.4.2		
Related to	Drive MD 1011: ACTUAL_VALUE_CONFIG, bit 3		

30250 MD number	**ACT_POS_ABS** Absolute encoder position at time of deactivation.		
Default setting: 0.0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DOUBLE	Applies from SW 3.6		
Significance:	The current position is stored in this MD (in internal format). It is used during POWER ON (or encoder activation) for: • Rotary absolute value encoders: to restore the current position (in conjunction with the potentially ambiguous position stored in the encoder) • Incremental encoders: for storing the actual value via POWER OFF when MD 34210: ENC_REFP_STATE ≠ 0 (i.e. as equivalent reference point). **Note:** This MD is changed internally in the control during traversing. Inputting an MD block that has been backed up earlier can therefore destroy the calibration of absolute encoders. For SW upgrades, we recommend backing up the MD block in the old SW release immediately prior to upgrading and then incorporating the new SW release without any axis motion in the meantime. Protection level 1 should be set for SW 3.6; protection level 2 is sufficient from SW 4. The encoder calibration should be verified explicitly (check/adjust) after SW upgrade!		

© Siemens AG, 2003. All rights reserved

30270 MD number	**ENC_ABS_BUFFERING[n]**; n=0,1 Position buffer in the encoder		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 5.2		
Significance:	By “1” parameterization, a mirroring of the value for $MA_REFP_MOV_DIST_CORR[n] in the EnDat encoder FEPROM is active (only for SIMODRIVE 611D and EnDat absolute encoders). At the same time, the traversing range extension of absolute encoders active per default is deactivated. This buffering is necessary for absolute encoders at addon axes. In addition to this, the extension to the traversing range can also be deactivated at the axes, for which the entire traversing range is uniquely represented by the absolute encoder multiturn information. It is not advisable to deactivate the traversing range extension for axes with an “uneven” (not a POWER Of 2) gear ratio and large traversing range, for which the absolute encoder position can be incorrect without warning on the next POWER ON if the traversing range extension is not working.		

34000 MD number	**REFP_CAM_IS_ACTIVE** Axis with reference point cam		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	Machine axes with only one zero mark across their entire traversing range or rotary axes with only one zero mark per revolution are not identified as a machine axis with reference cam by the MD: REF_CAM_IS_ACTIVE. A machine axis identified as a machine axis with reference cam accelerates, when the plus/minus travel key is pressed, to the velocity defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity).		
Further information	see Subsections 2.1.1 and 2.6.2		
MD irrelevant for ...	Linear measurement systems with distance-coded reference marks		

© Siemens AG, 2003. All rights reserved

34010 MD number	**REFP_CAM_DIR_IS_MINUS** Reference point approach in minus direction		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	**REF_CAM_DIR_IS_MINUS = 0** Reference point approach in plus direction **REF_CAM_DIR_IS_MINUS = 1** For incremental measurement systems: If the machine axis is positioned before the reference cam, it accelerates, when the travel key, to the velocity defined in MD 34020: REFP_VELO_SEARCH_CAM (reference point approach velocity) in the direction specified in MD: REFP_CAM_DIR_IS_MINUS. If the wrong travel key is pressed, the reference point approach does not start. If the machine axis is positioned on the reference mark, it accelerates to the velocity defined in MD 34020: REFP_VELO_SEARCH_CAM and travels in the opposite direction to that specified in MD: REFP_CAM_DIR_IS_MINUS. For linear measurement systems with distancecoded reference marks: If the machine axis has a reference cam (linear measurement systems with distance-coded reference marks do not need a reference cam) and the machine axis is positioned on the cam, it accelerates, regardless of the depressed plus/minus travel key, to the velocity defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the opposite direction to that specified in MD: REFP_CAM_DIR_IS_MINUS.		
Further information	see Subsections 2.1.1, 2.1.2, 2.2.1, 2.3.2 and 2.3.5		

34020 MD number	**REFP_VELO_SEARCH_CAM** Reference point approach velocity		
Default setting: 5000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm/min, rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The reference point approach velocity is the velocity at which the machine axis travels in the direction of the reference cam when the travel key is pressed (phase 1). This value should be set at a magnitude large enough for the axis to be stopped, before it reaches a hardware limit switch.		
Further information	see Section 2.1, Subsections 2.1.1, 2.1.2 and 2.1.3		
MD irrelevant for ...	Linear measurement systems with distancecoded reference marks		

34030 MD number	**REFP_MAX_CAM_DIST** Maximum distance to reference cam		
Default setting: 10000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	If the machine axis travels a maximum distance (defined in MD: REFP_MAX_CAM_DIST) REFP_MAX_CAM_DIST from the starting position in the direction of the reference cam without reaching the reference cam (“Reference point approach delay” interface signal (DB31, ... DBX12.7) is reset) and Alarm 20000 “Reference cam not reached” is output.		
Further information	See Subsection 2.1.1		
MD irrelevant for ...	Linear measurement systems with distancecoded reference marks		

© Siemens AG, 2003. All rights reserved

34040 MD number	**REFP_VELO_SEARCH_MARKER[n]** Reference point creep speed [encoder number]: 0, 1		
Default setting: 300	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm/min, rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	1) For incremental measurement systems: This is the velocity at which the axis travels between initial detection of the reference cam and synchronization with the first zero mark (phase 2). Direction of travel: opposite to the direction specified for cam detection (MD 34010: REFP_CAM_DIR_IS_MINUS) If MD 34050: REFP_SEARCH_MARKER_REVERSE (direction reversal on reference cam) is enabled, the axis travels at the velocity defined in MD 34020: REFP_VELO_SEARCH_CAM in the case of synchronization 2) For linear measurement systems with distance-coded reference marks: The axis crosses the two reference marks at this velocity. The max. velocity must be small enough that the time required to travel the smallest possible reference mark distance (x_{min}) on the linear measurement system is larger than one position controller cycle. In the following formula, the basic distance [multiple of the graduation cycle], graduation cycle [mm] and measured length [mm] yield the maximum velocity: $x_{min}\ [mm] = \frac{\text{Basic distance}}{2} * \text{Graduation cycle} - \frac{\text{Measured length}}{\text{Basic distance}}$ $\text{Max. velocity [m/s]} = \frac{x_{min}\ [mm]}{\text{Position controller cycle [ms]}}$ This limit also applies to the other measurement systems. Direction of travel: – as defined in MD: REFP_CAM_DIR_IS_MINUS; – if the axis is already positioned on the cam, the axis is traversed in the opposite action 3) Indirect measurement system with BERO on the loadside (preferred for spindles) At this velocity, the zero mark associated with the BERO is searched for (zero mark selection per VDI signal). The zero mark is accepted if actual velocity lies within the tolerance range defined by MD 35150: SPIND_DES_VELO_TOL, as deviation from the velocity specified in MD 34040: REFP_VELO_SEARCH_MARKER[n].		
Further information	see Sections 2.1 and 2.2, Subsections 2.1.1, 2.1.2, 2.1.3, 2.2.1 and 2.3.6		

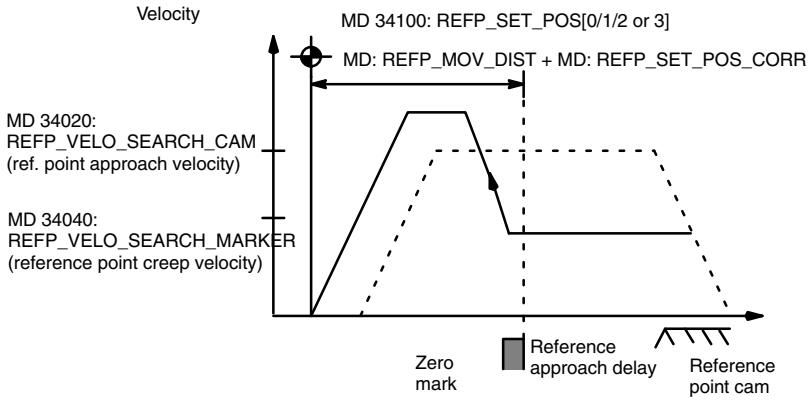
© Siemens AG, 2003. All rights reserved

34050 MD number	**REFP_SEARCH_MARKER_REVERSE[n]** Change of direction on reference cam [encoder number]: 0, 1		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	The MD: REFP_MARKER_REVERSE can be used to set the direction of the search for the zero mark: **REFP_SEARCH_MARKER_REVERSE = 0** Synchronization with falling reference cam signal edge The machine axis accelerates to the velocity defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the opposite direction to that defined in MD 34010: REFP_CAM_DIR_IS_MINUS (reference point approach in minus direction). If the reference cam is existed (IS “Reference point approach delay” (DB31, ... DBX12.7) is reset), the control synchronizes with the first zero mark. **MD: REFP_SEARCH_MARKER_REVERSE = 1** Synchronization with rising reference cam signal edge The machine axis accelerates to the velocity defined in MD 34020: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the opposite direction to that specified in the MD: REFP_CAM_DIR_IS_MINUS. When the axis leaves the reference cam (“Reference point approach delay” interface signal (DB31–48, DBX12.7) is reset), the machine axis decelerates to a halt and accelerates in the opposite direction towards the reference cam at the velocity defined in MD: REFP_VELO_SEARCH_MARKER. If the reference cam is reached (IS “Reference point approach delay” (DB31, ... DBX12.7) is reset), the control synchronizes with the first zero mark.		
Further information	See Subsection 2.1.2		
MD irrelevant for ...	Linear measurement systems with distancecoded reference marks		

34060 MD number	**REFP_MAX_MARKER_DIST[n]** Maximum distance to reference mark; Maximum distance to 2 reference marks for distancecoded measurement systems [encoder no.]: 0, 1		
Default setting: 20	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	For incremental measurement systems: If, after leaving the reference cam (“Reference point approach delay” interface signal is reset), the machine axis travels a distance defined in MD: REFP_MAX_MARKER_DIST without detecting the zero mark, the axis stops and alarm 20002 “Zero mark missing” is output. For linear measurement systems with distance-coded reference marks: If the machine axis travels a distance defined in MD: REFP_MAX_MARKER_DIST from the starting position without crossing two zero marks, the axis stops and alarm 20004 “Reference mark missing” is output.		
Application example(s)	If, on incremental measurement systems, the control is required to detect reliably that the same zero mark is always used for synchronization (to avoid detection of an incorrect machine origin), the maximum value in MD: REFP_MAX_MARKER_DIST must not exceed the distance between two reference marks.		
Further information	see Subsections 2.1.2 and 2.2.1		

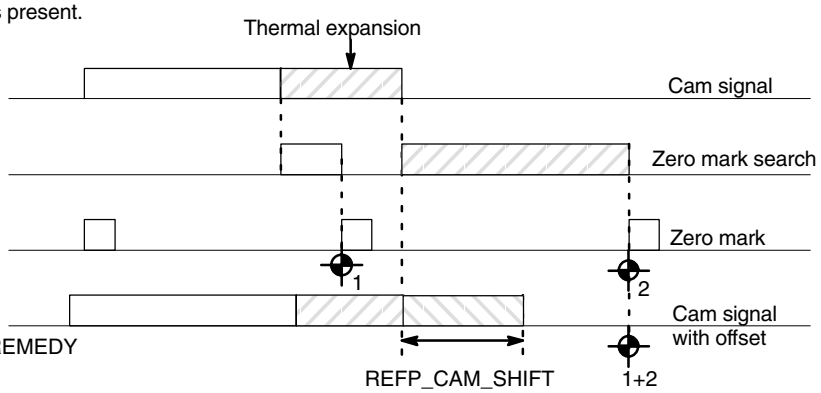
© Siemens AG, 2003. All rights reserved

34070 MD number	**REFP_VELO_POS** Reference point start velocity		
Default setting: 10000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm/min, rev/min	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	For incremental measurement systems: The axis travels at this velocity between the time of synchronization with the first zero mark and arrival at the reference point. For linear measurement systems with distance-coded reference marks: The axis travels at this velocity between the time of synchronization (crossing two zero marks) and arrival at the reference point.		
Further information	see Section 2.2, Subsections 2.1.2, 2.1.3 and 2.2.2		

34080 MD number	**REFP_MOVE_DIST[n]** Reference point distance encoder number]: 0, 1		
Default setting: –2.0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance: (graphic see Fig. 2-9):	For incremental measurement systems: Following synchronization with the first zero mark, the machine accelerates to the velocity defined in MD 34070: REFP_VELO_POS (reference point start velocity) and travels a distance calculated from the addition of MD: REFP_MOVE_DIST and MD 34090: REFP_MOVE_DIST_CORR (reference point offset). This distance precisely matches the distance between the zero mark detected in phase 2 and the reference point.		
			
	The value 6 in MD 34200: ENC_REFP_MODE refers to the distance traveled, after which the actual values between the measurement systems are aligned. The distance entered should greater than the existing backlash.		
Further information	See Subsection 2.1.3		
MD irrelevant for ...	Linear measurement systems with distance-coded reference marks		

© Siemens AG, 2003. All rights reserved

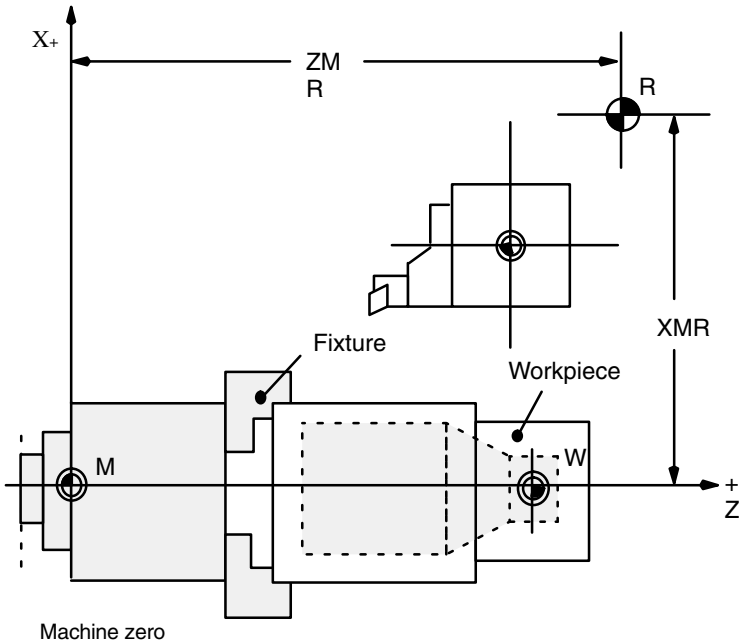
34090 MD number	**REFP_MOVE_DIST_CORR[n]** Reference point offset/absolute offset, distance-coded [Encoder number]: 0, 1		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after: Reset	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	● Incremental encoder with zero mark(s): After detection of the zero, the axis is moved away from the zero mark by the distance MD 34080: REFP_MOVE_DIST + REFP_MOVE_DIST_CORR. Once the axis has traveled this distance, it has reached the reference point. MD 34100: REFP_SET_POS is set as the actual value. During traversing REFP_MOVE_DIST+REFP_MOVE_DIST_CORR, the override switch and MD : JOG_INC_MODE_IS_CONT (permanent/job mode) are active ● Distance-coded measurement system: REFP_MOVE_DIST_CORR acts as the absolute offset. It defines the offset between the machine zero point and the first reference mark of the measurement system. ● Absolute value encoder: REFP_MOVE_DIST_CORR acts as the absolute offset. It defines the offset between machine zero and the zero point of the absolute measurement system. **Note:** This MD is changed by the control during calibration and modulo correction in conjunction with the absolute encoders! Manual entry or modification of this MD with a parts program should therefore be followed by a POWER ON reset so that the new value is activated and cannot be lost.		
Further information	See Subsection 2.1.3		

34092 MD number	**REFP_CAM_SHIFT** Electronic reference cam offset for incremental measurement systems with equidistant zero marks		
Default setting: 0.0	Minimum input limit: 0.0	Maximum input limit:	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm	
Data type: DOUBLE	Applies from SW 3.2		
Significance:	When the reference cam signal occurs, the zero mark search is not started immediately, but with a delay after the distance of REFP_CAM_SHIFT. The reproducibility of the zero mark search can thus be ensured even in the case of temperature-dependent expansion of the reference cam through defined selection of a zero mark. As the reference cam offset is calculated by the control in the interpolation cycle, the actual reference cam offset is is at least REFP_CAM_SHIFT and maximum REFP_CAM_SHIFT+(MD 34040: REFP_VELO_SEARCH_MARKER*interpolation cycle) The reference cam offset acts in the direction of zero mark search. The reference cam offset is active only if the cam MD 34000: REFP_CAM_IS_ACTIVE=1 is present.  		
Further information	See Subsection 2.1.2		

© Siemens AG, 2003. All rights reserved

34093 MD number	**REFP_CAM_MARKER_DIST** Reference cam/reference mark distance		
Default setting: 0.0	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: mm	
Data type: DOUBLE	Applies from SW 6.3		
Significance:	The indicated value is equivalent to the distance between departure from the reference cam and detection of the reference mark. If the values are too small, there is a risk that the determination of the reference point will be nondeterministic, due to temperature effects or fluctuations in the operating time of the cam signal. The path covered can be used as a guide for setting the electronic reference cam offset. The machine data is read-only.		
Related to	REFP_CAM_IS_ACTIVE, REFP_SHIFT_CAM		

© Siemens AG, 2003. All rights reserved

34100 MD number	**REFP_SET_POS[n]** Reference point value/destination point for distance-coded system		
Default setting: 0.0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after RESET	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	• Incremental encoder with zero mark(s): The position value which is set as the current axis position after detection of the zero mark and traversal of the distance REFP_MOVE_DIST + REFP_MOVE_DIST_CORR (relative to zero mark). The REFP_SET_POS for the reference point number, which is set as the instant that the edge of the reference cam signal (INTSIG DB31, ...DBX2.4–2.7) rises, is set as the axis position. • Distance-coded measurement system: Destination point which is approached when REFP_STOP_AT_ABS_MARKER is set to 0 (FALSE) and two zero marks have been crossed. • Absolute value encoder: REFP_SET_POS corresponds to the correct actual value at the calibration position. The reaction on the machine depends on the status of MD34210: ENC_REFP_STATE: When ENC_REFP_STATE = 1, the value of REFP_SET_POS is transferred as the absolute value. When ENC_REFP_STATE = 2 and REFP_STOP_AT_ABS_MARKER = 0 (FALSE), the axis approaches the destination point stored in REFP_SET_POS. The value of REFP_SET_POS that has been set via (INTSIG DB31, ...DBX2.4–2.7) is used.		
Fig. 12	 M Machine zero W Workpiece zero R Reference point XMR Reference point in X direction (MD: REFP SET POS [X]) ZMR Reference point in Z direction (MD: REFP SET POS [Z])		
Further information	See Chapter 2, Subsections 2.1.3, 2.2.2, 2.3.2, 2.3.3, 2.3.4 and 2.3.5		
Related to	“Reference point value 1 to 4” interface signal (DB31, ... DBX2.4 –2.7)		

© Siemens AG, 2003. All rights reserved

34102 MD number	**REFP_SYNC_ENCS** Measurement system adjustment active		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 3.1		
Significance:	Adjustment to the reference measurement system can be activated for all measurement systems of the relevant axis with this machine data. The adjustment is made during reference point approach or when calibrated absolute value encoders selected for the closed-loop control are switched on. Values: 0: No measurement system adjustment, measurement systems must be referenced individually 1: Adjustment of all measurement systems of the axis to the position of the referencing measurement system		
Further information	see Subsections 2.3.6, 2.4.1 and 2.6.1		

34104 MD number	**REFP_PERMITTED_IN_FOLLOWUP** Enable referencing in follow-up mode		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.1		
Significance:	In reference point approach mode and with G74, an axis can also be referenced in follow-up mode with the aid of an external movement.		
Related to	MD 34200: ENC_REFP_MODE		

34110 MD number	**REFP_CYCLE_NR** Axis sequence for channel-specific referencing		
Default setting: 1 ... 31 (dependent on axis no.)	Minimum input limit: –1	Maximum input limit:	8 (840D), 5 (FMNC/ 810D)
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		

© Siemens AG, 2003. All rights reserved

34110 MD number	**REFP_CYCLE_NR** Axis sequence for channel-specific referencing
Significance:	MD: REFP_CYCLE_NR = 0 ———> axisspecific referencing Axis-specific referencing is started separately for each machine axis with the "plus/minus travel keys" interface signal (DB31, ... DBX4.7 and 4.6). Up to 8 axes on the 840D and up to 5 axes on the FM-NC/810D can be referenced simultaneously. The following alternatives are provided for referencing the machine axes in a specific sequence: • The operator observes the correct sequence on start-up • The PLC user program checks the sequence on start-up or defines the sequence itself. • The channel-specific referencing function is used. MD: REFP_CYCLE_NR = 1 ———> channel-specific referencing The channel-specific referencing is started with IS "Activate referencing" (DB21, ... DBX18.5). The control acknowledges a successful start with the "referencing active" interface signal (DB21, ... DBX33.0). Each machine axis assigned to the channel can be referenced with channel-specific referencing (this is achieved internally on the control by simulating the plus/minus travel keys). The axis-specific MD: REFP_CYCLE_NR can be used to define the sequence in –1 means: The machine axis is started by channel-specific Referencing not and NC start is possible without referencing this axis. **Note:** A change from –1 or to –1 is not effective until after POWER ON. An entry of –1 for all axes in the channel can be made effective by setting MD 20700: REFP_NC_START_LOCK = 0. 0 means: The machine axis is started by channel-specific Referencing not and NC start is not possible without referencing this axis. 1 means: The machine axis is started by channel-specific Referencing started. 2 means: The machine axis is started by channel-specific Referencing started. if all machine axes identified by a 1 in MD: REFP_CYCLE_NR are referenced. 3 means: The machine axis is started by channel-specific Referencing started. if all machine axes identified by a 1 in MD: REFP_CYCLE_NR are referenced. 4 to 8 : As above for further machine axes. Setting the channel-specific MD 20700: REFP_NC_START_LOCK (NC start disable without reference point) to zero has the effect of entering –1 for all the axes of a channel. 4 to 31: (CPU type)
Further information	see Chapter 2
MD irrelevant for ...	Axis-specific referencing
Related to	IS "Activate referencing interface signal" (DB21, ... DBX1.0) IS "referencing active" (DB21, ... DBX33.0)

© Siemens AG, 2003. All rights reserved

34120 MD number	**REFP_BERO_LOW_ACTIVE** Polarity change of the BERO cam		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 5		
Significance:	The edge evaluation of a BERO connected to the digital 611D drive as a cam (ENC_REFP_MODE = 5) can be changed: REFP_BERO_LOW_ACTIVE = 0 BERO as cam switches on falling edge REFP_BERO_LOW_ACTIVE = 1 BERO as cam switches on rising edge Note: It may only be used in conjunction with ENC_REFP_MODE = 5 and the following SIMODRIVE 611 closedloop control modules: Performance 1 closed-loop control (1 axis) 6SN11180DG2*0AA1 Performance 1 closed-loop control (2 axes) 6SN1118–0DH2*–0AA1 Performance 2 closed-loop control (2 axes) 6SN11180DK230AA0		
Further information	See Subsection 2.7.1		

34200 MD number	**ENC_REFP_MODE[n]** Referencing mode [encoder number]: 0 ... max. no. of encoders –1		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 7	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The connected measurement systems can be subdivided into two groups for referencing with MD: ENC_REFP_MODE • ENC_REFP_MODE = 0: Transfer of MD 34100: REFP_SET_POS • ENC_REFP_MODE = 1: Referencing with incremental measurement systems: – incremental rotary measuring system – incremental linear measuring system (linear measurement system) – zero pulse on encoder track – (not for absolute encoders) • ENC_REFP_MODE = 2: BERO with 1edge detection; also possible with absolute value encoder. After referencing, the encoder is additionally designated as “adjusted”. • ENC_REFP_MODE = 3: Referencing with measuring systems with distance-coded reference markers (Heidenhain) using **two** reference markers • ENC_REFP_MODE = 4: BERO with 2 edge evaluation (only relevant for FM-NC) The positive and negative edges of the reference point BERO are traversed in succession and the relevant actual values are registered. The average value calculated is from the synchronization point. The two-edge evaluation compensated a possible drift or temperature-dependent expansion of the BERO. • ENC_REFP_MODE = 5: When the BERO is traversed, the zero mark search is started when the negative edge is detected and is synchronized with the next zero mark (see Section 2.7) • ENC_REFP_MODE = 6: Measuring system comparison to an encoder that has already been referenced (SW 3.2 and higher) • ENC_REFP_MODE = 7: BERO with configured approach speed for axis and spindle applications (SW 3.6 and higher) MD 34040: REFP_VELO_SEARCH_MARKER[n] (reference point deactivation velocity[Enc. No.]). • ENC_REFP_MODE = 8: Referencing with measuring systems with distance-coded reference markers (Heidenhain) using **four** reference markers		
Further information	See Subsection 2.7.1		

© Siemens AG, 2003. All rights reserved

34210 MD number	**ENC_REFP_STATE[n]** Status of absolute value encoder [encoder number]: 0 ... max. no. of encoders –1		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective IMMEDIATELY	Protection level: 4/7	Unit: –	
Data type: BYTE	Applies from SW 2.2		
Significance:	• Absolute value encoder: ENC_REFP_STATE = 0: Default when installing for the first time: Encoder is not adjusted. ENC_REFP_STATE = 1: Encoder adjustment enabled, encoder not yet adjusted. ENC_REFP_STATE = 2: Encoder is adjusted • Incremental encoder: ENC_REFP_STATE = 0: Default: no automatic referencing ENC_REFP_STATE = 1: Automatic referencing enabled, but encoder not yet referenced or not at exact stop Activate ENC_REFP_STATE = 2: Encoder is referenced and at exact stop; automatic referencing effective next time the encoder is activated.		
Application example(s)	The MD ENC_REFP_STATE can be altered by the person installing and by the operating system: • Absolute value encoder: – Alteration by the installer: The data must be set to “1” by the installer if adjustment of this encoder is required. – Alteration by the operating system: on successful adjustment from 1 ==> 2 if adjustment form 2 ==> 0 or 1 has become invalid SRAM losses and gear speed changes with change of ratio are identified by the operating system. **Changes to the mechanical components of the machine (e.g. encoder change, motor and encoder change, etc.) are not detected.** • Incremental encoder: – Alteration by the installer: The data must be set to “1” by the installer if automatic referencing is to be carried out. – Alteration by the operating system: for referenced axis and “axis at exact stop” from 1 ==> 2 if the reference position has become invalid or if the axis is not at exact stop from 2==> 1 Unlike the absolute value encoder, changes in position here are **not** detected during POWER OFF or when the encoder is inactive.		
Further information	see Section 2.3, Subsections 2.1.4, 2.3.2 and 2.3.3		
MD irrelevant for ...	Incremental encoders with zero mark (standard encoders) Linear measurement systems with distance-coded reference marks		

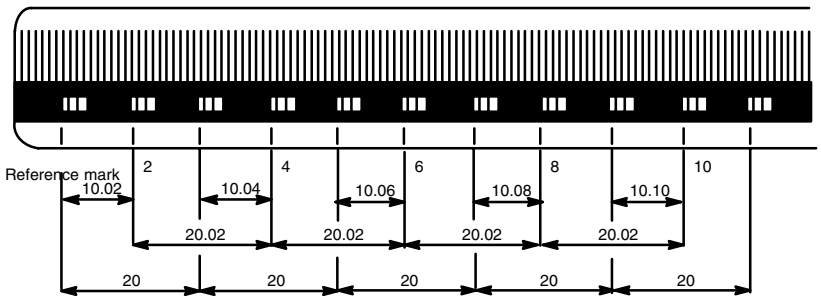
© Siemens AG, 2003. All rights reserved

34230 MD number	**ENC_SERIAL_NUMBER[n]**, n=0,1 Encoder serial number		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	The serial number of EnDat encoders can be output here. A “0” is displayed for encoders which do not have an encoder serial number. A manipulation of this MDs usually results in an automatic deadjustment of the absolute encoder ($MA_ENC_REFP_MODE drops back to “0“).		
Related to	MD 34232: EVERY_ENC_SERIAL_NUMBER		

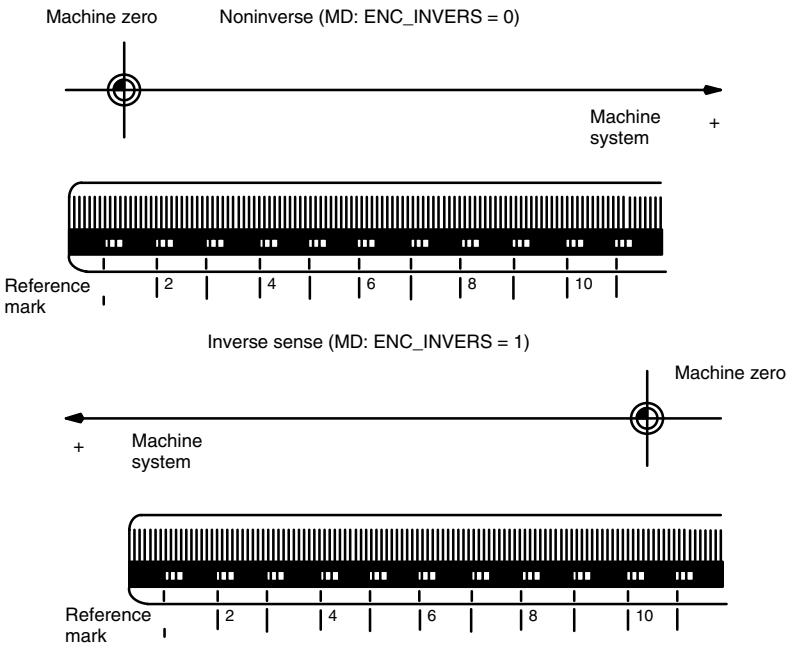
34232 MD number	**EVERY_ENC_SERIAL_NUMBER[n]**, n=0,1 Expansion of encoder serial number		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.3		
Significance:	Can be used to set the expansion of MD 34230: ENC_SERIAL_NUMBER digitally on the SIMODRIVE 611: FALSE: Only valid encoder serial numbers are entered in the machine data, i.e. a “0” returned from the drive (corresponds to invalid or unknown) the last valid encoder serial number is retained in the machine data (e.g. for structural rotary axes not always known on the machine). TRUE: The encoder serial number value supplied by the drive is written to the machine data every time the controller starts up. A validity check is not carried out. **Note for PROFIBUS DP drives:** As not every drive is capable of supplying the relevant parameters reliably and at the right time, this function is permanently assigned the code “FALSE” on the PROFIBUS DP drive. The setting “TRUE” therefore has no effect on PROFIBUS DP.		
Related to	MD 34230: ENC_SERIAL_NUMBER		

© Siemens AG, 2003. All rights reserved

34300 MD number	**ENC_REFP_MARKER_DIST[n]** Reference mark distance for distance-coded Scales [encoder no.]: 0 ... max. no. of encoders –1		
Default setting: 10	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	If the control detects that the distance between two reference marks exceeds 2 × ENC_REFP_MARKER_DIST and is thus false, the machine accelerates to half the speed defined in MD 34040: REFP_VELO_SEARCH_MARKER (reference point creep velocity) in the opposite direction to the depressed travel key (plus or minus) and crosses two reference marks. If the control still detects that the distance between two reference marks exceeds 2 * ENC_REFP_MARKER_DIST, the axis remains stationary and alarm 20003 “error in measurement system” is output.		
Further information	See Subsection 2.2.1		
MD irrelevant for ...	Measurement systems without distance-coded reference marks.		

34310 MD number	**ENC_MARKER_INC[n]** Differential distance between two reference marks for distance-coded scales [encoder no.]: 0 .. max. no. of encoders –1		
Default setting: 0,02	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The differences between two reference marks are defined variably, so that the position of the crossed reference marks can be determined accurately in linear measurement systems. The difference between two reference marks is entered in the MD: ENC_MARKER_INC. In the example in Fig. 13, this value is 20.02 – 20 = 0.02 [mm]		
Fig. 2-12	 DIADUR graduated glass scale with distance-coded reference marks Dimensions in mm for 20mm scale division The difference here between two reference marks is 0.02mm.		
MD irrelevant for ...	Incremental measurement systems		
Special cases, errors, ...	On linear measurement systems with distance-coded reference marks supplied by Heidenhain, the difference between two reference marks is always equal to one scale division.		

© Siemens AG, 2003. All rights reserved

34320 MD number	**ENC_INVERS[n]** Linear measurement system inverse to machine system [encoder no.]: 0 ... max. no. of encoders –1		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	• No effect with incremental measuring system irrelevant • With distance-coded measurement system: When setting the reference point, the actual position (determined by the distance-coded reference marks) on the linear measurement system is assigned to a precise machine axis position (in relation to the machine zero). This is achieved by entering the absolute offset on the linear measurement system in MD 34090: REFP_MOVE_DIST_CORR (reference point/absolute offset). Reference marker entered on the linear measurement system. MD ENC_INVERS must also be set in order to define whether the linear measurement system is installed in the non-inverse or inverse sense to the machine system.		
Fig. 14			
Further information	See Subsection 2.2.2		
MD irrelevant for ...	Incremental encoders without distance-coded reference marks.		

© Siemens AG, 2003. All rights reserved

34330 MD number	**REFP_STOP_AT_ABS_MARKER[n]** Distance-coded linear measurement system without destination point [encoder no.]: 0 ... max. no. of encoders –1		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: mm, degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	• Distance-coded measurement system: REFP_STOP_AT_ABS_MARKER = 0: At the end of the reference cycle, the position entered in MD 34100: REFP_SET_POS is approached (normal case for phase 2). REFP_STOP_AT_ABS_MARKER = 1: The axis is braked after the second reference mark has been detected (reduction in phase 2) • Absolute value encoder: MD REFP_STOP_AT_ABS_MARKER defines the response of an axis with valid calibration identifier (MD 34210: ENC_REFP_STATE = 2) with G74 or when a travel key is actuated in JOGREF: REFP_STOP_AT_ABS_MARKER = 0: Axis travels to the position entered in MD: REFP_SET_POS REFP_STOP_AT_ABS_MARKER = 1: Axis does not travel.		
Application example(s)	The MD: STOP_AT_ABS_MARKER must be enabled in order to inhibit the approach of a fixed destination point at the end of the reference point approach on linear measurement systems with distance-coded reference marks. In many cases, however, a fixed destination point is approached as on incremental measurement systems.		
MD irrelevant for ...	Incremental encoders with zero mark (standard encoders)		
Related to	MD 34100: REFP_SET_POS (reference point/destination point for distance-coded system)		

■

© Siemens AG, 2003. All rights reserved

Signal Descriptions 5

5.1 Channel-specific signals

5.1.1 Signals to channel

DB21, ... **DBX1.0** Data block	**Activate referencing** Signal(s) to channel (PLC –> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Channel-specific referencing is started with the “Activate referencing” interface signal. The control acknowledges a successful start with the “referencing active” interface signal Each machine axis assigned to the channel can be referenced with channel-specific referencing (this is achieved internally on the control by simulating the plus/minus travel keys). The axis-specific MD 34110: REFP_CYCLE_NR (Axis sequence for channel-specific Referencing) can be used to define the sequence in When all the axes are entered in MD: REFP_CYCLE_NR have reached their end points, the “All axes referenced” interface signal (DB21, ... DBX36.2) is enabled.	
Application example(s)	If the machine axes are to be referenced in a particular sequence, the following options are available: • The operator observes the correct sequence on start-up • The PLC user program checks the sequence on start-up or defines the sequence itself • The channel-specific referencing function is used.	
Related to	IS “referencing active” (DB21, ... DBX33.0) IS “All axes referenced” (DB21, ... DBX36.2)	

DB21, ... **DBX33.0** Data block	**Referencing active** Signal(s) to channel (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Channel-specific referencing has been started with the “Activate referencing” interface signal and successful start acknowledged with the “Referencing active” interface signal. Channel-specific referencing is operational.	
Signal state 0 or signal transition 1 ——> 0	• Channel-specific referencing is finished • Axis-specific referencing in progress • No referencing active	
Signal irrelevant for ...	Spindles	
Related to	IS “Activate referencing” (DB21, ... DBX1.0)	

© Siemens AG, 2003. All rights reserved

5.1.2 Signals from channel

DB21, ... **DBX36.2** Data block	**All axes referenced** Signal(s) from channel (NCK –> PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	All axes (linear axes and rotary axes) referenced. MD 20700: REFP_NC_START_LOCK (NC start disable without reference point) is zero. If two position measurement systems which would inhibit an NC start are connected to an axis, the active one must be referenced if the axis is to qualify as referenced. Only if this signal is present, NC Start command for parts program processing is accepted. The axes must have a reference point, if MD 34110: REFP_CYCLE_NR $\neq -1$ and the axis is not in the parking position (position measurement system inactive and controller enable canceled)		
Signal state 0 or signal transition 1 ——> 0	One or more axes on the channel have not been referenced.		
Special cases, errors, ...	The spindles of the channel have no effect on this interface signal.		
Related to	IS “Referenced/synchronized” 1 (DB31, ... DBX60.4) IS “Referenced/synchronized” 2 (DB31, ... DBX60.5)		

5.2 Axis/spindle-specific signals

5.2.1 Signals to axis/spindle

DB 31, ... **DBX2.4 to 2.7** Data block	**Reference point value 1 to 4** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	When the machine axis arrives at the reference point, the NCK is informed which coded reference cam is approached. The “Reference point value 1 to 4” interface signal must remain enabled until the reference point has been reached or until a new coded reference cam is approached. If the machine axis has arrived at the referenced point (the axis is stationary), the reference point value selected by the “Reference point value 1 to 4” interface signal from MD 34100: REFP_SET_POS is accepted by the control as the new reference position.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Signal irrelevant for ...	Linear measurement systems with distancecoded reference marks		
Application example(s)	On a machine tool with large traversing distances, four coded reference cams can be distributed over the entire distance traveled by the axis, four different reference points approached and the time required to reach a valid referenced point reduced.		
Special cases, errors, ...	If the machine axis has arrived at the reference point and none of the four Reference point value 1 to 4 interface signals are enabled, the value of the reference point is automatically set to 1.		
Related to	MD 34100: REFP_SET_POS (reference point value)		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX12.7** Data block	**Reference point approach delay** Signal(s) to axis/spindle (PLC –> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The machine axis is positioned on the reference cam.	
Signal state 0 or signal transition 1 ——> 0	The machine axis is positioned in front of the reference cam. A reference cam of appropriate length (to the end of the traversing range) has been used to prevent the machine axis from being positioned behind the referencing cam.	
Related to	“Reference point value 1 to 4” interface signal (DB31, ... DBX2.4 to 2.7)	

5.2.2 Signals from axis/spindle

DB 31, ... **DBX60.4** Data block	**Referenced/synchronized 1** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation:	Signal(s) updated:	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Axes: When, during a reference point approach, the machine axis has reached the reference point (incremental measurement systems) or destination point (linear measurement system with distance-coded reference marks), the machine axis is deemed to have been referenced and the “Referenced/synchronized 1” interface signal (depending on which position measurement system is active during referencing) is enabled. Spindles: Following POWER ON, a spindle is synchronized, at the earliest, after one spindle revolution of 360 degrees (the zero mark has been crossed or the BERO proximity switch addressed).	
Signal state 0 or signal transition 1 ——> 0	The machine axis/spindle with position measurement system 1 is not referenced/ synchronized.	
Related to	IS “Synchronized 1” (DB31, ... DBX1.5)	
References	/FB/, S1, “Spindles”	

DB 31, ... **DBX60.5** Data block	**Referenced/synchronized 2** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation:	Signal(s) updated:	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	Axes: When, during a reference point approach, the machine axis has reached the reference point (incremental measurement systems) or destination point (linear measurement system with distance-coded reference marks), the machine axis is deemed to have been referenced and the “referenced/synchronized 2” interface signal (depending on which position measurement system is active during referencing) is enabled. Spindles: Following POWER ON, a spindle is synchronized, at the earliest, after one spindle revolution of 360 degrees (the zero mark has been crossed or the BERO proximity switch addressed).	
Signal state 0 or signal transition 1 ——> 0	The machine axis/spindle with position measurement system 2 is not referenced/ synchronized. Axes: Alarm 21610 has been issued Spindles: Encoder limit frequency exceeded	
Related to	IS “Synchronized 2” (DB31, ... DBX1.6), MD 34102 = 0	
References	/FB/, S1, “Spindles”	

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

6 Example

None

■

7 Data Fields, Lists

7.1 Interface signals

DB number	Bit, byte	Name	Refer-ence
Mode group-specific			
11, ...	0.7	Mode group reset	K1
11, ...	1.2	Machine function REF	K1
11, ...	5.2	Active machine function REF	K1
Channel-specific			
21, ...	1.0	Activate referencing	
21, ...	28.7	(MMC ——> PLC) REF	K1
21, ...	33.0	Referencing active	
21, ...	35.7	Reset	K1
21, ...	36.2	All axes referenced	
Axis-specific			
31, ...	1.4	Follow-up mode (request)	A2
31, ...	1.5 / 1.6	Position measurement system 1/Position measurement system 2	A2
31, ...	2.4–2.7	Reference point value 1 to 4	
31, ...	4.6 / 4.7	Traversing keys minus/plus	H1
31, ...	12.7	Reference point approach delay	
31, ...	60.4 / 60.5	Referenced, synchronized 1/Referenced, synchronized 2	
31, ...	61.3	Follow-up mode active	A2
31, ...	64.6 / 64.7	Traverse command minus/plus	H1

© Siemens AG, 2003. All rights reserved

7.2 Machine data

Number	Identifier	Name	Reference
General ($MN_ ...)			
11300	JOG_INC_MODE_LEVELTRIGGRD	INC/REF in jog/continuous mode	H1
Channel-specific ($MC_ ...)			
20700	REFP_NC_START_LOCK	NC start disable without reference point	
Axis-specific ($MA_ ...)			
30200	NUM_ENCS	Number of encoders	G2
30240	ENC_TYP	Actual value encoder type	
30242	ENC_IS_INDEPENDENT	Encoder is independent	G2
30250	ACT_POS_ABS	Absolute encoder position at time of deactivation.	
30330	MODULO_RANGE	Magnitude of the modulo range	R2
30340	MODULO_RANGE_START	Starting position of modulo range	R2
30355	MISC_FUNCTION_MASK	Axis functions	R2
31122	BERO_DELAY_TIME_PLUS	BERO delay time in plus direction	S1
31123	BERO_DELAY_TIME_MINUS	BERO delay time in minus direction	S1
34000	REFP_CAM_IS_ACTIVE	Axis with reference cam	
34010	REFP_CAM_DIR_IS_MINUS	Reference point approach in minus direction	
34020	REFP_VELO_SEARCH_CAM	Reference point approach velocity	
34030	REFP_MAX_CAM_DIST	Maximum distance to reference cam	
34040	REFP_VELO_SEARCH_MARKER[n]	Reference point creep speed [encoder number]	
34050	REFP_SEARCH_MARKER_REVERSE[n]	Change of direction on reference cam [encoder number]	
34060	REFP_MAX_MARKER_DIST[n]	Maximum distance to reference mark; Maximum distance to 2 reference marks with distancecoded scales [encoder number]	
34070	REFP_VELO_POS	Reference point start velocity	
34080	REFP_MOVE_DIST[n]	Reference point distance/destination for distancecoded system [encoder number]	
34090	REFP_MOVE_DIST_CORR[n]	Reference point/absolute offset, distancecoded [encoder number]	
34092	REFP_CAM_SHIFT	Electronic reference cam shift for incremental measurement systems with equidistant zero marks.	
34093	REFP_CAM_MARKER_DIST	Reference cam/reference mark distance	
34100	REFP_SET_POS[n]	Reference point value [reference point number]	
34102	REFP_SYNC_ENCS	Actual value adjustment to the referencing measurement system	
34104	REFP_PERMITTED_IN_FOLLOWUP	Enable referencing in follow-up mode	
34110	REFP_CYCLE_NR	Axis sequence for channel-specific Referencing	
34120	REFP_BERO_LOW_ACTIVE	Polarity change of the BERO cam	
34200	ENC_REFP_MODE[n]	Referencing mode [encoder number]	

© Siemens AG, 2003. All rights reserved

Axis-specific ($MA_ ...)			
34210	ENC_REFP_STATE[n]	Status of absolute value encoder [encoder number]	
34220	ENC_ABS_TURNS_MODULO	Absolute value encoder range for rotary encoders	R2
34230	ENC_SERIAL_NUMBER	Encoder serial number	
34232	EVERY_ENC_SERIAL_NUMBER	Expansion of encoder serial number	
34300	ENC_REFP_MARKER_DIST[n]	Reference marker distance with distance-coded scales [encoder number]	
34310	ENC_MARKER_INC[n]	Differential distance between two reference markers with distance-coded scales [encoder no.]	
34320	ENC_INVERS[encoder]	Linear measuring system inverse to machine system [encoder number]	
34330	REFP_STOP_AT_ABS_MARKER[n]	Distancecoded linear measurement system without destination point [encoder number]	
35150	SPIND_DES_VELO_TOL	Spindle speed tolerance	S1
36302	ENC_FREQ_LIMIT_LOW	Encoder limit frequency resynchronization	A3
36310	ENC_ZERO_MONITORING	Zero mark monitoring	A3

7.3 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Spindles (S1)

1	**Brief Description**		**1/S1/1-3**
2	**Detailed Description**		**1/S1/2-5**
	2.1	Spindle modes	1/S1/2-5
	2.1.1	Spindle control mode	1/S1/2-7
	2.1.2	Spindle mode oscillation mode	1/S1/2-10
	2.1.3	Spindle positioning mode	1/S1/2-10
	2.1.4	Axis mode	1/S1/2-20
	2.1.5	Default mode setting	1/S1/2-23
	2.2	Referencing/synchronizing	1/S1/2-24
	2.3	Configurable gear adaption	1/S1/2-28
	2.3.1	Gear stage for spindles and gear change	1/S1/2-28
	2.3.2	Encoder on intermediate gear (SW 6.4 and higher)	1/S1/2-35
	2.3.3	Non-acknowledge gear stage change (SW 5.3 and higher)	1/S1/2-36
	2.3.4	Gear stage change with oscillation mode	1/S1/2-37
	2.3.5	Gear stage change at fixed position (SW 5.3 and higher)	1/S1/2-41
	2.4	Spindle auxiliary functions	1/S1/2-45
	2.4.1	No output of spindle auxiliary functions after block search (SW 5.3 and higher)	1/S1/2-45
	2.5	Selectable spindles	1/S1/2-48
	2.6	Programming	1/S1/2-51
	2.6.1	Programing from the parts program	1/S1/2-51
	2.6.2	Programming via synchronized actions	1/S1/2-54
	2.6.3	Programming spindle controls via the PLC with FC18	1/S1/2-55
	2.6.4	Special spindle movements via the PLC interface (SW 6.1 and higher)	1/S1/2-55
	2.6.5	Gear stage change in DryRun, Program Test and SERUPRO modes (SW 6.2 and higher)	1/S1/2-62
	2.6.6	Programming from external (PLC, MMC/HMI)	1/S1/2-64
	2.7	Spindle monitoring	1/S1/2-65
	2.7.1	Axis/spindle stationary (n < nmin)	1/S1/2-66
	2.7.2	Spindle in setpoint range	1/S1/2-66
	2.7.3	Minimum/maximum speed for the gear stage	1/S1/2-67
	2.7.4	Maximum encoder limit frequency	1/S1/2-67
	2.7.5	End point monitoring	1/S1/2-69

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions 1/S1/4-71

4 Data Descriptions (MD, SD) 1/S1/4-71

4.1 General machine data 1/S1/4-71

4.2 Channel-specific machine data 1/S1/4-73

4.3 Axis/spindle-specific machine data 1/S1/4-75
4.3.1 Referencing/synchronizing 1/S1/4-75
4.3.2 Intermediate gear 1/S1/4-76
4.3.3 Gear step change 1/S1/4-77
4.3.4 Assignments/settings for spindle and spindle speed 1/S1/4-81

4.4 Spindle-specific setting data 1/S1/4-92

5 Signal Descriptions 1/S1/5-97

5.1 Axis/spindle-specific signals 1/S1/5-97
5.1.1 Signals to axis/spindle 1/S1/5-97
5.1.2 Signals from axis/spindle 1/S1/5-103

6 Examples 1/S1/6-109

6.1 Example of automatic gear stage selection (M40) 1/S1/6-109

7 Data Fields, Lists 1/S1/7-111

7.1 Interface signals 1/S1/7-111

7.2 Machine data 1/S1/7-112

7.3 Setting data 1/S1/7-114

7.4 Alarms 1/S1/7-115

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

Application of spindles

The primary function of a spindle is to impart a rotary motion to a tool or workpiece, in order to facilitate machining.

Depending on the type of machine, the spindle must support the following functions in order to achieve this:

- Definition of a direction of rotation for the spindle (M3, M4)
- Definition of a spindle speed (S ...)
- Spindle stop without orientation (M5)
- Spindle stop with orientation/spindle positioning (SPOS, M19 and SPOSA)
- Gear change (M40 to M45)
- Spindle-axis functionality (spindle becomes rotary axis and vice versa)
- Thread cutting (G33, G34, G35)
- Tapping with and without compensating chuck (G331, G332)
- Revolutional feedrate (G95)
- Constant cutting rate G96, G961, G97, G971)
- Programmable spindle speed limits (G25, G26, LIMS)
- Position encoder assembly on the spindle or on the spindle motor
- Spindle monitoring for min. and max. speed as well as max. encoder limit frequency and end point monitoring of spindle
- Switching the position control (SPCON, SPCOF, M70) on/off
- Programming of spindle functions
  from the parts programs,
  via synchronized actions,
  via the PLC with FC18 or via the special spindle interface
  for simple spindle controls.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

Max. number of spindles

The spindles can be assigned to the available channels.

Table 2-1 Maximum number of spindles

SINUMERIK	Maximum number of spindles	Maximum number of channels
810D (CCU1)	1	1
810D (CCU2, SW 2.1 and lower)	2	2
840D (NCU 571)	2	1
840D (NCU 572/573)	5	2
840D (NCU 573, from SW 4.1)		
with 8MB DRAM	12 axes (spindles)	3[1)]
with 32MB DRAM	31 axes (spindles) (5 spindles/channel)	10
FM-NC (NCU 570)	2	1

With SW 4.4 and higher, it is possible to configure each axis existing in the channel as a spindle. The number of axes per channel depends on the control version.

2.1 Spindle modes

Spindle modes

The spindle can have the following modes:

- Control mode — See Subsection 2.1.1
- Oscillation mode — See Subsection 2.1.2
- Positioning mode — See Subsection 2.1.3
- Synchronous mode Synchronous Spindles (S3) — see /FB/ Part 2, Subsection 2.1.1
- Rigid tapping — see Feeds (V1), Subsection 2.1.5

References: /PA/, “Programming Guide Fundamentals”
/PAZ/, “Programming Guide: Cycles”

1) possibly due to memory: 2

© Siemens AG, 2003. All rights reserved

Axis mode

The spindle can be switched from spindle mode to axis mode (rotary axis) if the same motor is used for spindle and axis operation:

- Axis mode See Subsection 2.1.4

Changing the spindle mode

Interface signals or programming commands can be used to switch between the spindle modes and axis operation:

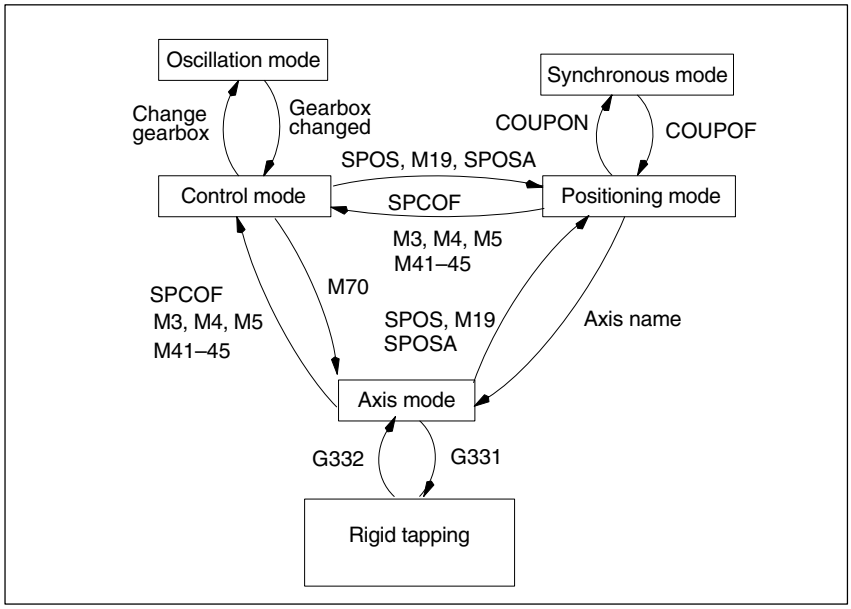


Fig. 2-1 Switching between spindle modes

- Open-loop control mode ——> Oscillation mode
  The spindle changes to oscillation mode if a new gear stage has been specified using the automatic gear stage selection (M40) in conjunction with a new S value or by M41 to M45. The spindle only changes to oscillation mode if the new gear stage is not equal to the current actual gear stage.
- Oscillation mode ——> control mode
  If the new gear stage is engaged, IS “Oscillation mode” (DB31, ... DBX84.6) is reset and IS “Gear changed” (DB31, ... When the new gear is engaged, the Oscillation mode interface signal (DB31, ... DBX84.6) is reset and the spindle is changed to control mode with the Gear changed interface signal (DB31, ... DBX16.3). The last programmed spindle speed (S value) is reactivated.
- Control mode ——> positioning mode
  To stop the spindle from rotation (M3 or M4) with orientation or to reorient it from standstill (M5), SPOS, M19 or SPOSA are used to switch to positioning mode.
- Positioning mode ——> control mode
  M3, M4 or M5 are used to change to the control mode if the orientation of the spindle is to be terminated. The last programmed spindle speed (S value) is reactivated.

© Siemens AG, 2003. All rights reserved

- Positioning mode ——> oscillation mode
  If the orientation of the spindle is to be terminated, M41 to M45 can be used to change to When the gear change is complete, the last programmed spindle speed (S value) and M5 (control mode) are reactivated.
- Positioning mode ——> axis mode
  If a spindle was stopped with orientation, the assigned axis name is used to program a change to axis mode. The gear stage is retained.
- Control mode ——> axis mode
  Switching from control mode into axis mode can be also achieved through programming of M70. In this case, a rotating spindle is decelerated in the same way as for M5, the position control activated and the zero parameter set selected.
- Axis mode ——> control mode
  To terminate axis mode, M3, M4 or M5 can be used to change to control mode. The last programmed spindle speed (S value) is reactivated.
- Axis mode ——> oscillation mode
  To terminate axis mode, M41 to M45 can be used to change to oscillation mode (only if the programmed gear stage is not equal to the actual gear stage). When the gear change is complete, the last programmed spindle speed (S value) and M5 (control mode) are reactivated.

2.1.1 Spindle control mode

When is control mode used?

The spindle is in control mode with the following functions:

- constant spindle speed S.... M3/M4/M5 and G93, G94, G95, G97, G971
- constant cutting rate G96/G961 S.... M3/M4/M5
- constant spindle speed S.... M3/M4/M5 and G33, G34, G35
- constant spindle speed S.... M3/M4/M5 and G63

Preconditions

- The spindle need not be synchronized
- A spindle position actual value encoder is not required for M3/M4/M5 in conjunction with time reciprocal feed coding (G93), feedrate in mm/min or inch/min (G94) and rigid tapping (G63).
  Feedrate in mm/min or inch/min (G94) and tapping with chuck (G63).
- A spindle position actual value encoder is a mandatory requirement for M3/M4/M5 in conjunction with revolutional feedrate (G95),
  constant cutting speed (G96, G961, G97 G971),
  thread cutting (G33, G34, G35),
  tapping (G331, G332) and
  activate position control (SPCON, M70).

© Siemens AG, 2003. All rights reserved

Position control ON/OFF

The spindle can be operated with or without position control:

- SPCON (position control ON)
- SPCOF (position control OFF)

Speed control mode

Speed control mode is especially suitable when a constant spindle speed is required, but the position of the spindle is unimportant (e.g. constant milling speed for even appearance to the workpiece surface).

- The speed control mode is activated in the parts program with M3, M4, M5 or with SPCOF.
- The “Control mode” interface signal (DB31, ... DBX84.7) is set.
- The IS “Position controller active” (DB31, ... DBX61.5) is reset if machining is carried out without position control.
- The acceleration is defined in MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL depending on the particular gear stage and should correspond to the physical conditions if possible.

“Position controller active” DB31, ... DBX61.5

Position control is particularly suitable when the position of the spindle is required to be tracked over a longer period or when synchronous spindle setpoint value linkage is to be activated.

- Position control mode is activated in the parts program with SPCON (spindle number).
- The IS “Position controller active” (DB31, ... DBX61.5) is enabled.
- The acceleration in position control mode is defined in MD 35210: GEAR_STEP_POSCTRL_ACCEL depending on the gear stage.

Independent spindle reset

The MD 35040: SPIND_ACTIVE_AFTER_RESET defines how the spindle behaves after a reset or program end (M2, M30):

- When SPIND_ACTIVE_AFTER_RESET is reset, the spindle immediately decelerates to a stop with the active acceleration. The last programmed spindle speed and direction of rotation are deleted.
- When SPIND_ACTIVE_AFTER_RESET is enabled, the past programmed spindle speed (S value) and the last programmed direction of spindle rotation (M3, M4, M5) are retained.
  The spindle is not decelerated.

  If prior to reset or end of program the constant cutting speed (G96, G961) is active, the current spindle speed (referred to 100% spindle override) is internally accepted as the spindle speed last programmed.
  The spindle can only be stopped with the IS “Delete distance-to-go/Spindle Reset” (DB31, ... DBX2.2).

  In SW 5.2 and higher, the direction of rotation is deleted in case of all alarms creating quick stop for the spindle. The last programmed spindle speed (S value) is kept.
  After the cause of the alarm is eliminated, the spindle must be restarted.

© Siemens AG, 2003. All rights reserved

(SW 5.3 and higher)

Spindle actual speed display and spindle behavior with G96, G961

"Axis/spindle stops" DB31, ... DBX61.4

The MD 36060: STANDSTILL_VELO_TOL is used to set the speed at which the spindle is considered "stationary". When doing so, the speed set in the machine data should be dimensioned such that the IS "Axis/spindle stationary" (DB31, ...DBX61.4) is reliably present at a standstill.

- If the IS "Axis/spindle stationary" is signaled and no position control is active for the spindle, zero is displayed for the actual speed on the MMC and zero is read with the system variable $AA_S[n].

For more detailed information on interface signals from and to axis/spindle (DB31, ...), please refer to:
References: /FB1/, A2, "Various Interface Signals and Functions"

Spindle behavior with G96, G961

Constant cutting rate

- – At the start of machining (transition from G0 to Gx), and after:
  – NC Stop,
  – G60 exact stop, modal,
  – G09 exact stop, nonmodal
  to perform the path start, it is waited until the actual speed has reached the tolerance range of the set speed. The signal "nIst=nSoll" (DB31, ... DBX83.5) is set.
- The signals "nIst = nSoll" (DB31, ... DBX83.5) and "Set speed limited" (DB31, ... DBX83.1) are also set to a defined state for high speed changes (face axis travels to position 0).
- When the speed falls below the minimum speed or if the signal "Axis/spindle stationary" is recognized (DB31, ... DBX61.4) the signal "nIst = nSoll" (DB31, ... DBX83.5) is reset. (e.g. for machine emergency strategy).
- A path operation which has started (G64, overgrinding) is not interrupted.

The influence of MD 35500: SPIND_ON_SPEED_AT_IPO_START is also significant for the spindle response.

Spindle behavior at the end of gear stage change

End of gear stage change "Gear changed" (DB31, ... DBX16.3)

- The IS "Gear changed" (DB31, ... DBX16.3) tells the NC that the new gear stage (IS "Actual gear stages A to C" (DB31, ... DBX16.0–16.2)) applies and oscillation mode is quitted. It does not matter whether IS "Oscillation speed" (DB31, ... DBX18.5) is still set. The actual gear stage should correspond to the set gear stage. Otherwise, alarm 22010 is signaled if MD 11410: SUPPRESS_ALARM_MASK, Bit 3 = 0. The actual gear stage reported is relevant for selection of the set of parameters.
- After the gear stage change has been acknowledged by the PLC (DB31, ...; DBX16.3), the spindle is in speed control mode (DB31, ... DBX84.7 = 1). DBX84.7 = 1). If a direction of rotation (M3, M4, M5 or FC18: "Start spindle rotation"), then the last speed and direction of rotation will become active again after the gear stage change.

© Siemens AG, 2003. All rights reserved

2.1.2 Spindle mode oscillation mode

Scope

Oscillation mode is activated for the spindle during the gear stage change. For a detailed description of the method of operation of the spindle mode oscillation mode, please refer to the Subsection 2.3.4 “Gear stage change with oscillation mode”.

2.1.3 Spindle positioning mode

When is positioning mode used?

The spindle positioning mode stops the spindle at the defined position and activates the position control, which remains active until it is deactivated. The spindle is in positioning mode with the following functions. The parameter [n] with, where n=spindle number, is not required for the main spindle.

- SPOS [n] = ...
- SPOS [n] = ACP(.....)
- SPOS [n] = ACN(.....)
- SPOS [n] = AC(.....)
- SPOS [n] = IC(.....)
- SPOS [n] = DC(.....)
- SPOSA [n] = ACP(.....)
- SPOSA [n] = ACN(.....)
- SPOSA [n] = AC(.....)
- SPOSA [n] = IC(.....)
- SPOSA [n]=DC(.....) identical to SPOSA [n]= .....
- M19 or M [n]=19

SPOS [n] = AC(.....)

Spindle positioning to an absolute position (0 to 359.999 degrees). The positioning direction is determined either by the current direction of spindle rotation (spindle is rotating) or the distancetogo.

SPOS [n] = IC(.....)

Spindle positioning to an incremental position (+/– 999999.99 degrees) in relation to the last programmed position. The positioning direction is defined by the sign of the distance to be traveled.

SPOS [n] = DC(.....)

Spindle positioning across the shortest path to an absolute position (0 to 359.999 degrees). The positioning direction is determined either by the current direction of spindle rotation (spindle is rotating) or automatically by the control (spindle is stationary).

© Siemens AG, 2003. All rights reserved

SPOS [n] =.....

Identical function to SPOS [n] = DC(.....).

SPOS [n] = ACP(.....)

Approaches the position from the plus direction.

When positioning form a negative direction of rotation, the speed is brought to a halt and accelerated in the opposite direction to execute the positive approach.

SPOS [n]=ACN(....)

Approaches the position from the negative direction.

When positioning form a positive direction of rotation, the speed is brought to a halt and accelerated in the opposite direction to execute the negative approach.

MD 20850 (SW 5.3 and higher)

In order to achieve consistency in the response at the VDI interface for “M19” and SPOS or SPOSA, it can be programmed to output the auxiliary function M19 at the VDI interface with SPOS and SPOSA and MD 20850: SPOS_TO_VDI = 1 is entered.
After activation the minimum duration of a SPOS/SPOSA block is increased to the time for output and acknowledgment of the auxiliary function by the PLC.

M19 (DIN 66025) (SW 5.3 and higher)

M19 can be used to position the spindle. The position and the position approach mode are read from SD 43240: M19_SPOS and SD 43250: M19_SPOSMODE. The positioning possibilities with M19 are identical to those of SPOS = < > <>.

M19 is output as an auxiliary function at the VDI interface as an alternative to M3, M4, M5 and M70. The M19 block remains active in the interpolator for the duration of the positioning (like SPOS).

Parts programs using M19 as a macro (e.g. DEFINE M19 AS SPOS = 0) or as a subroutine continue to remain executable. For the sake of compatibility to previous controls, the internal processing of M19 (NCK positions the spindle) can be disabled as shown in the following example:

```
MD 22000: AUXFU_ASSIGN_GROUP[0] = 4          ; Auxiliary function group: 4
MD 22010: AUXFU_ASSIGN_TYPE[0] = “M”         ; Auxiliary function type: “M“
MD 22020: AUXFU_ASSIGN_EXTENSION[0] = 0      ; AuxFuncExtension: 0
MD 22030: AUXFU_ASSIGN_VALUE[0] = 19         ; Auxiliary function value: 19
```

SD 43240 and SD 43250

The positioning data for M19 are stored in the axis-specific setting data SD 43240: M19_SPOS[n] (position) and SD 43250: M19_SPOSMODE[n] (position approach mode).

Positioning end (SW 5.1 and higher)

The positioning end can be programmed using the following commands:

- FINEA [Sn] — End of movement before reaching “Exact stop fine” (DB31, ... DBX60.7)
- COARSEA [Sn] — End of movement before reaching “Exact stop coarse” (DB31, ... DBX60.6)
- IPOENDA [Sn] — Motion ends when “IPO Stop” is reached

© Siemens AG, 2003. All rights reserved

Block change

Block change time settable (SW 6.2 and higher)

In addition to the existing programmable endofmotion criterion with FINEA, COARSEA, IPOENDA, for single-axis interpolation a new end-of-motion criterion can be programmed with IPOBRKA in the braking ramp (100–0%).

If the end-of-motion criteria are fulfilled for all spindles or axes acting in the block and, in addition, the end-of-block criterion is fulfilled for the path interpolation, the block change is carried out. This applies both to parts program blocks and for technology cycle blocks.

Note

For further explanations on the settable block change time of settable positioning axes for single axis interpolation, please refer to:
References: /FB/, P2, “Positioning Axes”

SPOS, M19 and SPOSA have the same functionality, but differ in their block change behavior:

- Programming with SPOS and M19

  The block change is carried out if all functions programmed in the block have reached their endofblock criterion (e.g. all auxiliary functions acknowledged by the PLC, all axes have reached their end points) and the spindle has reached its end point.

- Programming with SPOSA

  The block change occurs when all the functions programmed in the block (except the spindle) have acquired their block end criterion. If SPOSA is the only entry in the block, block change is performed immediately. The spindle positioning can range over several blocks (see WAITS).

Coordination

A coordination of the sequence of motions can be achieved with:

- WAITS for main spindle
- WAITS[n] for main spindles and other spindles
- WAITS[n, m, ...,q] for several spindles up to the maximum

The execution of the following blocks is suspended until the spindle or spindles programmed with SPOSA have reached their positions (see end position).

M function output (SW 5.3 and higher)

The output of the auxiliary function M19 to the VDI interface is carried out by:

- M[n]=19 always output to the interface
- SPOS[n] output of M19 to the interface with MD 20850 = 1
- SPOSA[n] output of M19 to the interface with MD 20850 = 1.

If M19 is programmed, the auxiliary function M [n]=19 is always output to the interface. In the case of SPOS and SPOSA, M19 is output to the interface depending MD 20850: SPOS_TO_VDI.

© Siemens AG, 2003. All rights reserved

Feedrate

The positioning velocity is configured by MD 35300: SPIND_POSCTRL_VELO and can be changed by programming or by using synchronized actions:

- FA[Sn] where n=spindle number
- FA[Sn]=0 the configured speed is activated

The speed is specified in [degrees/min].

Acceleration

The dynamic response during positioning can be modified by programming or by synchronized actions:

- ACC[Sn] Programming or synchronized action
- ACC[Sn]=0 The configured acceleration is active

n: Spindle number, 0 ... max. spindle number.

© Siemens AG, 2003. All rights reserved

Positioning from rotation

The spindle can be in speed control mode or in position control mode when positioning is started (by the SPOS, M19 or SPOSA command in the program). The following sequence is obtained:

- Case 1: Spindle in speed control mode, encoder limit frequency exceeded
- Case 2: Spindle in speed control mode, encoder limit frequency not exceeded
- Case 3: Spindle in position control mode
- Case 4: Spindle speed < position control activation speed

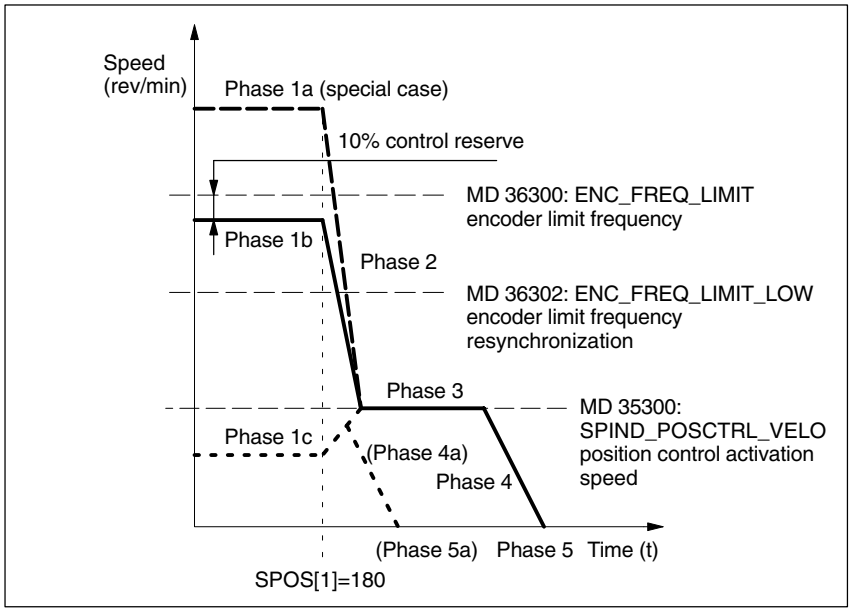


Fig. 2-2 Positioning from rotation

Note

The speed in MD 36302: ENC_FREQ_LOW must be greater than the position control activation speed (MD 35300: SPIND_POSCTRL_VELO).

Phase 1

Possible positioning from phase 1a:

The spindle is rotating at a higher speed than the encoder limit frequency. It is not synchronized.

Possible positioning from phase 1b:

Spindle rotates at a lower speed than the encoder limit frequency. It is synchronized.

© Siemens AG, 2003. All rights reserved

Note

If the position control is activated, the speed can only amount to 90% of the maximum speed of the spindle or the encoder limit frequency (10% control reserve required).

Possible positioning from Phase 1c:

The spindle is rotating at the programmed spindle speed, which is lower than that defined in MD 35300: SPIND_POSCTRL_VELO (position control activation speed). It is synchronized.

Phase 2

- Spindle speed > position control activation speed

  When the command SPOS, M19 or SPOSA comes into effect, the spindle starts decelerating with the position control activation speed defined in MD 35200: GEAR_STEP_SPEEDCTL_ACCEL. When the speed drops below the encoder limit frequency, the spindle is synchronized.

- Spindle speed < position control activation speed

  When SPOS, M19 or SPOSA is programmed, the spindle is switched to position control mode (if it is not yet in position control mode).
  MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode) is activated. The travel path to the destination point is calculated.

  The spindle travels to the programmed destination point optimally in terms of time. i.e., the destination point is approached at the highest possible velocity (but not higher than MD 35300: SPIND_POSCTRL_VELO). Depending on the appropriate supplementary conditions, the operational sequences in phases 2 –3 – 4 –5 or 2 –4a – 5a are executed.

Phase 3

- Spindle speed > position control activation speed

  When the position control activation speed defined in MD 35300: SPIND_POSCTRL_VELO is reached,

  – the position control is activated (if not already active),

  – the distance-to-go (to destination point) is calculated

  – the acceleration defined in MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode) activated or maintained

- Spindle speed < position control activation speed

  To reach the destination point, the spindle has accelerated to the speed set MD 35300: SPIND_POSCTRL_VELO (position control activation speed). This speed is not exceeded. The braking start point calculation detects at what instant the spindle can accurately approach the programmed position at the acceleration defined in MD 35210: GEAR_STEP_POSCTRL_ACCEL.

© Siemens AG, 2003. All rights reserved

Phase 4

- Spindle speed > position control activation speed

  The spindle brakes from the calculated “braking point” with MD 35210: GEAR_STEP_POSCTRL_ACCEL down to the destination point.

- Spindle speed < position control activation speed

  At the instant in time detected by the brake start point calculation in phase 3, the spindle brakes down to standstill at the acceleration set in MD 35210: GEAR_STEP_POSCTRL_ACCEL down to a standstill.

  Phase 4a:

  As command SPOS takes effect, the destination point is so close that the spindle can no longer be accelerated to the value in MD 35300: SPIND_POSCTRL_VELO. The spindle is braked down to standstill at the acceleration set in MD 35210: GEAR_STEP_POSCTRL_ACCEL.

Phase 5

- Spindle speed > position control activation speed

  The position control remains active and holds the spindle in the programmed position.

Note

The maximum encoder limit frequency of the spindle position actual value encoder is monitored by the control (overstepping is possible); in position control mode, the setpoint speed is reduced to 90% of the measurement system limit speed. In this case, the “Programmed speed too high” interface signal is enabled. If “MS limit frequency” exceeded is still detected after a reduction in the setpoint speed, then an alarm is output.

- Spindle speed < position control activation speed (Phase 5, 5a)

  The spindle is stationary and has reached the destination point. The position control is active and holds the spindle in the programmed position.

The IS “Position reached with exact stop coarse/fine” (DB31, ... DBX60.6 und DBX60.7) are set if the distance between the spindle actual position and the programmed position (spindle setpoint position) is less than the settings for the exact stop fine and coarse limits (defined in MD 36010: STOP_LIMIT_FINE and MD 36000: STOP_LIMIT_COARSE).

Note

The positioning operation is considered completed when the end-of-positioning criterion is signaled (see “Positioning end”). This requires “Exact stop fine”. This applies to SPOS, M19 or SPOSA from the parts program, synchronized actions and spindle positioning by the PLC using FC 18.

© Siemens AG, 2003. All rights reserved

Positioning from standstill

Positioning from standstill

A distinction is made between two cases with regard to positioning from standstill:

- Case 1: It is not synchronized. This is the case, if the spindle is to be positioned after switching on the control and drive (e.g. for a tool change). MD 31040: ENC_IS_DIRECT = 0.
- Case 2: It is synchronized. This is the case if, after switching on the control and drive, the spindle is to be rotated through a minimum of one revolution with M3 or M4 and then stopped with M5 (synchronization with the zero mark).

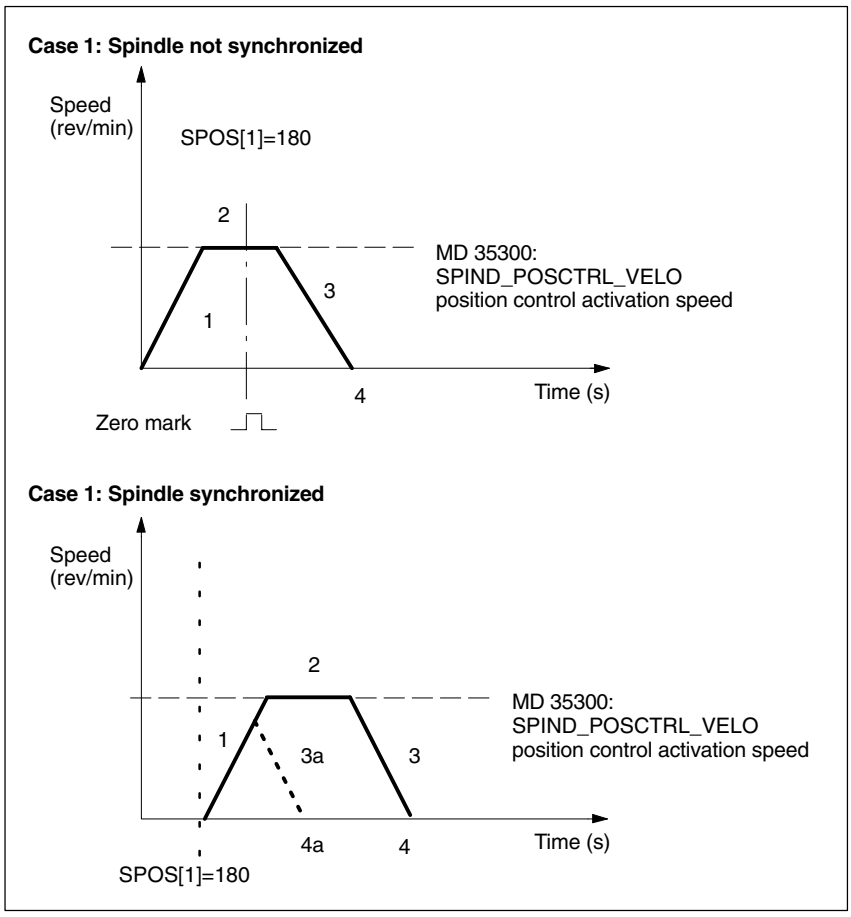


Fig. 2-3 Positioning with stationary spindle

© Siemens AG, 2003. All rights reserved

Phase 1

- Case 1: Spindle not synchronized

  If SPOS, M19 or SPOSA is programmed, the spindle will accelerate with the acceleration defined in MD 35200: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode). The direction of rotation is defined in MD 35350: SPIND_POSITIONING_DIR.

  Exception: If ACN, ACP, IC is used for positioning, the programmed direction of travel is activated.

  The spindle is synchronized with the next zero mark of the spindle position actual value encoder and switches to position control mode (see also Section 2.2). The zero mark in the path stored in MD 34060: REFP_MAX_MARKER_DIST is monitored to make sure it is detected (except for IC). If the speed entered in MD 35300: SPIND_POSCTRL_VELO (positioning speed) is reached but the spindle is not yet synchronized, the spindle continues to rotate at the positioning creep speed (the spindle is not accelerated further).

- Case 2: Spindle synchronized

  SPOS, M19 or SPOSA will switch the spindle to position control mode. The acceleration from MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode) is activated. The direction of rotation is defined by the programmed motion (ACP; ACN; IC; DC) or by the distance-to-go. The axis velocity defined in MD 35300: SPIND_POSCTRL_VELO (position control activation speed) is not exceeded. The travel path to the destination point is calculated.

  The spindle travels to the programmed destination point optimally in terms of time. i.e., the destination point is approached at the highest possible velocity (but not higher than MD 35300: SPIND_POSCTRL_VELO). Depending on the appropriate supplementary conditions, the operational sequences in phases 1 – 2 – 3 – 4 or 1 – 3a – 4a are executed.

Phase 2

- Case 1: Spindle not synchronized

  When the spindle is synchronized, the position control is activated. The spindle rotates at the speed defined in MD 35300: SPIND_POSCTRL_VELO until it detects the braking start point, at which point it can approach the programmed spindle position accurately with the defined acceleration.

- Case 2: Spindle synchronized

  To reach the destination point, the spindle has accelerated to the speed set MD 35300: SPIND_POSCTRL_VELO. This speed is not exceeded. The braking start point calculation detects at what instant the spindle can accurately approach the programmed position at the acceleration defined in MD 35210: GEAR_STEP_POSCTRL_ACCEL.

  At the instant in time detected by the brake start point calculation in phase 1, the spindle brakes down to standstill at the acceleration set in MD 35210: GEAR_STEP_POSCTRL_ACCEL down to a standstill.

© Siemens AG, 2003. All rights reserved

Phase 3

At the instant in time detected by the brake start point calculation in phase 1, the spindle brakes down to standstill at the acceleration set in MD 35210: GEAR_STEP_POSCTRL_ACCEL down to a standstill.

Phase 3a:

As command SPOS takes effect, the destination point is so close that the spindle can no longer be accelerated to the value in MD 35300: SPIND_POSCTRL_VELO. The spindle is braked down to a standstill at the acceleration set in MD 35210: GEAR_STEP_POSCTRL_ACCEL.

Phase 4, 4a

The spindle is stationary and has reached the destination point. The position control is active and holds the spindle in the programmed position. The IS “Position reached with exact stop coarse/fine” (DB31, ... DBX60.6 und DBX60.7) are set if the distance between the spindle actual position and the programmed position (spindle setpoint position) is less than the settings for the exact stop fine and coarse limits (defined in MD 36010: STOP_LIMIT_FINE and MD 36000: STOP_LIMIT_COARSE).

Phase 3:

At the instant in time detected by the brake start point calculation in phase 2, the spindle brakes down to standstill at the acceleration set in MD 35210: GEAR_STEP_POSCTRL_ACCEL down to a standstill.

Phase 4:

The spindle is stationary and has reached the destination point. The position control is active and holds the spindle in the programmed position. The IS “Position reached with exact stop fine/coarse” are set if the distance between the spindle actual position and the programmed position (spindle setpoint position) is less than the settings for the exact stop fine and coarse limits (defined in MD 36010: STOP_LIMIT_FINE and MD 36000: STOP_LIMIT_COARSE).

Abort: the positioning process

The positioning process can be interrupted with the “Delete distance-to-go/ spindle reset” interface signal. The positioning process is interrupted with each RESET (operator panel reset/“Delete distance-to-go/spindle RESET” interface signal), independent of MD 35040: SPIND_ACTIVE_AFTER_RESET (own spindle reset).

Special points to be noted

- The accelerations are defined in the following machine data:
  MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode)
  MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL (acceleration in speed control mode).
- The spindle override switch is active.
- The positioning (SPOS, M19 and SPOSA) is canceled with each Reset.
- Positioning is canceled with NC stop.
- The positioning velocity can also be programmed with FA[Sn].

© Siemens AG, 2003. All rights reserved

2.1.4 Axis mode

Why axis mode?

For certain machining tasks (e.g. on turning machines with end face machining), the spindle not only has to be rotated with M3, M4 and M5 and positioned with SPOS, M19 and SPOSA, but also addressed as an axis with its own identifier (e.g. C).

Preconditions

- The same spindle motor is used for spindle mode and axis mode.
- The same position measurement system or separate position measurement systems can be used for spindle mode and axis mode.
- A position actual value encoder is a mandatory requirement for axis mode.
- If the axis is not synchronized, e.g. M70 is programmed after POWER ON, first the axis must be referenced with G74. Only then does the mechanical position match the programmed one.
  Example:
  M70
  G74 C1=0 Z100
  G0 C180 X50

Configurable M function (SW 5.3 and higher)

The M function used to switch the spindle to axis mode can be configured in MD 20094: SPIND_RIGID_TAPPING_M_NR. In default condition, the value 70 is set.

Functionality

If the axis mode is active and the rotary axis referenced, all axis functions can be used.

References: /FB/, R2, “Rotary Axes”

The most important functions are:

- Programming with axis name
- Use of zero offsets (G54, G55, TRANS, ...)
- G90, G91, IC, AC, DC, ACP, ACN
- Use of kinematic transformations (e.g. transmit)
- Interpolation with other axes (path interpolation)
- Programming as a positioning axis

© Siemens AG, 2003. All rights reserved

Special points to be noted

- The feed override switch is active.
- IS “Reset” (DB21, ... DBX7.7) does not terminate the axis mode.
- Interface signals DBB16 to DBB19 and DBB82 to DBB91 in DB31, ... are irrelevant if the axis/no spindle interface signal (DB31, ... DBX60.0) is set to zero.
  DB31, ... are irrelevant if the “axis/no spindle” interface signal (DB31, ... DBX60.0) is set to zero.
- Axis mode can be activated in all gear stages. If the position actual value encoder is installed on the motor (indirect measurement system), the positioning and contouring accuracy can vary for the different gear stages.
- The gear stage cannot be changed when the axis mode is active. The spindle must be switched to control mode This is done using M41 ... M45 or M5, SPCOF.
- In axis mode, the machine data of the set of parameters with index zero are effective in order to carry out adaptation in this mode.

For further notes of the parameter block for the interpolation parameters for axis or spindle modes with 5 gear stages, see Section 2.3 Configurable gears.

Transition to axis mode

Transition to axis mode by programming:

- The spindle with its axis name or using M70 and the M function MD 20094: SPIND_RIGID_TAPPING_M_NR (SW 5.3 and higher).
- The following are the relevant machine data for changing the servo parameter set:
  MD 31050: DRIVE_AX_RATIO_DENOM (denominator measuring gearbox)
  MD 31060: DRIVE_AX_RATIO_NUMERA (numerator load gearbox)
  MD 32200: POSCTRL_GAIN (servo gain factor)
  MD 32452: BACKLASH_FACTOR (weighting factor for backlash)
  MD 32610: VELO_FFW_WEIGHT (weighting factor for feedforward control)
  MD 32800: EQUIV_CURRCTRL_TIME (equivalent time constant current circuit for feedforward control)
  MD 32810: EQUIV_SPEEDCTRL_TIME (equivalent time constant speed control circuit for feedforward control)
  MD 32910: DYN_MATCH_TIME (time constant for dynamic matching)
  MD 36012: STOP_LIMIT_FACTOR factor for exact stop coarse/fine and zero speed monitoring)
  MD 36200: AX_VELO_LIMIT (threshold value for velocity monitoring)
- The dynamic limits of the axis stored in the machine data are applicable in axis operation.
- The axis switches to the current feedforward control mode as designated by the MD and the commands FFWON and FFWOF.

For further notes on the servo parameter block, please refer to:
References: /FB1/,G2, “Velocities, Setpoint/Actual Value Systems, Closed-Loop Control”

© Siemens AG, 2003. All rights reserved

- When using resolution changes in (analog) drive actuators, the following NC program steps are required:
  - Change to **axis mode**
    SPOS= ...
    M5 Control enable off (from PLC)
    –> is output to PLC
    M70 Switchover actuator (from PLC due to M70),
    Control enable on (from PLC)
    C= ... NC travels to axes parameter set
  - Switch back to **spindle mode**
    C= ...
    M71 –> is output to PLC
    Control enable off (from PLC)
    Switchover actuator (from PLC)
    Internal switchover to spindle parameter set (1–5)
    in the NC, control enable on (from PLC)
    M3/4/5 or
    SPOS= ... NC travels with spindle parameter set

Change to spindle mode

- The interpolation parameter (set 1 ... 5) is selected according to the currently valid gear stage.
- The feedforward control function is always activated, except for tapping with compensating chuck. This is achieved by entering the absolute offset between the machine zero and the position of the 1st FFW_MODE (feedforward type) must always have a value different from 0.
- The feedforward control should always be operated with the value 100%, since alarms could otherwise be produced during positioning.

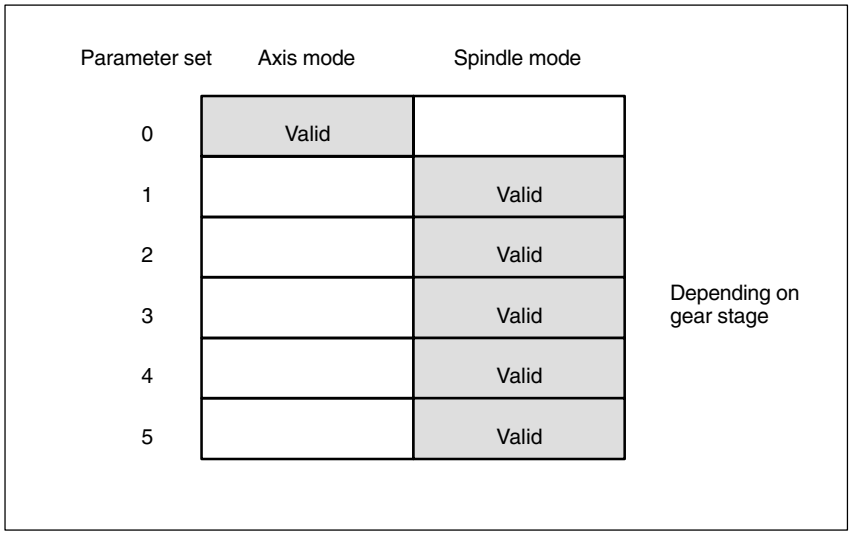


Fig. 2-4 Validity of parameter sets for axis and spindle modes

© Siemens AG, 2003. All rights reserved

2.1.5 Default mode setting

Machine data

With the default setting of the spindle operating mode, the basic spindle setting after POWER ON, NCSTART and RESET can be configured by means of machine data:
MD 35020: SPIND_DEFAULT_MODE and
MD 35030: SPIND_DEFAULT_ACT_MASK
SPIND_DEFAULT_ACT_MASK

Refer to Section 4.2 for information on setting the machine data.

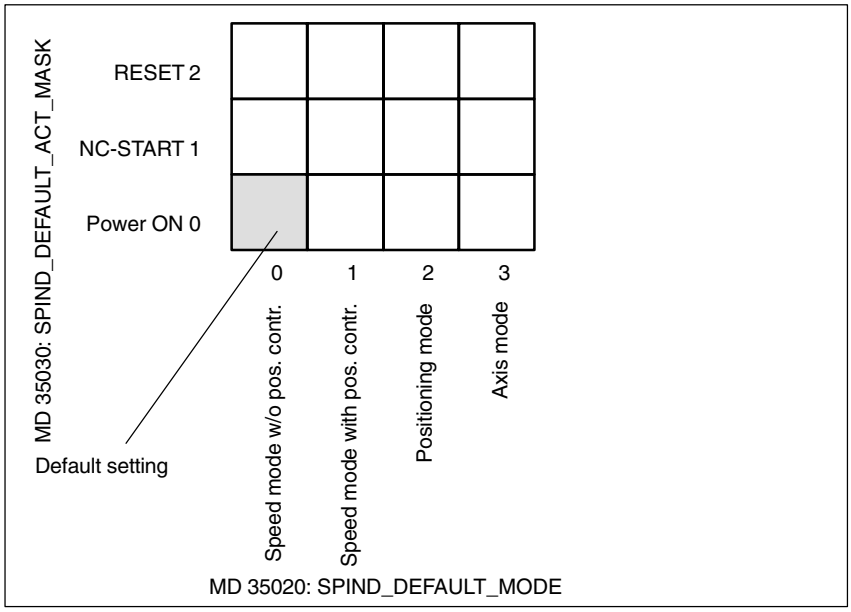


Fig. 2-5 Settable defaults for spindle operating mode

© Siemens AG, 2003. All rights reserved

2.2 Referencing/synchronizing

Note

For explanations on spindle synchronization and rotary axis referencing, please refer to:
References: /FB/, R1, “Reference Point Approach”

Why synchronize?

The control must be synchronized with the position measurement system on the spindle so that the control can know the exact position when switched on. This process is known as synchronization.

Only a synchronized spindle can be used for:

- Thread cutting
- Rigid tapping
- Programming as an axis.

Why reference?

The control must be synchronized with the position measurement system on the rotary axis so that the control can accurately detect the machine origin when switched on. This process is known as referencing. The sequence of operations required to reference an axis is known as reference point approach.

Only a referenced axis can approach a programmed position accurately on the machine.

Installation locations of the position measurement systems

The position measurement systems can be installed as follows:

- Direct on the motor in combination with a Bero proximity switch on the spindle as a zero mark encoder
- On the motor via a measurement gearbox in combination with a Bero proximity switch on the spindle as a zero mark encoder
- Directly on the spindle
- On the spindle via a measurement gearbox in combination with a Bero proximity switch on the spindle as a zero mark encoder (only with ratios not equal to 1:1).

Where two position measurement systems are provided, both of them can be installed at the same location or separately.

© Siemens AG, 2003. All rights reserved

Sequence synchronization

When the spindle is switched on, the spindle can be synchronized as follows:

- The spindle is started with a spindle speed (S value) and a spindle rotation (M3 or M4) and synchronized with the next zero mark of the position measurement system or with the next Bero signal.
- The spindle is to be positioned from the standstill using SPOS, M19 or SPOSA (see Subsection 2.1.3). The spindle synchronizes with the next zero mark of the position measurement system or with the next Bero signal. It is then positioned at the programmed position.
- The spindle can be synchronized from the movement (after M3 or M4) using SPOS, M19 or SPOSA.

  The responses are as follows:

  – SPOS = Pos, SPOS = DC(Pos) and SPOS = AC(Pos) retains the direction of motion and the position is approached.

  – With SPOS = ACN(Pos) or SPOS = ACP(Pos), the position is always approached with negative or positive direction of motion. If necessary, the direction of motion is inverted prior to positioning.
- If does not make any difference whether the procedure is initiated from the parts program, FC 18 or synchronized actions.
- Crossing the zero mark in JOG mode by means of direction keys in speed control mode.

Note

During synchronization of the spindle, all four possible reference point values and reference point offsets are effective as appropriate to the measurement system selected. The offset to the measurement system is analogous.
The following machine data must be observed:
MD 34080: REFP_MOVE_DIST (reference point distance/destination point for distance-coded system)
MD 34090: REFP_MOVE_DIST_CORR (reference point offset/absolute offset, distance-coded)
MD 34100: REFP_SET_POS (reference point value, irrelevant for distance-coded system).

Note

If a non-referenced spindle with SPOS=IC(...) and a path < 360 degrees is positioned, it can happen that the zero mark is not crossed and the spindle position is still not synchronized with the zero mark.

This can happen:

- After POWER ON
- By setting the axial IS “Resynchronize spindle when positioning 2 and 1” (DB31, ...DBX17.5 and 7.4)

© Siemens AG, 2003. All rights reserved

Special features for synchronization with BERO

The position falsification caused by the BERO signal delay can be corrected internally in the NC by entered a signal runtime compensation value.

The MD 31122: BERO_DELAY_TIME_PLUS or MD 31123: BERO_DELAY_TIME_MINUS contain a signal runtime compensation value (dead time) for positive or negative direction of movement together with the setting MD 34200: ENC_REFP_MODE = 2 or 7 (from SW 4.1).

- Setting MD 34200: ENC_REFP_MODE = 7 (from SW 3.6) carries out position synchronization only for a fixed velocity/speed defined in MD 34040: REFP_VELO_SEARCH_MARKER.

  The speed set in MD 34040 is also effective during referencing in JOG-REF mode and for the parts program with G74.

- Setting MD 34200: ENC_REFP_MODE = 2 carries out position synchronization without specifying a particular velocity/speed.

Note

Compensation of the signal runtime by the NC requires the use of type 611D drives.

The signal run times are preset on delivery so that the contents generally do not have to be changed.

Referencing sequence

If the spindle is to be programmed in axis mode directly after control power-up, it must be ensured that the axis is referenced. When the control is switched on, the spindle can be synchronized as follows (condition is one zero mark per revolution):

- For the procedure, please refer to:
  References: /FB/, R1, “Reference Point Approach”

- The rotary axis is referenced at the same time that the spindle is synchronized (see synchronization procedure) if the position measurement system used for the spindle is also used for the rotary axis.

Position measurement system, Spindle

The spindle can be switched from spindle mode to axis mode (rotary axis) if a single motor is used for spindle and axis mode. Please refer to Section 2.1 for details of how to switch the spindle between spindle and axis mode.

The spindle (spindle mode and axis mode) can be equipped with one or two position measurement systems. With two position measurement systems, it is possible to assign one position measurement system to the spindle and the other to the rotary axis, or to assign two position measurement systems to the spindle. Where two position measurement systems are provided, both are updated by the control, but only one can be active. The “position measurement system 1 and 2” interface signal (DB31, ... DBX1.5 and DBX1.6) is used to select the active position measurement system.

© Siemens AG, 2003. All rights reserved

The active position measurement system is required for the following functions:

- Position control of the spindle (SPCON)
- Spindle positioning (SPOS, M19 and SPOSA)
- Thread cutting (G33, G34, G35)
- Tapping without compensating chuck (G331, G332)
- Revolutional feedrate (G95)
- Constant cutting rate G96, G961, G97, G971)
- Spindle speed display
- Axis mode
- Synchronous spindle setpoint value linkage.

IS “Resynchronize spindle” DB31, ... DBX16.4

In the following cases, the spindle position measurement system must be resynchronized:

- The position encoder is on the motor, a Bero proximity switch is mounted on the spindle and a gear change is performed. The synchronization is triggered internally when the spindle is rotating in the new gear stage (see synchronization procedure).
- The machine has a selector switch for a vertical and horizontal spindle. Two different position encoders are used (one for the vertical spindle and one for the horizontal spindle), but only one actual value input is used on the control. When the system switches from the vertical to the horizontal spindle, the spindle must be resynchronized. This synchronization is initiated using the IS “Resynchronize spindle 1” (DB31, ... (DB31, ... DBX16.4) or the IS “Resynchronize spindle 2” (DB31, ... DBX16.5). The spindle must be in control mode.

© Siemens AG, 2003. All rights reserved

2.3 Configurable gear adaption

2.3.1 Gear stage for spindles and gear change

Why do we need gear stages?

Gear stages are used on spindles to step down the speed of revolution of the motor in order to generate a high torque at low spindle speeds.

Gear stages of spindles

5 gear stages can be configured for each spindle. If the spindle motor installed on the spindle has a direct transmission ratio (1:1) or an invariable transmission, MD 35010: GEAR_STEP_CHANGE_ENABLE must be set to zero. In this case, the 1st gear stage is active. This is the default setting.

Selection of the type of gear stage change

By configuring **MD 35010: GEAR_STEP_CHANGE_ENABLE** ("Gear stage change possible"), the GSC type can be defined as follows:

- 0: Spindle motor installed directly (1:1) or with a non-variable transmission ratio.
- 1: Spindle motor with a maximum of 5 gear stages
  The gear stage change is carried out in oscillation mode (refer to Subsection 2.3.4 Gear stage change in oscillation mode for details)
- 2: Spindle motor with a maximum of 5 gear stages
  Gear stage change at configured fixed position in SW 5.3 and higher).
  (see Subsection 2.3.5 Gear stage change at fixed position).

Requirement for a gear stage change:
It is activated bz the NC via MD 35010: GEAR_STEP_CHANGE_ENABLE **not equal to 0** and programmed via the parts program for M40 with S... or M41...M45. In principle, the gear stage change is only performed, however, is the requested gear stage is different from the active gear stage.

The section below explains who can request a gear stage change and how it is triggered, initiated and completed.

Ways of specifiying a gear stage change

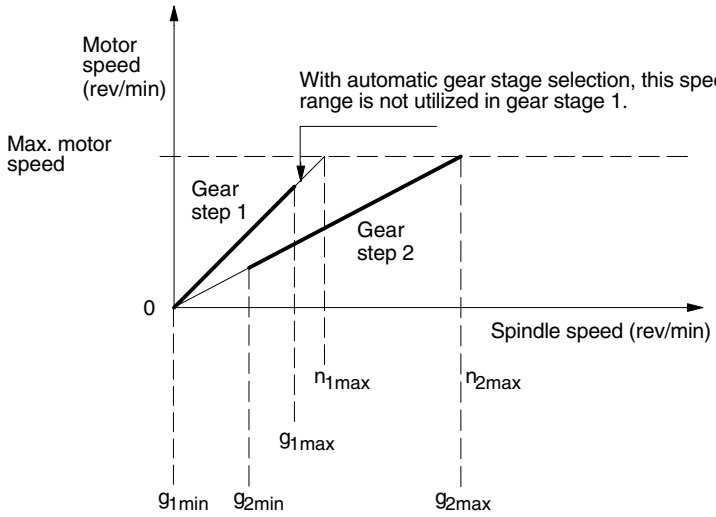
A gear stage change can be requested:

1. In the parts program
   automatically through programmed spindle speed for M40 without S or through programming with M41 to M45
2. By the PLC using the function module FC 18 (SW 4.4 or higher)
3. From synchronized actions with M40 and S or M41 to M45 (SW 6.2 and higher)
4. In reset status, by describing the VDI interface.
   Especially after a POWER ON, the NC can be notified of the currents gear stage.
5. In reset state on NC stop
   For a manually changes gear state directly at the installation location.

© Siemens AG, 2003. All rights reserved

Selection between two gear stages

Gear stage selection between two gear stages with specification of a maximum spindle speed is shown in the example below:



With MD 35110: GEAR_STEP_MAX_VELO:

- n_{1max} ...maximum spindle speed of 1st gear stage
- g_{1min} ... minimum spindle speed of 1st gear stage for automatic gear stage selection
- g_{1max} ...maximum spindle speed of 1st gear stage for automatic gear stage selection
- n_{2max} ...maximum spindle speed of 2nd gear stage
- g_{2min} ... minimum spindle speed of 2nd gear stage for automatic gear stage selection
- g_{2max} ...maximum spindle speed of 2nd gear stage for for automatic gear stage selection

Fig. 2-6 Gear stage change with selection between two gear stages

Parameters on gear stage change

On gear stage change, interpolation parameters and, in the standard case, also servo parameters sets of the position controller are activated according to MD 35590: PARAMSET_CHANGE_ENABLE.

Servo parameter sets of the position controller 1 to 6:
The servo parameter set are used quickly adapt the position controller to changed properties of the machine during operation on gear change of the spindle.

If the spindle is operating in axis mode, the new gear stage is stored internally and parameter set index “0” remains active in the servo. The new gear stage is activated when the spindle is **next** programmed.

Note

For further notes on control and servo parameter set, see:
References: /FB/, G2 “Velocities, Setpoint/Actual Value Systems, Closed-Loop Control”
/PGA/, Chapter 5, “Programmable Servo Parameter Set”

© Siemens AG, 2003. All rights reserved

The **Interpolation parameters** specify the gear stages 1 to 5 at the VDI interface for “Set gear stage A to C” (DB31, ... DBX82.0–82.2). Depending on “Actual gear stages A to C” (DB31, ... DBX16.0–16.2), the corresponding servo parameter set is activated.

One set of parameters, with the following assignment, is provided from the NC for each of the 5 gear stages:

Data block Spindle data	VDI inter-face	Parameter set index n	Date/contents of the data block
Data for axis mode	*	0	Monitoring
Data for 1st gear stage	000 001	1	M40 speed Min/max speed Acceleration Servo gain factor Transmission ratio
Data for 2nd gear stage	010	2	
Data for 3rd gear stage	011	3	
Data for 4th gear stage	100	4	
Data for 5th gear stage	101 110 111	5	

* the last active gear stage

The parameter sets of gear stages 1 to 5 can be configured with the following machine data:

	Speed min/max:
MD 35110: GEAR_STEP_MAX_VELO_LIMIT[n]	maximum for gear stage
MD 35120: GEAR_STEP_MIN_VELO_LIMIT[n]	minimum for gear stage
MD 35130: GEAR_STEP_MAX_VELO_LIMIT[n]	maximum for gear stage
MD 35140: GEAR_STEP_MIN_VELO_LIMIT[n]	minimum for gear stage
	Acceleration in:
MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL[n]	speed control mode
MD 35210: GEAR_STEP_POSCTRL_ACCEL[n]	position control mode
MD 35012: GEAR_STEP_CHANGE_POSITION[n]	gear stage change position

Changing parameter sets and formatting

The gear stage will also switch over the set of servo parameters if the machine data MD 35590: PARAMSET_CHANGE_ENABLE = 0 or = 1. When

- 0: parameter set changes **cannot be controlled**.
- 1: the servo parameter set is specified as priority **by the internal NC changeover** at the VDI interface.

If no influence to the parameter set change is possible, the following applies:
For axes, the 1st parameter set with index 0 is always active.
For spindles, the 2nd to 6th parameter set is always active depending on the gear stage **plus one** engaged.

On stipulation of the servo parameter set by the VDI interface, parameter sets 1 to 6 can be activated via indices 0 to 5. The following applies:

For the axes involved, the parameter set number the corresponds to the master spindle gear stage **incremented by one** is active. This corresponds to the parameter set numbers 2 to 6.
For spindles, the 2nd to 6th parameter set is always active depending on the gear stage plus one engaged.

When MD 35590: PARAMSET_CHANGE_ENABLE = 2, the servo parameter set is **stipulated by the PLC**. A gear stage can be specified by the PLC at any time.

© Siemens AG, 2003. All rights reserved

Sequence for changeover

If the new gear stage is preselected, the following sequence applies:
IS “Set gear step A to C” (DB31, ... DBX82.0–82.2) and IS “Change gear” (DB31, ... DBX82.3) are set. At the point when the “oscillation speed” interface signal (DB31, ... DBX18.5) is enabled, the spindle decelerates to a stop with the acceleration of oscillation or with the acceleration for speed control/position control.

At the latest when the spindle has come to a standstill (“Axis/spindle stationary” interface signal (DB31, ... Oscillation is started with the oscillation speed interface signal (DB31, ... DBX18.5) (see Subsection 2.3.4 Gear stage change in oscillation mode).

In principle, the new gear stage can also be engaged without oscillation (see Subsection 2.3.5 Gear stage change at fixed position).

When the new gear is engaged, the IS “Actual gear stages A to C” (DB31, ... DBX16.0–16.2) and IS “Gear changed” (DB31, ... DBX16.3).

End of gear stage change

The gear stage change is considered completed (spindle mode Oscillation mode is deselected) and the spindle is switched to the parameter block of the new actual gear stage. The IS “Change gear” (DB31, ... DBX82.3) is reset by the NCK, whereupon the PLC program resets the IS “Gear changed”.

The IS “Gear changed” (DB31, ... DBX16.3) informs the NC that the new gear stage is valid using IS “Actual gear stage A to C” (DB31, ... DBX16.0–16.2) and that oscillation mode is being terminated. It does not matter whether IS “Oscillation speed” (DB31, ... DBX18.5) is still set. The actual gear stage, which should correspond to the set gear stage, is relevant for selecting the parameter set. If this is not the case, Alarm 22010 is signalled on MD11410: SUPPRESS_ALARM_MASK, Bit 3 = 0.

After acknowledgement of the gear stage change by the PLC with IS “Gear changed” (DB31, ... DBX16.3) DBX16.3), the spindle is in speed control mode (DB31, ... DBX84.7).

For further notes on signal exchange between PLC and NC, see:
References: /FB1/, A2, “Various Interface Signals”, Section 2.7

1. Specifying a gear stage in the parts program

In the case of M40, the spindle must be in control mode for automatic gear stage selection with an S value. Otherwise, the gear stage change is rejected and Alarm 22000 “Gear stage change not possible” set.

Automatic selection for active M40
The gear stage is selected automatically by the control. The control checks which gear stage is possible for the programmed spindle speed (S value). If the suggested gear stage is not equal to the current (actual) gear step, the “Change gear” interface signal and the “Set gear stage A to C” interface signals are enabled.

When determining the matching gear stage, a gear stage change is requested only if the new speed lies in the permiisible speed range for the active gear stage.
If necessary the speed is limited to the maximum speed of the current gear stage or increased to the minimum speed of the current gear stage, and the IS “Programmed speed too high” or “Programmed speed too low” is enabled.

© Siemens AG, 2003. All rights reserved

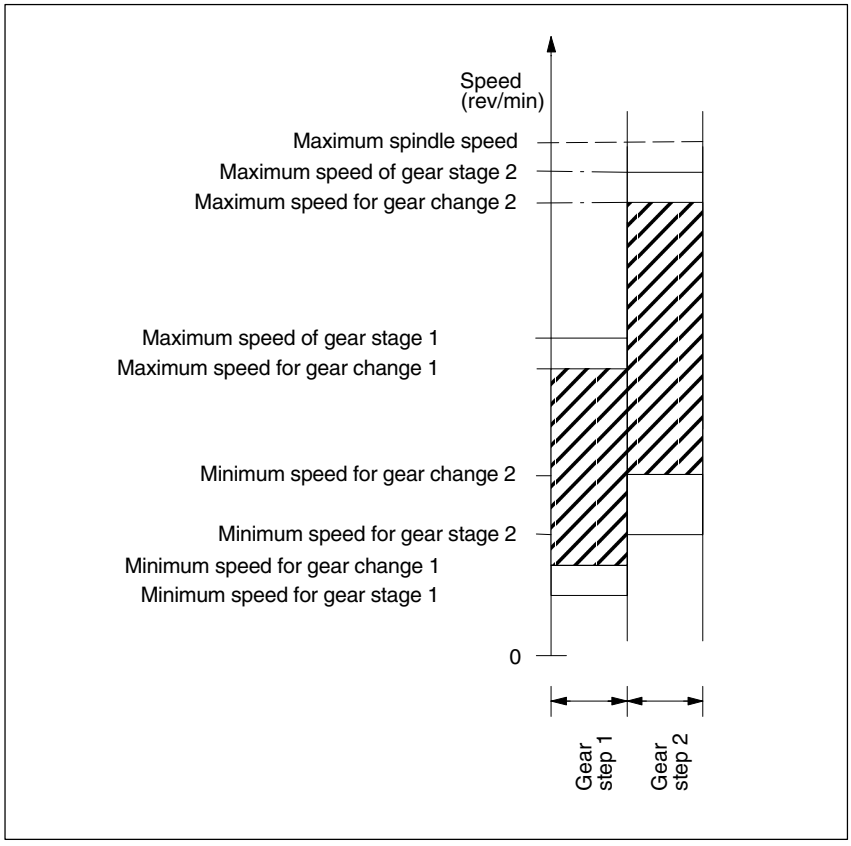


Fig. 2-7 Example for two “gear stages with overlapping speed ranges” for automatic gear stage change (M40)

Fixed gear stage specified with M41 to M45
The gear stage can be permanently defined in the parts program with M41 to M45. If a gear stage is defined by M41 to M45, which is different than the current (actual) gear stage, the IS “Switch gear” (DB31, ... DBX82.3) and “Set gear stage A to C” interface signals are enabled.
The programmed spindle speed (S value) then refers to this permanently defined gear stage.

If a spindle speed is programmed above the maximum speed for the fixed gear stage, the speed is limited to the maximum speed of the gear stage MD 35130: GEAR_STEP_MAX_VELO_LIMIT and “Programmed speed too high” (DB31, ... DBX83.0) is set.

Block change

When a gear stage change is programmed in the parts program, the gear change block remains active until the gear has been changed by the PLC (same effect as if IS “Read-in disable” (DB21, ...DBX6.1) were set).

© Siemens AG, 2003. All rights reserved

2. Specifying a gear stage from the PLC with FC18

The gear stage change can also be performed by function block FC18 during a parts program, in the reset state or in all operating modes.

If the speed and direction of rotation is specified with FC18, the NC can be requested to select the spped to match the gear stage. This corresponds to an automatic gear stage change with M40.

The gear stage is not changed if the spindle is positioned by FC18 or if traversing in axis mode.

For further note on function block FC18, please refer to:
References: /FB1/, P1, “FC18: SpinCtrl Spindle control”

3. Specifying a gear stage with synchronized actions (SW 6.2 and higher)

The gear stage change can be requested by synchronized actions:

- automatic gear stage selection with M40 with S or with
- specification of gear stages 1 to 5 with M41 to M45.

The gear stage is not changed if the spindle is positioned by synchronized actions or if traversing in axis mode.

For further information about how to program the gear stage change with synchroniyed actions, see Section 2.6 “Programming”.

4. Manual specification of a gear stage

Outside a parts program that is running, the gear stage can also be changed even without a request from the NC or the machine. This is the case, for example, when a gear stage is changed directly by hand.

To select the matching parameter sets, the NC must be informed of the current gear stage. For this to work, the control or the parts program must be in the reset state.

Supplementary conditions
Transfer of the gear stage to the NC is initiated when IS “Actual gear stage A to C” (DB31, ... DBX16.0–16.2) changes. These three bits must be set continuously during operation. Successful transfer is acknowledged with IS “Set gear stage A to C” (DB31, ... DBX82.0–82.2) to the PLC.

The IS “Gear changed” (DB31, ... DBX16.3) does not have to be set. If the position controller is active at the time when a new gear stage is specified by the PLC with IS “Actual gear stage A to C” (DB31, ... DBX16.0–16.2), it is deactivated for the duration of this gear change process.

NC stop during gear stage change

Spindle stop through NST “NC Stop” (DB21, ... DBX7.4) is not possible

if

- the spindle is in oscillation mode for a gear change
- IS “Gear changed” (DB31, ... DBX16.3) not yet present.

Note

Option for aborting:

IS “Delete distance-to-go/Spindle Reset” (DB31, ... DBX2.2) or IS “Gear changed” (DB31, ... DBX16.3) with corresponding checkback signal of the actual gear stage (IS “Actual gear stage” DB31, ... DBX16.0–16.2) changes.

© Siemens AG, 2003. All rights reserved

Spindle response after a gear stage change

How the spindle behaves once the gear stage has been changed depends on the following initial conditions:

- If the spindle was in the stop stage before the gear stage change (M5, FC18: “Stop rotate spindle”), in positioning or axis mode, M5 (spindle stop) is active after completion of the gear stage change.
- If a direction of rotation was programmed before the gear stage change (M3, M4, FC18: “Start spindle rotation”), then the last speed and direction of rotation will become active again after the gear stage change. The spindle accelerates at the new gear stage to the spindle speed last programmed (S value).

  The next block in the parts program can be executed.

Special points (SW 4.4 and higher)

The following points must be observed on gear stage change:

1. The gear stage change is not terminated by selecting IS “Ramp-up changeover to V/f mode” (DB31, ... DBX20.1). The setpoint 0 is output.

   The acknowledgment of the gear stage change is carried out as usual by the PLC signal “Gear changed” (DB31... DBX16.3).
2. The signal “Ramp function generator rapid stop” must be reset by the PLC before the gear stage change is ended by the PLC.
3. The gear stage change procedure is terminated without an alarm output on an NC reset. The gear stage signalled in IS “Actual gear stages A to C” (DB31, ... DBX16.0–16.2) is accepted by the NC.

Star/delta switchover with FC17 (SW 6.4 and higher)

As from SW 6.4, digital main spindle drives can be switched over in both directions from star to delta via FC17 even with the spindle running. This automatic switchover is controlled by a defined switchover logic in the FC17, which provides a user-programmable switchover time for the relevant spindle.

For further information on the function block FC17 please refer to:
References: /FB1/, P3, “FC17: Y/Delta Star/delta switchover”

Parameter set of the load gear ratio

In axis mode, it is possible to configure positive or **negative load gear factors** for each gear stage. The setting is defined in machine data MD 31060: DRIVE_AX_RATIO_NUMERA. The setting range is the same size for positive and negative load gear factors. It is not possible to enter the value 0. In this case, alarm 17095 is output.

Note

The reference will be lost if an indirect encoder is configured and the load gear ratio changes. The IS “Referenced/synchronized 1 or 2” (DB31, ... DBX60.4/60.5) is reset for the relevant measuring system.

© Siemens AG, 2003. All rights reserved

2.3.2 Encoder on intermediate gear (SW 6.4 and higher)

Application and functions

Using a configured intermediate gear, it is possible to adapt a variety of rotating tools. The intermediate gear at the tool side has a multiplicative effect on the motor/load gear. It is set via the machine data

MD 31066: DRIVE_AX_RATIO_NUMERA "numerator load gearbox"
DRIVE_AX_RATIO_DENOM "denominator load gearbox"

An encoder present on the tool side for the intermediate gear is configured using MD 31044: ENC_IS_DIRECT2 "Encoder on intermediate gear".

Change parameters for these machine data can be activated with "NewConfig" either using the commissioning software SinuCOM-NC or via the operator panel HMI per soft key. The existing motor/load gear, on the other hand, are active after Power On.

Tool change

If the intermediate gear is also changed on tool change, the user must also reconfigure the transmission ratio of the numerator and denominator via the machine data of the intermediate gear.

Example:
For a new tool inserted with a transmission ratio of 2:1, a matching intermediate gear is configued and is active immediately in the parts program with the command NEWCONF.

```
N05 $MA_DRIVE_AX_RATIO2–NUMERA[AX5] = 2
M10 $MA_DRIVE_AX_RATIO2–DENOM[AX5] = 1
N15 NEWCONF
```



Caution

It remains the task of the user to stop within the appropriate period in order to make changes to the machine data when required and then active activate a NewConfig.

Switchover

Changeover to a new transmission ratio is performed immediately by means of NewConfig. From a technological viewpoint, the associated mechanical changeover process takes some time, since a different intermediate header with rotating tool is replaced.

Note

At yero speed, changeover is jerk-free. The user is therefore responsible for taking appropriate precautions.

Applications in which changeover takes place during motions and which require a smooth or soft speed transition can be handles using existing speed setpoint filters.

For further details on control interdependencies, refer to:
References: /FB/, G2 "Velocities, Setpoint/Actual Value Systems, Closed-Loop Control"

© Siemens AG, 2003. All rights reserved

2.3.3 Non-acknowledge gear stage change (SW 5.3 and higher)

Mode change

A gear stage change not acknowledged cannot be interrupted by a mode change (e.g. switching over to JOG).
The switchover is delayed by the time entered in MD 10192: GEAR_CHANGE_WAIT_TIME as the maximum. If the gear stage change is not acknowledged within this time, the NC will provide one of the following two alarms:

- 16938: If the program interruption has occurred during the program status Running.
- 16939: In all of the other cases.
  During the wait phase, the selfclearing alarm 16940 is output if in MD 11411: ENABLE_AQLARM_MASK bit 1 = 1 is set.

Further events

Events that initiate reorganizing will also wait until a gear stage change is completed.
The time set in machine data MD 10192: GEAR_CHANGE_WAIT_TIME will now determine how long it is waited for the gear stage change. If this time expires without ending the gear stage change, the NC will also react with the alarms mentioned above.

The following events have an analog response:

- User ASUB
- Mode change
- Delete distance-to-go
- Axis change
- Activate PI user data
- Enable PI service machine data
- Switch over skip block, switch over Dry Run
- Editing in the modes
- Correction block alarms
- Overstore
- Rapid retraction with G33, G34, G35
- Subroutine level abortion, subroutine abortion

Response after POWER ON

The active gear stage on the machine can be specified by the PLC control after POWER ON and in the RESET state. The NCK will then select the appropriate set of parameters and check back the “Set gear stages A to C” (DB31, ... DBX82.0–82.2) to the PLC.

© Siemens AG, 2003. All rights reserved

2.3.4 Gear stage change with oscillation mode

What is oscillation?

Oscillation in this context means that the spindle motor rotates alternately in the clockwise and counterclockwise direction. This oscillation movement makes it easy to engage a new gear stage.

Oscillation mode

IS “Change gear” (DB31, ... DBX82.3) displays that a gear stage change is necessary. In principle, the new gear stage can also be engaged without oscillation

- MD 35010: GEAR_STEP_CHANGE_ENABLE must be set to 1.
- The “Oscillation mode” interface signal (DB31, ... DBX84.6) is set.
- The acceleration is defined in MD 35410: SPIND_OSCILL_ACCEL.

IS “Oscillation speed” DB31, ... DBX18.5

The spindle is in oscillation mode if a new gear stage was defined using the automatic gear stage selection (M40) or M41 to M45 (“Change gear” interface signal is enabled). The IS “Change gear” (DB31, ... DBX82.3) is only enabled when a new gear stage is selected that is not equal to the current actual gear stage.

If the “Oscillation speed” interface signal is enabled without defining a new gear stage, the spindle does not change to oscillation mode.

Oscillation is started with the oscillation speed interface signal (DB31, ... DBX18.5). The setting of the IS “Oscillation via PLC” (DB31, ... DBX18.4) distinguishes between:

- Oscillation via NCK
- Oscillation via PLC
- Oscillation with FC 18 (see Description of Functions P3).

Oscillation time

The oscillation time for oscillation mode can be defined in a machine data for each direction of rotation:

- Oscillation time in M3 direction (referred to as t1 in the following) in MD 35440: SPIND_OSCILL_TIME_CW
- Oscillation time in M4 direction (referred to as t2 in the following) in MD 35450: SPIND_OSCILL_TIME_CCW.

Oscillation through NCK

Phase 1:

IS “Oscillation speed” (DB31, ... DBX18.5) accelerates the spindle motor to the velocity defined in MD 35400: SPIND_OSCILL_DES_VELO (oscillation speed). The start direction is defined by MD 35430: SPIND_OSCILL_START_DIR. Time t1 (or t2) is started according to the start direction defined in MD 35430: SPIND_OSCILL_START_DIR. The time is always decisive, not the fact that the oscillation speed is reached.

© Siemens AG, 2003. All rights reserved

Phase 2:

When time t1 (t2) expires, the spindle motor accelerates in the opposite direction to the speed defined in MD 35400: SPIND_OSCILL_DES_VELO. Time t2 (t1) starts.

Phase 3:

When time t2 (t1) expires, the spindle motor accelerates in the opposite (same direction as phase 1) to the speed defined in the MD 35400: SPIND_OSCILL_DES_VELO. Time t1 (t2) starts. The process continues with phase 2.

Oscillation through the PLC

With the IS “Oscillation via PLC” (DB31, ... DBX18.5), the spindle motor accelerates (with the oscillation acceleration) to the speed defined in MD 35400: SPIND_OSCILL_DES_VELO (oscillation speed). The direction of rotation is defined by IS “Set speed CCW” (DB31, ... “Set direction of rotation clockwise” (DB31, ... DBX18.6). The oscillation movement and the two times t1 and t2 (for clockwise and counterclockwise rotation) must be simulated on the PLC.

Special points to be noted

Setting/resetting interface signals (IS) and machine data in oscillation mode

- To decelerate the spindle, the PLC user need not set the IS “Spindle stop” (DB31, ... DBX4.3). The spindle is brought to a standstill internally by the control when a gear stage change is requested.
- The IS “Gear changed” (DB31, ... DBX16.3) should always be used to stop the gear stage change.
- The “Oscillation speed” interface signal (DB31, ... DBX18.5) should be used to support mechanical engagement of the gear. It has no effect on the internal control mechanism for the gear stage change procedure and should therefore only be set as necessary.
- If the “Oscillation speed” interface signal (DB31, ... DBX18.5) is reset, the oscillation movement stops. However, the spindle remains in oscillation mode.
- The acceleration is defined in MD 35410: SPIND_OSCILL_ACCEL.
- If an indirect measuring system (motor encoder) is used, the synchronization gets lost.
  If MD 31050: ENC_IS_DIRECT = 0, the IS “Referenced/synchronized” (DB31, ... DBX60.4/5 = 0) is deleted automatically. The zero mark is referenced the next time it is crossed.

End of the oscillation mode

On completion of oscillation mode, the spindle is in openloop control mode again and automatically changes to the mode defined by SPCON or SPCOF.

All gear-specific limit values (min./max. speed etc.) correspond to the set values of the actual gear stage.

Note

For further explanations on the gear stage change, please refer to:
Subsection 2.3.5 “Gear stage change at fixed position (SW 5.3 and higher)”

© Siemens AG, 2003. All rights reserved

Functionality

Machine tools of the conventional design require a gear stage change of the spindle in oscillation mode.
Configuration with **MD 35010: GEAR_STEP_CHANGE_ENABLE = 1** the following procedure is implemented:

- Deceleration of the spindle. The braking action corresponds to an M5 movement.
- Output of VDI interface signal “Oscillation mode” (DB31, ... DBX84.6), “Change gear” (DB31, ... DBX82.3) and “Set gear step A to C” (DB31, ... DBX82.0–82.2). If the position control has been enabled, it is disabled (DB31, ... DBX61.5 = 0).
- The load gear can now “disengage”.
- The IS “Oscillation enable” (DB31, ... DBX18.5) can be set from the PLC. The spindle drive then performs an oscillation motion with preset values. The oscillation motion is to facilitate and accelerate the reengaging of the gear wheels.
- IS “Actual gear stages A to C” (DB31, ... DBX3.0) by the PLC.
- After the PLC reports to the NCK “Gear stage changed” (DB31, ... DBX16.3) the last active movement, if any, is continued. For indirect encoders (motor encoders), the referencing status is cleared (DB31, ... DBX60.4/5 = 0).

Block change

If the spindle has been switched over to oscillation mode, the IS “Gear change” (DB31, ... DBX82.3) is set, the parts program processing remains stopped. A new block is not executed. If oscillation mode is quitted using the IS “Gear switched” (DB31, ... DBX16.3) the execution of the parts program is continued as illustrated in Fig. 2-8. A new block is executed.

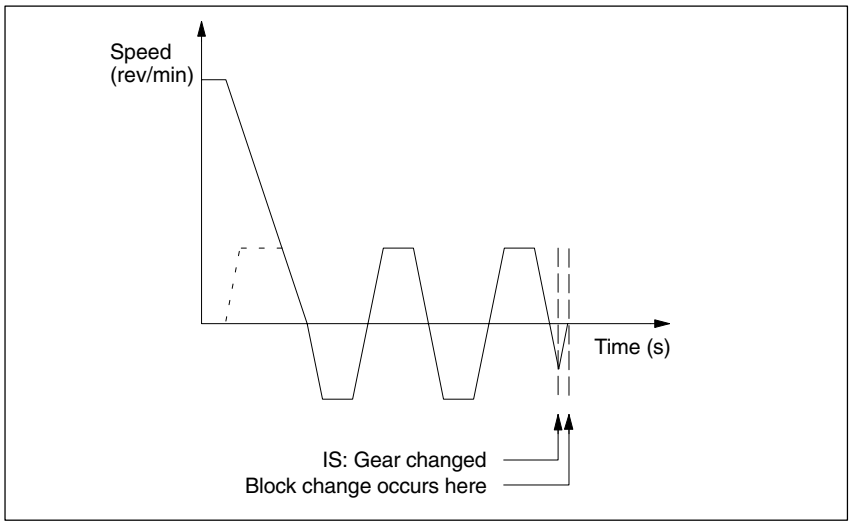
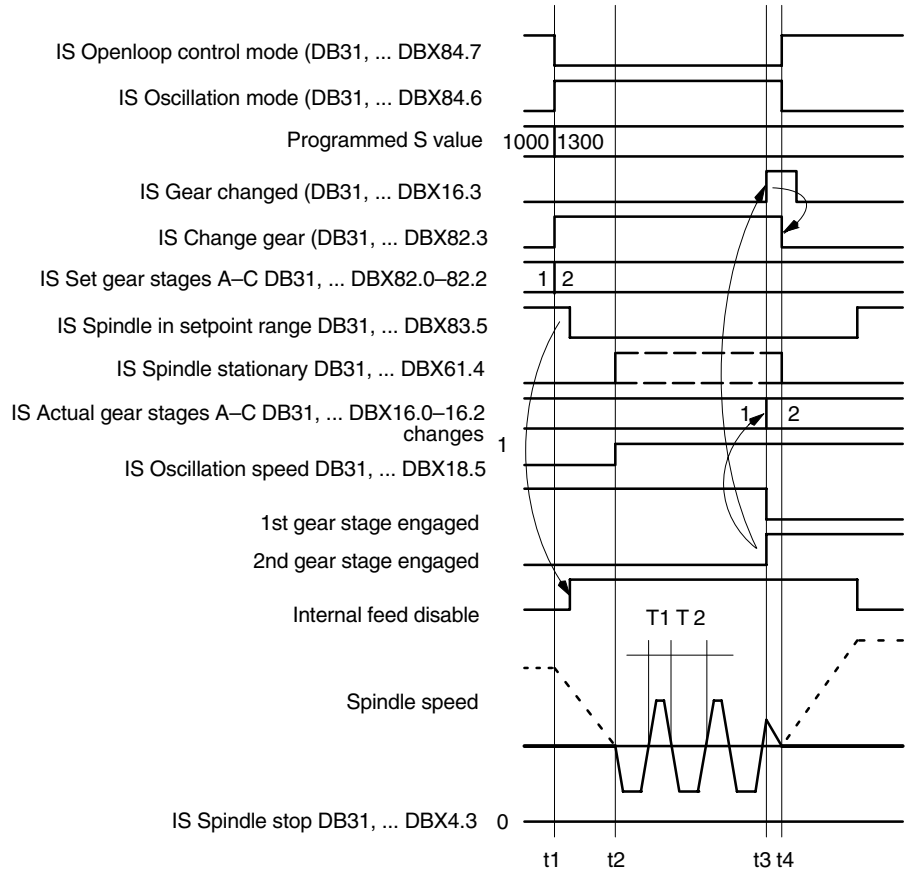


Fig. 2-8 Block change following oscillation mode

© Siemens AG, 2003. All rights reserved

Oscillation mode Typical time sequence for the gear stage with the spindle:



t1 When S1300 is programmed, the NCK detects a new gear stage (2nd gear stage), enables the change gear interface signal and inhibits execution of the next parts program block.

t2 The spindle is stationary and oscillation starts (oscillation via NCK). The oscillation speed interface signal must be enabled by the time t2.

t3 The new gear stage is engaged. The PLC user transmits the new (actual) gear stage to the NCK and enables the gear changed interface signal.

t4 At this point, the NCK cancels the change gear interface signal, terminates oscillation, enables execution of the next parts program block and accelerates the spindle to the new S value (S1300).

Fig. 2-9 Gear stage change with stationary spindle

© Siemens AG, 2003. All rights reserved

2.3.5 Gear stage change at fixed position (SW 5.3 and higher)

Application and advantages

Machine tools increasingly use standardized spindle drives, firstly to save technological dead time at a gear stage change and secondly to gain the cost benefits of using standardized components.

The function Gear stage change at fixed position supports the “directed gear stage change” of load gears that require a different activation from the NC. The gear stage change can in this case only be performed at a defined spindle position. An oscillation motion as required by conventional load gears is thus no longer necessary.

Sequence for gear stage change at fixed position

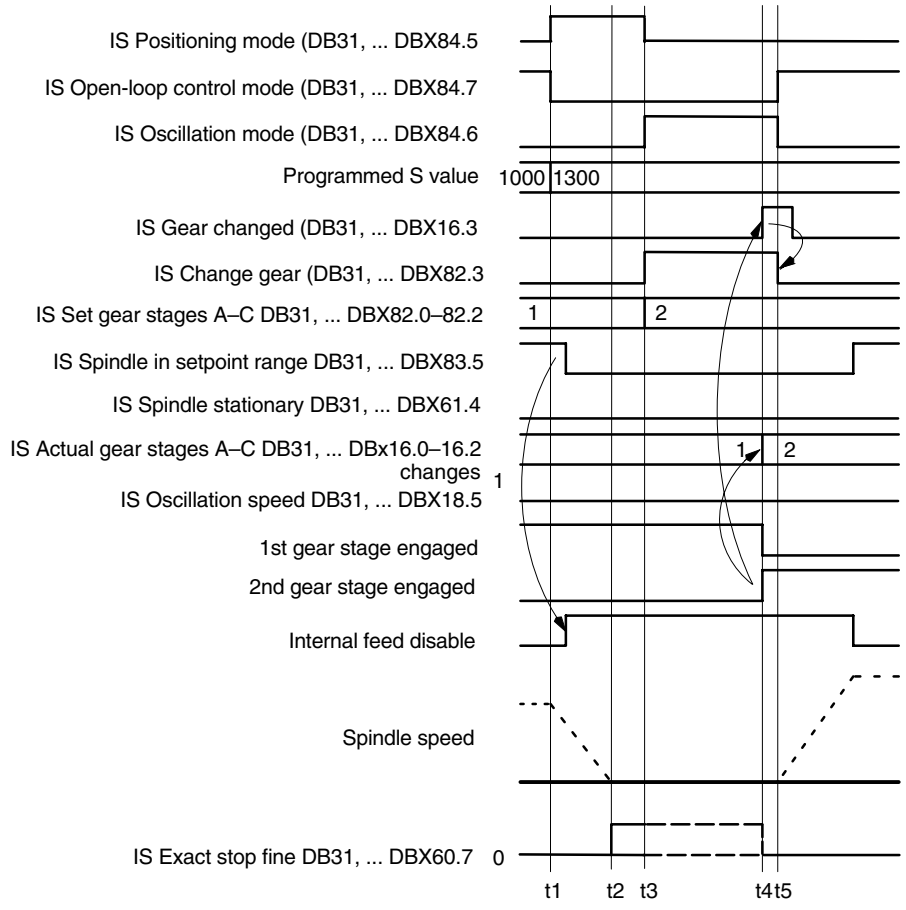
Gear stage change at a fixed position
Configuration with **MD 35010: GEAR_STEP_CHANGE_ENABLE = 2** carries out the following sequence:

- Positioning of the spindle from standstill or a movement to the position set in MD 35012. If the gear stage change is performed out of a movement, then the current direction of rotation is maintained. The spindle is in positioning mode during the positioning operation. The IS “Positioning mode” (DB31, ... DBX84.5) is output. If no reference is available (DB31, ... DBX60.4/5 = 0) or IS “Resynchroniye on positioning MS 1/2” (DB 31, ... DBX17.4/5) is set, the positioning operation will be extended by the zero mark search.
- After the GSC position configured in MD 35012 is reached, the timer set in MD 35310: SPIND_POSIT_DELAY_TIME is activated before the system switches over to oscillation mode and the normal GSC dialog commences.
- Output of VDI interface signal “Oscillation mode” (DB31, ... DBX84.6), “Change gear” (DB31, ... DBX82.3) and “Set gear step A to C” (DB31, ... DBX82.0–82.2).
- The position control is not disabled when an active measuring system with indirect encoder (motor encoder) MD 31040 is used: ENC_IS_DIRECT = 0. When an active measuring system with direct encoder (load encoder) is used, the position control is disabled (DB31, ... DBX61.5 = 0), since the flow of the force to the load is interrupted, making position control impossible.
- If position-controlled operation is not possible, then it can be disabled by resetting “Servo enabl“e (DB31, ... DBX2.1 = 0).
- Mechanical changing of the gear stage on the machine. No oscillation motion is required from the drive. The IS “Oscillation enable” (DB31, ... DBX18.5) and “Oscillation via PLC” (DB31, ... DBX16.3) does **not** have to be set. In principle, the oscillation movement is now still possible.
- IS “Actual gear stages A to C” (DB31, ... DBX3.0) by the PLC.
- After the message “Gear stage changed” DBX16.3) the last active movement, if any, is continued. For indirect encoders (motor encoders), the referencing status is cleared (DB31, ... DBX60.4/5 = 0). The spindle is in speed-control mode and the IS “Open-loop control mode” (DB31, ... DBX84.7) is output.

© Siemens AG, 2003. All rights reserved

Gear stage change at fixed position

Typical time sequence for the gear stage at fixed position:



t1 When S1300 is programmed, the NCK detects a new gear stage (2nd gear stage), enablesthe change gear interface signal Positioning mode and inhibits execution of the next parts program block.

t2 The spindle is stopped and exact stop is signaled.

t3 Gear stage change – wait time

t4 The new gear stage is engaged. The PLC user transmits the new (actual) gear stage to the NCK and enables the gear changed interface signal.

t5 At this point, the NCK cancels the change gear reverse, and enables execution of the next parts program block and accelerates the spindle to the new S value (S1300).

Fig. 2-10 Gear stage change with stationary spindle

Gear stage change at position MD 35012

The gear stage change position is configured in machine data MD 35012: GEAR_STEP_CHANGE_POSITION configured for each gear stage.

© Siemens AG, 2003. All rights reserved

GSC delay MD 35310

In SW 6.1 and higher, the timer set in machine data MD 3510: SPIND_POSIT_DELAY_TIME is activated before the gear stage change request IS “Oscillation mode” (DB31, ... DBX84.6), “Change gear” (DB31, ... DBX82.3) and “Set gear step A to C” (DB31, ... DBX82.0–82.2) is output.

Position identifiers/ position

The position is always approached on the shortest way (corresponds to DC). If no reference is available and if the spindle is at zero speed (e.g. after POWER ON), then the approach direction is determined by MD 35350: SPIND_POSITIONING_DIR.

If an adjustable gear stage change position is required, then this can be achieved by writing the machine data and by a subsequent NewConfig. The change in the MD value can be carried out using the parts program or via the MMC.

If the system is unable to reach the preset position, then alarm 22020 is signaled and the gear stage change dialog between NCK and PLC does not take place in order not to destroy the gears. As this alarm is serious, the parts program cannot be continued and the cause must be eliminated under all circumstances The cause for the abortion of positioning is from experience generally to be found in incorrect MD settings or incompatible PLC signals.

Velocity

The positioning speed is taken from MD 35300: SPIND_POSCTRL_VELO.

The IS “Spindle speed override“/“Feedrate override” at
(DB31, ... DBX17.0=0: DB 31, ... DBB19) and for
(DB31, ... DBX17.0=1: DB 31, ... DBB0) are effective as normal for positioning. The positioning speed can be changed proportionally through the program instruction OVRA[Sn].

Note: OVRA[Sn] is valid modally. After the gear stage change it should be set back to a value appropriate for the machining.

The parts program instruction FA[Sn] does not change the positioning speed during gear stage change.

Acceleration

The acceleration values are determined by the gear stage dependent
MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL and
MD 35210: GEAR_STEP_POSCTRL_ACCEL. The acceleration can be changed proportionally by the programming of ACC[Sn].
Note: ACC[Sn] is valid modally. After the gear stage change it should be set back to a value appropriate for the machining.

Speed-dependent acceleration

The “knee-shaped acceleration characteristic” is effective as in positioning with SPOS or FC18.

Jerk

It is currently not possible to limit the change in acceleration.

© Siemens AG, 2003. All rights reserved

End of positioning

The transition between the end of the positioning operation (DB31, ... DBX84.5) and start of oscillation mode (DB31, ... DBX84.6) is defined by reaching “Exact stop fine” (DB31, ... DB60.7) and the variable in MD 3510: SPIND_POSIT_DELAY_TIME.

The determination of the transition condition has an effect firstly on the gear stage change time and secondly on the accuracy of the approach to the preset gear stage change position.

Block change

The block change is stopped and the machining blocks are not started until the gear stage has been changed by the PLC (DB31, ... DBX16.3).

End of gear stage change

After the gear stage change has been ended, the spindle is in open-loop control mode again and will automatically change to the controller mode defined by SPCON or SPCOF.

All gearspecific limit values (min./max. speed of gear stage, etc.) correspond to the fed back values of the actual gear stage.

Supplementary conditions

- The spindle must possess at least one measuring system.
- Position-controlled operation must be possible and must have been activated.
- Generally, it must be possible to execute SPOS from the parts program, from a synchronized action or via FC18: “Start spindle positioning” without errors.

If not all requirements can be met, the described function cannot be used successfully.

Activation

The function for gear stage change at a fixed position is activated by setting the MD 35010: GEAR_STEP_CHANGE_ENABLE = 2.

© Siemens AG, 2003. All rights reserved

2.4 Spindle auxiliary functions

2.4.1 No output of spindle auxiliary functions after block search (SW 5.3 and higher)

Application

In conjunction with tool change routines, it can be useful if the programmed spindle values accumulated during a block search are not output in the action blocks but later, e.g. after a tool change.

Functionality

Programming the machine data MD 11450: SEARCH_RUN_MODE; with bit 2, the output options of the spindle auxiliary functions after block search are defined. The configuration offers the following options to choose from:

- Bit 2 = 0: previous behavior (compatibility mode), and with
- Bit 2 = 1: suppression of the output of the spindle auxiliary functions in the action blocks. No spindle auxiliary functions are output to the VDI interface, and no spindle setpoints are created for the drive. The programmed spindle values are always stored in system variables independently of the set configuration and can be evaluated later (e.g. after a tool change). This can be achieved with an ASUB, which can be started after output of the action blocks (DBx, ... 32.6 = 1). The accumulated programmed spindle values can be read and adapted in the ASUB.

Note

For explanations on the block search and on the action blocks, please refer to
References: /FB1/, K1,“Mode Group, Channel, Program Operation”, 2.5 Program Test
References: /FB1/, H2,“Auxiliary Functions to PLC”, 2.5 Behavior with Block Search

---

System variable for block search

The contents of the system variables $P_S, $P_DIR and $P_SGEAR can be lost after block search due to synchronization operations.
WB: Defined value ranges (**WB**) of the corresponding system variables.

Description	NCK variables
Collected spindle speed WB = {0...SMax}	$P_SEARCH_S[n]
Collected spindle rotation WB = {3, 4, 5, –5, –19, 70}	$P_SEARCH_SDIR[n]
Collected spindle gear stage M function WB = {40...45}	$P_SEARCH_SGEAR[n]
Collected spindle position WB = {0...MD 30330: MODULO_RANGE} Collected Traversing path WB = {–100.000.000...100.000.000}	$P_SEARCH_SPOS[n]
Collected position approach mode WB = {0...5} where 0=DC (default), 1=AC, 2=IC, 3=DC, 4=ACP, 5=ACN	$P_SEARCH_SPOSMODE[n]

© Siemens AG, 2003. All rights reserved

Spindle programming in ASUBs

The output of the collected spindle programming values can be carried out in an ASUB after block search and after output of the action blocks (DBx, ... DBX32.6=1).

Example

Block search to the contour with ASUB start after output of the action blocks, MD 11450: SEARCH_RUN_MODE for bit 2 = 1
(no output of spindle auxiliary functions in action blocks):

```
N05 M3 S200                    ;
N10 G4 F3                      ;
N15 SPOS = 111                 ; in the ASUB, spindle 1 is positioned to
                               ; 111 degrees
N20 M2 = 4 S2 = 300            ;
N25 G4 F3                      ;
N30 SPOS[2] = IC(77)           ; spindle 2 traverses incrementally by 77 degrees
N55 X10 G0                     ; target block
N60 G4 F10                     ;
N99 M30                        ;
```

ASUB

Sequence: After block search to N55, an ASUB is started:

```
PROC ASUP_SAVE
MSG ("output of spindle functions")
DEF INT SNR=1
AUSG_SPI:

M[SNR] = $P_SEARCH_SGEAR[SNR]  ; Output gear stage,

S[SNR] = $P_SEARCH_S[SNR]      ; Output speed (with M40
                               ; the matching gear stage is determined).
M[SNR] = $P_SEARCH_SDIR[SNR]   ; Output direction of rotation, positioning or
                               ; output axis operation
SNR = SNR+1                    ; Repeat output for the next spindle

REPEAT AUSG_SPI P=$P_NUM_SPINDLES–1        ;Repeat for all spindles
MSG("")
REPOSA
RET
```

Explanations regarding the example above

If the number of spindles is known, similar outputs can be written in one line, reducing the program execution time.
The output of $P_SEARCH_SDIR should be in a separate block, since spindle positioning or switching over to axis mode together with gear stage changing can produce an alarm.

Control response in the case of REPOS

If the started ASUB is completed with REPOSA, then spindle 1 remains at the position 111 degrees, whereas the response described above means that Spindle 2 is repositioned to the position 77 degrees.
If a different response is required, the program sequence, e.g. of “M3 S ...” and “SPOS = IC(...)” in the parts program for block search must be treated in a special way.
It can be determined whether block search is active in the ASUB by reading the system variables $P_SEARCH and comparing with the value “1”.

© Siemens AG, 2003. All rights reserved

With incremental positioning after speed control operation, the distance to be traveled is defined, but the end position attained is in certain cases only determined during the positioning operation itself.
This is the case, for example, with position compensation during crossing of the zero mark or when switching on the position control. For this reason, the distance programmed from the position as the REPOS position (REPOSA in the ASUB) is assumed to be zero.

With block search, the spindle programming is included in the new block search system variables, independently of the configuration.

Supplementary conditions

Collected S values

It is generally the case that the meaning of a programmed S value in the parts program is dependent on the current feed type of the G group of the “feed operation”:

G93, G94, G95, G97, G971:	S value is interpreted as speed
G96, G961:	S value is interpreted as a constant cutting rate.

If the feed operation is changed (e.g. for a tool change) before output of the system variable $P_SEARCH_S, then it must be ensured that the feed operation is restored to the original setting from the target block to avoid misinterpretation in the NC.

Collected direction of rotation

For the output of the direction of rotation, the system variable $P_SEARCH_SDIR is preset to the value –5 when the block search is started. This value is ineffective for output (NOP = No Operation).
This guarantees that the last spindle mode is not lost where the block search covers groups in which spindles are not programmed with direction of rotation, positioning or axis mode.
The programing of M19, SPOS and SPOSA is accumulated as “M–19” (internal M19) in the system variable $P_SEARCH_SDIR alternatively to M3, M4, M5 and M70.
The output of “M–19” runs as follows:

- The positioning data are read internally from the system variables $P_SEARCH_SPOS and $P_SEARCH_SPOSMODE.
  Both system variables can be written to, e.g. in order to enter corrections.

Note

Because of the assignments mentioned above (e.g. M[n] = $P_SEARCH_SDIR[n]), the values “–5” and “–19” generally remain hidden for the user and must only be observed with special evaluation of the system variables in the user program (ASUB).

Activation

Programming the machine data MD 11450: SEARCH_RUN_MODE, bit 2 = 1 suppresses the output of spindle auxiliary functions and spindle setpoint values after block search.

© Siemens AG, 2003. All rights reserved

2.5 Selectable spindles

Application

With SW 4.3 and higher, the “selectable spindles” function allows you to write parts programs with reference to the spindles used (“channel spindle, logical spindle”) regardless of the actual assignment of configured spindles (“physical spindles”) to a channel.

The physical spindles loaded or unloaded by “axis replacement” no longer have to be specified explicitly in the parts program.

Functionality

Each spindle is mapped unambiguously to a machine axis MD 35000: SPIND_ASSIGN_TO_MACHAX[AX ...] by a configurable number. This number is always applies to a spindle, regardless of the channel in which the spindle is actively handled.

The channel spindles can be switched over because an intermediate level is introduced between the logical spindle numbers used in the parts program and the physical spindles existing in the channel. This is achieved by assigning each logical spindle used in the parts program to a physical spindle in a table containing setting data (SD 42800: SPIND_ASSIGN_TAB[...]; spindle number converter).

The spindle number converter is effective in spindle programming by

- the parts program
- synchronized actions

The spindle number converter has no effect with PLC commands which use function block FC18. The physical spindle must always be addressed there within the context of the axis.

The logical spindles can be switched over by changing SD 42800: SPIND_ASSIGN_TAB[...]. The switchover can be initiated from the parts program, from the PLC and/or from the MMC.

Note

SD 42800: SPIND_ASSIGN_TAB[0] contains the logical master spindle. It is only used for display purposes.

This setting data is defined in the parts program by “SETMS (logical spindle)”.

The value 0 must be entered in SD 42800 for spindles which are not used.

System variables affected by the spindle switchover:
$P_S, $P_SDIR, $P_SMODE, $P_GWPS, $AC_SDIR, §AC_SMODE, $AC_MSNUM, $AA_S. See also:

Reference: /PGA/ “Programming Guide Advanced”

The converted, physical spindle number is always output as the address extension in the auxiliary function output.

© Siemens AG, 2003. All rights reserved

Supplementary conditions

- The switchable channel spindles are **not** a substitute for the axis replacement function.
- You can only switch spindles which have been assigned to the channel by parameterization.
- If spindles which are presently active in another channel are output for switchover, either the “AutoGet” function is triggered for the physical spindle or alarm 16105 “Assigned spindles do not exist” is output, depending on the configuration variant.
- If SD 42800: SPIND_ASSIGN_TAB[...] is defined from the PLC or the MMC, the channel whose table is modified must have the Reset state and the spindle to be switched must not be used in the parts program currently running. A synchronized response can be achieved by means of a STOPRE preprocessor stop.
- The multiple mapping of logical to physical spindles is not prevented in the NC. However, ambiguities can arise, according to the conversion table, when logical spindles are displayed on the MMC.
- The spindle conversion operates on spindles via FC 18.

Activation

SD 42800: SPIND_ASSIGN_TAB[...] is activated by enabling MD 20092: SPIND_ASSIGN_TAB_ENABLE=1.

Initial setting SD 42800

After POWER ON of the NC in startup switch position 1 (delete SRAM) the SD 42800 is in its basic setting. The numbers of the logical and physical spindles are identical.

SD 42800: SPIND_ASSIGN_TAB[1]=1
SD 42800: SPIND_ASSIGN_TAB[2]=2
SD 42800: SPIND_ASSIGN_TAB[3]=3
SD 42800: SPIND_ASSIGN_TAB[4]=4
SD 42800: SPIND_ASSIGN_TAB[5]=5
...

© Siemens AG, 2003. All rights reserved

Example

Assumptions: Spindle configurations:

Define spindle number and machine axis
MD 35000: SPIND_ASSIGN_TO_MACHAX [AX4] = 1
MD 35000: SPIND_ASSIGN_TO_MACHAX [AX5] = 2
MD 35000: SPIND_ASSIGN_TO_MACHAX [AX6] = 3
MD 35000: SPIND_ASSIGN_TO_MACHAX [AX7] = 5

Include machine axis in channel
MD 20070: AXCONF_MACHAX_USED[0] = 4
MD 20070: AXCONF_MACHAX_USED[1] = 5
MD 20070: AXCONF_MACHAX_USED[2] = 6
MD 20070: AXCONF_MACHAX_USED[3] = 7

Define master spindle
MD 20090: SPIND_DEF_MASTER_SPIND = ***1***

Spindle number converter

MD 20092: SPIND_ASSIGN_TAB_ENABLE = 1	Activate spindle number converter
SD 42800: SPIND_ASSIGN_TAB[0] = ***1***	Master spindle as configured
SD 42800: SPIND_ASSIGN_TAB[1] = 1	Initial setting of table
SD 42800: SPIND_ASSIGN_TAB[2]=2	
SD 42800: SPIND_ASSIGN_TAB[3]=3	
SD 42800: SPIND_ASSIGN_TAB[4] = 0	Logical spindle not assigned

M3 S1000	Address extension = ***1***, M***1***=3 S***1***=1000 is output The spindle configured with number “1” (no. of physical master spindle) rotates.
...	
...	
SD 42800: SPIND_ASSIGN_TAB[1] = 5	Assign logical spindle 1 to physical spindle 5
SD 42800: SPIND_ASSIGN_TAB[2] = 3	Assign logical spindle 2 to physical spindle 3 **Caution:** Physical spindle 3 has now been assigned twice. When programming logical spindles 2 and 3, physical spindle 3 Physical is always addressed. In the basic machine displays, both spindles rotate.

SETMS (2)	SD 42800:SPIND_ASSIGN_TAB[0] = 2 is defined internally by the NCK.
...	
M5	Master spindle = address extension = 2, The converted spindle number is output M3=5 The physical spindle configured with number “3” stops.
...	
GET (S4)	Alarm 16105, because logical spindle “4” cannot be converted.
..	
RELEASE (S1)	Channel spindle “1” = physical spindle “5” is enabled
..	
M30	

© Siemens AG, 2003. All rights reserved

2.6 Programming

2.6.1 Programing from the parts program

Note

Detailed information about programming the spindle can be found in
References: /PG/, “Programming Guide: Fundamentals”

SETMS

The spindle stored in MD 20090: SPIND_DEF_MASTER_SPIND (initial setting of master spindle in channel) is the master spindle.

SETMS(n)

The spindle with the number (n) is the master spindle (may differ from the initial setting in MD 20090: SPIND_DEF_MASTER_SPIND.

The master spindle must be defined for the following functions:

- G95 Revolutional feedrate
- G96 S../G961 S ... Constant cutting rate in m/min or feet/min
- G97/G971 Cancel G96/G961 and freeze last spindle speed
- G63 Tapping with compensating chuck
- G33/G34/G35 Thread cutting
- G331/G332 Rigid tapping
- G4 S ... Dwell time in spindle revolutions
- Programming of M3, M4, M5, S, SPOS, M19, SPOSA, M40, M41 to M45 and WAITS without specifying the spindle number.

In SW 6.4 and higher, the current machine spindle setting can be retained via RESET and START. The setting is defined in machine data
MD 20110: RESET_MODE_MASK and MD 20112: START_MODE_MASK.
References: /FB/, K1 “Channel, Program Operation, Reset Response”

M3
M1=3

Clockwise direction of rotation for the master spindle
Clockwise direction of rotation for spindle number 1

M4
M2=4

Counterclockwise direction of rotation for the master spindle
Counterclockwise direction of rotation for spindle number 2

M5
M1=5

Spindle stop without orientation for the master spindle
Spindle stop without orientation for spindle number 1

Note

The M functions M3, M4, M5 and M70 are not output in DB21, ... DBB194 and DBB202 if a spibdle is configured in a channel. These M functions are offered as extended M functions in DB21, ... DBB68 ff. and in the particular axis DB, DB31, ... DBB86 ff.

© Siemens AG, 2003. All rights reserved

S....	Spindle speed in rpm for the master spindle
S2=	Spindle speed in rpm for spindle number 2
SPOS =270 **SPOS[n] =270**	Spindle positioning for the master spindle or the spindle with number n to the position 270 degrees. or the spindle with number n to the position 270 degrees. The block change is only performed when the spindle is in position.
SPOSA=90 **SOSS[n]=90**	Spindle positioning for the master spindle or the spindle with number n to the position 90 degrees. The block change is executed immediately. Spindle positioning continues, regardless of further parts program processing, until the spindle has reached its position.
SPOS=DC(Pos) **SPOS[n]=DC(Pos)** **SPOSA=DC(Pos)** **SPOSA[n]=DC(Pos)**	The direction of motion is retained for positioning while in motion and the position approached. When positioning from standstill, the position is approached via the shortest path.
SPOS=ACN(Pos) **SPOS[n]=ACN(Pos)** **SPOSA=ACN(Pos)** **SPOSA[n]=ACN(Pos)**	The position is always approached with negative direction of motion. If necessary, the direction of motion is inverted prior to positioning.
SPOS=ACP(Pos) **SPOS[n]=ACP(Pos)** **SPOSA=ACP(Pos)** **SPOSA[n]=ACP(Pos)**	The position is always approached with positive direction of motion. If necessary, the direction of motion is inverted prior to positioning.
SPOS=IC(Pos) **SPOS[n]=IC(Pos)** **SPOSA=IC(Pos)** **SPOSA[n]=IC(Pos)**	The travel path is specified. The direction of travel is obtained from the sign in front of the travel path. If the spindle is in motion, the direction of travel is inverted as necessary to allow traversing in the programmed direction. If the zero mark is crossed during traversing, the spindle is automatically synchronized with the zero mark if no reference is available or, if a new one has been requested per interface signal.
M19 **M[n]=19**	Spindle positioning for the master spindle or the spindle with number n to the position 270 degrees. or the spindle with number n to the position entered in SD 43249: M19_SPOS. The block change is only performed when the spindle is in position.
M70 **M1=70**	Bring spindle to standstill and activate position control, select zero parameter set, activate axis mode for the master spindle or for the spindle with number 1
SPCON **SPCON(n)** **SPCON(n,m)**	Spindle position control for the master spindle ON Spindle position control mode for the spindle number n ON Spindle position control mode for the spindle number n and m ON
SPCOF **SPCOF(n)** **SPCOF(n,m)**	Spindle position control for the master spindle OFF, activate speed control mode for master spindle OFF, spindle position control mode for the master spindle OFF, activate speed control mode for spindle number n, as well as for spindle number n and m
FPRAON (S2)	Revolutional feedrate for spindle S2 on, derived from the master spindle
FPRAON (S2, A)	Revolutional feedrate for spindle S2 on, derived from axis A. The revolutional feedrate value must be specified with FA[Sn].

© Siemens AG, 2003. All rights reserved

FPRAOF (S2) Revolutional feedrate for spindle S2 off.

C30 G90 G1 F3600 Rotary axis C (spindle in axis mode) travels to the position at 30 degrees at a speed of 3600 degrees/min=10 rpm.

G25 S.... Programmable minimum spindle speed limitation
G25 S2... Programmable minimum spindle speed limitation for the spindle with number 2

G26 S.... Programmable maximum spindle speed limitation
G26 Sn... Programmable maximum spindle speed limitation for the spindle with number n

LIMS= Programmable maximum spindle speed limitation with G96, G961, G97

WAITS Parts program, synchronization command for master spindle

Execution of the following blocks is suspended until the spindle(s) programmed with SPOSA has/have reached their position(s) with exact stop fine.

Waits until the spindle is at standstill after M5.

WAITS(n) Synchronization command for spindle n
WAITS(n,m) Synchronization command for spindles n and m:

FA[Sn] Programming of positioning speed (axial feed) for spindles in [deg/min]. The value configured in MD35300: SPIND_POSCTRL_VELO is valid again when FA[Sn]=0.

OVRA[Sn] Programming of axial override value for the spindle n in [%]

ACC[Sn] Programming of the axial acceleration capacity of the spindle n in [%]

SPI(n) Axis functions for a spindle with SPI(n) (SW 6.1 and higher)
With SPI(n) (spino), a spindle number is converted to the data type AXIS according to the machine data MD35000: SPIND_ASSIGN_TO_MACHAX[ ].
SPI is used to program axis functions using the spindle number. The following statements with SPI are possible:

Examples Frame statements with SPI:
CTRANS( ), CFINE( ), CMIRROR( ), CSCALE( )

Velocity and acceleration values for following spindles with SPI:
FA[SPI(n)], ACC[SPI(n)], OVRA[SPI(n)]

System variable with SPI:
$P_PFRAME[SPI(1),TR] = 2.22 can be used, e.g. to write frames.
$P_PFRAME = CTRANS (X, axis value, Y, axis value, SPI(1), axis value)
$P_PFRAME = CSCALE (X, scale, Y, scale, SPI(1), scale)
$P_PFRAME = CMIRROR (S1, Y, Z)
$P_UBFR = CTRANS (A, 10) : CFINE (19, 0.1)

For further explanations regarding the programming of SPI, please refer to:
Reference: /PGA/ “Programming Guide Advanced”

M40 Automatic gear stage selection for the master spindle
M1=40 Automatic gear stage selection for spindle number 1

© Siemens AG, 2003. All rights reserved

M41 to M45 Select gear stage 1 to 5 for the master spindle
M1=41 to M1=45 Select gear stage 1 to 5 for spindle number 1

2.6.2 Programming via synchronized actions

M40 to M45 (SW 6.2 and higher)

M functions M40 to M45 can also be programmed in synchronized actions. Please note the following:

Programming of **M40...M45** in the parts program has no effect on the current status of the automatic gear stage change of the synchronized actions and viceversa.

When programming S values with **M40**, the automatic gear stage change acts separately for synchronized actions and the parts program.

For synchronized actions:

- M40 is deactivated after POWER ON. The gear stage is not adjusted if an S value is specified from a synchronized action.
- An M40 command programmed using synchronized actions always remains active for synchronized actions (modal) and is not reset on Reset.

M41...M45 selects the first to fifth gear stage according to the programming instructions in the parts program. Please note the following:

- An axis replacement is necessary in order to run the function.
- When the gear stage change has been performed, the spindle status is neutral (same response as M3, M4, M5 instructions).

Note

For further details explaining the programming of spindles and spindle movements from synchronized actions, please refer to:
Reference: /PGA/, “Programming Guide Advanced”
/FBSY/ , “Description of Functions, Synchronized Actions”

© Siemens AG, 2003. All rights reserved

2.6.3 Programming spindle controls via the PLC with FC18

Automatic gear stage change on FC18 (SW 6.2 and higher)

When the PLC specifies the direction of rotation and speed using FC18, the NCK can determine and select a gear stage to match the speed. This is equivalent to the M40 functionality when programming via the parts program.

The correct start code must be set when FC18 is called in a PLC user program, in order to activate the gear stage selection.

Note

For detailed explanations on how to program spindle controls via the PLC with FC18, please refer to:
References: /FB1/, P3, “Basic PLC Program”

2.6.4 Special spindle movements via the PLC interface (SW 6.1 and higher)

Why use a special spindle interface?

This function can be used to program the spindle via an axial PLC interface alternatively to the FC18. For the sake of simplicity, this results in a slightly reduced functionality. This functionality can be used preferably for simple control applications.

Functionality

Special VDI interface signals are provided to start and stop spindles outside the running parts program. To do so, the channel status and the program status need not be in the active mode. These states will occur, e.g. in case of Reset and in JOG mode.

The spindle concerned must have the state “Channel axis” or “Neutral axis” and must not be moved using the JOG keys or positioned by FC18 or synchronized actions. If these conditions are fulfilled, spindle jobs will be accepted via the DBB30 spindle interface.

The spindle specifications are kept beyond a mode change (e.g. from JOG mode to AUTOMATIC). With the start of the parts program, the spindle settings are accepted (direction of rotation, speed or cutting rate) into the parts program and can be modified again by parts program statements. In JOG mode, the spindle can be moved at the speed last programmed.

Multi-channel operation

In the case of multi-channel operation, the spindle started by the PLC becomes active in the channel that at the appropriate moment handles the spindle. This channel can be determined on side of the PLC by reading the IS “NC axis/spindle in channels A to D” DB31, ... DBX68.0–68.3.

© Siemens AG, 2003. All rights reserved

Spindle job

In order to start a spindle job, the channel handling the spindle must be in the acceptance status. A spindle job is always started with the LowHigh edge of a DBB30 signal.

Generally, the DBB30 start signals do not have any meaning in the static status and do not prevent the spindle to be programmed by FC18, synchronized actions, parts program or JOG traversing movements (e.g. when the STOP signal is statically on “1”).

Conditions for the acceptance status

Outside a running parts program, spindles can be started and stopped using the special VDI interface signals NCK –> PLC.

- To this aim, the **channel status** must be in the mode
  “interrupted” DB21, ... DBX35.6 = 1 or
  “Reset” DB21, ... DBX35.7=1 **and**
  the **program status** must be in the mode
  “interrupted” DB21, ... DBX35.6 = 1 or
  “aborted” DB21, ... DBX35.4 = 1.

These states will occur in case of Reset and in JOG mode. At the starting time, the spindle concerned must fulfill the following requirements:

- It must be in the status “Channel axis” or “Neutral axis” and must not be moved by means of the JOG keys.
- When the spindle is specified, no positioning movement may be carried out by FC18 or synchronized actions.

Spindle job outside the acceptance range

Low-high edges outside the acceptance range will be ignored. No alarm message is output by the NCK. It can be assumed that the acceptance range will be announced to the operator by the PLC program.

Spindle jobs outside the acceptance range can also be carried out using the functions FC18 or the ASUB.

SD 43200 Writing data

Setting data SD 43200: SPIND_S can be programmed as follows:

- Through speed programming
- Through direct programming in the parts program
- Through the MMC.

Note

The direct writing of the setting data comes into effect immediately and acts asynchronously to the parts program execution.

© Siemens AG, 2003. All rights reserved

Conditions for writing

The following conditions apply for writing data to the setting data SD 43200: SPIND_S:

Programming through:	Conditions for programming:
Speed programming	MD 35035 bit 4 = 1 must be set.
	The constant cutting rate G96, G961 must not be active.
	The IS “Constant cutting rate” DB31, ... DBX 84.0 = 0 must be set.
Direct programming in the parts program	A programmed S value and the value of the directly programmed SD can overtake as far as the time is concerned. In this case, after programming the SD you should use the statement STOPRE.
MMC: Direct programming in the parts programs	Only positive values including zero are accepted. Otherwise, an appropriate message is generated.

Spindle commands

The following basic logic functions can be specified for the spindle:

Basic logic functions:	Spindle commands:
Motion command: Spindle stop Spindle start clockwise rotation Spindle start counterclockwise rotation Select gear stage Spindle positioning	Independently of a running parts program: Input from PLC via axial VDI interface signal DB31, ... DBB30
Speed: MD 35035 bit 4 = 1 When the movement is started, the speed is read from SD 43200: SPIND_S	Speed settings from the parts program or FC18 are always written to the setting data SD 43200: SA_SPIND_S.
Setpoint speed: MD 35035 bit 5 = 1 The contents of SD 43200: SA_SPIND_S act as a setpoint speed	You can use the JOG keys to traverse the spindle with the speed from SD 43200: SA_SPIND_S.
Constant cutting rate: MD 35035 bit 8 = 1 is read from setting data SD 43202: SPIND_CONSTCUT_S	Settings of the constant cutting rate from the parts program, FC18 and synchronized actions are written to the setting data SD 43200: SA_SPIND_S.

Notes on MD 35035: SPIND_FUNCTION_MASK:
The following additionally applies to bit 4 and bit 8:

Programmed S values that are no programmed speed values are **not** written to the appropriate setting data.
This also applies, e.g. to an S value with revolutionrelated dwell time (G4).

If this conditions are fulfilled, spindle jobs will be accepted via the DBB30 interface (acceptance status).

The functionality of bits 0 to 2 is described in **Subsection 2.6.5** Gear stage change in DryRun, Program Test and SERUPRO modes.

© Siemens AG, 2003. All rights reserved

Motion specification

The direction of rotation is specified in DBB30 and transmitted by the NCK input signals at the axial VDI interface (PLC –> NCK) in data bytes DB31, ... DBB30 (JobShop).

Interface signals

• "Spindle stop" (corresponds to M5)	DB31, ... DBX30.0
• "Spindle start CW" (corresponds to M3)	DB31, ... DBX30.1
• "Spindle start CCW" (corresponds to M4)	DB31, ... DBX30.2
• "Select gear stage" (available soon)	DB31, ... DBX30.3 (reserved signal)
• "Spindle positioning" (corresponds to M19)	DB31, ... DBX30.4

Priorities

If several DBB30 signals are set at the same time, the following order of priorities is defined:

• Spindle stop (M5)	1st priority
• Spindle start/positioning mode (M3)	2nd priority
• Spindle start/positioning mode (M4)	3rd priority
• Spindle start/positioning mode (M19)	4th priority

Acknowledgment spindle start/stop

A spindle start can be detected at the VDI interface based on traversing commands output, such as "Traversing command minus" DB31, ... DBX64.6 = 1or "Traversing command plus" DB31, ... DBX64.7 = 1.

A spindle stop is executed if the IS Axis/spindle stationary "Axis/spindle stationary" DB31, ... DBX61.4 = 1 is signalled.

Direct acknowledgment signals for the DBB30 input signals will not be output.

Invert M3/M4

IS signal "Invert M3/M4" (DB31, ... DBX17.6 acts analogously to the programming of the direction of rotation for the signals
"Spindle start CW" DB31, ... DBX30.1 and
"Spindle start CCW" DB31, ... DBX30.2 of M3 or M4 via parts program, synchronized action or FC18.

© Siemens AG, 2003. All rights reserved

Setting the speed

Speed settings from the parts program, FC18 or synchronized actions are always written to the setting data SD 43200: SPIND_S and always from the usual sources.

Read SD 43200

When the movement is started, the speed is always read from the

setting data SD 43200: SPIND_S [r.p.m.]

with the positive edge of the start signals
“Spindle start CW” DB31, ... DBX30.1 and
“Spindle start CCW” DB31, ... DBX30.2

Note

Reprogramming of the setting data until the next positive edge of the start signals “Spindle start CW” DBB31, ... “Spindle start CCW” DB31, ... DBX30.2 remains inactive for the current spindle speed. The contents of the setting data is only accepted with the next spindle start signal.

Gear stage change and influence on the speed

In the current version, no gear stage change is triggered if the set speed is out of the speed range of the gear stage. The usual speed limitations and the speed increase to the set speed are active.

Setting the constant cutting rate

Settings of the constant cutting rate from the parts program, FC18 or synchronized actions are written to the setting data SD 43200: SPIND_S and always from the usual sources.

Read SD 43202

The constant cutting rate is read from the

setting data SD 43202: SPIND_CONSTCUT_S [m/min] or [feet/min]

with the positive edge of the start signals
“Spindle start CW” DB31, ... DBX30.1 and
“Spindle start CCW” DB31, ... DBX30.2

Note

Reprogramming of the setting data until the next positive edge of the start signals “Spindle start CW” DBB31, ... “Spindle start CCW” DB31, ... DBX30.2 remains inactive for the current constant cutting speed. The contents of the setting data is only accepted with the next spindle start signal.

© Siemens AG, 2003. All rights reserved

Condition

To make sure that settings for the constant cutting rate are active, the spindle concerned must be a master spindle in the channel handling the spindle. This condition is fulfilled if the IS “Constant cutting rate active” DB31, DBX84.0 = 1 is set at the axial VDI interface.

Writing from the parts program

When writing the appropriate value for the constant cutting rate from the parts program, it is interpreted as follows if the following is set/active in the 12th G group:

- G710 is active — metric
- G700 is — inch as [feet/min]

With G70, G71 and when writing from an external source (MMC), the setting in MD 10240: MN_SCALING_SYSTEM_IS_METRIC will decide how the programmed value is interpreted.

For further explanations on the metric/inch unit system, please refer to
References: /FB/, G2, “Velocities, Setpoint/Actual Value Systems, Closed-Loop Control”

Setting via FC18 synchronized actions

If the constant cutting rate is set via FC18, the decision how the speed value (bytes 8...11) is interpreted is decided by setting bit 6 in byte 2 in the area “Signals to concurring positioning axes”.

In the case of setting via synchronized actions, the feedrate type will decide how the S value is interpreted, analogously to the parts program.

Reading from parts program and synchronized actions

The programmed cutting rate value can be determined both in the parts program and in synchronized actions by reading the system variables $P_CONSTCUT_S and $AC_CONSTCUT_S.

Reading the programmed cutting rate value via the OPI interface is also possible.

System variables

WB: Defined range of values of the two new system variables.

Description	NCK variables
Constant cutting rate last programmed WB = {0, DBL_Max}	$P_CONSTRUCT_S[n]
Current constant cutting rate WB = {0, DBL_Max}	$AC_CONSTRUCT_S[n]

© Siemens AG, 2003. All rights reserved

Spindle-specific functions (SW 6.1 and higher)

MD 35035

The MD 35035: SPIND_FUNCTION_MASK can be used to set spindle-specific functions as follows from the parts program, FC18 and synchronized actions:

SW 6.2 and higher no gear stage change
with DryRun, program testing and SERUPRO

- Bit 0 = 1 Gear stage change for blocks with M40, M41 to M45 or via FC18 and synchronized actions are suppressed with
- Bit 1 = 1 Gear stage change for blocks with M40, M41 to M45 or via FC18 and synchronized actions are suppressed with program testing and SERUPRO.
- Bit 2 = 1 Gear stage change for programmed gear stage is performed after deselection of functions DryRun and SERUPRO.

SW 6.1 and higher Acceptance of programmed speed and cutting rate including settings.

- Bit 4 = 1 The programmed speed including speed settings programmed using FC18 and synchronized SD 43200: SPIND_S.
- Bit 5 = 1 The content of SD 4320: SA_SPIND_S acts as the set speed in JOG mode. You can use the JOG keys to traverse the spindle at the speed defined in SD 43200. If the content is zero, further JOG speed settings come into effect (see SD 41200: JOG:SPIND_SET_VELO).
- Bit 8 = 1 The programmed cutting rate including the settings via FC18 and the synchronized actions are accepted into the SD 43202: SPIND_S.

For further information about MD 35035: SPIND_FUNCTION_MASK, please see **Section 4.3** “Axis/spindle-specific machine data”

© Siemens AG, 2003. All rights reserved

2.6.5 Gear stage change in DryRun, Program Test and SERUPRO modes (SW 6.2 and higher)

Parts program, synchronized actions and FC18

The behavior of the gear stage change from parts programs, FC18 and synchronized actions for the DryRun, Program Test and SERUPRO (SearchRunByProgramTest) functions can be configured with the existing machine data MD 35035: SPIND_FUNCTION_MASK .

A gear stage change is not usually necessary for these functions and can therefore be suppressed as follows in MD 35035: SPIND_FUNCTION_MASK with bits 0 to 2:

- **Dry run feedrate (DryRun)**,

  Bit 0 = 0 Gear stages are activated even with the DryRun function active for parts program blocks with M40, M41 to M45 or via FC18 and synchronized action programming. (previous response)

  Bit 0 = 1 Gear stage change for blocks with M40, M41 to M45 or via FC18 and synchronized actions are suppressed with DryRun.

- **Program testing and SERUPRO,**

  Bit 1 = 0 Gear stages are activated even with the Program Test function active for parts program blocks with M40, M41 to M45 or via FC18 and synchronized action programming. (previous response)

  Bit 1 = 1 Gear stage change for blocks with M40, M41 to M45 or via FC18 and synchronized actions are suppressed with program testing and SERUPRO.

- **DryRun, program testing and SERUPRO**

  Bit 2 = 0 Gear stage change for programmed gear stage is performed subsequently on REPOS after deselection of functions DryRun, Program Test and SERUPRO.

  Bit 2 = 1 Gear stage change for programmed gear stage is performed after deselection of functions DryRun and SERUPRO if possible.

Gear stage change suppression

If a gear stage change is suppressed, the interpolator limits the programmed spindle speed to the permissible speed range of the active gear stage if necessary. The interface signals

IS “Setpoint speed increased” (DB31,...DBX83.2) and
IS “Setpoint speed limited” (DB31,...DBX83.1), which are generated on such a restriction, are suppressed. Monitoring by the PLC program is not necessary during DryRun.

When the gear stage change is suppressed, no new gear stage setpoint IS “Gear stage setpoint” (DB31,...DBX82.0–2) is output to the PLC.

The gear stage change request IS “Gear change” (DB31,...DBX82.3) is also suppressed. This ensures that no gear stage change information has to be processed by the PLC program.

© Siemens AG, 2003. All rights reserved

Determining the last active gear stage

System variable $P_GEAR returns the gear stage programmed in the parts program (which may not have been output to the PLC). System variable $AC_SGEAR can be used to read the last active gear stage from the parts program, synchronized actions and operator panel interface.

The DryRun function can be deselected within a running parts program. Once it has been deselected, the correct gear stage requested by the parts program must be determined and selected.

It cannot be assured that the remainder of the parts program will run without errors until the correct gear stage has been activated. Any necessary gear stage change is performed in the system REPOS started on deselection when the spindle is in speed control mode. A complete gear stage change dialog takes place with the PLC and the last programmed gear stage is activated.

If there is a mismatch between the gear stage programmed in the parts program and the actual gear stage returned over the VDI interface at REPOS, no gear stage change takes place.

The same applies to the SERUPRO function.

For further information about SERUPRO block search, please refer to
References: /FB/, K1, “Mode Group, Channels, Program Operation Mode”

Supplementary conditions

If the gear stage change is suppressed, the output spindle speed moves within the speed range specified by the current gear stage.

The following restrictions apply to the subsequent activation of a gear stage on REPOS:

- The gear stage change is not activated subsequently if the spindle in the deselection or target block is a command spindle (synchronized action) or PLC spindle (FC18).
- If the gear stage cannot be activated because the spindle is in position or axis mode or a link is active, alarm 22011“Channel%1 block%3 spindle% Change to programmed gear stage not possible” is signaled.

Example

Gear stage change in DryRun

```
                    1st gear stage (GS) for initial state;

N00 M3 S1000 M41            ; 1st GS is activated
M0                          ; Parts program stops

                    PI service: Activate DryRun;
                                ; (configuring)
N10 M42                     ; 2nd GS requested, no gear stage change takes place
N11 G0 X0 Y0 Z0             ; Position axes
N12 M0                      ; Parts program stops

                    PI service: Deactivate DryRun;

                                ; REORG and REPOS are performed
                                ; Gear stage change now takes place to 2nd gear stage
N20 G1 Z100 F1000           ;
...                         ;
N99 M30                     ; Parts program end
```

© Siemens AG, 2003. All rights reserved

2.6.6 Programming from external (PLC, MMC/HMI)

Traversing at revolutional feedrate can be selected via the axial setting data MD 43300: ASSIGN_FEED_PER_REV_SOURCE (revolutional feedrate for spindles) in JOG mode via the channelspecific setting data MD 42600: JOG_FEED_PER_REV_SOURCE (revolutional feedrate control in JOG mode). The following settings can be made via the setting data:

- >0: The machine axis number of the rotary axis/spindle from which the revolutional feedrate is to be derived.
- –1: The revolutional feedrate is derived from the master spindle of the particular channel in which the axis/spindle is active.
- 0: Function is deselected.

FPRAON (S2) Revolutional feedrate for spindle S2 on, derived from the master spindle

FPRAON (S2, A) Revolutional feedrate for spindle S2 on, derived from axis A. The revolutional feedrate value must be specified with FA[Sn].

FPRAOF (S2) Revolutional feedrate for spindle S2 off.

SPI(n) (SW 6.1 and higher) It is also possible to program SPI(n) instead of SPI(Sn).

© Siemens AG, 2003. All rights reserved

2.7 Spindle monitoring

Speed ranges

The spindle monitoring functions and the currently active functions (G94, G95, G96, G961, G97, G971, G33, G34, G35, G331, G332, etc.) define the admissible speed ranges of the spindle.

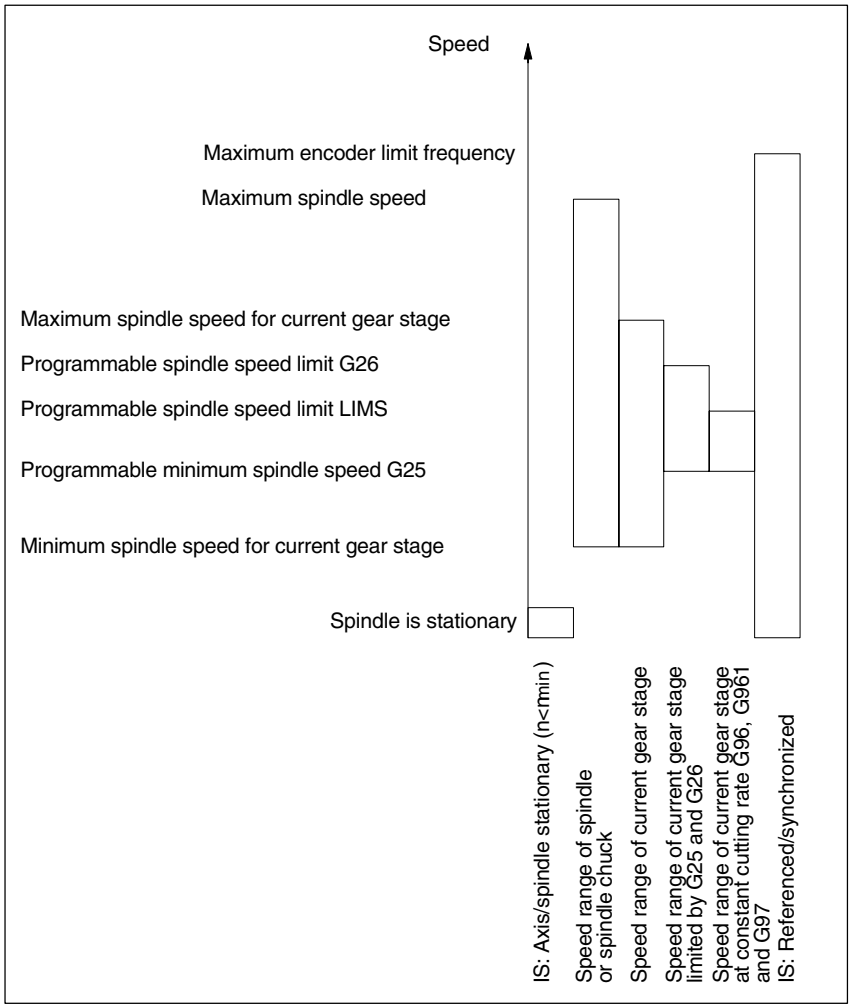


Fig. 2-11 Ranges for spindle monitoring/speeds

© Siemens AG, 2003. All rights reserved

2.7.1 Axis/spindle stationary ($n < n_{min}$)

Only when the axis/spindle is stationary**,** i.e. the actual spindle speed is below a value defined in MD 36060: STANDSTILL_VELO_TOL (maximum velocity/ speed “Axis/spindle stationary”), is it possible to perform certain functions on the machine, such as tool change, open machine doors, path feed enable, etc. If the spindle is not moving,

- IS “Axis/spindle stationary” (DB31, ... DBX61.4) is set to one.
- The next machining block is enabled
  MD 35510: SPIND_STOPPED_AT_IPO_START = 0 is enabled
  Exception: MD 35510: SPIND_STOPPED_AT_IPO_START =1
  path interpolation is not affected, traversing path axes not stopped

Monitoring is effective all spindle modes and axis mode.

2.7.2 Spindle in setpoint range

Function

The “Spindle in setpoint range” monitor checks whether the programmed spindle speed has been reached, whether the spindle is stationary (IS “Axis/spindle stationary”) or whether it is still in the acceleration or braking phase.

In the spindle mode control mode, the speed setpoint (programmed speed * spindle offset including the active limitations) is compared with the actual speed. If the deviation of the actual speed from the speed setpoint is greater than the spindle speed tolerance of MD 35150: SPIND_DES_VELO_TOL,

- the axial IS “Spindle in setpoint range” (DB31, ... DBX83.5) is set to zero
- The next machining block is not enabled
  MD 35500: SPIND_ON_SPEED_AT_IPO_START = 1 is enabled
  Exception: MD 35500: SPIND_STOPPED_AT_IPO_START =0 Path interpolation is not affected, traversing path axes not stopped

Tolerance range for speed setpoint

MD 35150: SPIND_DES_VELO_TOL = 0,1
The spindle actual speed may deviate +/– 10% compared to the set speed.

The spindle speed is obtained from the programmed speed and the current limits (taking the current limits into account.
A limitation or increase in the programmed speed is indicated by the axial IS “Set speed limited” (DB31, ... DBX83.1) or IS “Setpoint speed increased” (DB31, ... DBX83.2) is displayed and does **not** prevent the speed tolerance range being reached.

If the spindle lies inside the tolerance range, the axial IS “Spindle in set range” (DB31, ... DBX83.5) is set to 1 at the VDI interface. Special case: If the speed tolerance is set to zero, the axial IS “Spindle in set range” (DB31, ... DBX83.5) is permanently set to “1” and no path control takes place.

Speed change

The path control only takes place at the start of the traverse block and only if a speed change has been programmed. If the speed tolerance range is violated, e.g. due to an overload, the path control is not automatically brought to a standstill.

© Siemens AG, 2003. All rights reserved

2.7.3 Minimum/maximum speed for the gear stage

Minimum speed

The MD 35140: GEAR_STEP_MIN_VELO_LIMIT contains the minimum gear stage speed. It is not possible that the speed falls below this (set) speed if an S value is programmed, which is too small. Then, the “Setpoint speed increased” interface signal (programmed speed to low) is enabled.

The minimum gear stage speed is operative only in speed mode; the speed of the gear stage can only fall below the minimum limit through:

- Spindle override 0%
- M5
- S0
- IS “Spindle stop” (DB31, ... DBX8.3)
- IS “Servo enable” (DB31, ... DBX2.1)
- IS “Reset” (DB21, ... DBX7.7)
- IS “Delete distance-to-go/Spindle Reset” (DB31, ... DBX2.2)
- IS “Oscillation speed” (DB31, ... DBX18.5)
- “NC-STOP axes plus spindles” interface signal (DB21, ... DBX7.4)
- IS “Axis/spindle lock “(DB31, ... DBX1.3)
- “Delete S value” interface signal (DB31, ... DBX16.7)

Maximum speed

The MD 35130: GEAR_STEP_MAX_VELO_LIMIT defines the maximum speed for the gear stage. In the gear stage engaged, this set speed can never be exceeded. When the programmed spindle speed is limited, the IS “Set speed limited” (DV31, ... DBX83.1) (Programmed speed too high) is enabled.

2.7.4 Maximum encoder limit frequency



Warning

The maximum encoder frequency limit of the actual spindle position encoder is monitored by the control (the limit can be exceeded). It is the responsibility of the machine tool manufacturer to ensure that the configuration of the spindle motor, gears, measurement gears, encoder and machine data prevents the maximum speed of the actual spindle position encoder from being exceeded.

© Siemens AG, 2003. All rights reserved

Maximum encoder frequency exceeded

If the spindle speed reaches a speed (large S value programmed), which exceeds the maximum encoder limit frequency (the maximum mechanical speed limit of the encoder must not be exceeded), the synchronization is lost. The spindle continues to rotate, but with reduced functionality.

With the following functions, the spindle speed is reduced until the active measurement system is operating below the encoder limit frequency again:

- Thread cutting (G33, G34, G35)
- Tapping without compensating chuck (G331, G332)
- Revolutional feedrate (G95)
- Constant cutting rate G96, G961, G97, G971)
- SPCON (position controlled spindle operation).

If the encoder limit frequency is exceeded, the “Referenced/synchronized 1” or “Referenced / synchronized 2” interface signal is reset for the relevant measurement system and the “Encoder limit frequency 1 exceeded” or “Encoder limit frequency 2 exceeded” interface signals enabled.

If the spindle is in axis mode, the maximum encoder limit frequency must not be exceeded. The maximum speed (MD 32000: MAX_AX_VELO) must lie below the maximum encoder limit frequency; otherwise, alarm 21610 is output and the axis brought to a standstill.

Maximum encoder limit frequency exceeded

If the maximum encoder frequency limit has been exceeded and the speed subsequently falls below the maximum encoder limit frequency (smaller S value programmed, spindle offset switch changed, etc.), the spindle is automatically synchronized with the next zero mark or the next Bero signal. The new synchronization will always be carried out for the active position measuring system that has lost its synchronization and the max. encoder limit frequency has currently not been reached.

Special points to be noted

If the following functions are active, the maximum encoder frequency cannot be exceeded.

- Spindle mode positioning mode, axis mode
- Thread cutting (G33, G34, G35)
- Rigid tapping G331, G332 (does not apply to G63)
- Revolutional feedrate (G95)
- Constant cutting rate G96, G961, G97, G971)
- SPCON (position-controlled spindle operation).

© Siemens AG, 2003. All rights reserved

2.7.5 End point monitoring

End point monitoring

During positioning (the spindle is in positioning mode), the system monitors the distance from the spindle (with reference to the actual position) to the programmed spindle set position (destination point).

Two limit values can be defined in MD 36000: STOP_LIMIT_COARSE (exact stop coarse limit) and MD 36010: STOP_LIMIT_FINE as the incremental path starting at the spindle set position. The positioning of the spindle is always as accurate as the connected spindle measurement encoder, the backlash, the transmission ratio, etc.

Different exact stop windows for different parameter sets

In SW 5.1 and higher, different exact stop windows can be configured for different parameter sets. This makes it possible to work to different levels of accuracy in axis mode and spindle positioning. The exact stop window can be configured separately for each gear stage for spindle positioning.

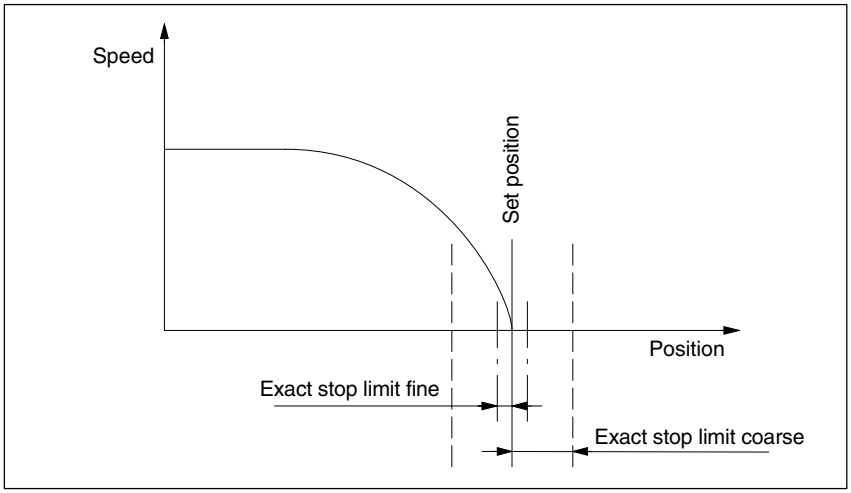


Fig. 2-12 Exact stop zones of a spindle

IS “Position reached with exact stop coarse/fine” DB31, ... DBX60.7 DB31, ...DBX60.6

The two limits defined by MD 36000: STOP_LIMIT_COARSE and MD 36010: STOP_LIMIT_FINE (exact stop limit coarse and fine) reached with the IS “Position reached with exact stop coarse” exact stop” (DB31, ... DBX60.7) and IS “Position reached with exact stop fine” (DB31, ... DBX60.6) are output to the PLC.

Block change for SPOS and M19

If the spindle is being positioned with SPOS or M19, the block change will be dependent on the end point monitoring with the IS “Position reached with exact stop fine”. All other functions programmed in the block must have achieved their end criterion (e.g. all auxiliary functions acknowledged by the PLC).

With SPOSA, the block change does not depend on the monitoring of the destination point.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

There are no supplementary conditions stipulated for this Description of Functions.

4 Data Descriptions (MD, SD)

4.1 General machine data

10192 MD number	**GEAR_CHANGE_WAIT_TIME** Wait time for of a gear stage change when reorganizing		
Default setting: 10	Minimum input limit: 0	Maximum input limit: 100000	
Changes effective after POWER ON	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	External events that initiate reorganizing will wait until a gear stage change is completed. The machine data MD 10192: GEAR_CHANGE_WAIT_TIME will now determine how long it is waited for the gear stage change. If this time expires without completing the gear stage change, the NCK will response with the following alarm. Amongst others, the following events result in reorganizing: • User ASUB • Mode change • Delete distance-to-go • Axis replacement • Activate user data or machine data • Switch over skip or dry run • Edit in modes • Correction block alarms • Overstore • Rapid retraction with G33, G34, G35 • Subroutine level abortion, subroutine abortion		
Related to	MD 35010: GEAR_STEP_CHANGE_ENABLE gear stage changeover possible		

© Siemens AG, 2003. All rights reserved

10714 MD number	**M_NO_FCT_EOP** M function for spindle active after RESET		
Default setting: –1	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Spindles for which a “2” is configured in MD 35040: SPIND_ACTIVE_AFTER_RESET are not reset with this M function once the parts program has been completed. The spindle thus remains active after the end of the parts program. Suggestion (JobSop): M32 M functions with predetermined meaning may not be configured in MD 10714: M_NO_FCT_EOP. These are: ● M0 to M5, ● M17, M30, ● M19, ● M40 to M45. ● M functions for spindle mode/axis mode switchover acc. to machine data MD 20094: SPIND_RIGID_TAPPING_M_NR (default: M70), ● M functions for nibbling/punching as configured via machine data MD 26008: NIBBLE_PUNCH_CODE and MD 26012: PUNCHNIB_ACTIVATION ● M functions for interrupt programming as configured via machine data MD 10804: EXTERN_M_NO_SET_INT and MD 10806: EXTERN_M_NO_DISABLE_INT (default: M96 and M97). ● For applied external language with MD 18800: MM_EXTERN_LANGUAGE M96 and M99 additionally.		
Related to	MD 35040: SPIND_ACTIVE_AFTER_RESET (spindle active via reset)		

© Siemens AG, 2003. All rights reserved

4.2 Channel-specific machine data

20090 MD number	**SPIND_DEF_MASTER_SPIND** Initial setting for master spindle on channel		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 5 (SINUMERIK 840D) 1 (SINUMERIK FMNC/810D)	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Each channel must have a master spindle for the following functions: ● G95 Revolutional feedrate ● G96/G961 S1→Spindle 1 Constant cutting rate in m/min or ft/min (for SINUMERIK FM-NC only) ● G97/G971 Cancel G96/G961 and freeze last spindle speed ● G63 Tapping with compensating chuck ● G33 Thread cutting ● G34 Thread lead increase (progressive velocity change) ● G35 Thread lead decrease (degressive velocity change) ● G331/G332 Rigid tapping ● G4 S1→Spindle 1 Dwell time in spindle revolutions The master spindle can also be programmed with the commands M3, M4, M5, S, SPOS, WAITS, SPOSA, M19, M40, M41 to M45 without specifying the spindle number. The MD 20090: SPIND_DEF_MASTER_SPIND, the reset value of the master spindle applies until a new master spindle is defined with the SETMS program command. The STEMS setting is deleted with NC start. Following an M02/M30 and new NC start, the spindle defined in SPIND_DEF_MASTER_SPIND is always the master spindle.		
Application example(s)	Two spindles (a main spindle and an auxiliary spindle as a rotating tool) are provided on a turning machine. The main spindle is declared as the master spindle in MD: SPIND_DEF_MASTER_SPIND. The SETMS program command can be used to declare the auxiliary spindle as the master spindle for drilling and face end machining (milling). It is therefore possible to program the auxiliary spindle with M3, M4 and M5 (with no address extension) or to perform rigid tapping.		

20092 MD number	**SPIND_ASSIGN_TAB_ENABLE** Enabling/disabling of spindle converter	
Default setting: 0,0,0,0, 0,0,0,0, 0,0,0,0, 0,0,0,0	Minimum input limit: 0	Maximum input limit: 1
Changes effective after RESET	Protection level: 7/7	Unit: –
Data type: BYTE	Applies from SW 4.3	
Significance:	Value 0: The spindle converter function is deactivated. The contents of SD 42800: SPIND_ASSIGN_TAB[..] are not evaluated. Value 1: The spindle converter is activated. Conversion from logical to physical spindle takes place. For more information, see SD 42800: SPIND_ASSIGN_TAB. Note: After “Delete SRAM” (start-up switch to position “1”), the spindle converter is deactivated.	
Special cases, errors, ...		
Related to	SD 42800: SPIND_ASSIGN_TAB	

© Siemens AG, 2003. All rights reserved

20850 MD number	**SPOS_TO_VDI** Output of the auxiliary function “M19” to the VDI interface for SPOS/SPOSA		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 7/7	Unit: –	
Data type: BYTE	Applies from SW SW 5.3		
Significance:	0: No “M19” is output to the VDI interface for SPOS and SPOSA. This saves the acknowledgment time of the M function. 1: No “M19” is output to the VDI interface for SPOS and SPOSA. Response corresponds to spindle position with “M19” from the parts program.		

22400 MD number	**S_VALUES_ACTIV_AFTER_RESET** S function via RESET active		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW SW1.1		
Significance:	0: After a reset, the various S values are equal to 0 and must therefore be reprogrammed. 1: The last S values set in main run are also effective after a reset.		

© Siemens AG, 2003. All rights reserved

4.3 Axis/spindle-specific machine data

4.3.1 Referencing/synchronizing

31122 MD number	**BERO_DELAY_TIME_PLUS** BERO delay time in plus direction		
Default setting: 0.000110	Minimum input limit: 0.0	Maximum input limit: 1.0	
Changes effective after NewConfig	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 4.1		
Significance:	In conjunction with the setting MD 34200: ENC_REFP_MODE (Referencing mode) = 2 or 7, a signal runtime compensation in the positive direction of movement with a BERO (zero mark), (see/FB1/, R1, Reference Point Approach). The typical overall delay time of the BERO detection range is entered for crossing in the positive direction. The time covers: the BERO edge delay time the signal digitization time the measured value processing time, etc. The times depend on the hardware used. The default value is typical for SIEMENS products. Customized calibration is necessary only in exceptional circumstances. Entering the minimum value “0.0” switches the compensation off; this makes sense in conjunction with MD 34200: ENC_REFP_MODE = 7. The machine data is available for each encoder.		
Related to	MD 34200: ENC_REFP_MODE (referencing mode) MD 34040: REFP_VELO_SEARCH_MARKER[n] (reference point creep speed [encoder no.])		

31123 MD number	**BERO_DELAY_TIME_MINUS** BERO delay time in minus direction		
Default setting: 0.000078	Minimum input limit: 0.0	Maximum input limit: 1.0	
Changes effective after NewConfig	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 4.1		
Significance:	In conjunction with the setting MD 34200: ENC_REFP_MODE (referencing mode) = 2 or 7, a signal runtime compensation in the negative direction of movement with a BERO (zero mark), (see/FB1/R1, Reference point approach). The typical overall delay time of the BERO detection range is entered for crossing in the negative direction. The time covers: the BERO edge delay time the signal digitization time the measured value processing time, etc. The times depend on the hardware used. The default value is typical for SIEMENS products. Customized calibration is necessary only in exceptional circumstances. Entering the minimum value “0.0” switches the compensation off; this makes sense in conjunction with MD 34200: ENC_REFP_MODE = 7. The machine data is available for each encoder.		
Related to	MD 34200: ENC_REFP_MODE (referencing mode) MD 34040: REFP_VELO_SEARCH_MARKER[n] (creep speed [encoder no.])		

© Siemens AG, 2003. All rights reserved

4.3.2 Intermediate gear

31044 MD number	**ENC_IS_DIRECT2[n]** Encoder at intermediate gear		
Default setting: 0.0 (FALSE)	Minimum input limit: 0.0 (FALSE)	Maximum input limit: 1.0 (TRUE)	
Changes effective after NewConfig	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.4		
Significance:	Encoder installation location behind load intermediate gear When using a load intermediate gear (e.g. for power tools, compare MD 31066 DRIVE_AX_RATIO2_NUMERA und MD 31064 DRIVE_AX_RATIO2_DENOM), the encoder installation location can be defined as “on drive” of the load intermediate gear: The position for encoder installation is configured through combination of the the two machine data MD 31040: ENC_IS_DIRECTE and MD 31044: ENC_IS_DIRECT2: For encoder installation “on drive of the load intermediate gear”, simultaneously set MD 31040: ENC_IS_DIRECT=TRUE **and** MD 31044: ENC_IS_DIRECT2=TRUE For encoder installation “on drive of the load intermediate gear”, set only MD 31040: ENC_IS_DIRECT=TRUE with MD 31044: ENC_IS_DIRECT2=FALSE. A parameter alarm is not issued for a non-defined combination. Set is MD 31044: ENC_IS_DIRECT2=TRUE with MD 31040: ENC_IS_DIRECT=FALSE.		
Related to	MD 31066: DRIVE_AX_RATIO2_NUMERA (numerator intermediate gear) MD 31064: DRIVE_AX_RATIO2_DENOM (denominator intermediate gear) MD 31040: ENC_IS_DIRECT (encoder is connected directly at the machine)		
References	/FB1/, Velocities, Setpoint/Actual Value Systems, Closed-Loop Control		

31064 MD number	**DRIVE_AX_RATIO2_DENOM** Intermediate gear		
Default setting: 1.0	Minimum input limit: 1.0	Maximum input limit: 2147000000	
Changes effective after NewConfig	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Denominator intermediate gear In conjunction with machine data MD 31066 DRIVE_AX_RATIO2_NUMERA, defines a intermediate gear that acts as a multiplier on the motor&load gearbox. The motor/load gearbox is described bz the machine data MD 31060: DRIVE_AX_RATIO_NUMERA (numerator load gearbox) MD 31050: DRIVE_AX_RATIO_DENOM (denominator load gearbox) The load intermediate gear is inactive for default values 1:1. For encoder installation, observe MD 31044: ENC_IS_DIRECT2! When the “Safety Integrated” function is active (Compare also MD 36901: SAFE_FUNCTION_ENABLE), use of the load intermediate gear is not permitted and is answered with Alarm 26014 “Channel %1 Axis %2 Hardware limit switch %3”. In this way, load intermediate gear is deactivated and the default values 1:1 set.		
Related to	MD 31066: DRIVE_AX_RATIO2_NUMERA (numerator intermediate gear) MD 31044: ENC_IS_DIRECT2 (encoder at intermediate gear) MD 31060: DRIVE_AX_RATIO_NUMERA (numerator load gearbox) MD 31050: DRIVE_AX_RATIO_DENOM (denominator load gearbox) MD 36901: SAFE_FUNKTION_ENABLE) (enable safe function)		
References	/FB1/, Velocities, Setpoint/Actual Value Systems, Closed-Loop Control		

© Siemens AG, 2003. All rights reserved

31066 MD number	DRIVE_AX_RATIO2_NUMERA Numerator intermediate gear		
Default setting: 1.0	Minimum input limit: –2147000000	Maximum input limit: 2147000000	
Changes effective after NewConfig	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.4		
Significance:	Numerator intermediate gear In conjunction with machine data MD 31064 DRIVE_AX_RATIO2_NUMERA, defines a intermediate gear that acts as a multiplier on the motor&load gearbox. The motor/load gearbox is described by the machine data MD 31060: DRIVE_AX_RATIO_NUMERA (numerator load gearbox) MD 31050: DRIVE_AX_RATIO_DENOM (denominator load gearbox)		
Related to	MD 31064: DRIVE_AX_RATIO2_DENOM (denominator intermediate gear) MD 31044: ENC_IS_DIRECT2 (encoder at intermediate gear) MD 31060: DRIVE_AX_RATIO_NUMERA (numerator load gearbox) MD 31050: DRIVE_AX_RATIO_DENOM (denominator load gearbox) MD 36901: SAFE_FUNKTION_ENABLE) (enable safe function)		
References	/FB1/, Velocities, Setpoint/Actual Value Systems, Closed-Loop Control		

4.3.3 Gear step change

35010 MD number	GEAR_STEP_CHANGE_ENABLE Gear stage change options extendable for fixed position (SW 5.3 and higher)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	0: Spindle drive is mounted to the spindle directly (1:1) or with a fixed transmission ratio. It is not possible to change the gear stage with M40 to M45. possible. 1: Spindle drive is mounted to the spindle via gears with selectable gear stages. The gear can have up to 5 gear stages, which can be selected using M40, M41 to M45. 2: As 1, but with gear stage change at programmed spindle position (SW 5.3 and		
Related to	MD 35110: GEAR_STEP_MAX_VELO (max. speed for automatic Gear stage change) MD 35120: GEAR_STEP_MIN_VELO (min. speed for automatic Gear stage change) GEAR_STEP_MAX_VELO and GEAR_STEP_MIN_VELO must cover the whole speed range.		

35012 MD number	GEAR_STEP_CHANGE_POSITION Gear stage change position		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: degrees	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	Gear stage change position The value range is limited internally to the programmed modulo range.		
Related to	MD 30330: MODULO_RANGE (Extent of modulo range) MD 35210: GEAR_STEP_POSCTRL_ACCEL (Acceleration in position control mode) MD 35300: SPIN_POSCTRL_VELO (Position control activation speed)		

© Siemens AG, 2003. All rights reserved

35110 MD number	**GEAR_STEP_MAX_VELO[n]** Maximum speed for gear stage change [gear stage number]: 0...5 (index 0 is irrelevant for spindles)		
Default setting: 500, 500, 1000, 2000, 4000, 8000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	GEAR_STEP_MAX_VELO defines the maximum speed of the gear stage for automatic gear stage change (M40). The gear stages must be defined by GEAR_STEP_MAX_VELO and MD 35120: GEAR_STEP_MIN_VELO_LIMIT in a way that avoids gaps in the programmable spindle speed range between the gear stages. **Incorrect** GEAR_STEP_MAX_VELO [gear stage1] =1000 GEAR_STEP_MIN_VELO [gear stage2] =1200 **Correct** GEAR_STEP_MAX_VELO [gear stage1] =1000 GEAR_STEP_MIN_VELO [gear stage2] =950		
Related to	MD 35010: GEAR_STEP_CHANGE_ENABLE (gear stage change is possible) MD 35120: GEAR_STEP_MIN_VELO (minimum speed for gear stage change) MD 35140: GEAR_STEP_MIN_VELO_LIMIT (minimum speed for gear stage) MD 35130: GEAR_STEP_MAX_VELO_LIMIT (maximum speed for gear stage)		

35120 MD number	**GEAR_STEP_MIN_VELO[n]** Minimum speed for gear stage change [gear stage number]: 0...5		
Default setting: 50, 50, 400, 800, 1500, 3000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	GEAR_STEP_MIN_VELO defines the minimum speed for gear stage for automatic gear stage change (M40). Refer to MD 35120: GEAR_STEP_MAX_VELO for more information.		
Related to	MD 35110: GEAR_STEP_MAX_VELO (maximum speed for gear stage change) MD 35010: GEAR_STEP_CHANGE_ENABLE (gear stage change is possible) MD 35140: GEAR_STEP_MIN_VELO_LIMIT (minimum speed for gear stage) MD 35130: GEAR_STEP_MAX_VELO_LIMIT (maximum speed for gear stage)		

© Siemens AG, 2003. All rights reserved

35130 MD number	**GEAR_STEP_MAX_VELO_LIMIT[n]** Maximum speed for gear stage [gear stage number]; 0 ...5		
Default setting: 500, 500, 1000, 2000, 4000, 8000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	GEAR_STEP_MAX_VELO_LIMIT defines for the gear stage. This speed can never be exceeded in the currently engaged gear stage.		
Special cases, errors, ...	• If position control is activated, the value is limited to 90% (control margin) • If an S value exceeding the maximum speed of the engaged gear stage is programmed, the speed is limited to the maximum speed of the gear stage (with gear stage selection M41 to M45) and the “Programmed speed too high” interface signal is enabled. • If an S value exceeding the maximum speed for the gear change is programmed, a new gear stage is selected (with automaticc selection M40). • If an S value exceeding the maximum speed of the highest gear stage is programmed, the speed is limited to the maximum speed of the gear stage (with automatic gear stage selection M40). • If an S value is programmed for which no suitable gear stage exists, a gear change is not triggered.		
Related to	MD 35010: GEAR_STEP_CHANGE_ENABLE (gear stage change is possible) MD 35110: GEAR_STEP_MAX_VELO (maximum speed for gear stage change) MD 35120: GEAR_STEP_MIN_VELO (minimum speed for gear stage change) MD 35140: GEAR_STEP_MIN_VELO_LIMIT (minimum speed for gear stage)		

35140 MD number	**GEAR_STEP_MIN_VELO_LIMIT[n]** Minimum speed for gear stage [gear stage number]: 0...5		
Default setting: 5, 5, 10, 20, 40, 80	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	GEAR_STEP_MIN_VELO_LIMIT defined the minimum speed for the gear stage. The speed cannot drop to below this value, even if a very low S value is programmed. The speed can only drop to below this minimum value as a result of the signals/commands/ states listed in Subsection 2.5.4. “Minimum/maximum speed for the gear stage”.		
MD irrelevant for ...	• Spindle mode oscillation mode • Spindle mode positioning mode, axis mode		
Application example(s)	Smooth operation of the motor is not assured below the minimum speed.		
Related to	MD 35010: GEAR_STEP_CHANGE_ENABLE (gear stage change is possible) MD 35110: GEAR_STEP_MAX_VELO (maximum speed for gear stage change) MD 35120: GEAR_STEP_MIN_VELO (minimum speed for gear stage change) MD 35130: GEAR_STEP_MAX_VELO_LIMIT (maximum speed for gear stage)		

© Siemens AG, 2003. All rights reserved

35200 MD number	**GEAR_STEP_SPEEDCTRL_ACCEL[n]** Acceleration in speed control mode [gear stage number]: 0...5		
Default setting: 30, 30, 25, 20, 15, 10	Minimum input limit: 2	Maximum input limit: ***	
Changes effective after NewConfig	Protection level: 2/7	Unit: rev./s^2	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	If the spindle is in **speed control mode**, the acceleration is entered in GEAR_STEP_SPEEDCTRL_ACCEL. The spindle is in speed control mode with the function SPCOF.		
Special cases, errors, ...	Thus the acceleration in speed control mode (GEAR_STEP_SPEEDCTRL_ ACCEL) can be set so that the current limit is reached.		
Related to	MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode) MD 35220: ACCEL_REDUCTION_SPEED_POINT (acceleration reduced by speed limit)		

35210 MD number	**GEAR_STEP_POSCTRL_ACCEL[n]** Acceleration in position control mode [gear stage number]: 0...5		
Default setting: 30, 30, 25, 20, 15,10	Minimum input limit: 2	Maximum input limit: ***	
Changes effective after NewConfig	Protection level: 2/7	Unit: rev./s^2	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The acceleration in position control mode must be set so that the current limit is not reached.		
Related to	MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL MD 35220: ACCEL_REDUCTION_SPEED_POINT		

© Siemens AG, 2003. All rights reserved

4.3.4 Assignments/settings for spindle and spindle speed

35000 MD number	**SPIND_ASSIGN_TO_MACHAX** Assignment between spindle no. and machine axis		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 5 840D/810D 2 FMNC	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	In order that the NCK/PLC user interface is the same for all axes and spindles, spindles are always mapped onto machine axes internally in the control. SPIND_ASSIGN_TO_MACHAX is set to define the spindle no. used for the machine axis.		
Application example(s)	An example of a milling machine with 3 machine axes (X, Y and Z) and a spindle: SPIND_ASSIGN_TO_MACHAX [AX1] = 0 ——> X SPIND_ASSIGN_TO_MACHAX [AX2] = 0 ——> Y SPIND_ASSIGN_TO_MACHAX [AX3] = 0 ——> Z SPIND_ASSIGN_TO_MACHAX [AX4] = 1 ——> Spindle number 1 is the 4th machine axis		
Related to	MD 30300: IS_ROT_AX (rotary axis/spindle) MD 30310: ROT_IS_MODULO (modulo conversion for rotary axis/spindle) This machine data must be set; otherwise, the alarms 4210 “Rotary axis declaration missing” and 4215 “Modulo axis declaration missing” are output.		

35020 MD number	**SPIND_DEFAULT_MODE** Basic spindle setting	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3
Changes effective after RESET	Protection level: 2/7	Unit: –
Data type: BYTE	Applies from SW 2	
Significance:	SPIND_DEFAULT_MODE activates the operating mode of the spindle at the time specified in MD 35030: SPIND_DEFAULT_ACT_MASK. The appropriate spindle operating modes can be selected with the following values: 0 Speed mode, position control deselected 1 Speed mode, position control activated 2 Positioning mode 3 Axis mode	
Related to	MD 35030: SPIND_DEFAULT_ACT_MASK (activate basic spindle position setting)	

35030 MD number	**SPIND_DEFAULT_ACT_MASK** Activate initial spindle setting	
Default setting: 0x00	Minimum input limit: 0	Maximum input limit: 0x03
Changes effective after RESET	Protection level: 2/7	Unit: HEX
Data type: BYTE	Applies from SW 2	
Significance:	SPIND_DEFAULT_ACT_MASK specifies the time that operating mode defined in MD 35020: SPIND_DEFAULT_MODE becomes effective. The basic spindle setting can be assigned the following values at the following points in time: 0 POWER ON 1 POWER ON and NC program start 2 POWER ON and RESET (M2/M30)	
Special cases, errors, ...	If MD 35040: SPIND_ACTIVE_AFTER_RESET = 1, the following supplementary conditions are applicable: – SPIND_DEFAULT_ACT_MASK should be set to 0 – If this is not possible, the spindle must be at standstill prior to activation.	
Related to	MD 35020: SPIND_DEFAULT_MODE (basic spindle setting) MD 35040: SPIND_ACTIVE_AFTER_RESET (spindle active via reset)	

© Siemens AG, 2003. All rights reserved

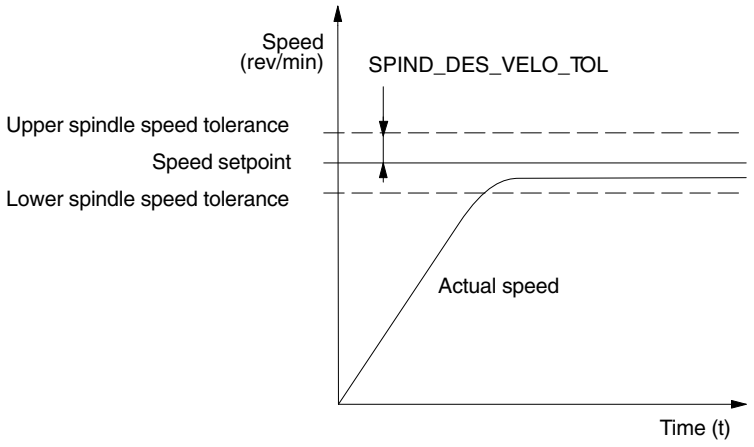
35035 MD number	**SPIND_FUNCTION_MASK** Setting spindle-specific functions		
Default setting: 0x110	Minimum input limit: 0	Maximum input limit: 0x137	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies from SW 6.1 bits 0 to 2, SW 6.1 and higher) bits 3 –11, SW 6.2 and higher)		
Significance:	SPIND_FUNCTION_MASK can be used to set spindle-specific functions. The machine data SPIND_FUNCTION_MASK is bit-coded; the following bits are occupied: Bit 0 = 1: Gear stage changes are suppressed when the DryRun function is active for block programming (M40, M41 to M45), and programming via FC18 and synchronized actions. Bit 1 = 1: Gear stage changes are suppressed when the Program Test function is active for block programming (M40, M41 to M45), and programming via FC18 and synchronized actions. Bit 2=1: Gear stage change for programmed gear stage is performed subsequently on REPOS after deselection of functions DryRun or Program Test. Bit 3: reserved Bit 4=1: The programmed speed is accepted into the SD 43200: SA_SPIND_S SA_SPIND_S (incl. speed settings via FC18 and synchronized actions). Programmed S values that are no programmed speed values are not written to the setting data SD 43200. This pertains, e.g. to the S value at constant cutting rate (G96, G961), S value at revolutionrelated dwell time (G4). Bit 5=1: The content of SD 43200: SA_SPIND_S acts as the set speed in JOG mode. If the content is zero, then other JOG speed settings become active (see SD 41200: JOG_SPIND_SET_VELO). Bit 6: reserved Bit 7: reserved Bit 8=1: The programmed cutting rate is accepted into the SD 43202: SA_SPIND_CONSTCUT_S (incl. settings via FC18 and synchronized actions). Programmed S values that are no programmed cutting rate values are not written to SD 43202.This pertains, e.g. to the S value at constant cutting rate (G96, G961) and to the S value at revolution-related dwell time (G4). Bit 9: reserved Bit 10: reserved Bit 11: reserved		
Special cases, errors, ...	Supplementary Conditions: Only programming SD 43200 and SD 43202 will not result in a change of the speed or cutting rate.		
Related to	MD 35020: SPIND_DEFAULT_MODE (basic spindle setting) MD 35040: SPIND_ACTIVE_AFTER_RESET (spindle active via reset) IS “interrupted” DB21, ... DBX35.6 or “Reset” DB21, ... DBX35.7 IS “aborted” DB21, ... DBX35.4		

© Siemens AG, 2003. All rights reserved

35040 MD number	**SPIND_ACTIVE_AFTER_RESET** Spindle active after reset		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	SPIND_ACTIVE_AFTER_RESET sets how the spindle responds to a reset (DB21, ... DBX7.7) and end of program (M2, M30). It is only active in the spindle control mode. SPIND_ACTIVE_AFTER_RESET = 0: Control mode: – Spindle stops, applies only for M2/M30 and MCP reset – Program is aborted, applies for M2/M30 Oscillation mode: – Alarm 10640 “Spindle cannot stop during gear change“ – Oscillation is not aborted – The axes are stopped – The program is aborted after a gear change or spindle reset and the alarm is cleared Positioning mode: – is stopped Axis mode: – is stopped ——————————————————————————————— SPIND_ACTIVE_AFTER_RESET = 1: Control mode: – The spindle does not stop – The program is aborted Oscillation mode: – Alarm 10640 “Spindle cannot stop during gear change“ – Oscillation is not aborted – The axes are stopped – The program is aborted after a gear change or spindle reset, the alarm is cleared and the spindle continues to rotate with the programmed M and S value. Positioning mode: – is stopped Axis mode: – is stopped ——————————————————————————————— SPIND_ACTIVE_AFTER_RESET = 2: The spindle stops on a RESET or at the end of the program unless the parts program is terminated using the M function configured in MD 10714: NO_FCT_EOP (default M32). The spindle can be stopped using the “Spindle reset/Delete distance-to-go” interface signal. ——————————————————————————————— The IS “Spindle Reset” (DB31, ... DBX2.2) is always active, independently of SPIND_ACTIVE_AFTER_RESET.		
MD irrelevant for ...	All spindle modes except control mode		
Related to	IS “Reset” (DB21, ... DBX7.7) IS “Spindle Reset” (DB31, ... DBX2.2)		

35100 MD number	**SPIND_VELO_LIMIT** Maximum spindle speed		
Default setting: 10 000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2/7	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	SPIND_VELO_LIMIT defines the maximum spindle speed, which the spindle (the spindle chuck with the workpiece or the tool) must not exceed. The NCK limits an excessive spindle speed setpoint to this value. If the maximum spindle actual speed is exceeded, even allowing for the spindle speed tolerance (MD 35150: SPIND_DES_VELO_TOL), there is a fault with the drive and the “Speed limit exceeded” interface signal (DB31, ... DBX83.0) is set. Alarm 22050 “maximum speed reached” is also output and all of the axes and spindles on the channel brought to a halt (provided the encoder is still functioning correctly).		
Related to	MD 35150: SPIND_DES_VELO_TOL (spindle speed tolerance) IS “Speed limit exceeded” (DB31, ... DBX83.0) Alarm 22050 “maximum speed reached”		

© Siemens AG, 2003. All rights reserved

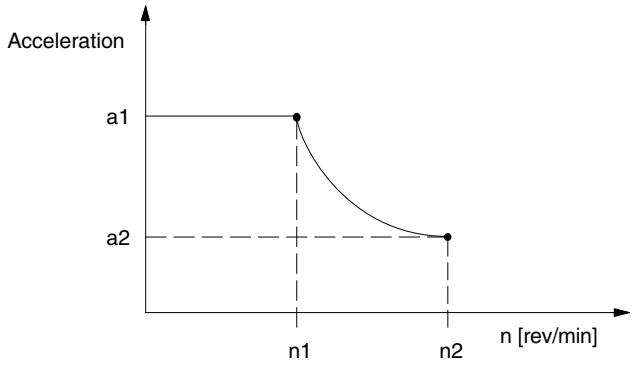
35150 MD number	**SPIND_DES_VELO_TOL** Spindle speed tolerance		
Default setting: 0,1 0,1 = 10%	Minimum input limit: 0	Maximum input limit: 1.0 1 = 100%	
Changes effective after RESET	Protection level: 2/7	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1 (modified in SW 6.4 and higher)		
Significance:	In spindle mode, the set speed (programmed speed x spindle offset taking limits into account) is compared to the actual speed. • If the actual speed deviates from the set speed by more that the SPIND_DES_VELO_TOL, the “spindle in setpoint range” interface signal (DB31, ... DBX83.5) is set to zero. • If the actual speed exceeds the maximum spindle speed (MD 35100: SPIND_VELO_LIMIT) by more than SPIND_DES_VERLO_TOL SPIND_VELO_LIMIT), IS “Speed limit exceeded” (DB31, ... DBX83.0) is set and Alarm 22050 “maximum speed reached” issued. All axes and spindles on the channel are brought to a standstill. The spindle set speed is obtained from the programmed speed and the current limits. A limitation or increase in the programmed speed is indicated by the axial IS “Set speed limited” (DB31, ... DBX83.1) or IS “Setpoint speed increased” (DB31, ... DBX83.2) and does not prevent reaching the speed tolerance range. VDI interface: If the actual spindle speed lies within the tolerance window, then the axial IS “Spindle in setpoint range” (DB31, ... DBX83.5) is set to “1”. This information corresponds to the setting in MD 35500: SPIND_ON_SPEED_AT_IPO_START or MD 35510: SPIND_STOPPED_SPEED_AT_IPO_START and can be used to delay the path interpolation. Speed change: The path control only takes place at the start of the traverse block and only if a speed change has been programmed. If the speed tolerance range is violated, e.g. due to an overload, the path control is not brought to a standstill.		
MD irrelevant for ...	Spindle oscillation mode Spindle mode positioning mode		
Fig. 4-1			
Application example(s)	MD 35150: SPIND_DES_VELO_TOL = 0,1. The spindle actual speed must not deviate more than +/– 10% compared to the set speed. Used for all axes and spindles to perform checking on a falling edge of the cam signal in the second phase of reference ponit approach when searching for the yero marker. For further information, refer to **Reference**: R1, “Reference Point Approach”		
Special cases, errors, ...	If the speed tolerance is set to zero, the axial VDI interface signal IS “Spindle in set range” (DB31, ... DBX83.5) is permanently set to “1” and no path control takes place.		
Related to	Further machine data: MD 35500: SPIND_ON_SPEED_AT_IPO_START, MD 35510: SPIND_STOPPED_AT_IPO_START, Alarm 22050 “Maximum speed reached” IS “Spindle in setpoint range” (DB31, ... DBX83.5) IS “Speed limit exceeded” (DB31, ... DBX83.0)		

© Siemens AG, 2003. All rights reserved

35160 MD number	**SPIND_EXTERN_VELO_LIMIT** Spindle speed limitation via PLC		
Default setting: 1000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: rev/min	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	A limit for the spindle speed is entered in SPIND_EXTERN_VELO_UNIT, which is taken into account when the IS “Velocity/speed limitation” (DB31, ... DBX3.6) is enabled. If the spindle speed is too high, the NCK will limit the it to this value.		

35220 MD number	**ACCEL_REDUCTION_SPEED_POINT** Speed limit for reduced acceleration		
Default setting: 1.0	Minimum input limit: 0.0	Maximum input limit: 1.0	
Changes effective after RESET	Protection level: 2/7	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The torque of a spindle is constant in the lower speed range and becomes smaller above a defined speed (upper speed range). The upper speed range with reducing torque starts at a speed which must be entered in MD: ACCEL_REDUCTION_SPEED_POINT.		
MD irrelevant for ...	Spindles with constant torque across the entire speed range		
Related to	MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode) MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL (acceleration in speed control mode) MD 35230: ACCEL_REDUCTION_FACTOR (reduced acceleration)		

© Siemens AG, 2003. All rights reserved

35230 MD number	**ACCEL_REDUCTION_FACTOR** Reduced acceleration		
Default setting: 0	Minimum input limit: 0.0	Maximum input limit: 0.95	
Changes effective after RESET	Protection level: 2/7	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The torque of a spindle is constant in the lower speed range and becomes smaller above a defined speed (upper speed range). The upper speed range with reducing torque starts at a speed defined in MD 35220: ACCEL_REDUCTION_SPEED_POINT (speed limit for reduced acceleration). ACCEL_REDUCTION_SPEED_POINT (speed limit for reduced acceleration). The value defined in ACCEL_REDUCTION_FACTOR is the maximum permissible acceleration for the maximum speed of the gear stage (MD 35130: GEAR_STEP_MAX_VELO_LIMIT). Caution: The reduced acceleration is entered as a percentage (%). Depending on whether the spindle is in position control mode or in speed control mode, the reduced acceleration relates either to the acceleration in position control mode (MD 35210: GEAR_STEP_POSCTRL_ACCEL) or to the acceleration of the active gear stage in the speed control mode (MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL).		
MD irrelevant for ...	Spindles with constant torque across the entire speed range		
Fig. 4-2	 a1 ... MD: GEAR_STEP_POSCTRL_ACCEL [degrees/s] a2 ... MD: ACCEL_REDUCTION_FACTOR [%] n1 ... MD: ACCEL_REDUCTION_SPEED_POINT[%] n2 ... MD: GEAR_STEP_MAX_VELO_LIMIT [rev/min]		
Related to	MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode) MD 35200: GEAR_STEP_SPEEDCTRL_ACCEL (acceleration in speed control mode) MD 35220: ACCEL_REDUCTION_SPEED_POINT (acceleration reduced by speed limit) MD 35130: GEAR_STEP_MAX_VELO_LIMIT (maximum speed for gear stage)		

© Siemens AG, 2003. All rights reserved

35300 MD number	**SPIND_POSCTRL_VELO** Position control activation speed		
Default setting: 500	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	When positioning a spindle that is not in position control mode, the position control is not activated until the spindle has reached the speed defined in MD: SPIND_POSCTRL_VELO. The speed can be changed with FA[Sn] from the parts program. Please refer to Subsection 2.1.3 Spindle “positioning mode” for a description of spindle operating characteristics under various supplementary conditions (positioning from rotation, positioning from standstill).		
Related to	MD 35350: SPIND_POSITIONING_DIR (direction of rotation during positioning from standstill), if no synchronization is available.		

35310 MD number	**SPIND_POSIT_DELAY_TIME** Positioning delay time		
Default setting: 0.0, 0.05, 0.1, 0.4, 0.8	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 6.1		
Significance:	The end of the positioning operation (exact stop fine) is followed by a delay corresponding to the set time. The position appropriate to the currently engaged gear stage is selected. The delay time is activated with ● a gear stage change at a defined spindle position. After the position configured in MD 35011: GEAR_STEP_CHANGE_POSTION is reached, a timer corresponding to the setting here is started. When this timer has run down, the closedloop position control is deactivated for an active, direct measuring system and IS “Change gear” (DB31, ... DBX82.3) and IS “Set gear step A to C” (DB31, ... DBX82.0–82.2). ● Block search on output of an accumulated positioning block (SPOS, SPOSA, M19).		
Related to	MD 35012: GEAR_STEP_CHANGE_POSITION (gear stage change position).		

35350 MD number	**SPIND_POSITIONING_DIR** Positioning direction of rotation for nonsynchronized spindle		
Default setting: 3	Minimum input limit: 3	Maximum input limit: 4	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	When SPOS or SPOSA is programmed, the spindle is switched to position control mode and accelerates with the acceleration defined in MD 35210: GEAR_STEP_POSCTRL_ACCEL (acceleration in position control mode) if the spindle is not synchronized. The direction of rotation is defined by MD 35350: SPIND_POSITIONING_DIR. SPIND_POSITIONING_DIR = 3 ——> Clockwise direction SPIND_POSITIONING_DIR = 4 ——> Counterclockwise direction		
Related to	MD 35300: SPIND_POSCTRL_VELO (position control activation speed)		

© Siemens AG, 2003. All rights reserved

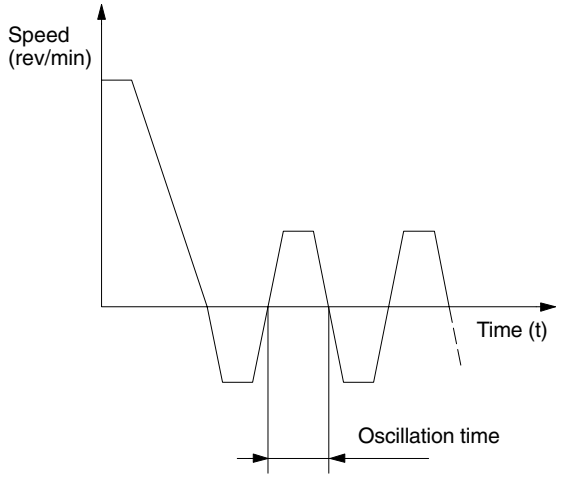
35400 MD number	**SPIND_OSCILL_DES_VELO** Oscillation speed		
Default setting: 500	Minimum input limit: 0	Maximum input limit: value in MD: GEAR_STEP_MAX_VELO_LIMIT	
Changes effective after NewConfig	Protection level: 2/7	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	During oscillation, the “oscillation speed” interface signal (DB31, ... DBX18.5) is used to select a motor speed for the spindle motor. This motor speed is defined in MD: SPIND_OSCILL_DES_VELO. The motor speed defined in this MD is independent of the current gear stage. In the AUTOMATIC and MDA displays, the oscillation speed is displayed in the “spindle setpoint” window until the gear is changed.		
MD irrelevant for ...	All spindle modes except oscillation mode		
Application example(s)	The to and fro motion of the spindle motor makes it easier to engage a new gear stage because the teeth on the gear wheels can mesh with each other better.		
Special cases, errors, ...	The oscillation acceleration defined in MD 35410: SPIND_OSCILL_ACCEL is valid for the oscillation speed defined in this MD.		
Related to	MD 35410: SPIND_OSCILL_ACCEL (acceleration during oscillation) IS “Oscillation via PLC” (DB31, ... DBX18.4) IS “Oscillation speed” (DB31, ... DBX18.5)		

35410 MD number	**SPIND_OSCILL_ACCEL** Oscillation acceleration		
Default setting: 16	Minimum input limit: 2	Maximum input limit: ***	
Changes effective after NewConfig	Protection level: 2/7	Unit: rev./s^2	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The acceleration specified is only effective for the output of the oscillation speed (MD 35400: SPIND_OSCILL_DES_VELO) to the spindle motor. The oscillation speed is selected using the “oscillation speed” interface signal.		
MD irrelevant for ...	All spindle modes except oscillation mode		
Related to	MD 35400: SPIND_OSCILL_DES_VELO (Oscillation speed) IS “Oscillation speed” (DB31, ... DBX18.5) IS “Oscillation via PLC” (DB31, ... DBX18.4)		

35430 MD number	**SPIND_OSCILL_START_DIR** Oscillation start direction		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 4	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	With the “Oscillation speed” interface signal, the spindle motor accelerates to the speed specified in MD 35400: SPIND_OSCILL_DES_VELO. **The start direction** is defined by SPIND_OSCILL_START_DIR if “oscillation via PLC” is not enabled. SPIND_OSCILL_START_DIR = 0 ——> Start direction is last direction of rotation SPIND_OSCILL_START_DIR = 1 ——> Start direction counter to last direction of rotation SPIND_OSCILL_START_DIR = 2 ——> Start direction counter to last direction of rotation SPIND_OSCILL_START_DIR = 3 ——> Start direction is M3 SPIND_OSCILL_START_DIR = 4 ——> Start direction is M4		
MD irrelevant for ...	All spindle modes except oscillation mode		
Related to	MD 35400: SPIND_OSCILL_DES_VELO (Oscillation speed) IS “Oscillation speed” (DB31, ... DBX18.5) IS “Oscillation via PLC” (DB31, ... DBX18.4)		

© Siemens AG, 2003. All rights reserved

35440 MD number	**SPIND_OSCILL_TIME_CW** Oscillation time for M3 direction		
Default setting: 1.0	Minimum input limit: 0 0 means a time of **one** interpolation cycle (MD 10070: IPO_SYSCLOCK_TIME_RATIO)	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The oscillation time defined here is active in the M3 direction (see Fig. 4-3 for MD 35450: SPIND_OSCILL_TIME_CCW).		
MD irrelevant for ...	• All spindle modes except oscillation mode • Oscillation via PLC (“Oscillation via PLC” interface signal (DB31, ... DBX18.4) enabled)		
Related to	MD 35450: SPIND_OSCILL_TIME_CCW (oscillation time for M4 direction) MD 10070: IPO_SYSCLOCK_TIME_RATIO (interpolator clock) IS “Oscillation speed” (DB31, ... DBX18.5) IS “Oscillation via PLC” (DB31, ... DBX18.4)		

35450 MD number	**SPIND_OSCILL_TIME_CCW** Oscillation time for M4 direction		
Default setting: 0,5	Minimum input limit: 0 0 means a time of **one** interpolation cycle (MD 10070: IPO_SYSCLOCK_TIME_RATIO)	Maximum input limit: plus	
Changes effective after NewConfig	Protection level: 2/7	Unit: s	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The oscillation time defined here is active in the M4 direction (see Fig. below).		
MD irrelevant for ...	• All spindle modes except oscillation mode • Oscillation via PLC (“Oscillation via PLC” interface signal (DB31, ... DBX18.4) enabled)		
Fig. 4-3			
Related to	MD 35440: SPIND_OSCILL_TIME_CW (oscillation time for M3 direction) MD 10070: IPO_SYSCLOCK_TIME_RATIO (interpolator clock) IS “Oscillation speed” (DB31, ... DBX18.5) IS “Oscillation via PLC” (DB31, ... DBX18.4)		

© Siemens AG, 2003. All rights reserved

35500 MD number	**SPIND_ON_SPEED_AT_IPO_START** Feed enable with spindle in setpoint range		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Data type: BYTE	Applies from SW 4.2		
Significance:	Bit 0: The path interpolation is not affected Bit 1: The feed interpolation is not enabled (positioning axes continue traversing) until the spindle has reached the specified speed (tolerance range is defined via MD 35150).		
	SW 4.2 and higher: Byte = 0: The path interpolation is not affected Byte = 1: The feed interpolation is not enabled (positioning axes continue traversing) until the spindle has reached the specified speed (tolerance range is defined via MD 35150). If a measuring system is active, then the actual speed is monitored, otherwise the setpoint speed. Traveling axes in continuous-path mode (G64) are not stopped. Byte = 2: In addition to 1., In continuous-path mode (G64) when a new machining block is loaded after rapid traverse (G0), the path velocity is decelerated if the spindle is not within the speed setpoint range. Byte = 3: No longer available in SW 5.3 and higher.		
Application example(s)	When the spindle is in the acceleration phase (programmed setpoint speed not yet reached), the path feed must generally be disabled as follows: • IS “Spindle in setpoint range” (DB31, ... DBX83.5) is evaluated and IS “Feed disable” (DB21, ... DBX6.0) enabled The positioning axes are also stopped. • MD 35500 is set to 1 (see above) • MD 35500 is set to 2 (see above) • In blocks with G0 the influencing factor set by the MD is not active. • The path feed specification activated by MD 35500 take effect according to the: – Specification of speed – Specification of direction of rotation – NC Start It ceases to act after the speed tolerance window has been reached at least once.		
Related to	MD 35150: SPIND_DES_VELO_TOL (spindle speed tolerance) IS “Spindle in setpoint range” (DB31, ... DBX83.5)		

© Siemens AG, 2003. All rights reserved

35510 MD number	**SPIND_STOPPED_AT_IPO_START** Feed enable with stationary spindle		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after RESET	Protection level: 2/7	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	When a spindle is stopped (M5), the path feed is disabled (positioning axes continue traversing) if the MD SPIND_STOPPED_AT_IPO_START is enabled and the spindle is in the control mode. If the spindle has come to a halt (IS “Axis/spindle stopped” (DB31, ... DBX61.4) set), path feed is enabled.		
Application example(s)	With MD 35500: SPIND_ON_SPEED_AT_IPO_START and SPIND_STOPPED_AT_IPO_START can be used to control the path feed according to the actual spindle speed (control mode): ● If the spindle is in the acceleration phase (programmed setpoint not yet reached), the path feed is disabled. ● If the actual speed deviates from the speed setpoint by less than the spindle speed tolerance (MD 35150: SPIND_DES_VELO_TOL), the path feed is enabled. ● If the spindle is in the braking phase, the path feed is disabled. ● If the spindle has been reported as stationary (“axis/spindle stationary” interface signal DB31, ... DBX61.4), the feed is enabled. ● In blocks with G0, the modification is not active.		
Related to	MD 35500: SPIND_ON_SPEED_AT_IPO_START (feed enable for spindle in setpoint range)		

© Siemens AG, 2003. All rights reserved

4.4 Spindle-specific setting data

42800 SD number	**SPIND_ASSIGN_TAB** Spindle number converter		
Default setting: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18}, {0, 1, 2, ..., }	Minimum input limit: 0	Maximum input limit: MD_SPIND_ASSIGN_TAB_LENGTH	
Changes effective immediately	Protection level: 7 / 7	Unit:	
Data type: BYTE	Applies from SW 4.3		
Significance:	The spindle converter sets the programmed (= logical) spindle number to the physical (= internal, configured) spindle number. The index of the setting data (SD) corresponds to the programmed spindle number or the programmed address extension. The SD contains the physical spindle which actually exists.		
Application example(s)	Spindles 1, ..., 3 are always programmed in the channel. The spindles which are actually available in the channel are entered by the MMC, PLC or parts program in the assignment table before the program is started. SD 42800: SPIND_ASSIGN_TAB[1]=15 SD 42800: SPIND_ASSIGN_TAB[2]=18 SD 42800: SPIND_ASSIGN_TAB[3]=3 SD 42800: SPIND_ASSIGN_TAB[4]=0 M1=3 S1=1000 ; phys. spindle 15 rotates at 1000 rpm in pos. direction. SPOS[2]=0 ; phys. spindle 18 positioned to zero degrees. M3=5 ; Stop for phys. spindle 3. R1=$AA_S[4] ; Alarm 16105 reported because phys. spindle cannot be determined.		
Special cases, errors, ...	Notes: • The zero index (SPIND_ASSIGN_TAB[0]) is only used to display the master spindle selected in the channel and must not be overwritten. • Changes to the spindle converter take effect immediately. It is therefore **not** advisable to change the spindle converter from MMC or PLC for a spindle used in the parts program while the program is running. • After “delete SRAM”, the numbers of the logical and physical spindles are identical.		
Related to	MD 10710: PROG_SD_RESET_SAVE_TAB (SW 5.3 and higher)		

© Siemens AG, 2003. All rights reserved

43200 SD number	**SPIND_S** Specification of the spindle speed		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective immediately	Protection level: 7 / 7	Unit: rpm	
Data type: DOUBLE	Applies from SW 6		
Significance:	The value SD 43200: SPIND_S is evaluated by the VDI IS “Spindle start CW” DB31, ... DBX30.1 and “Spindle start CCW” DB31, ... DBX30.2.		
SD irrelevant for ...	Only programming SD 43200 will not result in a speed change. If SD 43200 is programmed with negative values, an alarm is signaled. SD 43200: SPIND_S will not be read if feedrate type G96 or G961 is active for the master spindle of the channel handling the spindle (constant cutting rate). In this case, the interface signal (IS) “Const. Current control loop active (DB31, ... DBX84.0 = 1 is set at the VDI interface.		
Application example(s)	Example: SD 43200: SA_SPIND_S[S1] = 600 If the positive edge is detected at one of the above mentioned VDI signal, spindle 1 is started at a speed of 600 rpm.		
Special cases, errors, ...	Speed and direction of rotation cannot be changed by setting the DBB30 input signals while the program is running. The speed and direction of rotation specified via the DBB30 spindle interface can always be modified by programmed spindle values from the parts program, ASUB, FC18 and/or synchronized actions.		
Related to	SD 43202: SPIND_CONSTCUT_S (setting of the constant cutting rate for the master spindle) MD 35035: SPIND_FUNCTION_MASK (setting of spindlespecific functions)		

43202 SD number	**SPIND_CONSTCUT_S** Specification of the constant cutting rate for the master spindle		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective immediately	Protection level: 7 / 7	Unit: m/min (feet/min)	
Data type: DOUBLE	Applies from SW 6		
Significance:	The value SD 43202: SPIND_CONSTCUT is evaluated by the VDI IS “Spindle start CW” DB31, ... DBX30.1 and “Spindle start CCW” DB31, ... DBX30.2. If SD 43202 is programmed from the parts program, the value is interpreted as follows, depending on the 12th G group: ● If G710 is active metric in [m/min]. ● If G700 is set in inch as [feet/min]. ● With G70 and G71 and when writing from an external source (MMC), the setting in MD 1240: SCALING_SYSTEM_IS_METRIC will decide on how the programmed value is interpreted /R3/. To make sure that a constant cutting rate is active, the following is required: ● The spindle concerned must be the master spindle in the channel handling the spindle. ● The feedrate type in the channel handling the spindle must be G96 or G961 active. The above mentioned requirements are fulfilled if the IS “Const. Current control loop active” (DB31, ... DBX84.0 = 1 is set at the VDI interface.		
SD irrelevant for ...	Only programming SD 43202 will not result in a change of the constant cutting rate. If SD 43202 is programmed with negative values, an alarm is signaled. SD 43202: SPIND_CONSTCUT_S will not be read if the appropriate prerequisites are not fulfilled. In this case, the SD 43200 SPIND_S is read.		
Special cases, errors, ...	The programmed cutting rate value can be determined in the parts program and in synchronized actions by reading the system variables $P_CONSTCUT_S and $AC_CONSTCUT_S.		
Related to	SD 43200: SPIND_S (definition of the spindle speed) MD 35035: SPIND_FUNCTION_MASK (setting of spindle-specific functions)		

© Siemens AG, 2003. All rights reserved

43210 SD number	**SPIND_MIN_VELO_G25** Progr. spindle speed limitation G25)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective immediately	Protection level: MMCMD 9220	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The spindle speed is not permitted to exceed the maximum spindle speed limit entered in SPIND_MIN_VELO_G25. The NCK limits an excessively small spindle speed setpoint to this value. References The spindle speed can only fall below the minimum limit through: ● Spindle override 0% ● M5 ● S0 ● IS “Spindle stop” (DB31, ... DBX8.3) ● IS “Remove servo enable” (DB31, ... DBX2.1) ● IS “Reset” (DB21, ... DBX35.7) ● IS “Spindle Reset” (DB31, ... DBX2.2) ● IS “Oscillation speed” (DB31, ... DBX18.5) ● Delete S value		
SD irrelevant for ...	Spindle modes other than control mode (SPOS, M19, SPOSA)		
Application example(s)	Mounted in the spindle is a tool balanced with water (e.g. grinding wheel) which is no longer in balance at excessively low speeds (the balancing weight in the form of water runs out of the chambers).		
Special cases, errors, ...	The value SPIND_MIN_VELO_G25 can be changed using: ● G25 S .... in the parts program ● Operator commands via MMC The value in SPIND_MIN_VELO_G25 is retained after a reset or power down.		
Related to	SD 43220: SPIND_MAX_VELO_G26 SD 43230: SPIND_MAX_VELO_LIMS (progr. spindle speed limitation with G96/G961) MD 10710: $MN_PROG_SD_RESET_SAVE_TAB (SW 5.3 and higher)		

43220 SD number	**SPIND_MAX_VELO_G26** Progr. spindle speed limitation G26)		
Default setting: 1000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective immediately	Protection level: MMCMD 9220	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	A maximum spindle speed is entered in SPIND_MAX_VELO_G26, which the spindle must not exceed. The NCK limits an excessive spindle speed setpoint to this value.		
SD irrelevant for ...	All spindle modes except control mode.		
Special cases, errors, ...	The value in SD: SPIND_MIN_VELO_G26 can be altered by means of: ● G26 S .... in the parts program ● Operator commands via MMC The value in SPIND_MIN_VELO_G26 is retained after a reset of power down.		
Related to	SD 43210: SPIND_MIN_VELO_G25 (progr. spindle speed limitation G25) SD 43230: SPIND_MAX_VELO_LIMS (progr. spindle speed limitation with G96/G961) MD 10710: $MN_PROG_SD_RESET_SAVE_TAB		

© Siemens AG, 2003. All rights reserved

43230 SD number	**SPIND_MAX_VELO_LIMS** Progr. spindle speed limit G96/G961		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective immediately	Protection level: MMCMD 9220	Unit: rpm	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	At constant cutting rate (G96/G961 and G97) an extra limitation entered in SPIND_MAX_VELO_LIMS is operate in addition to the continually applied limitations. Furthermore, SPIND_MAX_VELO_LIMS can be defined in the parts program with LIMS=... for the master spindle.		
SD irrelevant for ...	All spindle modes except for G96/G961 and G97 (constant cutting speed)		
Application example(s)	On parting and very small diameters of the workpiece to be machined, the spindle with the workpiece (turning machine) accelerates higher and higher at constant cutting rate (G96/G961), theoretically reaching an infinitely high set speed in the position of the traverse axis X=0. In cases such as this, the spindle rotates up to its maximum spindle speed in the current gear stage (limited, where applicable, by G26). SPIND_MAX_VELO_LIMS can be programmed as an additional spindle speed limit with G96, G961 and G97.		
Special cases, errors, ...	The value SD 43210: SPIND_MIN_VELO_LIMS can be changed using: ● LIMS S .... in the parts program ● Operator commands via MMC The value SPIND_MIN_VELO_LIMS is retained after reset of power down.		
Related to	SD 43220: SPIND_MAX_VELO_G26 (maximum spindle speed) SD 43210: SPIND_MIN_VELO_G25 (minimum spindle speed) MD 10710: $MN_PROG_SD_RESET_SAVE_TAB (SW 5.3 and higher)		

43240 MD number	**M19_SPOS** Position for spindle positioning with M19		
Default setting: 0	Minimum input limit: –10000000.0	Maximum input limit: 10000000.0	
Changes effective immediately	Protection level: 7 / 7	Unit: degrees	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	Spindle position in [DEGREES] for spindle positioning with M19. Preset positions must lie within the value range. The value range is limited internally to the programmed modulo range. Preset paths (SD 43250: M19_SPOSMODE = 2) can be positive or negative and are limited only by the input format.		
Related to	The position approach mode is determined in SD 43250: M19_SPOSMODE.		

43250 MD number	**M19_SPOSMODE** Spindle positioning approach mode for spindle positioning with M19		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 5	
Change effective after: immediate	Protection level: 7 / 7	Unit: –	
Data type: DWORD	Applies from SW 5.3		
Significance:	Spindle position approach mode for spindle positioning with M19. Where the following applies: 0: Approach DC (default) position by shortest path. 1: Approach AC position. 2: IC Travel incrementally (as distance), plus/minus sign gives the travel direction. 3: Approach DC position by shortest path. 4: ACP position with positive direction of approach. 5: ACN position with negative direction of approach.		

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Signal Descriptions

5

5.1 Axis/spindle-specific signals

5.1.1 Signals to axis/spindle

DB 31, ... **DBX2.2** Data block	**Spindle reset/Delete distance-to-go** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal transition 0 ——> 1	Independent of MD 35040: SPIND_ACTIVE_AFTER_RESET, spindle reset has the following effects: Control mode: – Spindle stops – Program continues – Spindle continues to run with the following M and S program Oscillation mode: – Oscillation is interrupted – Axes continue traversing – Program is continued with current gear stage – With the following M value and larger S value, the “Programmed speed too high” (DB31, ... DBX83.1) interface signal is enabled if necessary. Positioning mode: – is stopped Axis mode: – is stopped		
Signal state 0 or signal transition 1 ——> 0	No effect		
Related to	MD 35040: SPIND_ACTIVE_AFTER_RESET (own spindle reset) IS “Reset” (DB21, ... DBX7.7) IS “Delete distance-to-go” (DB31, ... DBX2.2) is a different name for the same signal		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX16.7** Data block	**Delete S value** Signal(s) to axis/spindle (PLC –> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal transition 0 ——> 1	Control mode: – Spindle stops – Program continues – The spindle continues to run with the following S value, if M3 or M4 was active Oscillation mode, axis mode, positioning mode: – Signal is not effective for the appropriate function. However, if Control mode is selected again, a new S value must be programmed.	
Signal state 0 or signal transition 1 ——> 0	No effect	
Application example(s)	Terminate traversing motion on account of an external signal (e.g. sensing probe)	

DB 31, ... **DBX16.5, 16.4** Data block	**Resynchronize spindle 1 and 2** Signal(s) to axis/spindle (PLC –> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal transition 0 ——> 1	The spindle should be resynchronized, as the synchronization between the position measurement system on the spindle and the 0 degree position has been lost.	
Signal state 0 or signal transition 1 ——> 0	No effect	
Signal irrelevant for ...	All spindle modes except control mode	
Application example(s)	The machine has a selector switch for a vertical and horizontal spindle. **Two** different position encoders are used (one for the vertical spindle and one for the horizontal spindle), but only **one** actual value input is used on the control. When the system switches from the vertical to the horizontal spindle, the spindle must be resynchronized. The synchronization is triggered with the “Resynchronize spindle 1 or 2” interface signal.	
Related to	IS “Referenced/synchronized” 1 (DB31, ... DBX60.4) IS “Referenced/synchronized” 2 (DB31, ... DBX60.5)	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX16.3** Data block	**Gear changed** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	When the new gear is engaged the “Actual gear stage A to C” and “Gear changed” interface signals are enabled. This informs the NCK that the correct gear stage has been successfully engaged. The gear stage change is complete (spindle oscillation mode is deselected), the spindle accelerates in the new gear stage to the last programmed spindle speed and the next block in the parts program can be executed. The “Change gear” interface signal is reset by the NCK, which causes the PLC application to reset the “Gear changed” signal.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Signal irrelevant for ...	All spindle modes except oscillation mode		
Special cases, errors, ...	If the PLC user reports back to the NCK with a different actual gear stage than issued by the NCK as the set gear stage, the gear change is still treated as having been successfully completed and the actual gear stage A to C is activated.		
Related to	IS “Actual gear stages A to C” (DB31, ... DBX16.2 to 16.0) IS “Set gear step A to C” (DB31, ... DBX82.2 to 82.0) IS “Change gear” (DB31, ... DBX82.3) IS “Oscillation speed” (DB31, ... DBX18.5)		

DB 31, ... **DBX16.2 – 16.0** Data block	**Actual gear stage A to C** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 (leveloperated)	When the new gear is engaged the “Actual gear stage A to C” and “Gear changed” interface signals are enabled. This informs the NCK that the correct gear stage has been successfully engaged. The gear change is complete (spindle oscillation mode is deselected), the spindle accelerates in the new gear stage to the last programmed spindle speed and the next block in the parts program can be executed. The specification of the actual gear stage is coded. One set of parameters, with the following assignment, is provided for each of the 5 gear stages:		
	Parameter block no.	VDI interface	Data of the data block / Contents
	0	–	Data for axis mode: Servo gain factor, Monitoring
	1	000 001	Data for 1st gear stage: M40 speed Min/max speed, Acceleration
	2	010	Data for 2nd gear stage: etc.
	3	011	Data for 3rd gear stage
	4	100	Data for 4th gear stage
	5	101 110 111	Data for 5th gear stage
Special cases, errors, ...	If the PLC user reports back to the NCK with a different actual gear stage than issued by the NCK as the set gear stage, the gear change is still treated as having been successfully completed and the actual gear stage A to C is activated.		
Related to	IS “Set gear step A to C” (DB31, ... DBX82.0 to DBX82.2) IS “Change gear” (DB31, ... DBX82.3) IS “Gear changed” (DB31, ... DBX16.3) IS “Oscillation speed” (DB31, ... DBX18.5) Parameter sets for gear stages		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX17.6** Data block	**Invert M3/M4** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The direction of rotation of the spindle motor changes with the following functions: ● M3 ● M4 ● M5 ● SPOS/M19/SPOSA from the movement; not applicable to SPOS/M19/SPOSA from the standstill.		
Application example(s)	The machine has a selector switch for a vertical and horizontal spindle. The mechanical design incorporates **one gear wheel more** on the horizontal spindle than on the vertical spindle. The direction of rotation must therefore be changed on the vertical spindle if the spindle is always to rotate clockwise with M3.		

DB 31, ... **DBX17.5, 17.4** Data block	**Resynchronize spindle during positioning 2 and 1** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 4.0	
Signal state 1	The spindle is to be resynchronized during positioning.		
Signal state 0 or signal transition 1 ——> 0	No effect		
Signal irrelevant for ...	All spindle modes except for positioning mode		
Application example(s)	The spindle has an indirect measurement system and slipping may occur between the motor and the clamp. If the signal=1 when the positioning process is started, the old reference is deleted and the zero mark searched for again before the end position is approached.		
Related to	IS “Referenced/synchronized” 1 (DB31, ... DBX60.4) IS “Referenced/synchronized” 2 (DB31, ... DBX60.5)		

DB 31, ... **DBX18.7, 18.6** Data block	**Set counterclockwise/Set clockwise** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	When the “Oscillation via PLC” interface signal is enabled, the two interface signals “Direction of rotation setpoint counterclockwise and Direction of rotation setpoint clockwise” can be used to set the direction of rotation for the oscillation speed. The times for the oscillation movement of the spindle motor are defined by enabling the “Direction of rotation setpoint counterclockwise and clockwise” interface signals for a corresponding length of time.		
Signal irrelevant for ...	All spindle modes except oscillation mode		
Application example(s)	See “Oscillation via PLC” interface signal		
Special cases, errors, ...	● If both of the interface signals are enabled simultaneously, no oscillation speed is output. ● If neither of the interface signals are enabled simultaneously, no oscillation speed is output.		
Related to	IS “Oscillation via PLC” (DB31, ... DBX18.4) IS “Oscillation speed” (DB31, ... DBX18.5)		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX18.5** Data block	**Oscillation speed** Signal(s) to axis/spindle (PLC –> NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	If the gear stage is to be changed (“Change gear” interface signal (DB31–48, DBX82.3) is enabled), the spindle changes to oscillation mode. The spindle decelerates to a standstill with different acceleration levels, according to the point when the “Oscillation speed” interface signal (DB31–48, DBX18.5) was enabled: 1. The “Oscillation speed” interface signal is enabled **before** the “Change gear” interface signal is enabled by the NCK. The spindle is brought to a standstill with the acceleration during oscillation (MD: SPIND_OSCILL_ACCEL). Oscillation commences immediately when the spindle is stationary. 2. The “Oscillation speed” interface signal is enabled **before** the “Change gear” inter signal is enabled by the NCK and when the spindle is stationary. The position controller is switched off. The spindle decelerates with the acceleration in speed control mode. When the “Oscillation speed” is enabled, the spindle starts oscillation with the oscillation acceleration (MD:SPIND_OSCILL_ACCEL). With the IS “Oscillation via PLC” (DB31, ... DBX18.4) **not set**, automatic oscillation is performed in the NCK with IS “Oscillation speed”. The two times for the directions of rotation are entered in MD: in SPIND_OSCILL_TIME_CW (oscillation time for M3 direction) and SPIND_OSCILL_TIME_CCW (oscillation time for M4 direction). If the “Oscillation via PLC” interface signal is **enabled**, the “Oscillation speed” interface signal effects output of a speed only in conjunction with the “Direction of rotation setpoint counterclockwise and clockwise” interface signal. The oscillation, i.e. the continuous change of the direction of rotation, is performed by the PLC application with the “Direction of rotation setpoint counterclockwise and clockwise” interface signal (oscillation via PLC).	
Signal state 0 or signal transition 1 ——> 0	The spindle does not oscillate.	
Signal irrelevant for ...	All spindle modes except oscillation mode	
Application example(s)	The oscillation speed is used to facilitate the engagement of a new gear stage.	
Related to	IS “Oscillation via PLC” (DB31, ... DBX18.4) IS “Set direction of rotation counterclockwise” (DB31, ... DBX18.7) IS “Set direction of rotation clockwise” (DB31, ... DBX18.6)	

DB 31, ... **DBX18.4** Data block	**Oscillation via PLC** Signal(s) to axis/spindle (PLC –> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	If the “Oscillation via PLC” interface signal is **not enabled**, the “oscillation speed” interface signal triggers automatic oscillation on the NCK. The two times for the directions of rotation are entered in MD 35440: SPIND_OSCILL_TIME_CW (oscillation time for M3 direction) and MD 35450: SPIND_OSCILL_TIME_CCW (oscillation time for M4 direction). If the “Oscillation via PLC” interface signal is **enabled**, the “Oscillation speed” interface signal effects output of a speed only in conjunction with the “Direction of rotation setpoint counterclockwise and clockwise” interface signal. The oscillation, i.e. the continuous change of the direction of rotation, is performed by the PLC application with the “Direction of rotation setpoint counterclockwise and clockwise” interface signal (oscillation via PLC).	
Application example(s)	If the new gear stage cannot be engaged in spite of several attempts by the NCK, the system can be switched to oscillation via PLC. Both of the times can then be altered by the PLC user. This assures a reliable change of the gear stage, even with unfavorable gear wheel positions.	
Related to	MD 35440: SPIND_OSCILL_TIME_CW (oscillation time for M3 direction) MD 35450: SPIND_OSCILL_TIME_CCW (oscillation time for M4 direction) IS “Oscillation speed” (DB31, ... DBX18.5) IS “Set direction of rotation counterclockwise” (DB31, ... DBX18.7) IS “Set direction of rotation clockwise” (DB31, ... DBX18.6)	

© Siemens AG, 2003. All rights reserved

Signals for the special spindle interface (SW 6 and higher)

DB 31, ... **DBX30.4** Data block	**Spindle positioning** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 6	
Signal state 1 or signal transition 0 ——> 1	The IS “Spindle positioning” (corresponds to M19 ’Position for spindle positioning’).		
Application example(s)	To stop the spindle from rotation (M3 or M4) with orientation or to reorient it from standstill (M5), SPOS, SPOSA or M19 are used to switch to positioning mode.		
Related to	MD 20850: SPOS_TO_VDI (output of “M19” with SPOS/SPOSA to the VDI) SD 43240: M19_SPOS (position for spindle positioning using M19) IS “Active spindle mode ’Positioning mode” (DB31, ... DBX84.5)		

DB 31, ... **DBX30.3** Data block	**Select gear stage** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) reserved from SW 6	
Signal state 1 or signal transition 0 ——> 1	This function is available soon. For this reason, this IS is preliminarily reserved.		
Application example(s)	One parameter set each is provided for each of the 5 gear stages. When changing to the spindle mode, the corresponding parameter set 1 to 5 is selected according to the gear stage engaged. A gear stage can be preselected as follows: • M40 automatically by the programmed spindle speed • M41 to M45 permanently by the parts program • By the PLC using the function module FC 18 • In reset status, by programming the VDI interface		
Related to	MD 35010: GEAR_STEP_CHANGE_ENABLE (gear stage change is possible) MD 35590: PARAMSET_CHANGEENABLE (parameter set specification possible from the PLC) IS “Change gear” (DB31, ... DBX82.3)		

DB 31, ... **DBX30.2** Data block	**Spindle start CCW** Signal(s) to axis/spindle (PLC –> NCK)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 6	
Signal state 1 or signal transition 0 ——> 1	If the IS “Spindle start CCW” (corresponds to M4 ’Spindle rotation CCW for the master spindle’).		
Application example(s)	On the sositive edge of the start signal “Spindle start counterclockwise” DB31, ... DBX30.2 the content of SD 43200: SPIND_S is read and comes into effect.		
Related to	“Set direction of rotation counterclockwise” (DB31, ... DBX18.7) “Set direction of rotation clockwise” (DB31, ... DBX18.6) IS “Invert M3/M4” (DB31, ... DBX17.6)		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX30.1** Data block	**Spindle start CW** Signal(s) to axis/spindle (PLC –> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 6
Signal state 1 or signal transition 0 ——> 1	If the IS “Spindle start CCW” (corresponds to M3 ’Spindle rotation CW for the master spindle’).	
Application example(s)	On the positive edge of the start signal “Spindle start counterclockwise” DB31, ... DBX30.1 the content of SD 43200: SPIND_S is read and comes into effect.	
Related to	“Set direction of rotation counterclockwise” (DB31, ... DBX18.7) “Set direction of rotation clockwise” (DB31, ... DBX18.6) IS “Invert M3/M4” (DB31, ... DBX17.6)	

DB 31, ... **DBX30.0** Data block	**Spindle stop** Signal(s) to axis/spindle (PLC –> NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 6
Signal state 1 or signal transition 0 ——> 1	If the IS “Spindle start CCW” (corresponds to M5 ’Spindle stop for the master spindle’).	
Application example(s)	A spindle stop is executed if the IS Axis/spindle stationary “Axis/spindle stationary (DB31, ... DBX61.4 = 1 is signalled.	
Related to	IS “Axis/spindle stationary” (DB31, ... DBX17.6)	

5.1.2 Signals from axis/spindle

DB 31, ... **DBX60.0** Data block	**Spindle/no axis** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The machine axis is operated as a spindle in the following spindle modes: • Control mode • Oscillation mode • Positioning mode • Rigid tapping • Synchronous mode The IS to axis (DB31, ... DBB12 to 15) and from axis (DB31, ... DBB74 to 81) are not valid. The IS to spindle (DB31, ... DBB16 to 19) and from spindle (DB31, ... DBB82 to 91) are valid.	
Signal state 0 or signal transition 1 ——> 0	The machine axis is operated as an axis The IS to axis (DB31, ... DBB12 to 15) and from axis (DB31, ... DBB74 to 81) are valid. The IS to spindle (DB31, ... DBB16 to 19) and from spindle (DB31, ... DBB82 to 91) are not valid.	
Application examples	If a spindle is sometimes also used as a rotary axis on a machine tool (turning machine with spindle/C-axis or milling machine with spindle/rotary axis for rigid tapping), the “Spindle/no axis” signal can be used to determine whether the machine axis is in axis mode or spindle mode.	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX82.3** Data block	**Change gear** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	A gear stage can be defined as follows: ● permanently by the parts program (M41 to M45) ● automatically by the programmed spindle speed (M40) M41 to M45: ● The gear stage can be permanently selected using M41 to M45. If a gear stage is set that is not the current (actual) gear stage, IS “Switch gear state” (DB31, ... Change gear interface signal (DB31, ... DBX82.3) and “Set gear stage A to C” interface signals are enabled. M40: ● M40 in the parts program causes the gear stage to be selected automatically by the control. The control checks which gear stage is suitable for the programmed spindle speed (S value). If the suggested gear stage is not equal to the current (actual) gear step, the “Change gear” interface signal and the “Set gear stage A to C” interface signals are enabled. ● While the signal = 1, the channel displays the operating message “waiting for gear change”.	
Special cases, errors, ...	The “Change gear” interface signal is only enabled when a new gear stage is selected that is **not equal** to the current actual gear stage.	
Related to	IS “Set gear step A to C” (DB31, ... DBX82.0 to 82.2) IS “Actual gear stages A to C” (DB31, ... DBX16.0 to D16.2) IS “Gear changed” (DB31, ... DBX16.3)	

DB 31, ... **DBX82.2 – 82.0** Data block	**Set gear stage A to C** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	A gear stage can be defined as follows: ● permanently by the parts program (M41 to M45) ● automatically by the programmed spindle speed (M40) M41 to M45: ● The gear stage can be permanently selected using M41 to M45. If a gear stage is If M41 to M45 specify a gear stage that is not the current (actual) gear stage, IS “Change gear” and IS the Change gear interface signal (DB31–48, DBX82.3) and the “Set gear stage A to C” interface signal are enabled. M40: ● M40 in the parts program causes the gear stage to be selected automatically by the control. The control checks which gear stage is suitable for the programmed spindle speed (S value). If a gear stage is found, which is not equal to the current (actual) gear step, the “Change gear” (DB31, ... DBX82.3) and the IS ”Set gear stages A to C” are enabled. The set gear stage is output in coded form: 1st gear stage 0 0 0 (C B A) 1st gear stage 0 0 1 2nd gear stage 0 1 0 3rd gear stage 1 0 1 4th gear stage 1 0 0 5th gear stage 1 0 1 Invalid value 1 1 0 Invalid value 1 1 1	
Signal irrelevant for ...	Other spindle modes except oscillation mode	
Related to	IS “Change gear” (DB31, ... DBX82.3) IS “Actual gear stages A to C” (DB31, ... DBX16.0 to 16.2) IS “Gear changed” (DB31, ... DBX16.3)	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX83.7** Data block	**Actual direction of rotation clockwise** Signal(s) from axis/spindle (NCK –> PLC)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	When the spindle is rotating, “Actual direction of rotation clockwise” = 1 signals that the direction of rotation is CLOCKWISE. The actual direction of rotation is derived from the spindle position measurement encoder.		
Signal state 0 or signal transition 1 ——> 0	When the spindle is rotating, “Actual direction of rotation clockwise” = 0 signals that the direction of rotation is COUNTERCLOCKWISE.		
Signal irrelevant for ...	• stationary spindle (no evaluation of the direction of rotation possible) • spindle without a position measurement encoder		
Related to	IS “Spindle stationary” (DB31, ... DBX61.4)		

DB 31, ... **DBX83.5** Data block	**Spindle in setpoint range** Signal(s) from axis/spindle (NCK –> PLC)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The “Spindle in setpoint range” interface signal reports whether the programmed, and possibly limited spindle speed has been reached. In the spindle control mode, the speed setpoint (programmed speed + spindle override including limitations) is compared with the actual speed. If the actual speed deviates from the set speed by **less** than the spindle speed tolerance in MD 35150: SPIND_DES_VELO_TOL, the “Spindle in setpoint range” interface signal is enabled.		
Signal state 0 or signal transition 1 ——> 0	The “Spindle in setpoint range” interface signal reports whether the spindle is still in the acceleration or braking phase. In the spindle control mode, the speed setpoint (programmed speed * spindle override) is compared with the actual value. If the actual speed deviates from the set speed by **more** than the spindle speed tolerance in SPIND_DES_VELO_TOL, the “Spindle in setpoint range” interface signal is reset.		
Signal irrelevant for ...	All spindle modes except for speed mode (control mode).		
Application example(s)	The path feed must generally be disabled when the spindle is in the acceleration phase (programmed speed setpoint not yet reached). This can be performed as follows: • The “Spindle in setpoint range” interface signal is evaluated and the “Feed disable” (DB21, ... DBX6.0). The positioning axes are also stopped. • The MD 35500: SPIND_ON_SPEED_AT_IPO_START (feed enable with spindle in setpoint range) is enabled and the NCK then internally evaluates whether the spindle- is within setpoint range. The path feed is only enabled if the spindle is within the setpoint range. The positioning axes are never stopped by this function.		
Related to	MD 35500: SPIND_DES_VELO_TOL (spindle speed tolerance)		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX83.2** Data block	**Setpoint speed increased** (programmed speed too low) Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	If a spindle speed (1/min) or a constant cutting speed (m/min or ft/min) is programmed, the value is below one of the following limits: • Minimum speed of the specified gear stage • Minimum spindle speed • Speed limitation through VDI • Progr. spindle speed limitation G25) • Progr. spindle speed limitation with G96 The spindle speed is limited to the minimum limit.	
Signal state 0 or signal transition 1 ——> 0	If a spindle speed (1/min) or a constant cutting speed (m/min or ft/min) is programmed, the value has not fallen below any of the limits.	
Application example(s)	The “Setpoint speed“ increased interface signal can be used to detect that the programmed speed cannot be achieved. The PLC user can accept this state as permissible and enable the feed, or he can disable the path feed and/or the complete channel the “Spindle in set-point range” interface signal is processed.	

Data block	Signal(s) from axis/spindle (NCK –> PLC)	
DB 31, ... **DBX83.1** Data block	**Setpoint speed limited** (programmed speed too high) Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	If a spindle speed (1/min) or a constant cutting speed (m/min or ft/min) is programmed, one of the following limits has been exceeded: • Speed of the selected gear stage • Maximum spindle speed • Speed limitation through VDI • Progr. spindle speed limitation G26) • Progr. spindle speed limitation with G96 The spindle speed is limited to the maximum value	
Signal state 0 or signal transition 1 ——> 0	If a spindle speed (1/min) or a constant cutting speed (m/min) or ft/min) is programmed, none of the limit values have been exceeded.	
Application example(s)	The “Setpoint speed limited” interface signal can be used to detect that the programmed speed cannot be achieved. The PLC user can accept this state as permissible and enable the feed, or he can disable the path feed and/or the complete channel the “Spindle in set-point range” interface signal is processed.	

DB 31, ... **DBX83.0** Data block	**Speed limit exceeded** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	If the actual speed exceeds the maximum spindle speed MD 35100: SPIND_VELO_ LIMIT by more than the spindle speed tolerance MD 35150: SPIND_DES_VELO_TOL the “Speed limit exceeded” interface signal is enabled and alarm 22050 “Maximum speed reached” is output. All axes and spindles on the channel are brought to a standstill.	
Related to	MD 35150: SPIND_DES_VELO_TOL (spindle speed tolerance) MD 35100: SPIND_VELO_LIMIT (max. spindle speed) Alarm 22050 “maximum speed reached“	

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX84.7** Data block	**Active spindle control mode** Signal(s) from axis/spindle (NCK –> PLC)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The spindle is in control mode with the following functions: ● Spindle direction of rotation specification M3/M4 or spindle stop M5 ● M41 ...M45 or automatic gear stage changeover		
Related to	"Spindle oscillation mode" interface signal (DB31, ... DBX84.6) IS "Spindle oscillation mode 'Positioning mode'" (DB31, ... DBX84.5)		

DB 31, ... **DBX84.6** Data block	**Active spindle mode oscillation mode** Signal(s) from axis/spindle (NCK –> PLC)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The spindle is in oscillation mode if a new gear stage was defined using the automatic gear stage selection (M40) or M41 to M45 ("Change gear" interface signal is enabled). The "Change gear" interface signal is only enabled when a new gear stage is selected that is **not equal** to the current actual gear stage.		
Related to	"Spindle control mode" interface signal (DB31, ... DBX84.7) IS "Spindle oscillation mode 'Positioning mode'" (DB31, ... DBX84.5) IS "Change gear" (DB31, ... DBX82.3)		

DB 31, ... **DBX84.5** Data block	**Active spindle positioning mode** Signal(s) from axis/spindle (NCK –> PLC)		
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The spindle is in positioning mode with the following functions: ● SPOS [n] = ... ● SPOS [n] = AC(.....) ● SPOS [n] = IC(.....) ● SPOS [n] = DC(.....) ● SPOS [n] = ACN(.....) ● SPOS [n] = ACP(.....) ● SPOSA [n] = ... ● SPOSA [n] = AC(.....) ● SPOSA [n] = IC(.....) ● SPOSA [n] = DC(.....) ● SPOSA [n] = ACN(.....) ● SPOSA [n] = ACP(.....) The command parameter [n] is not required for the master spindle.		
Related to	"Spindle control mode" interface signal (DB31, ... DBX84.7) "Spindle oscillation mode" interface signal (DB31, ... DBX84.6)		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX84.3** Data block	**Rigid tapping active** Signal(s) from axis/spindle (NCK –> PLC)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The spindle is switched internally to axis mode during the rigid tapping function (G331/G332). For rigid tapping, the spindle speed is also programmed with S .... in rpm, however the direction of rotation is stored as the leading sign for the thread lead. None of the spindle-specific interface signals, such as the following, are updated: “Spindle reset” “Synchronize spindle” “Invert M3/M4” “Spindle in setpoint range” “Programmable speed too high”	
Application example(s)	Certain functions should not be used during rigid tapping, such as: ● IS “Servo enable” (DB31, ... DBX2.1) reset. ● IS “Set feed halt” (DB31, ... DBX8.3) ● Reset ● When activating the EMERGENCY STOP during rigid tapping, it should be remembered that the tool and workpiece are locked together.	

DB 31, ... **DBB86, DBB87** Data block	**M function for spindle** Signals from axis/spindle (NCK –> PLC), axis-specific	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The M functions are output axis-specific in the DBs 31, ... and channel-specific in the DBs 21, ... Selected “M functions for the spindle” are **output to the PLC in the M function for spindle** interface signal. It is irrelevant whether the spindle has been programmed with or without an address extension and on which channel it has been programmed. The following selected M functions are output here: ● M3 ● M4 ● M5 ● M70	
Related to	IS “S functions for Spindle” (DB31, ... DBB88 to 91), axis-specific IS “M functions for Spindle” (DB21, ... DBB58 and DBB68 to DBB97), axis-specific	

DB 31, ... **DBB88–91** Data block	**S function for spindle** Signals from axis/spindle (NCK –> PLC), axisspecific	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The S functions are output axisspecific in the DBs 31, ... and channel-specific in the DBs 21, ... Selected “S functions for the spindle” are **output to the PLC in the S function for spindle** interface signal. It is irrelevant whether the spindle has been programmed with or without an address extension and on which channel it has been programmed. The following selected S functions are output here: ● S .... as the spindle speed in rpm (programmed value) ● S .... as the constant cutting speed in m/min or ft/min The following S functions are **not** output here: ● S .... as the programmed Spindle speed limitation G25) ● S .... as the programmed Spindle speed limitation G26) ● S .... as the spindle speed in rpm if a spindle is not defined on the control ● S .... as the dwell time in spindle revolutions	
Related to	IS “M functions for Spindle” (DB31, ... DBB86 to 87), axis-specific IS “S functions for Spindle” (DB21, ... DBB60 and DBB98 to DBB115), axis-specific	

■

© Siemens AG, 2003. All rights reserved

6 Examples

6.1 Example of automatic gear stage selection (M40)

To illustrate the contents of the new block search variables (SW 5.3 and higher):
Assumptions for automatic gear stage selection (M40):

S0...500 → 1st gear stage
S501..1000 → 2nd gear stage
S1001..2000 → 3rd gear step

Contents of the system variables:

```
$P_SEARCH_S          ; accumulated S value
$P_SEARCH_DIR        ; accumulated direction of rotation
$P_SEARCH_GEAR       ; Collected gear stage
```

Collected	S value:	direction of rotation:	gear stage:
	; 0/last speed	–5	40/last gear stage
N05 G94 M40 M3 S1000	; 1000	3	40
N10 G96 S222	: 222	3	40
N20 G97	; f (PlanAxPosPCS)	* 3	40
N30 S1500	; 1500	3	40
N40 SPOS=0**	; 1500	–19	40
N50 M19**	; 1500	–19	40
N60 G94 G331 Z10 S300	; 300	–19	40
N70 M42	; 300	–19	42
N80 M4	; 300	4	42
N90 M70	; 300	70	42
N100 M3 M40	; 300	3	40
N999 M30			

* f (PlanAxPosPCS): The speed depends on the current position of the transverse axis in the workpiece coordinate system.

** ($P_SEARCH_SPOS and $P_SEARCH_SPOSMODE are described)

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

7

7.1 Interface signals

DB number	Bit, byte	Name	Refer-ence
Axis-specific			
31, ...	0	Feed override	V1
31, ...	1.7	Override active	V1
31, ...	1.6	Position measurement system 2	A2
31, ...	1.5	Position measurement system 1	A2
31, ...	1.4	Follow-up mode	A2
31, ...	1.3	Axis/spindle disable	A2
31, ...	2.2	Spindle reset/Delete distance-to-go	A2
31, ...	2.1	Servo enable	A2
31, ...	3.6	Velocity/spindle speed limitation	A3
31, ...	16.7	Delete S value	
31, ...	16.5	Resynchronize spindle 2	
31, ...	16.4	Resynchronize spindle 1	
31, ...	16.3	Gear changed	
31, ...	16.2–16.0	Actual gear stage A to C	
31, ...	17.6	Invert M3/M4	
31, ...	17.5	Resynchronize spindle during positioning 2	
31, ...	17.4	Resynchronize spindle during positioning 1	
31, ...	18.7	Direction of rotation setpoint left	
31, ...	18.6	Direction of rotation setpoint right	
31, ...	18.5	Oscillation speed	
31, ...	18.4	Oscillation via PLC	
31, ...	19.7–19.0	Spindle offset H – A	V1
31, ...	30.4	Spindle positioning (SW 6 and higher)	
31, ...	30.3	Select gear stage (reserved in SW 6 and higher)	
31, ...	30.2	Spindle start CCW rotation (SW 6 and higher)	
31, ...	30.1	Spindle start CW rotation (SW 6 and higher)	
31, ...	30.0	Spindle stop (SW 6 and higher)	
31, ...	60.7	Position reached with exact stop fine	B1
31, ...	60.6	Position reached with exact stop coarse	B1
31, ...	60.5	Referenced/synchronized 2	R1
31, ...	60.4	Referenced/synchronized 1	R1
31, ...	60.3	Encoder limit frequency exceeded 2	A3
31, ...	60.2	Encoder limit frequency exceeded 1	A3
31, ...	60.0	Axis/no spindle	

© Siemens AG, 2003. All rights reserved

DB number	Bit, byte	Name	Reference
31, ...	61.7	Current controller active	A2
31, ...	61.6	Speed controller active	A2
31, ...	61.5	Position controller active	A2
31, ...	61.4	Axis/spindle stationary ($n < n_{min}$)	A2
31, ...	82.3	Change gear	
31, ...	82.2–82.0	Set gear stage AC	
31, ...	83.7	Actual direction of rotation clockwise	
31, ...	83.5	Spindle in setpoint range	
31, ...	83.2	Setpoint speed increased	
31, ...	83.1	Setpoint speed limited	
31, ...	83.0	Speed limit exceeded	
31, ...	84.7	Active spindle control mode	
31, ...	84.6	Active spindle mode oscillation mode	
31, ...	84.5	Active spindle positioning mode	
31, ...	84.3	Rigid tapping active	
31, ...	86 and 87	M function for spindle	
31, ...	88–91	S function for spindle	

7.2 Machine data

Number	Identifier	Name	Reference
General ($MN_ ...)			
10192	GEAR_CHANGE_WAIT_TIME	Wait time for acknowledgment of a gear stage change when reorganizing (≥ SW 5.3)	
10714	M_NO_FCT_EOP	M function for spindle active after RESET (SW 6.4 and higher)	
12060	OVR_SPIND_IS_GRAY_CODE	Spindle override with Gray coding	V1
12070	OVR_FACTOR_SPIND_SPEED	Evaluation of spindle speed override switch	V1
12080	OVR_REFERENCE_IS_PROG_FEED	Override reference velocity	V1
Channel-specific ($MC_ ...)			
20090	SPIND_DEF_MASTER_SPIND	Initial setting for master spindle on channel	
20092	SPIND_ASSIGN_TAB_ENABLE	Enabling/disabling of spindle converter	
20850	SPOS_TO_VDI	Output of auxiliary function “M19” to the VDI interface (SW 5.3 and higher)	
22400	S_VALUES_ACTIVE_AFTER_RESET	S function via RESET active	
Axis-specific ($MA_ ...)			
30300	IS_ROT_AX	Rotary axis	R2
30310	ROT_IS_MODULO	Modulo conversion	R2
31044	ENC_IS_DIRECT2	Encoder on intermediate gear (≥ SW 6.4)	
31050	DRIVE_AX_RATIO_DENOM	Denominator load gearbox	G2
31060	DRIVE_AX_RATIO_NUMERA	Numerator load gearbox	G2
31064	DRIVE_AX_RATIO_DENOM	Encoder on intermediate gear (≥ SW 6.4)	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
Axis-specific ($MA_ ...)			
31066	DRIVE_AX_RATIO2_NUMERA	Encoder on intermediate gear gearbox (SW 6.4 and higher)	
31070	DRIVE_ENC_RATIO_DENOM	Denominator measuring gearbox	G2
31080	DRIVE_ENC_RATIO_NUMERA	Numerator measuring gearbox	G2
31122	BERO_DELAY_TIME_PLUS	BERO delay time in plus direction	
31123	BERO_DELAY_TIME_MINUS	BERO delay time in minus direction	
32200	POSCTRL_GAIN	Servo gain factor	G2
32800	EQUIV_CURRCTRL_TIME	Equivalent time constant current control circuit for feedforward control	K3
32810	EQUIV_SPEEDCTRL_TIME	Equivalent time constant speed control circuit for feedforward control	K3
32910	DYN_MATCH_TIME	Time constant for dynamic matching	G2
34040	REFP_VELO_SEARCH_MARKER	Reference point creep speed	R1
34060	REFP_MAX_MARKER_DIST	Monitoring of zero mark distance	R1
34080	REFP_MOVE_DIST	Reference point distance/destination point for distancecoded system	R1
34090	REFP_MOVE_DIST_CORR	Ref. point offset/absolute offset, distancecoded	R1
34100	REFP_SET_POS	Reference point value	R1
34200	ENC_REFP_MODE	Referencing mode	R1
35000	SPIND_ASSIGN_TO_MACHAX	Assignment of spindle to machine axis	
35010	GEAR_STEP_CHANGE_ENABLE	Gear stage change options extendable for fixed position (≥ SW 5.3)	
35012	GEAR_STEP_CHANGE_POSITION	Gear stage change position (≥ SW 5.3)	
35020	SPIND_DEFAULT_MODE	Basic spindle setting	
35030	SPIND_DEFAULT_ACT_MASK	Activate initial spindle setting	
35035	SPIND_FUNCTION_MASK	Setting spindle-specific functions (≥ SW 6)	
35040	SPIND_ACTIVE_AFTER_RESET	Spindle active after reset	
35100	SPIND_VELO_LIMIT	Maximum spindle speed	
35110	GEAR_STEP_MAX_VELO[n]	Maximum speed for gear change	
35120	GEAR_STEP_MIN_VELO[n]	Minimum speed for gear change	
35130	GEAR_STEP_MAX_VELO_LIMIT[n]	Maximum speed of gear stage	
35140	GEAR_STEP_MIN_VELO_LIMIT[n]	Minimum speed of gear stage	
35150	SPIND_DES_VELO_TOL	Spindle speed tolerance	
35160	SPIND_EXTERN_VELO_LIMIT	Spindle speed limitation via PLC	
35200	GEAR_STEP_SPEEDCTRL_ACCEL[n]	Acceleration in speed control mode	
35210	GEAR_STEP_POSCTRL_ACCEL[n]	Acceleration in position control mode	
35220	ACCEL_REDUCTION_SPEED_POINT	Speed limit for reduced acceleration	
35230	ACCEL_REDUCTION_FACTOR	Reduced acceleration	
35300	SPIND_POSCTRL_VELO	Position control activation speed	
35350	SPIND_POSITIONING_DIR	Positioning direction of rotation for nonsynchronized spindle	
35400	SPIND_OSCILL_DES_VELO	Oscillation speed	
35410	SPIND_OSCILL_ACCEL	Oscillation acceleration	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
Axis-specific ($MA_ ...)			
35430	SPIND_OSCILL_START_DIR	Oscillation start direction	
35440	SPIND_OSCILL_TIME_CW	Oscillation time for M3 direction	
35450	SPIND_OSCILL_TIME_CCW	Oscillation time for M4 direction	
35500	SPIND_ON_SPEED_AT_IPO_START	Feed enable with spindle in setpoint range	
35510	SPIND_STOPPED_AT_IPO_START	Feed enable with stationary spindle	
35590	PARAMSET_CHANGE_ENABLE	Parameter set definition possible from PLC	A2
36060	STANDSTILL_VELO_TOL	Threshold velocity “Axis/spindle stationary”	A3
36200	AX_VELO_LIMIT	Threshold value for velocity monitoring	A3

7.3 Setting data

Number	Identifier	Name	Refer-ence
Spindle-specific ($SS_ ...)			
42600	JOG_FEED_PER_REF_SOURCE	Revolutional feedrate control in JOG mode	V1
42800	SPIND_ASSIGN_TAB	Spindle number converter	
42900	MIRROR_TOOL_LENGTH	Mirror tool length offset	W1
42910	MIRROR_TOOL_WEAR	Mirror wear values of tool length compensation	W1
42920	WEAR_SIGN_CUTPOS	Mirror wear values of machining plane	W1
42930	WEAR_SIGN	Invert sign of all wear values	W1
42940	TOOL_LENGTH_CONST	Retain the assignment of tool length compo-nents when changing the machining plane (G17 to G19)	W1
43200	SPIND_S	Specification of the spindle speed (SW 6 and higher)	
43202	SPIND_CONSTCUT_S	Specification of a constant cutting rate for the master spindle (SW 6 and higher)	
43210	SPIND_MIN_VELO_G25	Progr. spindle speed limit G25	
43220	SPIND_MAX_VELO_G26	Progr. spindle speed limit G26	
43230	SPIND_MAX_VELO_LIMS	Progr. spindle speed limit G96/G961	
43240	M19_SPOS	Spindle position for spindle positioning with M19 (SW 5.3 and higher)	
43250	M19_SPOSMODE	Spindle positioning mode for spindle position-ing with M19 (SW 5.3 and higher)	
43300	ASSIGN_FEED_PER_REF_SOURCE	Revolutional feedrate for positioning axes/ spindles	V1, P2

© Siemens AG, 2003. All rights reserved

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Feeds (V1)

1	**Brief Description**		**1/V1/1-3**
2	**Detailed Description**		**1/V1/2-7**
	2.1	Path feed F	1/V1/2-7
	2.1.1	Type of feed G93, G94, G95	1/V1/2-9
	2.1.2	Type of feed G96, G961, G97, G971	1/V1/2-11
	2.1.3	Feed with G33, G34, G35 (thread cutting)	1/V1/2-14
	2.1.4	Feed for G63, tapping with compensation chuck	1/V1/2-23
	2.1.5	Feed for G331/G332, rigid tapping	1/V1/2-23
	2.2	Feed FA for positioning axes	1/V1/2-24
	2.2.1	Feed type for positioning feed – G94	1/V1/2-25
	2.3	Supplementary conditions for feedrate programming	1/V1/2-25
	2.4	Feed control	1/V1/2-26
	2.4.1	Feed disable and feed/spindle stop	1/V1/2-26
	2.4.2	Feed override on machine control panel	1/V1/2-27
	2.4.3	Programmable feed override	1/V1/2-32
	2.4.4	Dry run feed	1/V1/2-33
	2.4.5	Multiple feed values in one block	1/V1/2-34
	2.4.6	Fixed feed values (840D, 810D)	1/V1/2-39
	2.4.7	Feed for chamfer/rounding FRC, FRCM (SW 5.2 and higher)	1/V1/2-40
	2.4.8	Blockwise (nonmodal) feed FB (SW 5.3 and higher)	1/V1/2-42
	2.4.9	Programmable single-axis dynamic response (SW 5.1 and higher)	1/V1/2-43
3	**Supplementary Conditions**		**1/V1/4-49**
4	**Data Descriptions (MD, SD)**		**1/V1/4-49**
	4.1	General machine data	1/V1/4-49
	4.2	Channel-specific machine data	1/V1/4-59
	4.3	Axis/spindle-specific machine data	1/V1/4-61
	4.4	Channel-specific setting data	1/V1/4-62
5	**Signal Descriptions**		**1/V1/5-65**
	5.1	Channel-specific signals	1/V1/5-66
	5.1.1	Signals to channel	1/V1/5-66

© Siemens AG, 2003. All rights reserved

5.2 Axis/spindle-specific signals 1/V1/5-73
5.2.1 Signals to axis/spindle 1/V1/5-73
5.2.2 Signals from axis/spindle 1/V1/5-79

6 Example 1/V1/6-81

6.1 Feedrate programming for chamfer/rounding FRC, FRCM 1/V1/6-81

7 Data Fields, Lists 1/V1/7-83

7.1 Interface signals 1/V1/7-83

7.2 Machine data 1/V1/7-84

7.3 Setting data 1/V1/7-85

7.4 Alarms 1/V1/7-85

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

Types of feed

The feed determines the machining speed (axis or path velocity) and is observed in every type of interpolation even where allowance is made for tool offsets on the contour or on the tool center point path (depending on G commands.

The following types of feed allow optimum adaptation to the various technological applications (turning, milling, drilling, etc.):

- Rapid traverse feed (G0)
- Time-reciprocal feed (G93)
- Linear feed (G94)
- Revolutional feedrate (G95)
- Constant cutting rate (G96), (G961; SW 5.3 and higher)
- Constant speed (G97), (G971; SW 5.3 and higher)
- Feedrate on thread cutting (G33), (G34, G35; SW 5.2 and higher)
- Feed for tapping with compensating chuck (G63)
- Feed for rigid tapping (G331, G332)
- Feed for chamfering/rounding FRC, FRCM (SW 5.2 and higher)
- Blockwise (nonmodal) feed FB (SW 5.3 and higher)

Programmable runin/runout path with G33 (SW 5 and higher)

In SW 5.1 of the SINUMERIK 840D, the thread runin/runout paths can be programmed. The thread axis is accelerated and decelerated within the defined distance.

Note

If a very short distance is specified, the axis can be overloaded.

Axis assignment for feeds

The feeds can be assigned variably to the axes according to the technological requirements. The following versions are supported:

- Separate feeds for working plane and infeed axis
- Variable axis assignment for path feed
- Feed for positioning axes

© Siemens AG, 2003. All rights reserved

Feed control

In order to adapt the system to changing requirements during machining or for test purposes, the programmed feedrate can be changed on the machine control panel, operator panel, PLC or using a program command.

Feed interpolation

In order to be able to define the feed path, the feedrate programming as per DIN 66025 has been expanded to include linear and cubic paths. The cubic paths can be programmed direct or as interpolating splines.
The following feed profiles can be programmed:

- FNORM: Response as per DIN 66025 (default setting).
  An F value programmed in the block is specified as constant over the entire path and applies as a modal value thereafter.
- FLIN: An F value programmed in the block is positioned from the current value at the start of the block to the end of the block linearly across the path distance and applies as a modal value thereafter.
- FCUB: The blockwise programmed F value (relative to the end of the block) are connected by a spline. The spline start and ends at a tangent to the previous or subsequent define feed. If the F address is missing in a block, the last programmed F value is used instead.
- FPO: The F address [syntax: F=FPO (..., ..., ...)] designates the feed response over a polynomial from the current value to the end of the block in which it has been programmed. The end value then applies as a modal value.

MD 20172: $MC_COMPRESS_VELO_TOL allows the definition of a tolerance for the path feed, if FLIN and FCUB are used in conjunction with compression COMPON.

For further information on programmable feed paths, please see
Reference: /PGA/, Programming Guide Advanced

Feed for chamfering/ rounding FRC, FRCM (SW 5.2 and higher)

The machining conditions can change significantly during surface transitions to chamfer/rounding. The chamfer/rounding contour elements therefore need their own optimized feed values in order to achieve the required surface finish.
You can program the feed for chamfer/rounding with

- FRC (non-modal) or
- FRCM (modal).

Blockwise feed FB (SW 5.3 and higher)

A separate feed can be specified for a single block with the command FB. During this block the previously active path feed is overwritten, after the block the previously active modal path feed becomes active once more.

© Siemens AG, 2003. All rights reserved

Programmable single-axis dynamic response (SW 5.1 and higher)

The dynamic response of **single axes** can be changed directly through the programming:

- Percentage acceleration correction (ACC) in the parts program and in synchronized actions
- Programmable movement end criterion: FINEA (Exact stop fine), COARSEA (Exact stop coarse), IPOENDA (Interpolator stop) in the parts program and in synchronized actions
- Programmable servo parameter sets (SCPARA) in the parts program and in synchronized actions.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Detailed Description 2

2.1 Path feed F

Path feed F

The path feed represents the geometrical total of the velocity components in the participating axes. It is therefore generated from the individual motions of the interpolating axes.

The standard setting uses the axial velocities of the geometry axes which have been programmed. The “FGROUP” command can be used to include other geometry and/or synchronized axes in the calculation of the path feed.

The path feed F determines the machining speed and is observed in every type of interpolation even where allowance is made for tool offsets. The value programmed under the address F remains in the program until a new F value or a new type of feedrate is programmed.

Value range for path feed F

References: /PG/, “Programming Guide: Fundamentals”
/FB/, G2, “Velocities, Setpoint/Actual Value Systems, ClosedLoop Control”

F value at PLC interface

The F value of the current path feed is always entered in the channel-specific PLC interface for auxiliary functions (DB21, ... DBB158 to 193).

The related interface signals (modification signal, F value) are described in:
References: /FB/, H2, “Output of Auxiliary Functions to PLC”

Feed with transition circle

Please refer to the following documentation for information:
References: /PA/, “Programming Guide Fundamentals”

© Siemens AG, 2003. All rights reserved

Feedrate for internal radius and external radius path sections

For circular blocks or spline blocks with curvature in the same direction and tool radius offset activated (G41/G42), the programmed feedrate can act on the center point path or on the contour (depending on the internal radius or external radius path sections).
A group of G commands is provided for this purpose.

- CFTCP: programmed feed acting on the center point path.
- CFC: programmed feed acting on the contour.
- CFCIN: programmed feed acting only on the contour with a concave spline.

References: /PA/, “Programming Guide Fundamentals”

Maximum path velocity

The maximum path velocity is obtained from the maximum velocities of the linear and/or rotary axes involved (MD 32000: MAX_AX_VELO), i.e. the axis with the lowest maximum velocity determines the maximum path velocity. This cannot be exceeded.

If G0 is programmed, traversing is at the path velocity resulting from the MAX_AX_VELO limitation.

Limit speed for Path axes

In addition, a limit speed can be programmed for path axes (geometry and synchronous axes) using the command FL[x]= ... the limit for the particular axis.
References: /PG/, “Programming Guide: Fundamentals”

Separate feeds can therefore be programmed for the working plane and infeed axis. This means that a feedrate is specified for both pathrelated interpolation and for the infeed axis. The infeed axis is designated as the axis perpendicular to the selected machining plane. The infeed axisspecific feed can be programmed to limit the axis velocity and therefore the path velocity. No coordinate rotations through frames should be included, i.e. the infeed axis must be an axis of the standard coordinate system. This function can be used to compensate for the fact that a cutter has a lower cutting performance on the face side than across the cutter circumference.

Programming example:

```
. .. .. G94 ....     Selection of feed type (mm/min)
X30 Y20 F200         Path feedrate = 200 mm/s
FL[Z]=50 Z30         Maximum feedrate for Z axis: 50 mm/s
```

© Siemens AG, 2003. All rights reserved

2.1.1 Type of feed G93, G94, G95

Activation

The feed types G93, G94 and G95 are active for the G functions of group 1 (except G0) in the automatic modes.

G94 or G95 can be used to travel in JOG mode.
References: /FB/, H1, “Jogging with the handwheel”

Reciprocal feed (G93) (valid for 840D/810D only)

The inverse time feedrate is used when it is easier to program the duration, rather than the feedrate, for retraction of a block.

Formula

The inverse time feedrate is calculated from the following formula:

$$F = \frac{v}{s};\quad F\ [\text{rev/min}] \qquad (4)$$

with F: Inverse time feedrate
v: Desired path velocity in mm/min or inches/min
s: Path length in mm or inches

Programming example:
N10 G1 G93 X100 Y200 F2; the programmed path is traveled in 0.5 minute.

Note

G93 may not be used when G41/G42 is active. If the block length varies greatly from block to block, a new F value should be programmed in each block for G93.

Linear feed (G94)

The linear feedrate is programmed in the following units relative to a linear or rotary axis:

- [mm/min, degrees/min] on standard metric systems
- [inch/min, degrees/min] on standard imperial systems

Revolutional feedrate (G95)

The revolutional feedrate is programmed in the following units relative to a master spindle:

- [mm/rev] on standard metric systems
- [inch/rev] on standard imperial systems
- [degrees/rev] on a rotary axis

© Siemens AG, 2003. All rights reserved

The path velocity is calculated from the actual speed of the spindle according to the following formula:

Formula

$$V = n * F \quad (5)$$

with V: Path velocity
n: Speed of master spindle
F: Programmed revolutional feed

Note

The programmed F value is deleted when the system switches between the feed types G93, G94 and G95.

In the JOG mode, the response of the axis/spindle also depends on the setting data in SD 41100: JOG_REV_IS_ACTIVE (revolutional feedrate for JOG active).

- If this setting data is active, an axis/spindle always travels with the revolutional feedrate MD 32050: JOG_REV_VELO (revolutional feedrate for JOG) or MD 32040: JOG_REV_VELO_RAPID (revolutional feedrate for JOG with rapid traverse overlay) depending on the master spindle.
- If the setting is not active, the response of the axis/spindle depends on the setting data SD 43300: ASSIGN_FEED_PER_REV_SOURCE (revolutional feedrate for positioning axes/spindles).
- If the setting data is not active, the response of a geometry axis, on which a frame with rotation is active, depends on the channel-specific setting data SD 42600: JOG_FEED_PER_REV_SOURCE (in the JOG mode revolutional feedrate for geometry axes, on which the frame with rotation is active).

"Revolutional feedrate active" DB31, ... DBX62.2

A programmed, active revolutional feedrate (G95) is displayed using this interface signal.

Alarms

- If no F value is programmed, alarm 10860 "No feedrate programmed" is output. The alarm is not generated with G0 blocks.
- If a negative path velocity is programmed, alarm 14800 "Programmed path velocity smaller than or equal to zero" is output.
- If a revolutional feedrate (G95) is programmed but no master spindle has been defined, alarm 10810 "No master spindle defined" is output.

© Siemens AG, 2003. All rights reserved

2.1.2 Type of feed G96, G961, G97, G971

Constant cutting rate (G96, G961)

The constant cutting rate is used on turning machines to keep the cutting conditions constant, independently of the work diameter of the workpiece. This allows the tool to be operated in the optimum cutting performance range and therefore increases its service life.

Selection of G96, G961:
When programming G96, G961, the corresponding S value is interpreted as the cutting rate in m/min or ft/min along the transverse axis. If the workpiece diameter decreases during machining, the speed is increased until the constant cutting speed is reached.

When G96, G961 is first selected in the part program, a constant cutting rate must be entered in mm/min or ft/min, when the command is reselected, a new cutting rate may be entered.

With G96, the control system will automatically switch to revolutional feedrate (as with G95), i.e. the programmed feedrate F is interpreted in mm/rev or inch/rev.

When programming G961, linear feedrate is selected automatically (as with G94). A programmed feedrate F is interpreted in mm/min or inch/min.

Depending on the programmed cutting rate S_{Speed}, either (S_{G96}) or (S_{G961}) and the partoriented actual value of the transverse axis (radius r), the control system determines the spindle speed using the following formula:

Formula

$$n = \frac{S_{Speed}}{2 * \pi * r} \qquad (6)$$

π = circle constant

Example

S_{G96} = 230 m/min
– when r = 0.2 m → n = 183.12 rev/min
– when r = 0.1 m → n = 366.24 rev/min
⇒ The smaller the workpiece diameter, the higher the speed.

For G96, G961, a geometry axis must be defined as the transverse axis.
The transverse axis, the position of which affects the speed of the master spindle, is defined via the channel-specific MD 20100: DIAMETER_AX_DEF (geometry axis with transverse axis function).

The function G96, G961 requires that the machine zero and the workpiece zero of the transverse axis are in the turning center of the spindle.

Constant speed (G97, G971)

G97, G971 deactivates the “Constant cutting rate function” (G96, G961) and saves the last calculated spindle speed. With G97, the feedrate is interpreted as a revolutional feedrate (as with G95).

When programming G971, linear feedrate is selected (as with G94). The feedrate F is interpreted in mm/min or inch/min.

When G97, G971 is active, an S value can be reprogrammed to define a new spindle speed. This will not modify the cutting rate programmed at G96, G961.

© Siemens AG, 2003. All rights reserved

G97, G971 can be used to avoid speed variations in movements along the transverse axis without machining (e.g. cutting tool).

Note

G96, G961 is only active when the workpiece is being machined (G1, G2, G3, spline interpolation, etc., where the feed F is active).

The response of the spindle speed for active G96, G961 and G0 blocks can be defined in the channel-specific MD 20750: ALLOW_G0_IN_96 (G0 logic with G96, G961).

When constant cutting rate G96, G961 is selected, no gear stage change can take place.

The spindle override switch acts on the calculated spindle speed.
A DRF offset on the facing axis does not influence the spindle speed setpoint calculation.

At the start of machining (after G0) and after NC Stop, G60,G09, ...the path start waits for “nAct= nSet”.

The interface signals “nAct = nSet” and “Set speed limited are not modified by internal speed settings.

When the speed falls below the minimum speed or if the signal “Axis/spindle stationary” is recognized, “nAct =nSet” is reset.

A path operation which has started (G64, overgrinding) is not interrupted.

Spindle speed limitation with G96, G961

For the function “Constant cutting rate”, in SD 43230: SPIND_MAX_VELO_LIMS (spindle speed limitation with G96/G961) and in the parts program (for the master spindle) with the programming command “LIMS”, a maximum spindle speed can be set. The most recently changed value (LIMS or SD) is active. LIMS is operative with G96, G961, G97.

The speed limit set with LIMS remains stored after the control is switched off, depending on MD 10710: PROG_SD_RESET_SAVE_TAB[n] (setting data to be updated). When G96, G961, G97 are reactivated, the spindle speed limitation is also activated.

The maximum permissible spindle speed defined by means of G26 or SD 43220: SPIND_MAX_VELO_G26 (maximum spindle speed) cannot be exceeded.

Incorrect programming that would cause one of the speed limits (G26 or SPIND_MAX_VELO_G26) to be exceeded enables the “Programmed speed too high” interface signal (DB31, ... DBX83.0) is set.

In order to ensure smooth rotation with large part diameters, the spindle speed is not permitted to fall below a minimum level. This speed is defined using SD 43210: SPIND_MIN_VELO_G25 (minimum spindle speed) and MD 35140: GEAR_STEP_MIN_VELO_LIMIT for each gear step.

The minimum spindle speed can be changed in the parts program with G25. Incorrect programming that would cause one of the spindle speed limits (G25 or SPIND_MAX_VELO_G25) to be exceeded enables the Set speed too high interface signal (DB31, ... DBX83.2) is set.

© Siemens AG, 2003. All rights reserved

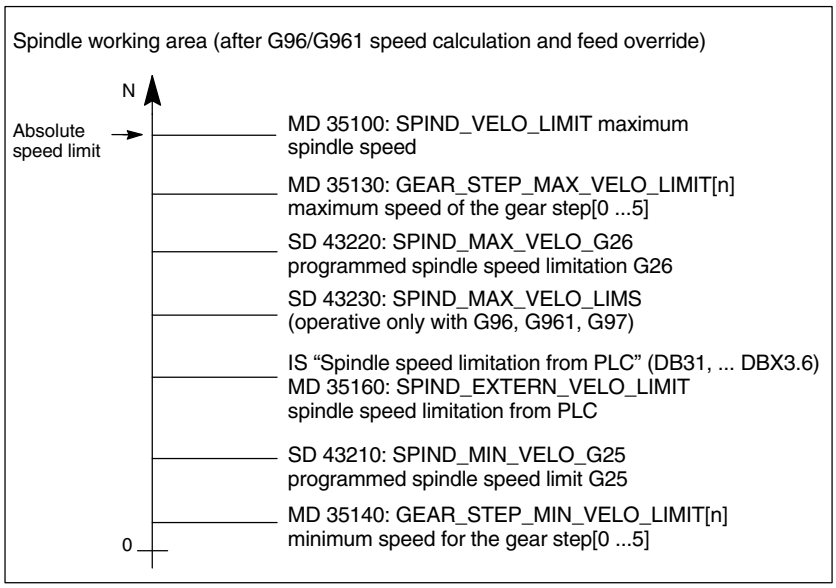


Fig. 2-1 Spindle speed limitations

The various spindle speed limits are illustrated in the figure above. For more detailed information on this and on the method of functioning of the setting data, see:

References: /FB1/, S1, “Spindles”, Spindle monitoring, setting data

Master spindle switchover with G96, G961

With SW 6.2 and lower, the cutting rate programmed under Sxxx is interpreted as speed for a masterslave switchover where the master spindle is rotating with G96 Sxxx.

With SW 6.3 and higher, a master spindle that changes when G96, 961 is active continues to operate at the speed of the preceding master spindle. This corresponds to a transition from B96 to G97. The master spindle newly defined with SETMS executes the “Constant cutting rate” function generated in this way.

Alarms

Constant cutting rate G96, G961

- If no F value is programmed, alarm 10860 “No feedrate programmed” is output. The alarm is not generated with G0 blocks.
- If a negative path velocity is programmed, alarm 14800 “Programmed path velocity smaller than or equal to zero” is output.
- If no transverse axis is defined for active G96/G961 in MD 20100: DIAMETER_AX_DEF (geometry axis with facing axis functions) when G96/G97 is active, alarm 10870 “No facing axis defined” is output.
- If a negative maximum spindle speed is programmed with the LIMS program command when G96, G961 is active, alarm 14820 “Negative maximum spindle speed programmed for G96, G961” is output.
- If no constant cutting rate is programmed when G96, G961 is selected for the first time, alarm 10900 “No S value programmed for constant cutting rate” is output.

© Siemens AG, 2003. All rights reserved

2.1.3 Feed with G33, G34, G35 (thread cutting)

Application of G33

The function G33 can be used to machine threads with constant lead of the following type:

References: /PG/, “Programming Guide: Fundamentals”
/PAZ/, “Programming Guide: Cycles”

Speed S, feedrate F, thread lead

A revolutional feedrate [mm/revolution] is used for G33 threads. The revolutional feedrate is defined by programming the thread lead increase [mm/revolution]. The axis velocities for the thread length are calculated from the programmed spindle speed S and the thread lead:

Feed F [mm/min] = speed S [rev/min] * thread lead [mm/rev]

At the end of the acceleration ramp, the position coupling between the spindle actual value (spindle setpoint with SPCON on master spindle) and the axis setpoint is established. At this moment, the position of the axis in relation to the zero mark of the spindle (including zero mark offsets) is as if the axis had accelerated abruptly at the start of the block when the thread start position (zero mark plus SF) was crossed. Compensation is made for the following error of the axis.

Minimum spindle speed

In order to ensure smooth rotation at low speeds, the spindle speed is not permitted to fall below a minimum level. This speed is defined using SD 43210: SPIND_MIN_VELO_G25 (minimum spindle speed) and in MD 35140: GEAR_STEP_MIN_VELO_LIMIT for each gear step. The minimum spindle speed can be changed in the parts program with G25.

NC stop, single block

NC stop and single block (even at the block boundary) are only active after completion of thread chaining. All subsequent G33 blocks and the first subsequent non-G33 block are traversed like a single block.

Premature abortion without destruction

In SW 4.1 and higher, it is possible to interrupt thread cutting without destruction before the end point is reached. This can be done by activating a retraction motion.

Thread cutting with ROT frame

With ROT frame and G33, G34, G35, alarm 10607 “Thread with frame not executable” is activated if the rotation causes a change in the thread length and thus the lead. Rotation **around the thread axis** is permissible.

Alarm 10607 “Thread with frame not executable” can be suppressed by setting bit 12 in machine data MD 11410: SUPPRESS_ALARM_MASK if the ROT statement is intentionally used in the application.

All other frames are accepted by the NC without alarm. Attention is drawn to the lead-changing effect of SCALE.

© Siemens AG, 2003. All rights reserved

Programmable runin and runout path with G33, G34 and G35 (SW 5.1 and higher)

Functionality

The parts program statement **DITS** (Displacement Thread Start) and **DITE** (Displacement Thread End) can be used to set the acceleration distance (DITS) and/or the deceleration distance (DITE) for thread cutting.

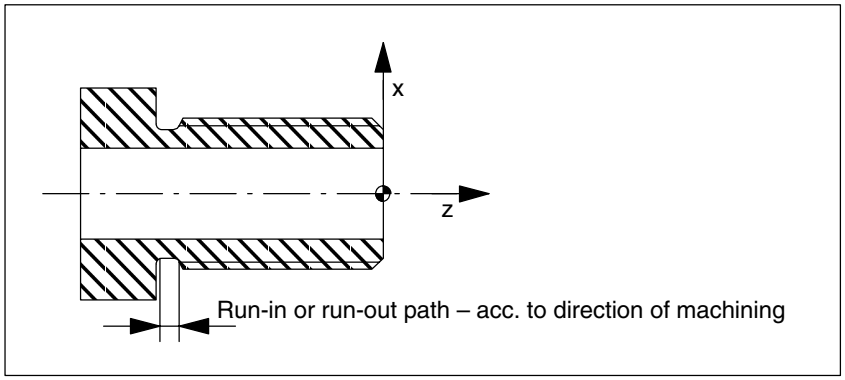


Fig. 2-2 Run-in or run-out path too short

- **Short run-in path**:
  Due to the collar on the thread run-in, only little room is left for the tool (T) start ramp. This must therefore be set shorter via DITS.
- **Short run-out path**:
  Because of the lip at the start of the thread, there is only a small amount of space for the tool start ramp. This results in a risk of collision between the workpiece and the tool edge. The tool braking ramp can be set shorter via DITE. Due to the inertia of the mechanical system, however, a collision can nevertheless occur.
  Remedy: Program a shorter thread, reduce the spindle speed.

The programmed run-in and run-out path has an exclusively acceleration increasing effect on the path. If one of the two paths is set larger than the thread axis needs with active acceleration, the thread axis is accelerated or decelerated with maximum acceleration.

Activation

The DITS and DITE functions are always active during thread cutting.

Example:

```
N...
N59 G90 G0 Z100 X10 SOFT M3 S500
N60 G33 Z50 K5 SF=180 DITS=1 DITE=3   ;Rounding begins at Z=53
N61 G0 X20
```

Alarms

If the run-in and/or run-out path is very short, the thread axis accelerates at a greater rate than allowed for in the configuration. The axis is then overloaded in terms of acceleration and

technology alarm 22280 “Programmed runin path too short” is generated if bit 2 is set in MD 11411: ENABLE_ALARM_MASK.

Alarm 22280 for thread runin or runout is purely an information alarm and has no effect on parts program execution.

© Siemens AG, 2003. All rights reserved

SD 42010:

Only paths, and not positions, are programmed with DITS and DITE. The parts program statements are related to the setting data SD 42010: THREAD_RAMP_DISP[0,1] defining the following acceleration response of the axis on thread cutting:

- **SD 42010 = < 0 to –1** is identical to the previous behavior.
  Starting/braking of the feed axis is carried out with the configured acceleration. Jerk according to current BRISK/SOFT programming.
- **SD 42010 = 0**
  Abrupt starting/braking of the feed axis on thread cutting.
- **SD 42010 = > 0**
  The thread run-up/deceleration distance is set.
  To avoid technology alarm 22280, the acceleration limits of the axis must be observed in case of very small runin and runout paths.

Note

DITE acts as a rounding clearance at the end of the thread. This achieves a smooth change in the axis movement.

Compatibility

MD 20650: THREAD_START_IS_HARD is omitted and replaced by
SD 42010: THREAD_RAMP_DISP[0] or
SD 42010: THREAD_RAMP_DISP[1].

The response of the new setting data with
SD 42010: THREAD_RAMP_DISP[0] = 0 and
SD 42010: THREAD_RAMP_DISP[1] = 0 is identical to the previous
MD 20650: THREAD_START_IS_HARD = 1
and
SD 42010: THREAD_RAMP_DISP[0] = –1 and
SD 42010: THREAD_RAMP_DISP[1] = –1 is identical to the previous
MD 20650: THREAD_START_IS_HARD = 0

The default setting of MD 20650: THREAD_START_IS_HARD=1 is valid up to SW 4.4.

The default setting of SD 42010: THREAD_RAMP_DISP[0;1] = –1 is valid from SW 5.1.

Supplementary conditions

When a block containing the command DITS and/or DITE is loaded into the interpolator, the path programmed in DITS is copied into SD42010 THREAD_RAMP_DISP[0] and the path programmed in DITE is copied into SD42010 TEAD_RAMP_DISP[1].

The programmed run-in path is handled in accordance with the current setting (inch, metric).

If no run-in/deceleration path is programmed before or in the first thread block, it is determined by the current contents of SD 42010 THREAD_RAMP_DISP[0,1].

In case of RESET, the setting data SD 42010: THREAD_RAMP_DISP[0,1] is set to the value –1.

© Siemens AG, 2003. All rights reserved

G34, G35 linear progressive/degressive thread lead change (SW 5.2 and higher)

Application G34, G35

The functions can be used to produce self-shearing threads.

Functionality

The thread lead increase (G34) defines the numerical increase in the lead value. A larger lead results in a larger distance between the thread grooves on the workpiece. The velocity of the thread axis thus increases with supposed constant spindle speed.
The opposite is similarly true for the thread lead decrease (G34).

The following definitions have been made for the thread lead changes with respect to the new G codes:

- G34 thread lead **increase** corresponds to progressive change
- G35 thread lead **decrease** corresponds to degressive change

Both functions – G34 and G35 – imply the functionality of G33 and additionally provide the option of programming an absolute lead change value for the thread under F. If the start and end lead of a thread is known, the thread lead change can be determined using the following equation:

Formula

$$F = \frac{|k^2_e - k^2_a|}{2 * l_G} \qquad (7)$$

where the following applies:

F: the thread lead change to be programmed [mm/rev^2]
k_e: lead of axis target point coordinate of the thread axis [mm/rev]
k_a: thread start lead (programmed under I, J or K) [mm/rev]
l_G: thread length [mm]

The absolute value of F must be applied to G34 or G35 according to the desired lead increase or lead decrease.

When the thread length l_G, lead change F and initial lead k_a are known, the lead increase at the end of block k_e can be determined as follows by modifying the formula (7):

Formula

$$k_e = \sqrt{k^2_a + F * 2 * l_G} \qquad \text{valid with G34 (increasing lead).} \qquad (8)$$

Formula

$$k_e = \sqrt{k^2_a - F * 2 * l_G} \qquad \text{valid with G34 (decreasing lead).} \qquad (9)$$

Note

If the formula (9) results in a negative root expression, the thread cannot be machined!

In this case, the NC signals alarm 10605 or alarm 22275.

© Siemens AG, 2003. All rights reserved

Sample program

Thread cutting G33 with decreased thread lead G35

```
N1608 M3 S10                        ; Spindle speed
N1609 G0 G64 Z40 X216               ; Approach starting point
N1610 G33 Z0 K100 SF=R14            ; Thread with constant lead 100 mm/rev
N1611 G35 Z–220 K100 F17.045455     ; Thread lead decrease 17.045455 mm/rev2
                                    ; Lead at block end 50 mm/rev
N1612 G33 Z–240 K50                 ; Execute thread block without jerk
N1613 G0 X218                       ;
N1614 G0 Z40                        ;
N1616 M17                           ;
```

Suppress special alarms

Any lead changes that would overload the thread axis when G34 is active or which would result in an axis standstill when G35 is active, are detected during block preparation in time. Alarm 10604 “Thread lead increase too high” or alarm 10605 “Thread lead decrease too high” is activated if bit 10 in MD 11410: SUPPRESS_ALARM_MASK is not set.

On thread cutting, certain practical applications require a correction of the spindle speed. In this case, the operator will base his correction on the permissible velocity of the thread axis.

By setting bit 10 of machine data MD 11410: SUPPRESS_ALARM_MASK, it is possible to suppress output of alarms 10604 and 10605 generated as a function of monitoring.

Block preparation continues as normal when alarms are suppressed. The following situations are monitored cyclically when the thread is machined (interpolation):

- Exceeding of maximum velocity of thread axis
- Reaching of axis standstill at G35.

In these cases,
alarm 22270 “Maximum velocity of thread axis reached” or
alarm 22275 “Zero velocity of thread axis reached” is output.

Alarms

Thread cutting G33, G34, G35

- The following alarms are output when programming is incorrect:
  Alarm 10604 “Thread lead **increase** too high”
  Alarm 10605 “Thread lead **increase** too high”
  Alarm 10607 “Thread with frame (ROT) not executable”
  Alarm 16005 “Illegal path for retract path”
  Alarm 16710 “Master spindle not programmed”
  Alarm 16720 “Thread lead is zero”
  Alarm 16730 “Incorrect parameter”
  Alarm 16740 “No geometry axis programmed”
- If the spindle speed is too high with active G33, G34, G35, e.g. because the spindle speed override is set to 200%, alarm 22270 “Spindle speed on thread cutting too high” is output.
  When the rapid traverse rate of the thread axis is exceeded, alarm 22270 is signaled. It is possible to reduce the spindle speed using the spindle override switch to prevent serious alarms.

References: /FB1/, D1, “Diagnostic Tools”, General Machine Data
/DA/, “Diagnostics Guide”

© Siemens AG, 2003. All rights reserved

Stop for thread cutting

Note

The non-destructive interrupt function should only be used for thread cutting, not for tapping with G33.

Retraction motion

The retraction motion (liftfast) for thread cutting is controlled by the following keywords:

- **LFON** ⇒ Enable liftfast for thread cutting
- **LFOF** ⇒ Disable liftfast for thread cutting

These G functions can always be programmed. The MD 20150: GCODE_RESET_VALUES.

The following sources can initiate a retraction motion during thread cutting:

- Fast inputs (programming with SETINT LIFTFAST for “LIFTFAST” option)
- NC Stop
- Alarms that implicitly initiate the NC stop.

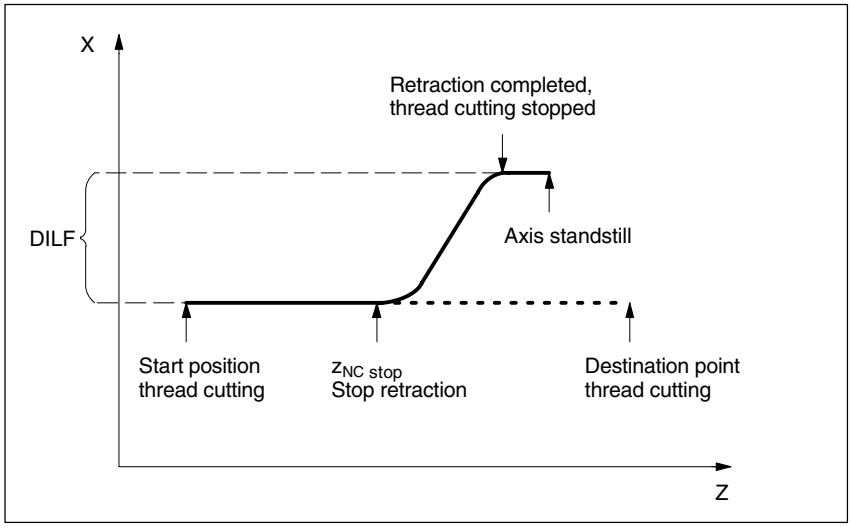


Fig. 2-3 Interruption of G33 through retraction motion

Retraction path

The retraction path can be configured in MD 21200: LIFTFAST_DIST. If required, this path in the parts program can be changed by writing DILF at any point.

After NC reset, the value entered in MD 21200: LIFTFAST_DIST is always active (default value).

© Siemens AG, 2003. All rights reserved

Retraction direction (up to and including SW 4.2)

The type of defining the retraction direction is set in MD 20660: THREAD_AUTO_LIFTFASTANGEL (not included in SW 4.3 and higher):

1: Retraction angle and direction are determined internally on initiation of the-retraction motion. It is alway orthogonal to the axis for which the thread pitch was programmed.

   The following have no effect on the retraction direction:

   - ALF programming,
   - the direction of travel of the thread axis (right/left thread),
   - the transition from cylindrical to conical thread,

   provided that the axis that determines the thread lead is not changed.

0: The retraction angle is defined by the **ALF** programming and is orthogonal to the tangent of the interpolated contour at the time of retraction.

   The transition from the cylindrical to conical thread changes the retraction-direction according to the change in the tangent at the contour.

For information of programming ALF, refer to:

References: /PGA/, Programming Guide Advanced

The retraction direction depends on the current machining plane. G18 is an expedient setting.

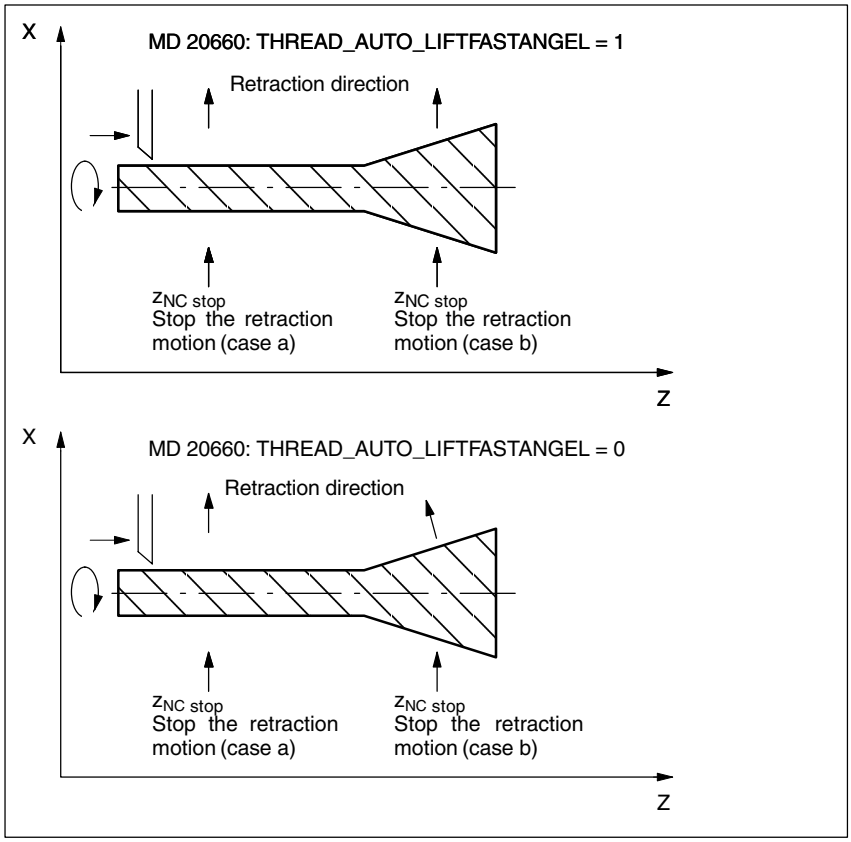


Fig. 2-4 Retraction direction (machining outer thread)

© Siemens AG, 2003. All rights reserved

Retraction direction (SW 4.3 and higher)

The type of defining the retraction direction is controlled in conjunction with the variable **ALF** using the following vocabulary words:

- **LFTXT**

  The plane of the retraction movement is determined by the path tangent and the tool direction. This G code (default setting) is used to program the response on a fast lift.

  For information of programming ALF, refer to:

  Reference: /PGA/, Programming Guide Advanced
  /FB/, K1, Mode Group, Channel, Program Operation Mode (Interrupt routines and LIFTFAST, Fast retraction).

- **LFWP**

  The plane of the retraction movement is the active working plane selected with G codes G17, G18 or G19. The direction of the retraction plane is independent of the path tangent. This allows a fast lift to be programmed parallel to the axis.

These G functions can always be programmed. The MD 20150: GCODE_RESET_VALUES.

In the plane of the retraction movement, **ALF** is used, as before, to program the direction in discrete steps of 45 degrees. With **LFTXT**, retraction in the tool direction was defined for ALF=1.

With **LFWP** the direction in the working plane is derived from the following assignment:

G17:	X/Y plane	ALF=1	Retraction in X direction
		ALF=3	Retraction in Y direction
G18:	Z/X plane	ALF=1	Retraction in Z direction
		ALF=3	Retraction in X direction
G19:	Y/Z plane	ALF=1	Retraction in Y direction
		ALF=3	Retraction in Z direction

Note

The extension for programming the plane of the retraction movement can be used independently of thread cutting.

Retraction speed

The retraction is made at the maximum axis speed. This can be configured in the MD (MD 32000: MAX_AX_VELO).

Retraction acceleration, retraction jerk

The acceleration is made with the maximum permissible values. This can be configured in the MD (MD 32300: MAX_AX_ACCEL).

© Siemens AG, 2003. All rights reserved

Example

```
N55 M3 S500 G90 G18                ; Set active machining plane
...
N65 MSG (thread cutting)
MM_THREAD:
N67 $AC_LIFTFAST=0                 ;Reset before the start of the thread
N68 G0 Z5
N69 X10
N70 G33 Z30 K5 LFON DILF=10 LFWP ALF=7
                                   ;Enable fast lift for
                                   ;Thread cutting
                                   ;Retraction distance=10 mm
                                   ;Retraction plane Z/X (because of G18)
                                   ;Retraction direction –X (with ALF=3
                                   ;Retraction direction +X)
N71 G33 Z55 X15
N72 G1                             ;Select thread cutting
N69 IF $AC_LIFTFAST GOTOB MM_THREAD   ;if thread cutting
                                      ;was interrupted
N90 MSG ("")
...
N70 M30
N55 M3 S500 G90 G0 X0 Z0
...
N87 MSG (Tapping)
N88 LFOF                           ;Disable fast retraction before tapping
N89 CYCLE ...                      ;Tapping cycle with G33
N90 MSG ("")
...
N99 M30
```

Control response

During POWER ON and Reset, the retraction path is set with the configured path (MD) and the status of **LFON** or **LFOF** and **LFTXT** or **LFWP** through MD 20150: GCODE_RESET_VALUES of the corresponding G group.

Smoothed actual values (SW 6.3 and higher)

If low-resolution encoders are being used, smoothed actual values can be applied for constant coupled path and axis motions. Machine data MD 34990: ENC_ACTVAL_SMOOTH_TIME can be used to change the time constant for better smoothed actual values for:

- Thread cutting with feed on G33, G34, G35
- Revolutional feedrate on G95, G96, G97, FRAPON
- Display of actual positions and actual speed/velocity

or change speed. The larger the time constant, the better the smoothing of the actual values and the longer the overtravel.

© Siemens AG, 2003. All rights reserved

2.1.4 Feed for G63, tapping with compensation chuck

Application

G63 is a subfunction for tapping threads using a tap with compensating chuck. A position encoder is not required.

Speed S, feed F, thread lead

With G63, a speed S must be programmed for the spindle and a feed F for the infeed axis (axis for thread length).
The programmer has to calculate the feed F from the speed S and the thread lead:

Feed F [mm/min] = speed S [rev/min] * thread lead [mm/rev]

Alarms

The following alarms are output when programming is incorrect:
Alarm 16700 “Incorrect feed type”
Alarm 16710 “Master spindle not programmed”

If a path axis is stopped when G63 is active, e.g. through an axisspecific “Feed hold”, alarm 22200 “Axis stop with tapping” is output.

2.1.5 Feed for G331/G332, rigid tapping

Application

G331 (tapping) and G332 (tapping retraction) can be used to tap a thread without a compensating chuck (rigid tapping) if the spindle is technically capable of operating in position control mode.

Speed S, feed F, thread lead

With G331 and G332, a revolutional feedrate is active [mm/rev]. The revolutional feedrate is defined by programming the thread lead increase [mm/revolution].
The axis velocities for the thread length are calculated from the programmed spindle speed S and the thread lead:

Feed F [mm/min] = speed S [rev/min] * thread lead [mm/rev]

Alarms

The following alarms are output when programming is incorrect:
Alarm 16700 “Incorrect feed type”
Alarm 16720 “Thread lead is zero” and Alarm 16760 “S value missing”

Note

For further information about programming G63/G331/G332, refer to:
References: /PG/, “Programming Guide: Fundamentals”
/PAZ/, “Programming Guide: Cycles”

© Siemens AG, 2003. All rights reserved

2.2 Feed FA for positioning axes

Feed FA for positioning axes

The axis-specific feed FA defines the velocity of a positioning axis.

One feed can be defined for each axis in a parts program block. A maximum of five feeds per block can be defined for positioning axes.

The value programmed at the FA address for the defined axis remains active in the program until a new FA value is programmed.

If no feed FA is programmed, the axis travels with the value defined in MD 32060: POS_AX_VELO (reset value for positioning axis velocity).

At the program end or reset, the FA value is reset to the last value programmed or to the value entered in MD: POS_AX_VELO, depending on the setting in MD 22410: F_VALUES_ACTIVE_AFTER_RESET (F function active via RESET).

The time set in MD 32000: MAX_AX_VELO (maximum axis velocity) cannot be exceeded.

Value range for positioning feed FA

References: /PG/, “Programming Guide: Fundamentals”
/FB/, G2, “Velocities, Setpoint/Actual Value Systems, ClosedLoop Control”

F value at PLC interface

The F values of the positioning axes of the current block are entered in the PLC interface for auxiliary functions (DB21, ... DBB158 to DBB193).

Please refer to the following documentation for the channelspecific interface signals (modification signal, machine axis number of positioning axis, F value):
References: /FB/, S5, “Synchronized Actions”

In addition, the F values of the positioning axes are entered in the axis-specific PLC interface (DB31, ... DBB78 – DBB81).
The values are retained until they are overwritten by new ones.

Application:
Monitoring of the programmed F values by the PLC user program and selection of an axis-specific feed override from the PLC when a limit value is exceeded.

Note

It is recommended to avoid output of the F functions to the PLC where possible, as this can lead to reductions in speed in contouring mode.
The parameters are defined in MD 22240: AUXFU_F_SYNC_TYPE (output time for F functions).
References: /FB/, S5, “Synchronized Actions”

© Siemens AG, 2003. All rights reserved

2.2.1 Feed type for positioning feed – G94

Linear feed (G94)

The feed for positioning axes is always traversed with the G94 feed type.
The linear feed is programmed in the following units in relation to a linear axis or rotary axis:

- [mm/min, degrees/min] on standard metric systems
- [inch/min, degrees/min] on standard imperial systems

Alarm

When a negative axis velocity is programmed, alarm 14810 “Negative axis velocity programmed for positioning axis” is output.

2.3 Supplementary conditions for feedrate programming

Unit of measurement

The unit of measurement used for the feedrates depends on the value entered in MD 12240: SCALING_SYSTEM_IS_METRIC (standard metric/inch system on the control) and the axis type entered in MD 30300: IS_ROT_AX (rotary or linear axis).

Standard setting for feed type

G94 is displayed on the screen as the standard setting. The reset setting (standard programmed setting) for the feed type is only displayed at the start of the parts program.
The reset setting is defined with MD 20150: GCODE_RESET_VALUES (reset setting of the G groups).

Activation of the F values

Individual basic frames can be deleted with MD 22140: F_VALUES_ACTIVE_AFTER_RESET (F function active via RESET) can be used to define whether the most recently programmed F values are to be active after a reset.

- Programmed path feed: F=....
- Programmed feed change for path feed: OVR=...
- Programmed positioning feed: FA=....
- Programmed feed change for positioning feed: OVRA[U]=...

Please refer to the following documentation for further information on the syntax.
References: /PG/, “Programming Guide: Fundamentals”

Spindle positioning

When G95, G96, G961, G97, G971, G33, G35 are active, no spindle positioning should take place, since the derived path feed = 0 after spindle positioning has been completed.
⇒ If the programmed axis position has not been reached, the block cannot be terminated.

© Siemens AG, 2003. All rights reserved

2.4 Feed control

Programming and controlling the feedrate

The following figure illustrates the possibilities for programing and controlling the feedrate.
For further information on
– Feed programming, see Sections 2.1, 2.2, 2.3
– Feed override on the machine control panel, see Subsection 2.4.2
– Programmable feed override, see Subsection 2.4.3
– Activation of a dry run feed, see Subsection 2.4.4

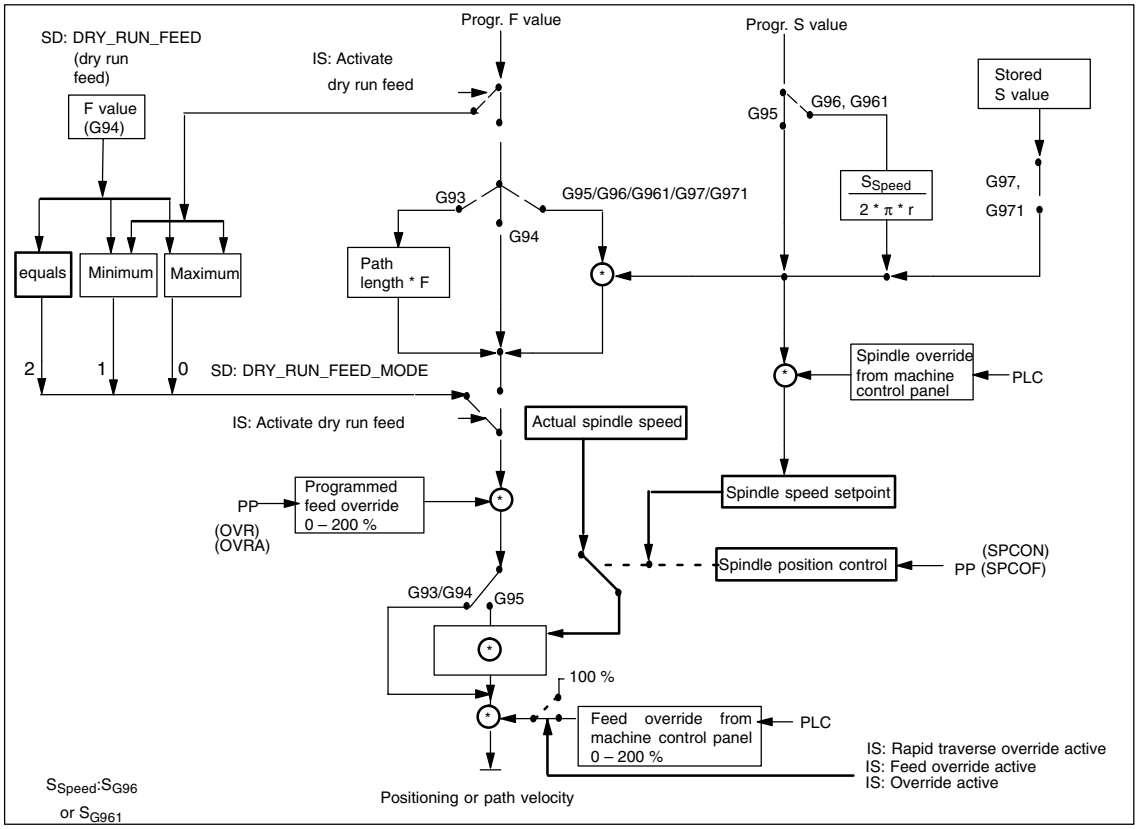


Fig. 2-5 Programming and controlling the feedrate

2.4.1 Feed disable and feed/spindle stop

General

The feed disable or feed/spindle stop brings the axes to a standstill with adherence to the braking characteristics and the path contour (exception: G33 block).

References: /FB/, B1, “Continuous-Path Mode, Exact Stop and Look Ahead”

“Feed disable” DB21, ... DBX6.0

IS “Feed disable” (DB21, ... DBX6.0) shutsdown all axes (geometry and auxiliary axes) of a channel in all modes.

© Siemens AG, 2003. All rights reserved

When the following functions are active, the channel-specific feed disable is

– inactive with G33, G34, G35
– active with G63
– active with G331, G332

"Feed hold" DB21, ...DBX12.3 geometry axes

IS "Feed hold" (DB21, ... DBX12.3 and the following signals) for geometry axes 1, 2 and 3 are used to stop the axes in JOG mode.

"Feed hold" DB31, ...DBX4.3 axisspecific

IS "Feed hold" (DB31, ... DBX4.3) stops the relevant machine axis.

In automatic mode:

- If the "Feed hold" is performed for a path axis, all the axes traversed in the current block and all axes participating in the axis group are stopped.
- If the "Feed hold" is performed for a positioning axis, only this axis is stopped.

Only the current axis is stopped in JOG mode.

The axis-specific Feed Hold is active when
– G33, G34, G35 are active (in this case, contour deviations will occur)
– active with G63
– G331, G332 are active

"Axis/spindle disable" DB31, ... DBX1.3

If "Axis/spindle lock" (DB 31, ...DBX1.3) is active, the axial PLC interlocks

"No servo enable" or "Feed hold" are not active.

The axial and channel-specific override, however, are active.

"Spindle stop" DB31, ... DBX4.3

The interface signal IS "Spindle hold" (DB31, ... DBX4.3) stops the relevant spindle.

When the following functions are active, the spindle stop is
– active with G33, G34, G35 (contour deviations can arise, depending on the dynamic characteristic values)
– active with G63
– inactive with G331, G332

2.4.2 Feed override on machine control panel

General

The operator can use the feed override switch to increase or decrease the path feed relative to the programmed percentage with immediate effect. The feeds are multiplied by the override values.

© Siemens AG, 2003. All rights reserved

An override between 0 and 200% can be programmed for the path feed.

The rapid traverse override switch is used to reduce the traversing velocity when testing a parts program.

An override between 0 and 100% can be programmed for the rapid traverse.

The feed can be changed axisspecifically for positioning axes. The override can be between 0 and 200%.

The spindle override can be used to modify the spindle speed and the cutting rate (with G96, G961). The override can be between 0 and 200%.

The override is not permitted to exceed the machine-specific acceleration and speed limits or generate a contour error.

The feed override can be changed separately for path and positioning axes.

The overrides act on the programmed values or on the limits (e.g. G26, LIMS for spindle speed).

A sequence of G63 blocks is performed without consideration of the feed override. A sequence of G63 blocks comprises directly consecutive blocks with the G code G63. It begins with the first G63 block and ends with the first path motion block which is not G63.

Channel-specific "Feed override" DB21, ...DBB4 "Rapid traverse override" DB21, ...DBB5

One enable signal and one byte is provided on the PLC interface for the override factor for each of the feed types.

IS "Feedrate override" (DB21, ... DBB4)

IS "Feed override active" (DB21, ... DBX6.7)

IS "Rapid traverse override" (DB21, ... DBB5)

IS "Rapid traverse override active" (DB21, ... DBX6.6)

The interface for the override can be provided by the PLC in binarycoded or Graycoded format.
The MD12020: OVR_FEED_IS_GRAY_CODE (path feed override switch Graycoded) and MD 12040: OVR_RAPID_IS_GRAY_CODE (rapid traverse override switch binarycoded) are used to define whether the binary or Gray codes are active.

The following permanent assignment applies to binary coding:

Code	Override factor
00000000	0.00 ≙ 0%
00000001	0.01 ≙ 1%
00000010	0.02 ≙ 2%
00000011	0.03 ≙ 3%
00000100	0.04 ≙ 4%
.	.
.	.
.	.
01100100	1.00 ≙ 100%
.	.
.	.
.	.
11001000	2.00 ≙ 200%

© Siemens AG, 2003. All rights reserved

With Graycoding, the override factors corresponding to the switch setting are entered in MD 12030: OVR_FACTOR_FEEDRATE [n] (evaluation of path feed override switch) or MD 12050: OVR_FACTOR_RAPID_TRA [n] (evaluation of rapid traverse override switch).

An active feed override acts on all path axes assigned to the current channel.

An active rapid traverse override acts on all axes moving with rapid traverse and assigned to the current axis.

If no dedicated rapid traverse override switch is used, the system can be switched between rapid traverse override and feed override. In this case, feed overrides above 100% are limited to 100% rapid traverse override.

The override which is to be active can be selected from the PLC or operator panel. When rapid traverse override is activated from the operator panel, the PLC basic program transmits the “Feed override for rapid traverse selected” interface signal (DB21, ... DBX25.3) to the “Rapid traverse override active” interface signal (DB21, ... and copies the “Feed override” interface signal (DB21, ... DBB4) into the “Rapid traverse override” interface signal (DB21, ... DBB5). DBB5).

When activated from the PLC, the PL C user program is required to enable the **“Rapid traverse override active” interface signal (DB21, DBX6.6)** and **copy** the interface signal for the feed override **(DB21, ... DBB4)** in to the interface signal for rapid traverse override **(DB21, ... DBB5)**.

When the following functions are active, the channel-specific feed and rapid traverse overrides are:
- G33, G34, G35 inactive
- G63 inactive
- inactive with G331, G332

Reference speed for path feed override

From SW 4, it is possible to specify the reference speed for the path feed override that deviates from the standard value. The machine data MD 12082: OVR_REFERENCE_IS_MIN_FEED is used to make this selection.

© Siemens AG, 2003. All rights reserved

Axisspecific "Feed override" DB31, ...DBB0

One enable signal and one byte for the feed override factor are available on the PLC interface for each positioning axis.

IS "Feedrate override" (DB31, ... DBB0)
IS "Override active" (DB31, ... DBX1.7)

The interface for the feed override can be provided by the PLC in binary-coded or Gray-coded format.
The MD 12000: OVR_AX_IS_GRAY_CODE (axis feed override switch Gray-coded) is used to define whether the binary or Gray codes are active.

The following permanent assignment applies to binary coding:

Code	Override factor
00000000	0.00 ≙ 0%
00000001	0.01 ≙ 1%
00000010	0.02 ≙ 2%
00000011	0.03 ≙ 3%
00000100	0.04 ≙ 4%
.	.
.	.
.	.
01100100	1.00 ≙ 100%
.	.
.	.
.	.
11001000	2.00 ≙ 200%

With Gray-coding, the override factors corresponding to the switch setting are entered in MD 12010: OVR_ FACTOR_AX_ SPEED [n] (evaluation of axis feed override switch).

The axis-specific feed override is inactive when
- G33, G34, G35 inactive
- G63 is inactive (The offset is fixed by the NC to 100%).
- G331, G332 are inactive (the NC sets the override permanently to 100%)

"Spindle override" DB31, ...DBB0

One enable signal and one byte for the spindle override factor.
IS "Spindle override" (DB31, ... DBB19), up to SW 2, DBB0.
IS "Override active" (DB31, ... DBX1.7)

The interface for the spindle override can be provided by the PLC in binary-coded or Gray-coded format.

The MD 12060: OVR_SPIND_IS_GRAY_CODE (spindle offset switch is Gray-coded) is used to define whether the binary or Gray codes are active.

© Siemens AG, 2003. All rights reserved

The following permanent assignment applies to binary coding:

Code	Override factor
00000000	0.00 ≙ 0%
00000001	0.01 ≙ 1%
00000010	0.02 ≙ 2%
00000011	0.03 ≙ 3%
00000100	0.04 ≙ 4%
.	.
.	.
.	.
01100100	1.00 ≙ 100%
.	.
.	.
.	.
11001000	2.00 ≙ 200%

With Gray-coding, the override factors corresponding to the switch setting are entered in MD 12070: OVR_FACTOR_SPIND_SPEED [n] (evaluation of spindle override switch).

The spindle override is active when
- G33, G34, G35 active (if the spindle is in position control, the override switch can be operated).
- G63 inactive (the NC sets the override permanently to 100%)
- G331, G332 active.

Limiting the correction factor

On binary-coded interfaces, the maximum override factors can be further limited for path feed, axis feed and spindle speed using MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override factors). Please see Section 4.1 for further information.

Override active DB21, ...DBX6.6 DB21, ... DBX6.7 DB31, ... DBX7.7

The override values set with the selection switch on the machine control panel are immediately active for all operating modes and machine functions, provided that the “Rapid traverse override active”, (DB21, ... DBX6.6) “Feed override active” (DB21, ... DBX6.6) or IS “Override active” (DB31, ... DBX1.7).

An override factor of 0% acts as a feed disable.

Override inactive

When the override is inactive (i.e. the above interface signals are set to “0”), the override factor “1” is used internally on the NC (i.e. the override is 100%). In this case, the value entered in the PLC interface has no effect.

Exceptions are the zero setting for a binary interface and the 1st switch setting for a Gray-coded interface. In these cases, the override factors entered in the PLC interface are used. With a binary interface, the override factor = 0. With a Gray-coded interface, the value entered in the machine data for the 1st switch setting is output as the override value. This should be initialized with “0”.

An override factor of 0% acts as a feed disable.

© Siemens AG, 2003. All rights reserved

Spindle override reference

The MD 12080: OVR_REFERENCE_IS_PROG_FEED (override reference velocity) specifies whether the spindle override

- refers to the speed limited by MD or SD, or
- to the programmed speed.

2.4.3 Programmable feed override

Function

The programmable feed override can be used to change the velocity level of path and positioning axes by means of a command in the parts program.

A separate feed override can be programmed for positioning axes.

Programming

The feed override can be changed with the following commands:

OVR=	Feed change for path feed F
OVRA[X1]= ...	Feed change for positioning feed FA

The programmable range is from 0 to 200%.
The standard setting = 100%.

Activation

The “Rapid traverse/feed override active” (DB21, ... DBX6.6) or IS “Override active” (DB31, ... DBX1.7) interface signals do not relate to the programmable feed override. The programmable feed override remains active when these signals are deactivated.

The active override is calculated from the product of the programmable feed override and the feed override from the machine control panel.

The standard setting is 100%. It applies when no feed override has been programmed, and after reset if MD 22410: F_VALUES_ACTIVE_AFTER_RESET (F function active via reset) is not enabled.

OVR is also active for rapid traverse (G0). However, it is automatically limited to a maximum of 100%.

OVR is not active with G33, G34, G35.

© Siemens AG, 2003. All rights reserved

2.4.4 Dry run feed

Application

The dry run feed is used when testing parts programs without machining the workpiece in order to allow the program or program sections to execute with an increased path feed, for example.

"Dry run feedrate selected" DB21, ... DBX24.6

The dry run feed can be activated from the PLC or operator panel.
When activated from the operator panel, the "Dry run feed selected" interface signal (DB21, ... DBX24.6) is enabled and transmitted by the PLC basic program to the "Activate dry run feed" interface signal (DB21, ... DBX0.6).

When selected on the PLC, the "Activate dry run feed" interface signal is required to be set by the PLC user program.

The execution of the program is triggered with G94.

The dry run feed also take precedence over the feeds for G93, G95 and G33. In this case, the programmed feedrate is compared to the dry run feedrate in SD 42100: DRY_RUN_FEED and the axis then traversed at the higher of the two feedrates. Standard solution up to SW 5 and higher.

Dry run feed change

The dry run feed (SD 42100: DRY_RUN_FEED) can be changed on the operator panel in the "Parameter" user area.
No changes can be made after the program has started.

Evaluation

The effect of SD 42100: DRY_RUN_FEED can be controlled by another setting data SD 42101: DRY_RUN_FEED_MODE in SW 6 and higher.

SD 42101: DRY_RUN_ FEED_ MODE	Activation As long as interface signal "Activate dry run feedrate" is active, the feedrate value set in SD 42100: DRY_RUN_FEED is operative instead of the programmed feedrate in the following way:
0	The maximum value from SD 42100: DRY_RUN_FEED and the programmed velocity is operative. This corresponds to the default setting and the system response in SW 5 and lower.
1	The minimum value from SD 42100: DRY_RUN_FEED and the programmed velocity is operative. (SW 6 and higher.)
2	Setting data SD 42100: DRY_RUN_FEED is directly operative, irrespective of the programmed velocity. (SW 6 and higher.)
3 – 9	reserved
10	As for configured value 0 except for thread cutting (G33, G34, G35) and tapping (G331, G332, G63). These functions are executed as programmed.
11	As for configured value 1 except for thread cutting (G33, G34, G35) and tapping (G331, G332, G63). These functions are executed as programmed.
12	As for configured value 2 except for thread cutting (G33, G34, G35) and tapping (G331, G332, G63). These functions are executed as programmed.

© Siemens AG, 2003. All rights reserved

A dry run feed can be selected in the automatic modes and activated on interruption of an automatic mode or end of a block.

SW 4.2 and higher Previously, a dry run feedrate could only be activated by interrupting Automatic mode or at the end of a block. If MD 10704: DRYRUN_MASK = 1 is enabled, the dry run feedrate can also be activated during execution of the program (in the parts program block).

Note

Activation during machining triggers an internal reorganization task on the control which causes the axes to be stopped for a short time. This can affect the surface finish of the workpiece being machined.

2.4.5 Multiple feed values in one block

Application The functionality described in the following is used primarily for grinding, but is not restricted to it.

References: /FB/, W4, Grinding
/FB/, P5, Oscillation

Functionality The function “Multiple feeds in one block” can be used to activate **6** different **feed values** of an NC block, a **dwell time** or a **retraction** motion synchronously depending on the external digital and/or analog inputs.

Retraction Retraction is initiated by an amount defined before in the IPO cycle.

Signals The HW input signals are combined in one input byte for the function “Multiple feed in one block”. A permanent functional assignment applies within the byte:

Table 2-1 Input byte for “Multiple feeds in one block”

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Input no.	I7	I6	E5	I4	I3	I2	E1	I0
Feed address	F7	F6	F5	F4	F3	F2	ST	SR

I7 to I2 Activate feeds F7 to F2
I1 Activate the dwell time ST (in seconds)
I0 Activate the retraction motion SR

© Siemens AG, 2003. All rights reserved

Priorities

The signals are polled in ascending order, starting at I0.
In this way, the retraction motion (SR) has the highest priority, the feed F7 the lowest.

SR and ST end the feed motions that were activated with F2 – F7. SR also terminates ST, i.e. the complete function.

The highest priority signal determines the current feed. MD 21230: MULTFEED_STORE_MASK (input signals for the function store “Multiple feeds in one block”) can be used to define the response on loss of the highest priority input in each case (F2 – F7).

The end-of-block criterion is satisfied when:

- the programmed end position is reached
- the retraction motion ends (SR)
- the dwell time elapses (ST)

Delete distance-to-go

A retraction motion or dwell time causes the distancetogo-to-be deleted.

Hardware assignment

The above input byte can be assigned a maximum of two digital input bytes or comparator input bytes of the NCK I/Os using the channel-specific MD 21220: MULTFEED_ASSIGN_FASTIN (assignment of the input byte of the NCK I/Os for “Multiple feeds in one block”).

MD 21220: MULTFEED_ASSIGN_FASTIN can still be used to invert the input bits.

If a 2nd byte is entered, the contents of the 1st and 2nd byte are ORed before being used.

© Siemens AG, 2003. All rights reserved

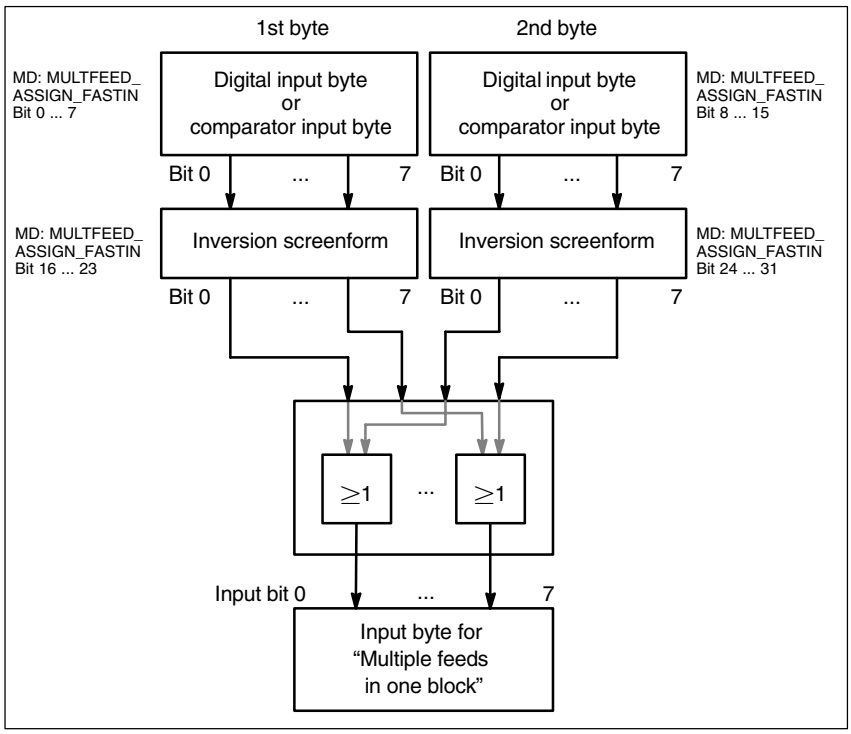


Fig. 2-6 Signal routing for "Multiple feeds in one block"

The routing of the digital input bytes and the parameterizing of the comparators are described in

References: /FB/, A2 "Various Interface Signals"

Programming path motion

The path feed is programmed under the address F and remains valid until an input signal is present. This value acts modally.

F2= ... to F7= ... can be used in addition to the path feed to program up to 6 further feeds in the block.

The numerical expansion indicates the bit number of the input that activates the feed when changed.

E.g. F7=1000 ; 7 corresponds to input bit 7

Bits 2 to 7 are permissible for numerical expansion of the feed.The programmed values act **non-modally**. The path feed programmed under F applies in the next block.

Dwell (sparking out time) and retraction path are programmed under separate addresses in the block:

ST=... dwell (for grinding sparking out time)
SR=... retraction path

These addresses apply non-modally.

Programming axial motion

The axial feeds are programmed under address FA and remain valid until an input signal is present. They act modally.

© Siemens AG, 2003. All rights reserved

FMA[2,x]= ... to FMA[7,x]= ... can be used to program up to 6 further feeds per axis in the block.

The first expression in square brackets indicates the bit number of the input that activates the feed when changed. The second expression indicates the axis to which the feed applies.

E.g. FMA[3,Y]=1000 ; Axial feed for Y axis,
; 3 corresponds to input bit 3

Bit 2 to 7 are permissible for the numerical expansion of the axial feed.The values programmed under FMA are active nonmodally. The feed programmed under FA applies to the next block.

Dwell (sparking out time) and retraction path can also be defined for a single axis:

STA[x]= ... Axis-specific dwell (sparking out time)
SRA[x]=... Axis-specific retraction path

The expression in square brackets indicates the axis for which the sparking out time and retraction path apply.

STA[X]=2.5 ;Sparking out time is 2.5 seconds
SRA[X]=3.5 ;Retraction path is 3.5 (units e.g.: mm)

These addresses apply nonmodally.

If a sparking out time (dwell) or retraction path is programmed for an axis feed on account of an external input, this axis must not be programmed as a PSOA axis in this block (positioning axis beyond end of block).

When the input for the sparking out time or retraction path is activated, the distancetogo for the path axes or the particular single axis is deleted and the dwell or retraction is started.

Note

The unit for the retraction path refers to the current valid unit of measurement (mm or inch).

The reverse stroke is always made in the opposite direction to the current motion. SR/SA always programs the value for the reverse stroke. No sign is programmed.

It is also possible to poll the status of an input for synchronous commands of various axes.

Look Ahead is also active for multiple feeds in one block. In this way, the current feed is restricted by the Look Ahead value.

© Siemens AG, 2003. All rights reserved

Application

Typical applications are, for example:

- Analog or digital calipers
  Depending on whether the external inputs are analog or digital, various feedrate values, a dwell and a retraction path can be activated. The limit values are defined via the setting data.
- Switching from infeed to working feed via proximity switch

Example

Internal grinding of a conical ring, where the actual diameter is determined using calipers and, depending on the limits, the feed value required for roughing, finishing or fine finishing is activated. The position of the calipers also provides the end position. Thus, the block end criterion is determined not only by the programmed axis position of the infeed axis but also by the calipers.

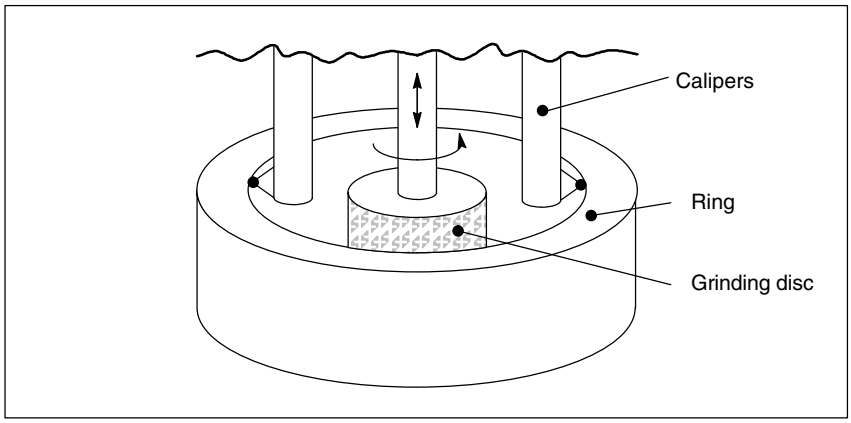
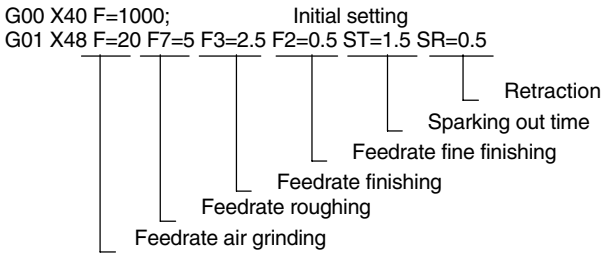


Fig. 2-7 Calipers

Parts programs



© Siemens AG, 2003. All rights reserved

Note

The axial feed/path feed (F value) is the 100% feed. The “Multiple feed values in one block” interface signal (F2 to F7 value) can be used to obtain feeds that **are less than or equal to** the axial feed/path feed.

The functionality “Multiple feeds in one block” is available only in conjunction with the Synchronized Actions function.

References: /FB/, S5, Synchronized Actions

2.4.6 Fixed feed values (840D, 810D)

Function

The machine data can be used to define 4 fixed feed values, which can be activated via the interface signal. The function is possible in the AUTOMATIC and JOG modes.

Response in AUTOMATIC mode

The fixed feed which has been selected is used instead of the programmed feed. The following MDs and interface signals can be used to select fixed feeds for path/geometry axes:

MD 12202: $MN_PERMANENT_FEED[n] (fixed feeds for linear axes)
MD 12204: $MN_PERMANENT_ROT_AX_FEED[n] (fixed feeds for rotary axes)
IS “Activate fixed feed for path/geometry axes” (DB21, ... DBX29.0– 29.3)

The contour travels at the activated fixed feed, instead of using the programmed feed.

Behavior in JOG mode

The fixed feed selected via the interface signal is used instead of the set JOG speed.
The direction of travel is specified using the interface signals.
Fixed feeds can be selected for path/geometry axes using the following interface signals.

MD 12202: $MN_PERMANENT_FEED[n] (fixed feeds for linear axes)
MD 12204: $MN_PERMANENT_ROT_AX_FEED[n] (fixed feeds for rotary axes)
IS “Activate fixed feed for machine axes” (DB31, ... DBX3.2–3.5)

The axis travels at the activated fixed feed, instead of at the set JOG/JOG rapid traverse speed.

© Siemens AG, 2003. All rights reserved

Supplementary conditions

- The fixed feed is not active for spindles, positioning axes and tapping.
- The override depends on MD 12200: $MN_RUN_OVERRIDE_0 (traverse with override 0) when traversing with fixed feed.
- When fixed feed is selected, it is not possible to activate DRF offset.
- The fixed feeds are always linear feedrate values. Revolutional feedrates are converted to linear feeds internally.

Interface signals

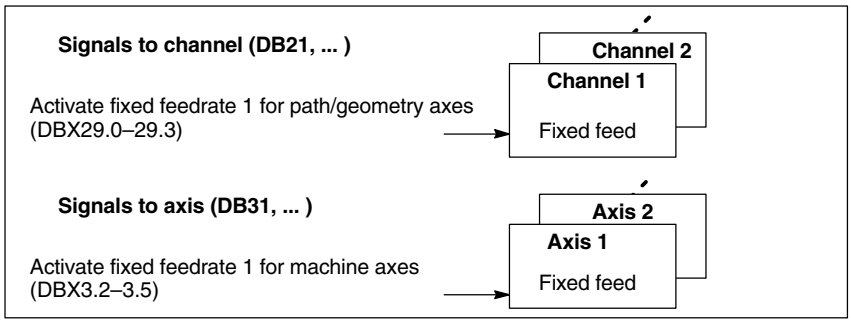


Fig. 2-8 Overview of interface signals for permanent feed (for more information, see Chapter 5)

2.4.7 Feed for chamfer/rounding FRC, FRCM (SW 5.2 and higher)

General

The machining conditions can change significantly during surface transitions to chamfer/rounding. The chamfer/rounding contour elements therefore need their own optimized feed values in order to achieve the required surface finish.

Up to SW 5.1, the feed for chamfer/rounding was taken from the preceding block or following block.

Function

You can program the feed for chamfer/rounding with

- FRC (non-modal) or
- FRCM (modal).

The feed value is interpreted according to the type of feed active:

- G94, G961, G971: Feedrate in mm/min, inch/min or °/min
- G95, G96, G97: Revolutional feedrate in mm/rev or inch/rev

© Siemens AG, 2003. All rights reserved

The FRC/FRCM value is taken according to MD 20201: CHFRND_MODE_MASK:

- Bit=0: FRC/FRCM from following block (default setting)
- Bit0 = 1: FRC/FRCM from preceding block (recommended setting).

Reason: The feed type (G94, G95, G96, G961 ...) and thus the conversion to the internal format must be consistent within the block for F and FRC/FRCM.

Empty blocks

When chamfer or rounding is active, the possible number of blocks containing no traversing information is limited. The maximum number is defined in MD 20200: CHFRND_MAXNUM_DUMMY_BLOCKS.

The possible blocks without traversing information in the compensation plane are pure dummy commands and are called empty blocks. For this reason, they may only be written between two blocks with traversing information.

Supplementary conditions

- Feed interpolation FLIN and FCUB is not possible for chamfer/rounding.
- FRC/FRCM has no effect if a chamfer is traversed with G0; it is possible to program this, in accordance with the F value, without receiving an error message.
- FRC is only active if a chamfer/rounding operation is also programmed in the block or if RNDM was activated.
- FRC overwrites the F or FRCM value for chamfer/rounding in the current block.
- Must be greater than zero.
- FRCM=0 activates the feedrate programmed in F for the chamfer/rounding.
- If FRCM is programmed, the FRCM value must be programmed again, equivalent to F, $\leftrightarrow$ on a G94, G95 change etc. If only F is reprogrammed and FRCM is $>$ 0, before the feed type is changed, error message 10860 appears (no feed programmed).

© Siemens AG, 2003. All rights reserved

2.4.8 Blockwise (nonmodal) feed FB (SW 5.3 and higher)

General

A separate feed can be specified for a single block with the command FB (Feed Block). For this block the previously active path feed is overwritten. After this block the previously active modal path feed becomes active once more.

Function

The blockwise feed is programmed with FB=<value>.

The feed value is interpreted according to the type of feed active:

- G94, G961, G971: Feedrate in mm/min, inch/min or °/min
- G95, G96, G97: Revolutional feedrate in mm/rev or inch/rev

Supplementary conditions

- The programmed value of FB= <value> must be greater than zero.
- If no traversing motion is programmed in the block (e.g.: computation block), the FB has no effect.
- If no explicit feed for chamfering/rounding is programmed, then the value of FB also applies for any contour element chamfering/rounding in this block.
- The path velocity profile programmed with FLIN or FCUB does not act with the rotational feed for G95 and with constant cutting speed for G96/G961 and G97/G971.
- Feed interpolations FLIN, FCUB, .. are possible without restriction.
- Simultaneous programming of FB and FD (handwheel travel with feed overlay) or F (modal path feed) is not possible.

© Siemens AG, 2003. All rights reserved

2.4.9 Programmable single-axis dynamic response (SW 5.1 and higher)

Single axes

Single axes can be programmed:

- **In the parts program**
  POS[Axis]= ... POSA[Axis]= ... SPOS [n] = ... SPOSA [Axis] = ...
  OS[Axis]= ... OSCILL[Axis]= ...
- **In synchronized actions** (command axes)
  EVERY ... DO POS[Axis]= ... SPOS [Spindle] = ... MOV[Axis]=...
- **In the PLC**
  FC15/ FC16 and FC18

Dynamic response of an axis/spindle

The dynamic response of an axis is dependent on

- **the preset feed value** (MD 32060: POS_AX_VELO)
  In parts program through FA[axis]= ... , or through
  percentage feedrate override OVRA[axis]= ...
  Programmable in synchronized actions with FA[axis]= ...
  Can be changed by the PLC through setting of FRate or overwriting of the axis override.
- **aceleration value** (MD 32300: MAX_AX_ACCEL)
  Programmable indirectly in the parts program through overwriting of the MD with subsequent newConfig, or directly through a percentage acceleration override ACC[axis]= ...
  In synchronized actions, indirectly by overwriting of the MD (no newConfig possible), or it can be modified by initiating an ASUP or it can be directly programmed via a percentage acceleration correction ACC[axis]= ... (cannot be specified from the PLC).
  The PLC has the same possibilities as the synchronized actions.
- **the acceleration pattern**
  Programmable in the parts program with BRISKA(axis), SOFTA(axis), DRIVEA(axis) and JERKA(axis).
  Not programmable in synchronized actions. (Only indirectly with ASUP)
  Cannot be preset by the PLC. (Only indirectly with ASUP)
- **the selected servo parameter set**
  A set of parameters describes the most important setting adjusting possibilities used in servo. It comprises the following axis machine data:

31050: DRIVE_AX_RATIO_DENOM[n]	Parameter block n for 0 to 5
31060: DRIVE_AX_RATIO_NUMERA[n]	Parameter block n for 0 to 5
32200: POSCTRL_GAIN[n]	Parameter block n for 0 to 5
32452: BACKLASH_FACTOR[n]	Parameter block n for 0 to 5
32610: VELO_FFW_WEIGHT[n]	Parameter block n for 0 to 5
32800: EQUIV_CURRCTRL_TIME[n]	Parameter block n for 0 to 5
32810: EQUIV_SPEEDCTRL_TIME[n]	Parameter block n for 0 to 5
32910: DYN_MATCH_TIME[n]	Parameter block n for 0 to 5
36012: STOP_LIMIT_FACTOR[n]	Parameter block n for 0 to 5
36200: AX_VELO_LIMIT[n]	Parameter block n for 0 to 5

For further information on parameter blocks and programming, refer to:
References: /FB/, G2, “Closed-Loop Control”
/PG/, Chapter 5 “Path response” and Chapter 7 “Feedrate control”

© Siemens AG, 2003. All rights reserved

Dynamic criteria and precontrol

With dynamic response criteria, it must be distinguished from where they were set:

- From the parts program or
- From a main run interpolation (synchronized action or PLC default)

Note

Dynamic response changes which are performed in a parts program, have no effect on command axis or PLC axis motions.

Changes from synchronized actions have no reverse effects on motions from a parts program.

Traverse with precontrol ON/OFF

The type of precontrol and which path axes are to be pre-controlled is defined:

- Programmable in the parts program with FFWON/ FFWOF for axes selected in the machine data.
- Only indirectly programmable in synchronized actions. (ASUP)
- Only indirectly programmable from the PLC. (ASUP)

Up to SW 5.2, a single axis movement is ended when “Exact stop FINE” is reached.

Percentage acceleration correction

ACC[axis]

With ACC[axis]= 0 .. 200, the aceleration defined in MD 32300: MAX_AX_ACCEL can be modified in a range from 0% – 200% in parts programs and synchronized actions.
ACC[axis]= <value>

Axis	Channel axis name (X, Y ....), spindle (S1, ...)
Value	Percentage of MD 32300: MAX_AX_ACCEL. (0 <= value <= 200)

The current acceleration value can be called with the system variables **$AA_ACC[axis]**. It is determined by:

$AA_ACC[Axis]= Content(MD 32300: MAX_AX_ACCEL[Axis]) * ACC[axis]/100

! **Important**

The aceleration offset programmed with ACC[] is always considered as specified above for the output in $AA_ACC. Output of $AA_ACC in the parts program takes place at a **different time** than outputting in a synchronized action.

The value described in the parts program is only considers in the system variables $AA_ACC as described in the parts program, if ACC was not changed in the meantine by a synchronized action.

The value described in the synchronized action is only considers in the system variables $AA_ACC as described in the synchronized action, if ACC was not changed in the meantine by a parts program.

© Siemens AG, 2003. All rights reserved

Main run axes

Main run axes (MR axes) are axes that are interpolated by the main run and can be:

- command axes (activated by synchronized actions)
- PLC axes (started by PLC via function block)
- asynchronous oscillating axes (setting data or from parts program)
- neutral axes

!

Important

Depending on whether $AA_ACC is programmed in a parts program or in a synchronized action, the ACC value is output for the NC axes or the mainrun axes.

Variable $AA_ACC must always be queried in the mode in which the acceleration was written (either parts program or synchroniyed action).

Examples

After RESET, the last programmed value remains.
In the parts program

```
...
N80 G01 POS[X]=100 FA[X]=1000 ACC[X]=90 IPOENDA[X]
...
```

or **via synchronized actions**

```
...
N100 EVERY $A_IN[1] DO POS[X]=50 FA[X]=2000 ACC[X]=140 IPOENDA[X]
...
```

Acceleration factor is written in the part program:

```
...
ACC[X]=50
RO $AA_ACC[X]

IF (RO <> $MA_ MAX_AX_ACCEL[X] * 0,5
SETAL(61000)
ENDIF
```

Acceleration factor is set by the synchronized action:

```
WHEN TRUE DO ACC[X]=25 $R1 = $AA_ACC[X]
G4 F1

IF (RO <> $MA_ MAX_AX_ACCEL[X] * 0.25
SETAL(61001)
ENDIF

M30
```

© Siemens AG, 2003. All rights reserved

Movement end criterion for single axes

Similar to the block change criterion for path interpolation (G601, G602 and G603) it is also not possible to program the movement end criterion for single-axis interpolation in a parts program or in synchronized actions for main run axes: Command/PLC axes

Programmable criterion:	End-of-motion on reaching
• **FINEA[axis]**	of “Exact stop fine”
• **COARSEA[axis]**	of “Exact stop coarse”
• **IPOENDA[axis]**	from “Interpolator Stop” (IPO-Stop)
Axis	Channel axis name (X, Y ....), spindle (S1, ...)

Depending on which movement end criterion is set, parts program blocks and technology cycle blocks with single-axis movements will be ended more or less quickly.
The same applies for PLC positioning instructions via FC15/16/18.

The set end of movement criterion can be scanned by system variable **$AA_MOTEND[axis]**.

$AA_MOTEND[axis] = 1	Movement end with “Exact stop fine”.
$AA_MOTEND[axis] = 2	Movement end with “Exact stop coarse”.
$AA_MOTEND[axis] = 3	Movement end with “IPO stop”.

!

Important

Depending on whether $AA_MOTEND is programmed in a parts program or a synchronized action, the MOTEND value is output for the NC axes or the mainrun axes.

After RESET, the last programmed value remains.

Example:

```
...
N80 G01 POS[X]=100 FA[X]=1000 ACC[X]=90 IPOENDA[X]
...
```

or via synchronized actions

```
...
N100 EVERY $A_IN[1] DO POS[X]=50 FA[X]=2000 ACC[X]=140 IPOENDA[X]
```

Note

For further information on the block change and endofmotion criteria with FINEA, COARESA and IPOENDA, please refer to:

References: /FB/, P2, “Positioning Axes”, block change

Programmable servo parameter set

SCPARA[axis] = ...

SCPARA[axis]= ... can be used to program the parameter set (consisting of MDs) in the parts program and in synchronized actions (formerly only via PLC).

© Siemens AG, 2003. All rights reserved

SCPARA[Axis]=<value>

axis	Channel axis name (X, Y ....), spindle (S1, ...)
value	Desired parameter set (1<= value <=6)

DB3n DBB9 bit3
To prevent conflicts between PLC and NC user requests, a further bit is defined at the PLC–>NCK interface:
DB3n DBB9 bit3 “Parameter set selection by SCPARA disabled”.

The PLC user is thus able to arrange orderly processing when using both PLC parameter set switching and selections from synchronized actions or parts programs.
When Bit 3 is changed (0=>1 or 1=>0) the selection is stored in Bit 0–2.

Note

Up to SW 5.2, the servo parameter set can only be selected through the PLC (DB3n DBB9 bits 0–2). With G33, G34, G35, G331 or G332, the parameter set for the axes is selected by the control system according to the gear stage of the master spindle.

Important

If parameter set selection via SCPARA is disabled, there is no error message if the latter is programmed nevertheless.

The current parameter set can be scanned by system variable **$AA_SCPAR[axis]**.

Example

```
...
N100 SCPARA[X]=3;          The 3rd parameter set is selected for the
                           X axis
...
```

© Siemens AG, 2003. All rights reserved

Supplementary conditions

- With **different movement end criteria** parts program blocks will be ended more or less quickly. This can lead to side effects for technology cycles and PLC user parts.
- If the **servo parameter set** is to be **changed** both in a parts program or synchronized action and in the PLC, the PLC user program must be extended.
- After POWER ON the following **basic setting values** are set:
  - Percentage acceleration correction for all single-axis interpolations 100%
  - Movement end criterion for all single-axis interpolations FINEA
  - Servo parameter set from the NC 1
- When the **operating mode is changed** from AUTO => JOG, the programmed dynamic response changes remain valid.
- In case of **RESET,** the last programmed value remains for the parts program specifications. The presets for the mainrun interpolations are not changed.
- **Block search:**
  The last movement end criterion programmed for an axis is collected and output in an action block. The last block with a programmed movement end criterion processed by the block search serves as a container for all programmed movement end criteria of all axes.

  Example:
  ```
  N01 G01 POS[X]=20 POS[Y]=30 IPOENDA[X] IPOENDA[Y]
  N02 POS[Z]=55 FINEA[Z]
  N03 $A_OUT[1]=1
  N04 POS[X]=100 COARSEA[X]
  N05 ... ...
  TARGET:                    ; Block search destination
  ```

  In this example N04 serves as a container for all programmed movement end criteria. Two action blocks are saved. The first action block outputs the digital output (N03) and the second sets COARSEA for the X axis, IPOENDA for the Y axis and FINEA for the Z axis.
  The same applies for the programmed servo parameter set. The last programmed acceleration override is effective from the first approach block.

■

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

Several feeds in one block

The function “Several feeds in one block is only available in combination with the function Synchronized actions.
References: /FB/, S5, /FBSY/, Synchronized Actions

Feed interpolation

This function is also available for SINUMERIK 810DE (CCU1) in SW 3.2 and higher.

Inverse time feedrate

This function is also available for SINUMERIK 810DE (CCU1) in SW 3.2 and higher.

4 Data Descriptions (MD, SD)

4.1 General machine data

10704 MD number	**DRYRUN_MASK** Activating the dry run feedrate		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies from SW 4.2		
Significance:	If DRYRUN_MASK = 1 is enabled, the dry run feedrate can also be activated during execution of the program (in the parts program block).		
Related to	SD 42100: DRY_RUN_FEED		

© Siemens AG, 2003. All rights reserved

12000 MD number	**OVR_AX_IS_GRAY_CODE** Path feed override switch Graycoded		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	This machine data is used to adapt the axis feed override switch to the interface coding of the PLC interface. 1: The 5 least significant bits of the “feed override” interface signal (DB31, ... DBB0) are interpreted as a Gray code. The value which is read corresponds to a switch setting. It is used as a pointer for selection of the correct override factor from the table of MD 12010: OVR_FACTOR_AX_SPEED [n] 0: The feed override byte of the PLC interface is interpreted as a binary 8bit representation of the override value in percent.		
Related to	IS “Feedrate override” (DB31, ... DBB0), (axis-specific) MD 12010: OVR_FACTOR_AX_SPEED [n] (evaluation of axis feed override switch)		

© Siemens AG, 2003. All rights reserved

12010 MD number	**OVR_FACTOR_AX_SPEED** [**n**] Evaluation of the axis feed override switch		
Default setting: see table	Minimum input limit: 0.00	Maximum input limit: 2.00	
Changes effective after POWER ON	Protection level: 2	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		

Significance:

The factors for evaluating the individual settings on the axis feed override switch on a Gray-coded interface are required to be entered in this machine data.
The factors should be entered in the order of the switch settings (1st factor → 1st keyswitch position, 2nd factor → 2nd switch setting, etc.).
An axis feed override switch with up to 31 switch settings can be used. Any factors can be assigned, however it is recommended to assign zero to the first switch setting.
It is recommended to initialize the 1st switch setting with zero.

Switch settings	Axial feed override factor (standard values)	Index[n]
1	0.0	0
2	0.01	1
3	0.02	2
4	0.04	3
5	0.06	4
6	0.08	5
7	0.10	6
8	0.20	7
9	0.30	8
10	0.40	9
11	0.50	10
12	0.60	11
13	0.70	12
14	0.75	13
15	0.80	14
16	0.85	15
17	0.90	16
18	0.95	17
19	1.00	18
20	1.05	19
21	1.10	20
22	1.15	21
23	1.20	22
24	1.20	23
25	1.20	24
26	1.20	25
27	1.20	26
28	1.20	27
29	1.20	28
30	1.20	29
31	1.20	30

Index [n] of the machine data is coded as follows:
[switch setting]: 0–30

MD irrelevant for ...	MD 12000: OVR_AX_IS_GRAY_CODE = 0
Related to	IS “Feedrate override” (DB31, ... DBB0), (axis-specific)

© Siemens AG, 2003. All rights reserved

12030 MD number	**OVR_FACTOR_FEEDRATE[n]** Evaluation of the path feed override switch		
Default setting: see table	Minimum input limit: 0.00	Maximum input limit: 2.00	
Changes effective after POWER ON	Protection level: 2	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		

Significance:

The factors for evaluating the individual settings on the path feed override switch on a Gray-coded interface are required to be entered in this machine data.
The factors should be entered in the order of the switch settings (1st factor → 1st keyswitch position, 2nd factor → 2nd switch setting, etc.).
A path feed override switch with up to 31 switch settings can be used. Any factors can be assigned, however it is recommended to assign zero to the first switch setting.
It is recommended to initialize the 1st switch setting with zero.

Switch settings	Feed override factor (standard values)	Index[n]
1	0.0	0
2	0.01	1
3	0.02	2
4	0.04	3
5	0.06	4
6	0.08	5
7	0.10	6
8	0.20	7
9	0.30	8
10	0.40	9
11	0.50	10
12	0.60	11
13	0.70	12
14	0.75	13
15	0.80	14
16	0.85	15
17	0.90	16
18	0.95	17
19	1.00	18
20	1.05	19
21	1.10	20
22	1.15	21
23	1.20	22
24	1.20	23
25	1.20	24
26	1.20	25
27	1.20	26
28	1.20	27
29	1.20	28
30	1.20	29
31	1.20	30

The 31st switch setting has a special function for the velocity control:
The setting for the 31st override factor defines the dynamic reserve which the velocity control maintains for increasing the path feed.
The setting should correspond to the highest override factor actually used.
The function of the 31st value is identical to the effect of MD 12100: OVR_FACTOR_LIMIT_BIN
(limit for binary-coded override switch) when a binary-coded interface is used.
References: /FB/, B1, “Continuous-Path Mode, Exact Stop and Look Ahead”

Coding of machine data index [n]:
[switch setting]: 0–30

MD irrelevant for ...	MD 12020: OVR_FEED_IS_GRAY_CODE = 0
Related to	IS “Feedrate override” (DB21, ... DBB4)

© Siemens AG, 2003. All rights reserved

12050 MD number	**OVR_FACTOR_RAPID_TRA[n]** Evaluation of the rapid traverse override switch		
Default setting: see table	Minimum input limit: 0.00	Maximum input limit: 1.00	
Changes effective after POWER ON	Protection level: 2	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		

Significance:

The factors for evaluating the individual settings on the rapid traverse override switch on a Gray-coded interface are required to be entered in this machine data.The factors should be entered in the order of the switch settings (1st factor →1st switch setting, 2nd factor → 2nd switch setting, etc.).
A rapid traverse override switch with up to 31 switch settings can be used. Any factors can be assigned, however it is recommended to assign zero to the first switch setting.
It is recommended to initialize the 1st zero to the first switch setting.

Switch settings	Rapid traverse override (standard values)	Index[n]
1	0.0	0
2	0.01	1
3	0.02	2
4	0.04	3
5	0.06	4
6	0.08	5
7	0.10	6
8	0.20	7
9	0.30	8
10	0.40	9
11	0.50	10
12	0.60	11
13	0.70	12
14	0.75	13
15	0.80	14
16	0.85	15
17	0.90	16
18	0.95	17
19	1.00	18
20	1.00	19
21	1.00	20
22	1.00	21
23	1.00	22
24	1.00	23
25	1.00	24
26	1.00	25
27	1.00	26
28	1.00	27
29	1.00	28
30	1.00	29
31	1.00	30

Coding of machine data index [n]:
[switch setting]: 0–30

MD irrelevant for ...	MD 12040: OVR_RAPID_IS_GRAY_CODE = 0
Related to	IS “Rapid traverse override” (DB21, ... DBB5)

© Siemens AG, 2003. All rights reserved

12070 MD number	**OVR_FACTOR_SPIND_SPEED[n]** Evaluation of the spindle override switch		
Default setting: see table	Minimum input limit: 0.00	Maximum input limit: 2.00	
Changes effective after POWER ON	Protection level: 2	Unit: factor	
Data type: DOUBLE	Applies from SW 1.1		

Significance:

The factors for evaluating the individual settings on the spindle override switch on a Gray-coded interface are required to be entered in this machine data.
The factors should be entered in the order of the switch settings (1st factor → 1st Switch position, 2nd factor → 2nd switch setting, etc.).
A spindle override switch with up to 31 switch settings can be used. Any factors can be assigned, however it is recommended to assign zero to the first switch setting.

Switch settings	Spindle override factor (standard values)	Index[n]
1	0.50	0
2	0.55	1
3	0.60	2
4	0.65	3
5	0.70	4
6	0.75	5
7	0.80	6
8	0.85	7
9	0.90	8
10	0.95	9
11	1.00	10
12	1.05	11
13	1.10	12
14	1.15	13
15	1.20	14
16	1.20	15
17	1.20	16
18	1.20	17
19	1.20	18
20	1.20	19
21	1.20	20
22	1.20	21
23	1.20	22
24	1.20	23
25	1.20	24
26	1.20	25
27	1.20	26
28	1.20	27
29	1.20	28
30	1.20	29
31	1.20	30

The 31st switch setting has a special function for the velocity control:
The setting for the 31st override factor defines the dynamic reserve which the velocity control maintains for increasing the spindle feed. The setting should correspond to the highest override factor actually used. The function of the 31st value is identical to the effect of the MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binarycoded override switch) when a binarycoded interface is used.
References: /FB/, B1, “Continuous-Path Mode, Exact Stop and Look Ahead”
Coding of machine data index [n]:
[switch setting]: 0–30

MD irrelevant for ...	MD 12060: OVR_SPIND__IS_GRAY_CODE = 0
Related to	IS “Spindle override” (DB31, ... DBB0)

© Siemens AG, 2003. All rights reserved

12020 MD number	OVR_FEED_IS_GRAY_CODE Path feed override switch Graycoded		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	This machine data is used to adapt the path feed override switch to the interface coding of the PLC interface. 1: The 5 least significant bits of the “feed override” interface signal are interpreted as a Gray code. The value which is read corresponds to a switch setting. It is used as a pointer for selection of the correct override factor from the table of MD 12030: OVR_FACTOR_FEEDRATE [n]. 0: The feed override byte of the PLC interface is interpreted as a binary 8bit representation of the override value in percent.		
Related to	IS “feedrate override” (DB21, ... DBB4) MD 12030: OVR_FACTOR_FEEDRATE [n] (evaluation of path feed override switch)		

12100 MD number	OVR_FACTOR_LIMIT_BIN Limit for binary-coded override switch		
Default setting: 1.2	Minimum input limit: 0.00	Maximum input limit: 2.00	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	This machine data can be used as an additional limit for the override factor when using the binary-coded interface for path, axis and spindle feed. The maximum values • 200% for channel-specific feed override • 100% for channel-specific rapid traverse override • 200% for axis-specific feed override • 200% for spindle override are replaced with the limit value entered in MD: OVR_FACTOR_LIMIT_BIN, if this value is lower. Example: OVR_FACTOR_LIMIT_BIN = 1.20 ⇒ Maximum override factor for • channel-specific feed override =120% • channel-specific rapid traverse override =100% • axis-specific feed override =120% • spindle override =120% This value also defines the dynamic reserve maintained by the velocity control for increasing the path and spindle feedrate. **References:** /FB/, B1, “Continuous-Path Mode, Exact Stop and Look Ahead”		

© Siemens AG, 2003. All rights reserved

12040 MD number	OVR_RAPID_IS_GRAY_CODE Rapid traverse override switch Gray-coded		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	This machine data is used to adapt the rapid traverse override switch to the interface coding of the PLC interface. 1: The 5 least significant bits of the “Rapid traverse override” PLC interface signal are interpreted as a Gray code. The value which is read corresponds to a switch setting It is used as a pointer for selec ting the valid correction factor from the Table of MD 12050: OVR_FACTOR_RAPID_TRA. [n] 0: The rapid traverse override byte of the PLC interface is interpreted as a binary 8bit of the override value in percent.		
Related to	IS “Rapid traverse override” (DB21, ... DBB5) MD 12050: OVR_FACTOR_RAPID_TRA[n] (evaluation of rapid traverse override switch)		

12060 MD number	OVR_SPIND_IS_GRAY_CODE Spindle override switch Gray-coded		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	This machine data is used to adapt the path feed override switch to the interface coding of the PLC interface. 1: The 5 least significant bits of the “Spindle override” interface signal are interpreted as a Gray code. The value which is read corresponds to a switch setting. It Itis used as a pointer for the selection of the correct override factor from the table of MD 12070: OVR_FACTOR_SPIND_SPEED [n]. 0: The spindle override byte of the PLC interface is interpreted as a binary 8bit of the override value in percent.		
Related to	IS “Spindle override” (DB31, ... DBB0) MD 12070: OVR_FACTOR_SPIND_SPEED [n] (evaluation of spindle override switch)		

12080 MD number	OVR_REFERENCE_IS_PROG_FEED Override reference velocity		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/4	Unit: –	
Data type: BOOLEAN	Applies from SW 2.1		
Significance:	In this MD, it is entered whether the spindle override given by the interface signal refers to the speed limited by MD/SD or to the programmed speed. 1: Spindle override acts with reference to the programmed speed (programmed speed ≙ Spindle correction 100%) 0: Spindle override acts on the speed limited by MD or SD (limited speed by MD/SD ≙ spindle override 100%)		
Related to	A speed limitation is effected by the following MDs or SDs: MD 35100: SPIND_VELO_LIMIT Maximum spindle speed MD 35130: GEAR_STEP_MAX_VELO_LIMIT Maximum speed of gear step MD 35160: SPIND_EXTERN_VELO_LIMIT Spindle speed limitation by PLC SD 43220: SPIND_MAX_VELO_G26 Maximum spindle speed SD 43230: SPIND_MAX_VELO_LIMS Spindle speed limitation with G96		

© Siemens AG, 2003. All rights reserved

12082 MD number	**OVR_REFERENCE_IS_MIN_FEED** Defining the reference for the path override		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Change effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 4.1		
Significance:	The reference speed for the path feed override specified via the machine control panel can be set differently from the standard. 0: Default settings: the override is relative to the programmed feed. 1: Special case: The override is relative to the programmed feed or to the path feed limit, depending on which resulting value is lower. In this way, even for a great feed reduction (due to the permissible axis dynamics), the effect of the override value (in the range 0 to 100%) is always visible.		

12200 MD number	**RUN_OVERRIDE_0** Traversing with override 0		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 7 / 2	Unit: –	
Data type: BOOLEAN	Applies from SW 4.1		
Significance:	**= 0** Override 0 **is active** and means brake (conventional mode, safety function) When using handwheels, bit 0 and bit 1 are specified for geometry and contour handwheel via $MA_HANDWH_STOP_COND for machine axis and $MC_HANDW_CHAN_STOP_COND to determine whether the pulses are collected. **= 1** Override 0 **is not active**, i.e. traversing with handwheels and, in JOG mode, with fixed feeds is also possible for 0% override.		
Related to	$MA_HANDWH_STOP_COND $MC_HANDW_CHAN_STOP_COND		

12202 MD number	**PERMANENT_FEED**[**n**] Fixed feeds for linear axes		
Default setting:	Minimum input limit:	Maximum input limit:	
Changes effective after RESET	Protection level: 7 / 2	Unit: mm/min, Inch/min	
Data type: DOUBLE	Applies from SW	840D SW4.1 810D SW2.1	
Significance:	In **AUTOMATIC** mode: When a fixed feed is activated via the interface signal, the fixed feed is used instead of the programmed feedrate. In **JOG** mode: When a fixed feed is activated via the interface signal and a travel key is used for the linear axis, traversing is performed with the fixed feed in the selected direction. n = 0, 1, 2, 3 means fixed feed 1, 2, 3, 4		
Special cases, errors, ...	The maximum velocity defined by $MA_MAX_AX_VELO is active. The effect of the override depends on $MN_RUN_OVERRIDE_0.		
Related to	$MN_RUN_OVERRIDE_0		

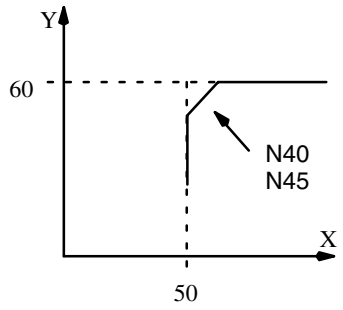
© Siemens AG, 2003. All rights reserved

12204 MD number	**PERMANENT_ROT_AX_FEED**[**n**] Fixed feeds for rotary axes		
Default setting:	Minimum input limit:	Maximum input limit:	
Changes effective after POWER ON	Protection level: 7/2	Unit: deg/min	
Data type: DOUBLE	Applies from SW	840D SW4.1 810D SW2.1	
Significance:	In **AUTOMATIC** mode: When a fixed feed is activated via the interface signal, the fixed feed is used instead of the programmed feedrate. In **JOG** mode: When a fixed feed is activated via the interface signal and a travel key is used for the linear axis, traversing is performed with the fixed feed in the selected direction. n = 0, 1, 2, 3 means fixed feed 1, 2, 3, 4		
Special cases, errors, ...	The maximum velocity defined by $MA_MAX_AX_VELO is active. The effect of the override depends on $MN_RUN_OVERRIDE_0.		
Related to	$MN_RUN_OVERRIDE_0		

© Siemens AG, 2003. All rights reserved

4.2 Channel-specific machine data

20172 MD number	**COMPRESS_VELO_TOL** Maximum permissible path feed for compression		
Default setting: 1000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: mm/min	
Data type: DOUBLE	Applies from SW 3.2		
Significance:	The value indicates the maximum permissible deviation for the compression for the path feed. The larger the value, the more short blocks can be compressed into a long block. The maximum number of compressible blocks is limited by the size of the spline buffer.		
Related to	$MA_COMPRESS_POS_TOL[AXn] $MC_COMPRESS_BLOCK_PATH_LIMIT		
References	Programming Guide		

20200 MD number	**CHFRND_MAXNUM_DUMMY_BLOCKS** Dummy blocks for chamfer/rounding		
Default setting: 3	Minimum input limit: 0	Maximum input limit: 15	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	Specifies the maximum number of blocks without traversing information in the correction plane (dummy blocks) that may be present with active chamfer/rounding between two block with traversing information.		
Application example(s)	 . . . N30 X50 N35 Y60 B5 N40 M62 N45 H13 N50 X110 } 2 blocks without traversing information		

20201 MD number	**CHFRND_MODE_MASK** Properties for chamfer/rounding		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 5.2		
Significance:	Specifies properties for chamfer/rounding. The MD is coded as follows: Bit 0: = 0: Take feed from following block (default value) = 1: Take feed from preceding block (recommended value) Bits 1 – 7: Not assigned		
Application examples	see Section 6.1		

© Siemens AG, 2003. All rights reserved

20750 MD number	**ALLOW_G0_IN_G96** G0 logic in G96		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	This machine data is used to define the speed response of the spindle in G0 blocks when a constant cutting speed (G96) is selected. 1: In a G0 block, the spindle speed is kept constant at the last value of the previous block which was not equal to G0. Before a succeeding block which does not contain G0, the spindle speed is accelerated to a value belonging to the facing axis position of the next block. 0: In a G0 block, the spindle speed changes according to the facing axis position.		
MD irrelevant for ...	Feed type $\neq$ G96		

21220 MD number	**MULTFEED_ASSIGN_FASTIN** Assignment of the input byte of the NCK I/Os for “Multiple feeds in one block”		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2/7	Unit: HEX	
Data type: DWORD	Applies from SW 2.1		
Significance:	The MD: MULTFEED_ASSIGN_FASTIN (assignment of the input bytes of the NCK I/Os for “Multiple feeds in one block”), at most two digital input bytes or comparator input byte of the NCK I/Os can be assigned to the input byte for the “Multiple feeds in one block” function. The assigned input signals can still be inverted with the machine data. The MD is coded as follows: Bit 0 – 7: No. of 1st digital input byte used or comparator input byte Bit 8 – 15: No. of 2nd digital input byte used or comparator input byte Bit 16 – 23: Inversion screenform for describing the 1st byte Bit 24 – 31: Inversion screenform for describing the 2nd byte Bit=0: do not invert Bit=1: invert If a 2nd byte is entered, the contents of the 1st and 2 bytes are ORed before being used. The number for the digital inputs should be specified as follows: 1: For the onboard byte 2 – 5: For external bytes The number for a comparator input byte should be specified as follows: 128: For comparator 1 (corresponds to 80Hex) 129: For comparator 2 (corresponds to 81Hex)		
Application example(s)	For the function “Multiple feeds in one block”, the external digital input byte 3 is to be used as the first byte and the input byte of comparator 2 as the second byte. Inversion is also to be carried out: Bit 0, 2, 3 of the digital input byte Bit 0, 1, 5, 7 of the comparator input byte $\Rightarrow$ MD: MULTFEED_ASSIGN_FASTIN=A30D8103 (in hexadecimal format)		

© Siemens AG, 2003. All rights reserved

21230 MD number	**MULTFEED_STORE_MASK** Storing input signals for the function “Multiple feeds in one block”		
Default setting: 0	Minimum input limit: ***	Maximum input limit: ***	
Changes effective after POWER ON	Protection level: 2/7	Unit: HEX	
Data type: BYTE	Applies from SW 2.1		
Significance:	The priority of the signals for feeds F2 – F7 of the function “Multiple feeds in one block” decreases as the bit number increases in the input byte. The highest priority signal determines the current feed. The MD: MULTFEED_STORE_MASK (store input signals of the function “Multiple feeds in one block”) can be used to specify the response when the highest priority input drops out: set bit 2 – 7 has the effect that the associated feed (F2 to F7) that has been selected by the highest priority input signal in each case is retained, even if the input signal drops out and a lower priority is present. The MD is coded as follows: Bit 0 – 1: No significance Bit 2 – 7: Storage response of the feed signals Bit 8 – 31: Reserved		

22410 MD number	**F_VALUES_ACTIVE_AFTER_RESET** F function active after reset		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 1.1		
Significance:	1: The last programmed F, FA, OVR and OVRA values are also active after RESET. 0: The various values are set to the default values after reset.		
Related to	MD 22240: AUXFU_F_SYNC_TYPE Output time of the F functions		

4.3 Axis/spindle-specific machine data

34990 MD number	**ENC_ACTVAL_SMOOTH_TIME** Smoothing time constant for actual values		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0.5	
Changes effective after RESET	Protection level: 2 / 7	Unit: s	
Data type: DOUBLE	Applies from SW 6.3		
Significance:	If low-resolution encoders are being used, smoothed actual values can be applied for constant coupled path and axis motions. The larger the time constant, the better the smoothing of the actual values and the longer the overtravel. Smoothed actual values are used for: • Thread cutting with feed on G33, G34, G35 • Revolutional feedrate on G95, G96, G97, FRAPON • Display of actual position and actual speed/velocity		

© Siemens AG, 2003. All rights reserved

4.4 Channel-specific setting data

42010 SD number	**THREAD_RAMP_DISP[0,1]** Acceleration behavior of the feed axis when thread cutting		
Default setting: –1	Minimum input limit: –1	Maximum input limit: 999999	
Changes effective immediately	Protection level: 7 / 7	Unit: mm/inch	
Data type: DOUBLE	Applies from SW 5.1		
Significance:	The SD is effective for thread cutting with G33 (G34, G35). It has two elements, defining the behavior during acceleration of the thread axis (1st element) and overshoot during deceleration (2nd element). The values have the same attributes for thread runin and runout: –1: Starting/deceleration of the thread axis is performed with configured acceleration. Jerk according to current BRISK/SOFT programming. The behavior is compatible with the previous machine data MD 20650: THRED_START_IS_HARD = FALSE. 0: Abrupt starting/deceleration of feed axis in thread cutting. The behavior is compatible with the previous machine data MD 20650: THRED_START_IS_HARD = TRUE. >0: The maximum thread starting/deceleration distance is defined. The defined distance can cause an acceleration overload on the axis. The SD is written from the block when DITR (Displacement Thread Ramp) is programmed. The default values (–1) are enabled for both elements of the SD on NC Reset and end of parts program. **Example:** $SC_THREAD_RAMP_DISP[0]=2 Runin path 2 mm **Note:** The MD 10710: PROG_SD_RESET_SAVE_TAB can be set so that the value written by the parts program is retained after the reset.		
Special cases, errors, ...	The SD is described when programming DITS (index 0) and DITE (index 1) from the block. MD 20650: THREAD_START_IS_HARD is omitted and replaced by SD 42010:		
Related to	MD 10710: PROG_SD_RESET_SAVE_TAB With the DITS and DITE (Displacement Threat Start/End) parts program instructions.		

42100 SD number	**DRY_RUN_FEED** Dry run feed		
Default setting: 5000	Minimum input limit: 0	Maximum input limit: plus	
Changes effective immediately	Protection level: MMCMD 9220	Unit: mm/min	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The feedrate for the active dry run is required to be entered in this setting data. The setting data can be changed via the operator panel in the operating area “Parameters”. The entered dry run feed is always interpreted as a linear feed (G94). If the dry run feed is activated via the PLC interface, the dry run feed is used as the path feed instead of the programmed feed following a reset. If the programmed speed is larger than the speed stored here, the programmed speed is used for traversing. SW 5 and lower. In SW 6 and higher, the dry run feedrate corresponds to the setting in SD 42101.		
Application example(s)	Program testing		
Related to	IS “Activate dry run feedrate”, (DB21, ... DBX0.6) IS “Dry run feedrate selected” (DB21, ... DBX24.6) MD 10710 $MN_PROG_SD_RESET_SAVE_TAB SD 42101: DRY_RUN_FEED_MODE (SW 6 and higher)		

© Siemens AG, 2003. All rights reserved

42101 MD number	**DRY_RUN_FEED_MODE** Mode for test run velocity		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 12	
Changes effective IMMEDIATELY	Protection level: 7 / 7	Unit: –	
Data type: BYTE	Applies from SW 6		
Significance:	This data can be parameterized to select the mode of functioning of the test run velocity specified in setting data SD 42100: DRY_RUN_FEED. The following values can be selected: 0: The maximum value from setting data SD 42100: DRY_RUN_FEED and the programmed velocity is operative. This is the default setting and matches the system response in SW 5 and lower. 1: The minimum value from setting data SD 42100: DRY_RUN_FEED and the programmed velocity is operative. 2: Setting data SD 42100: DRY_RUN_FEED is directly operative irrespective of the programmed velocity. 3 – 9: reserved 10: As for configured value 0 except for thread cutting (G33, G34, G35) and tapping (G331, G332, G63). These functions are executed as programmed. 11: As for configured value 1 except for thread cutting (G33, G34, G35) and tapping (G331, G332, G63). These functions are executed as programmed. 12: As for configured value 1 except for thread cutting (G33, G34, G35) and tapping (G331, G332, G63). These functions are executed as programmed.		
MD irrelevant for ...	Deselected dry run feedrate		
Related to	SD 42100: DRY_RUN_FEED		

42110 SD number	**DEFAULT_FEED** Default value for path feed		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after POWER ON	Protection level: /	Unit: mm/min	
Data type: DOUBLE	Applies from SW 5.2		
Significance:	Default value for path feed rate. The evaluation of the setting data for start of a parts program considers the feedrate type active at this time, see MD 20150 GCODE_RESET_VALUE or MD 20154 EXTERN_GCODE_RESET_VALUE		
Application example(s)			
Related to			

© Siemens AG, 2003. All rights reserved

42600 MD number	**JOG_FEED_PER_REV_SOURCE** In the JOG mode, revolutional feedrate for geometry axes on which a frame with rotation acts		
Default setting: 0	Minimum input limit:	Maximum input limit:	
Changes effective after POWER ON	Protection level: 7	Unit:	
Data type: DWORD	Applies from SW 3		
Significance:	0= No revolutional feedrate active >0= Machine axis index of rotary axis/spindle, from which the revolutional feedrate is derived –1= Revolutional feedrate derived from the master spindle of the channel in which the axis/spindle is active		
Related to	SD 53000: ASSIGN_FEED_PER_REV_SOURCE (revolutional feedrate for position axes/spindles)		

43300 MD number	**ASSIGN_FEED_PER_REV_SOURCE** Revolutional feedrate for positioning axes/spindles		
Default setting: 0	Minimum input limit: 0	Maximum input limit:	
Changes effective after POWER ON	Protection level: 7	Unit:	
Data type: DWORD	Applies from SW 3		
Significance:	In SW 5 and higher, the MD no longer refers directly to a machine axis, but to the logical machine axis image. 0= No revolutional feedrate active >0= Logical machine axis index (see $MN_AXCONF_LOGIC_MACHAX_TAB) of the rotary axis/spindle from which the revolutional feedrate is derived. Up to and including SW 4: machine axis index. –1= Revolutional feedrate derived from the master spindle of the channel in which the axis/spindle is active		
Related to	SD 52011: JOG_FEED_PER_REV_SOURCE (in JOG mode revolutional feedrate for geometry axes on which a frame with rotation acts) MD 10710 $MN_PROG_SD_RESET_SAVE_TAB		

■

© Siemens AG, 2003. All rights reserved

5 Signal Descriptions

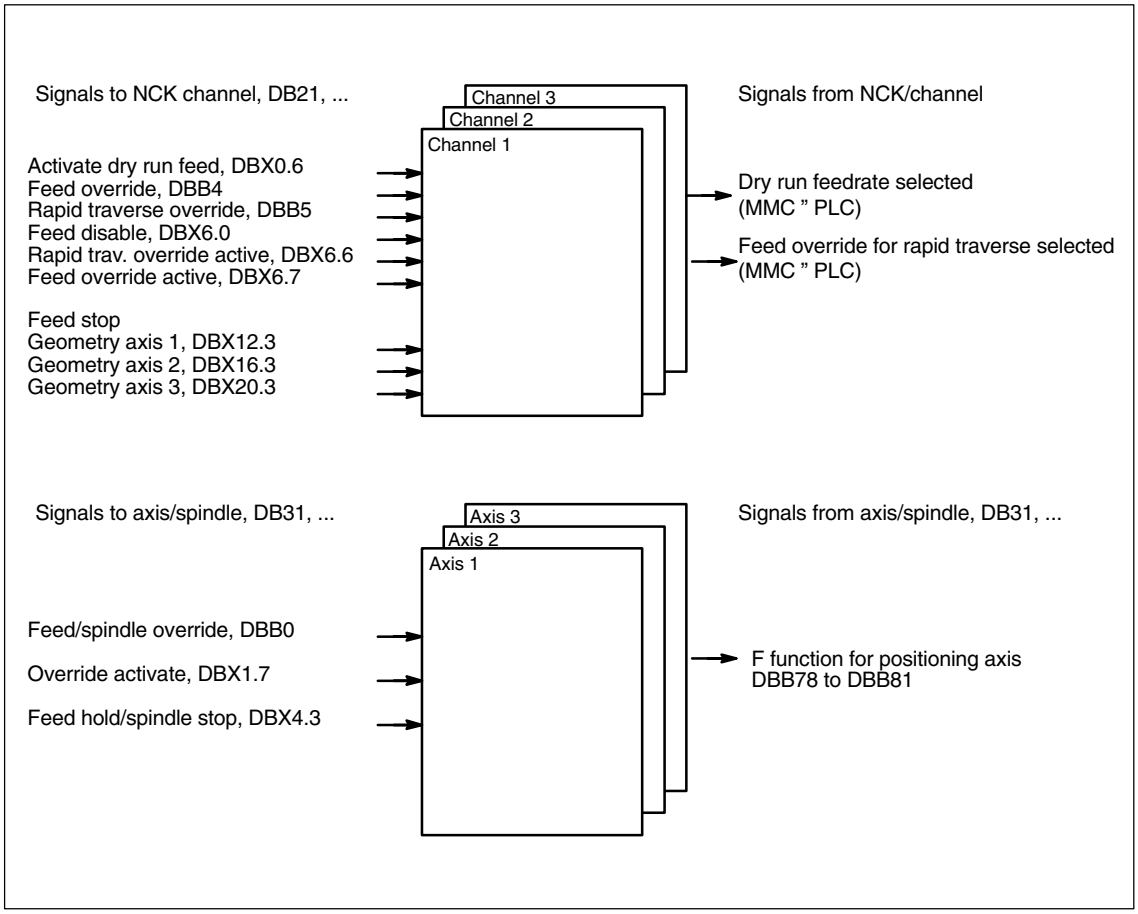


Fig. 5-1 PLC interface signals for feeds

© Siemens AG, 2003. All rights reserved

5.1 Channel-specific signals

5.1.1 Signals to channel

DB21, ... **DBX0.6** Data block	**Activate dry run feed** Signal(s) to channel (PLC → NCK)	
Edge evaluation: yes	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The dry run feed defined in SD 42100: DRY_RUN_FEED is used for traversing instead of the programmed feed (with G01, G02, G03) if the dry run feed is larger than the programmed feed. The dry run feed is active after a reset. This interface signal is evaluated on an NC start when the channel is in the “Reset” state. The dry run feed can be activated from the PLC or operator panel. When activated from the operator panel, the “Dry run feed selected” PLC interface signal is enabled and transmitted by the PLC basic program to the “Activate dry run feed” interface signal. When selected on the PLC, the “Activate dry run feed” interface signal is required to be set by the PLC user program.	
Signal state 0 or signal transition 1 ——> 0	The programmed feed is used for traversing. Active after reset.	
Application example(s)	Testing a workpiece program with an increased feedrate.	
Special cases, errors, ...	If the signal changes to “0” within a G33 block, the programmed feed is not activated until the end of the block, since an NC stop was not triggered.	
Related to	IS “Dry run feedrate selected” (DB21, ... DBX24.6) SD 42100: DRY_RUN_FEED (dry run feed)	

DB21, ... **DBB4** Data block	**Feed override** Signal(s) to channel (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The feed override can be defined via the PLC in binary or Gray coding. With binary coding, the feed value is interpreted in %. 0% to 200% feed changes are possible, in accordance with the binary value in the byte. The permanent assignment is as follows: Codes — Feed override factor 00000000 — 0.00 ≙ 0% 00000001 — 0.01 ≙ 1% 00000010 — 0.02 ≙ 2% 00000011 — 0.03 ≙ 3% . — . 11001000 — 2.00 ≙ 200% Binary values > 200 are limited to 200%. The MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded correction switch) can be used to further limit the maximum feed correction. The table continues on the next page.	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBB4** Data block	**Feed override** Signal(s) to channel (PLC → NCK)
Signal state 1 or signal transition 0 ——> 1	With Gray coding, the individual switch settings are assigned to the following codes:

Switch settings	Generate	Feed override factor (standard values)
1	00001	0.0
2	00011	0.01
3	00010	0.02
4	00110	0.04
5	00111	0.06
6	00101	0.08
7	00100	0.10
8	01100	0.20
9	01101	0.30
10	01111	0.40
11	01110	0.50
12	01010	0.60
13	01011	0.70
14	01001	0.75
15	01000	0.80
16	11000	0.85
17	11001	0.90
18	11011	0.95
19	11010	1.00
20	11110	1.05
21	11111	1.10
22	11101	1.15
23	11100	1.20
24	10100	1.20
25	10101	1.20
26	10111	1.20
27	10110	1.20
28	10010	1.20
29	10011	1.20
30	10001	1.20
31	10000	1.20

Table 5-1 Gray coding for feed override

The feed override factors specified in the table are stored in the NC machine data 12030: OVR_FACTOR_FEEDRATE [n]. The table contains the default settings.
The number of possible switch settings on standard machine control panels are described in the configuring guides for the 840D/810D and FMNC.

Related to	IS “Feed override active” (DB21, ... DBX6.7) MD 12030: OVR_FACTOR_FEEDRATE [n] (evaluation of path feed override switch) MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override switch)

© Siemens AG, 2003. All rights reserved

DB21, ... **DBB5** Data block	**Rapid traverse override** Signal(s) to channel (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The rapid traverse override can be defined via the PLC in binary or Gray coding. With binary coding, the rapid traverse override is interpreted in %. 0% to 100% feed changes are possible, in accordance with the binary value in the byte. The permanent assignment is as follows: Codes — Rapid traverse override factor 00000000 — 0.00 ≙ 0% 00000001 — 0.01 ≙ 1% 00000010 — 0.02 ≙ 2% 00000011 — 0.03 ≙ 3%. . — . 01100100 — 1.00 ≙ 100% Binary values > 100 are limited to 100%. The MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override switch) can be used as an additional limit for the maximum rapid traverse override. The table continues on the next page.	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBB5** Data block	**Rapid traverse override** Signal(s) to channel (PLC → NCK)
	With Gray coding, the individual switch settings are assigned to the following codes:

Switch settings	Generate	Rapid traverse override factor (standard values)
1	00001	0.0
2	00011	0.01
3	00010	0.02
4	00110	0.04
5	00111	0.06
6	00101	0.08
7	00100	0.10
8	01100	0.20
9	01101	0.30
10	01111	0.40
11	01110	0.50
12	01010	0.60
13	01011	0.70
14	01001	0.75
15	01000	0.80
16	11000	0.85
17	11001	0.90
18	11011	0.95
19	11010	1.00
20	11110	1.00
21	11111	1.00
22	11101	1.00
23	11100	1.00
24	10100	1.00
25	10101	1.00
26	10111	1.00
27	10110	1.00
28	10010	1.00
29	10011	1.00
30	10001	1.00
31	10000	1.00

Table 5-2 Gray coding for rapid traverse override

The rapid traverse override factors specified in the table are stored in MD 12050: OVR_FACTOR_RAPID_TRA[n]. The table contains the default settings.
The number of possible switch settings on standard machine control panels are described in the configuring guides for the 840D/810D and FMNC.

Related to	IS “Rapid traverse override active” (DB21, ... DBX6.6) MD 12050: OVR_FACTOR_RAPID_TRA[n] (evaluation of rapid traverse override switch) MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override switch)

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX6.0** Data block	**Feed disable** Signal(s) to channel (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The signal is active on a channel in all operating modes. • The signal disables all axes (geometry and synchronized) interpolating relative to each other, as long as G33 (thread) is not active. All axes are brought to a standstill with adherence to the path contour. When the feed disable is canceled (0 signal), the interrupted parts program is continued. • The signal triggers a feed disable for all positioning axes. Traversing axes are brought to a standstill under controlled braking (ramp stop). No alarm is output. • The position control is retained, i.e. the following error is eliminated. • If a travel request is issued for an axis with an active “Feed disable”, the request is retained and executed immediately the “Feed disable” is canceled. If the axis is interpolating in relation to others, this also applies to these axes.	
Signal state 0 or signal transition 1 ——> 0	• The feed is enabled for all axes on the channel. • If a travel request (“Travel command”) exists for an axis or group of axes when the “Feed disable” is canceled, this is executed immediately.	
Application example(s)	Stop machining by selecting FEED OFF on the machine control panel.	
Special cases, errors, ...	The feed disable is inactive when G33 is active.	

DB21, ... **DBX6.6** Data block	**Rapid traverse override active** Signal(s) to channel (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The rapid traverse override between 0 and a maximum of 100% entered in the PLC is channel-specific. The rapid traverse override is specified in MD 12040: OVR_RAPID_IS_GRAY_CODE (rapid traverse override switch Gray-coded), MD 12050: OVR_FACTOR_RAPID_TRA[n] (evaluation of rapid traverse override switch)	
Signal state 0 or signal transition 1 ——> 0	The rapid traverse override entered in the PLC interface is ignored. When the rapid traverse override is inactive, the NC always uses 100% as the internal override factor. Exceptions are the zero setting for a binary interface and the 1st switch setting for a Graycoded interface. In these cases, the override factors entered in the PLC interface are used. With a binary interface, the override factor is 0. With a Gray-coded interface, the value entered in the machine data for the 1st switch setting is output as the override value.	
Application example(s)	The override value is generally selected using the rapid traverse override switch on the machine control panel. The “Rapid traverse override active” interface signal can be used to enable the rapid traverse override switch from the PLC user program, e.g. with the keyswitch, during start-up of a new NC program.	
Special cases, errors, ...	The rapid traverse override is inactive when G33, G63, G331 or G332 are active.	
Related to	IS “Rapid traverse override” (DB21, ... DBB5)	

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX6.7** Data block	**Feed override active** Signal(s) to channel (PLC → NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The feed override between 0 and a maximum of 200% entered in the PLC interface is active for the path feed and therefore automatically for the related axes. In JOG mode, the feed override acts directly on the axes. The override factor is specified in the MD 12020: OVR_FEED_IS_GRAY_CODE (path feed override switch Gray-coded), MD 12030: OVR_FACTOR_FEEDRATE [n] (path feed override switch Gray-coded)		
Signal state 0 or signal transition 1 ——> 0	The feed override entered in the PLC interface is ignored. When the feed override is inactive, the NC always uses 100% as the internal override factor. Exceptions are the zero setting for a binary interface and the 1st switch setting for a Gray-coded interface. In these cases, the override factors entered in the PLC interface are used. With a binary interface, the override factor = 0. With a Gray-coded interface, the value entered in the machine data for the 1st switch setting is output as the override value.		
Application example(s)	The override value is generally selected using the feed override switch on the machine control panel. The “Feed override active” interface signal can be used to enable the feed override switch from the PLC user program, e.g. with the keyswitch, during start-up of a new NC program.		
Special cases, errors, ...	The feed override is inactive when G33, G63, G331 or G332 are active.		
Related to	IS “feedrate override” (DB21, ... DBB4)		

DB21, ... **DBX12.3 16.3 20.3** Data block	**Feed hold (Geometry axis 1 to 3)** Signal(s) to channel (PLC → NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The signal is only active in JOG mode. • The signal stops the geometry axis. Traversing axes are brought to a standstill under controlled braking (ramp stop). No alarm is output. • The position control is retained, i.e. the following error is eliminated. • If a travel request is issued for an axis with an active “Feed hold”, the request is retained and executed immediately the “Feed hold” is canceled.		
Signal state 0 or signal transition 1 ——> 0	• The feed is enabled for the geometry axis. • If a travel request (“travel command”) exists for a geometry axis when the “Feed hold” is canceled, this is executed immediately.		

DB21, ... **DBX24.6** Data block	**Dry run feedrate selected** Signal(s) to channel MMC → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	Dry run feed is selected. Instead of the programmed feed, the dry run feed entered in SD 42100: DRY_RUN_FEED is used. When activated from the operator panel, the dry run feed signal is automatically entered in the PLC interface and transmitted by the PLC basic program to the “Activate dry run feed” PLC interface signal.		
Signal state 0 or signal transition 1 ——> 0	Dry run feed is not selected. The programmed feed is active.		
Related to	IS “Activate dry run feedrate” (DB21, ... DBX0.6) SD: DRY_RUN_FEED (dry run feed)		

© Siemens AG, 2003. All rights reserved

DB21, ... **DBX25.3** Data block	**Feed override for rapid traverse selected** Signal(s) to channel MMC → PLC)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The feed override switch is also to be active as rapid traverse override switch. Override values above 100% are limited to the maximum value for 100% rapid traverse override. The “Feed override for rapid traverse selected” interface signal is automatically entered in the PLC interface from the operator panel and transferred to the “Rapid traverse override active” PLC interface signal by the PLC basic program. IS “Feedrate override” (DB21, ... DBB4) IS “Rapid traverse override” (DB21, ... DBB5).		
Signal state 0 or signal transition 1 ——> 0	The feed override switch is not to be activated as rapid traverse override switch.		
Application example(s)	The signal is used when no separate rapid traverse override switch is available.		

DB 21, 22, ... **DBX29.0** **DBX29.1** **DBX29.2** **DBX29.3** Data block	**Activate fixed feed 1 for path/geometry axes** **Activate fixed feed 2 for path/geometry axes** **Activate fixed feed 3 for path/geometry axes** **Activate fixed feed 4 for path/geometry axes** Signal(s) to channel (PLC —> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 840D SW4.1, 810D SW2.1	
Description	These signals select/deselect the fixed feed function and specify which fixed feed is to be active for path/geometry axes.		

Bit 3	Bit 2	Bit 1	Bit 0	Significance
0	0	0	0	Fixed feed is deselected
0	0	0	1	Fixed feed 1 is selected
0	0	1	0	Fixed feed 2 is selected
0	1	0	0	Fixed feed 3 is selected
1	0	0	0	Fixed feed 4 is selected

Related to	$MN_PERMANENT_FEED[n] $MN_RUN_OVERRIDE_0

DB 31, 32, ... **DBX3.2** **DBX3.3** **DBX3.4** **DBX3.5** Data block	**Activate fixed feed 1 for machine axes** **Activate fixed feed 2 for machine axes** **Activate fixed feed 3 for machine axes** **Activate fixed feed 4 for machine axes** Signal(s) to axis (PLC —> NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 840D SW4.1, 810D SW2.1	
Description	These signals select/deselect the fixed feed function and specify which fixed feed is to be active for machine axes.		

Bit 5	Bit 4	Bit 3	Bit 2	Significance
0	0	0	0	Fixed feed is deselected
0	0	0	1	Fixed feed 1 is selected
0	0	1	0	Fixed feed 2 is selected
0	1	0	0	Fixed feed 3 is selected
1	0	0	0	Fixed feed 4 is selected

Related to	$MN_PERMANENT_FEED[n] $MN_RUN_OVERRIDE_0

© Siemens AG, 2003. All rights reserved

5.2 Axis/spindle-specific signals

5.2.1 Signals to axis/spindle

<table>
<tr><td>DB 31, ...
DBB0
Data block</td><td colspan="3">Feed override (axis-specific)
Signal(s) to axis (PLC → NCK)</td></tr>
<tr><td>Edge evaluation: no</td><td colspan="2">Signal(s) updated: cyclically</td><td>Signal(s) valid from SW: 1.1</td></tr>
<tr><td>Signal state 1 or signal transition 0 ——> 1</td><td colspan="3">The axis-specific feed override can be defined via the PLC in binary or Gray coding.
With binary coding, the feed value is interpreted in %. 0% to 200% feed changes are possible, in accordance with the binary value in the byte.

The permanent assignment is as follows:
Codes — Axis-specific feed override factor
00000000 — 0.00 ≙ 0%
00000001 — 0.01 ≙ 1%
00000010 — 0.02 ≙ 2%
00000011 — 0.03 ≙ 3%
•
•
•
01100100 — 1.00 ≙ 100%
•
•
•
11001000 — 2.00 ≙ 200%
Binary values > 200 are limited to 200%.
The MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override switch) can be used as an additional limit for the maximum axis-specific feed override.

The table continues on the next page.</td></tr>
</table>

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBB0** Data block	**Feed override (axis-specific)** Signal(s) to axis (PLC → NCK)
Signal state 1 or signal transition 0 ——> 1	With Gray coding, the individual switch settings are assigned to the following codes:

Switch settings	Generate	Axial feed override factor (standard values)
1	00001	0.0
2	00011	0.01
3	00010	0.02
4	00110	0.04
5	00111	0.06
6	00101	0.08
7	00100	0.10
8	01100	0.20
9	01101	0.30
10	01111	0.40
11	01110	0.50
12	01010	0.60
13	01011	0.70
14	01001	0.75
15	01000	0.80
16	11000	0.85
17	11001	0.90
18	11011	0.95
19	11010	1.00
20	11110	1.05
21	11111	1.10
22	11101	1.15
23	11100	1.20
24	10100	1.20
25	10101	1.20
26	10111	1.20
27	10110	1.20
28	10010	1.20
29	10011	1.20
30	10001	1.20
31	10000	1.20

Table 5-3 Gray coding for axis-specific feed override

The factors specified in the table for the axial feed override are stored in the NCMD 12010: OVR_FACTOR_AX_SPEED [n]. The table contains the default settings.
The number of possible switch settings on standard machine control panels are described in the configuring guides for the 840D/810D and FMNC.

Related to	IS “Override active” (DB31, ... DBX1.7) MD 12010: OVR_FACTOR_AX_SPEED [n] (evaluation of axis feed override switch) MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override switch)

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBB19** Data block	**Spindle override** Signal(s) to spindle (PLC → NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	The spindle override can be defined via the PLC in binary or Gray coding. The override value determines the percentage of the programmed speed setpoint output to the spindle. With binary coding, the override is interpreted in %. 0% to 200% speed changes are possible, in accordance with the binary value in the byte. The permanent assignment is as follows: Codes — Spindle override factor 00000000 — 0.00 ≙ 0% 00000001 — 0.01 ≙ 1% 00000010 — 0.02 ≙ 2% 00000011 — 0.03 ≙ 3% • • • 01100100 — 1.00 ≙ 100% • • • 11001000 — 2.00 ≙ 200% Binary values > 200 are limited to 200%. The MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override switch) can be used as an additional limit for the maximum spindle override. The table continues on the next page.		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBB19** Data block	**Spindle override** Signal(s) to spindle (PLC → NCK)
Signal state 1 or signal transition 0 ——> 1	With Gray coding, the individual switch settings are assigned to the following codes:

Switch settings	Generate	Spindle override factor (standard values)
1	00001	0.5
2	00011	0.55
3	00010	0.60
4	00110	0.65
5	00111	0.70
6	00101	0.75
7	00100	0.80
8	01100	0.85
9	01101	0.90
10	01111	0.95
11	01110	1.00
12	01010	1.05
13	01011	1.10
14	01001	1.15
15	01000	1.20
16	11000	1.20
17	11001	1.20
18	11011	1.20
19	11010	1.20
20	11110	1.20
21	11111	1.20
22	11101	1.20
23	11100	1.20
24	10100	1.20
25	10101	1.20
26	10111	1.20
27	10110	1.20
28	10010	1.20
29	10011	1.20
30	10001	1.20
31	10000	1.20

Table 5-4 Gray coding for spindle override

The factors for the spindle specified in the table are stored in the MD 12070: OVR_FACTOR_SPIND_SPEED [n]. The table contains the default settings. The number of possible switch settings on standard machine control panels are described in the configuring guides for the 840D/810D and FMNC.

Related to	IS “Override active” (DB31, ... DBX1.7) MD 12070: OVR_FACTOR_SPIND_SPEED [n] (evaluation of spindle override switch) MD 12100: OVR_FACTOR_LIMIT_BIN (limit for binary-coded override switch)

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX1.7** Data block	**Override active** Signal(s) to axis/spindle (PLC → NCK)		
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1	
Signal state 1 or signal transition 0 ——> 1	Feed override active: • The axis-specific feed override between 0 and a maximum of 200% entered in the PLC interface is used. The rapid traverse override is specified in MD 12000: OVR_AX_IS_GRAY_CODE (feed rate override switch Gray-coded), (spindle override switch Gray-coded), MD 12010: OVR_FACTOR_AX_SPEED [n] (evaluation of axis feed override switch). Spindle override active: • The spindle override of 0 to a maximum of 200% is used. The override factor is specified in the MD 12060: OVR_SPIND_IS_GRAY_CODE (spindle override switch Gray-coded), MD 12070: OVR_FACTOR_SPIND_SPEED [n] (evaluation of spindle override switch).		
Signal state 0 or signal transition 1 ——> 0	The existing axis-specific feed override or spindle override is not active. If the feed override is inactive, “100%” is used as the internal override factor. Exceptions are the zero setting for a binary interface and the 1st switch setting for a Gray-coded interface. In these cases, the override factors entered in the PLC interface are used. With a binary interface, the override factor = 0. With a Gray-coded interface, the value entered in the machine data for the 1st switch setting is output as the override value.		
Application example(s)	The override value is generally specified using the axis-specific feed override switch or the spindle override switch on the machine control panel. The “Feed override active” signal can be used to enable the feed override switch from the PLC user program, e.g. with the key switch, during start-up of a new NC program.		
Special cases, errors, ...	• The spindle override is always accepted with 100% in the spindle “oscillation mode”. • The spindle override acts on the programmed values before the limits intervene (e.g. G26, LIMS ...). • When the following functions are active, the feed override is – inactive with G33 – inactive with G63 (the override is set to 100% on the NC) – inactive with G331, G332 (the override is set to 100% on the NC) • The spindle override is active when – inactive with G63 (the override is set to 100% on the NC)		
Related to	“Feed/spindle override” interface signal (DB31, ... DBB0)		

© Siemens AG, 2003. All rights reserved

DB 31, ... **DBX4.3** Data block	**Feedrate hold/spindle stop (axis-specific)** Signal(s) to axis/spindle (PLC → NCK)	
Edge evaluation: no	Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The signal is active in all modes. Feed hold: • The signal triggers a feed hold for the axis. Traversing axes are brought to a standstill under controlled braking (ramp stop). No alarm is output. • The signal triggers a feed hold for all path axes interpolating relative to each other when the “Feed hold” is activated for any one of these axes. In this case, all the axes are brought to a stop with adherence to the path contour. When the feed hold signal is canceled, execution of the interrupted parts program is resumed. • The position control is retained, i.e. the following error is eliminated. • If a travel request is issued for an axis with an active “Feed hold”, the request is retained and executed immediately the “Feed hold” is canceled. If the axis is interpolating in relation to others, this also applies to these axes. If the axis is interpolating in relation to others, this also applies to these axes. Spindle stop: • The spindle is brought to a standstill along the acceleration characteristic. • In positioning mode, activation of the “Spindle stop” signal interrupts the positioning process. The above response applies with respect to individual axes.	
Signal state 0 or signal transition 1 ——> 0	Feed hold: • The feed is enabled for the axis. • If a travel request (“travel command”) is active when the “Feed hold” is canceled, this is executed immediately. Spindle stop: • The speed is enabled for the spindle. • The spindle is accelerated to the previous speed setpoint with the acceleration characteristic or, in positioning mode, positioning is resumed.	
Application example(s)	Feed hold: • The traversing motion in the machine axes is not started when “Feed hold” is active, if, for example, certain operating states that do not permit axis motion (e.g. door not closed) prevail. Spindle stop: • Change a tool • Enter auxiliary functions (M, S, H, T, D and F functions) during setup.	
Special cases, errors, ...	Spindle stop is inactive when G331 or G332 are active.	

© Siemens AG, 2003. All rights reserved

5.2.2 Signals from axis/spindle

DB 31, ... **DBX62.2** Data block	**Revolutional feedrate active** Signal(s) from axis (NCK → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	For programming of G95 (revolutional feedrate) in JOG mode or AUTOMATIC mode.		
Application example(s)			
Related to	SD 41100: JOG_REV_IS_ACTIVE (revolutional feedrate for JOG active) SD 42600: JOG_FEED_PER_REV_SOURCE (in JOG mode revolutional feed for geometry axes on which a frame with rotation is active) SD 43300: ASSIGN_FEED_PER_REV_SOURCE (revolutional feedrate for positional axes/spindles) MD 32040: JOG_REV_VELO_RAPID (revolutional feedrate for JOG mode with rapid traverse overlay) MD 32050: JOG_REV_VELO (revolutional feedrate for JOG mode)		

DB 31, ... **DBB78–81** Data block	**F function for positioning axis** Signal(s) from axis (NCK → PLC)		
Edge evaluation: no		Signal(s) updated: cyclically	Signal(s) valid from SW: 1.1
Signal state 1 or signal transition 0 ——> 1	The F value of a positioning axis programmed in the current block is entered in the axis-specific PLC interface signal. The assignment between the DB number and machine axis number is performed using the axis name. The value is retained until it is overwritten by another. Format: binary number in real format.		
Application example(s)	Modification of the programmed F value by the PLC, e.g. by overwriting the defined axis-specific feed override.		
Related to	MD 22240: AUXFU_F_SYNC_TYPE Output time of the F functions		

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

6 Example

6.1 Feedrate programming for chamfer/rounding FRC, FRCM

Example 1: Feed from following block

MD 20201: CHFRND_MODE_MASK Bit0 = 0: Take feed from following block (default setting)

```
N10   G0 X0 Y0 G17 F100 G94
N20   G1 X10 CHF=2              ; Chamfer N20–N30 at F=100 mm/min
N30   Y10 CHF=4                 ; Chamfer N30–N40 at FRC=200 mm/min
N40   X20 CHF=3 FRC=200         ; Chamfer N40–N60 at FRCM=50 mm/min
N50   RNDM=2 FRCM=50
N60   Y20                       ; Modal rounding N60–N70 at
                                ; FRCM=50 mm/min
N70   X30                       ; Modal rounding N70–N80 at
                                ; FRC=100 mm/min
N80   Y30 CHF=3 FRC=100         ; Chamfer N80–N90 at FRCM=50 mm/min
                                ; (modal)
N90   X40                       ; Modal rounding N90–N100 at
                                ; F=100 mm/min (deselection of FRCM)
N100  Y40 FRCM=0                ; Modal rounding N100–N120 with G95
                                ; FRC=1 mm/rev
N110  S1000 M3
N120  X50 G95 F3 FRC=1
...
M02
```

Example 2: Feed from preceding block

MD 20201: CHFRND_MODE_MASK Bit0 = 1: Take feed from preceding block (recommended setting)

```
N10   G0 X0 Y0 G17 F100 G94
N20   G1 X10 CHF=2              ; Chamfer N20–N30 at F=100 mm/min
N30   Y10 CHF=4 FRC=120         ; Chamfer N30–N40 at FRC=120 mm/min
N40   X20 CHF=3 FRC=200         ; Chamfer N40–N60 at FRC=200 mm/min
N50   RNDM=2 FRCM=50
N60   Y20                       ; Modal rounding N60–N70 at
                                ; FRCM=50 mm/min
N70   X30                       ; Modal rounding N70–N80 at
                                ; FRCM=50 mm/min
N80   Y30 CHF=3 FRC=100         ; Chamfer N80–N90 at FRC=100 mm/min
N90   X40                       ; Modal rounding N90–N100 at
                                ; FRCM=50 mm/min
N100  Y40 FRCM=0                ; Modal rounding N100–N120
                                ; at F=100 mm/min
N110  S1000 M3
N120  X50 CHF=4 G95 F3 FRC=1    ; Chamfer N120–N130 with G95
                                ; FRC=1 mm/rev
N130  Y50                       ; Mod. rounding N130–N140 at F=3 mm/rev
N140  X60
...
M02
```

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

7

7.1 Interface signals

DB number	Bit, byte	Name	Refer-ence
Channel-specific			
21, ...	0.6	Activate dry run feed	
21, ...	4	Feed override	
21, ...	5	Rapid traverse override	
21, ...	6.0	Feed disable	
21, ...	6.6	Rapid traverse override active	
21, ...	6.7	Feed override active	
21, ...	12.3	Feed hold, geometry axis 1	
21, ...	16.3	Feed hold, geometry axis 2	
21, ...	20.3	Feed hold, geometry axis 3	
21, ...	24.6	Dry run feedrate selected	
21, ...	25.3	Feed override for rapid traverse selected	
21, ...	29.0 29.1 29.2 29.3	Activate fixed feed 1 for path/geometry axes Activate fixed feed 2 for path/geometry axes Activate fixed feed 3 for path/geometry axes Activate fixed feed 4 for path/geometry axes	
Axis/spindle-specific			
31, ...	0	Feed/spindle override	
31, ...	1.7	Override active	
31, ...	3.2 3.3 3.4 3.5	Activate fixed feed 1 for machine axis Activate fixed feed 2 for machine axis Activate fixed feed 3 for machine axis Activate fixed feed 4 for machine axis	
31, ...	4.3	Feed hold/spindle stop	
31, ...	62.2	Revolutional feedrate active	
31, ...	78–81	F function for positioning axis	
31, ...	83.1	Programmed speed too high	S1

© Siemens AG, 2003. All rights reserved

7.2 Machine data

Number	Identifier	Name	Refer-ence
General ($MN_ ...)			
10704	DRYRUN_MASK	Activating the dry run feedrate	
10710	PROG_SD_RESET_SAVE_TAB[n]	Setting data to be updated	K1
11410	SUPPRESS_ALARM_MASK	Mask for suppressing special alarms	D1
12000	OVR_AX_IS_GRAY_CODE	Path feed override switch Gray-coded	
12010	OVR_FACTOR_AX_SPEED[n]	Evaluation of the axis feed override switch	
12020	OVR_FEED_IS_GRAY_CODE	Path feed override switch Gray-coded	
12030	OVR_FACTOR_FEEDRATE[n]	Evaluation of the path feed override switch	
12040	OVR_RAPID_IS_GRAY_CODE	Rapid traverse override switch Gray-coded	
12050	OVR_FACTOR_RAPID_TRA[n]	Evaluation of the rapid traverse override switch	
12060	OVR_SPIND_IS_GRAY_CODE	Spindle override switch Gray-coded	
12070	OVR_FACTOR_SPIND_SPEED[n]	Evaluation of the spindle override switch	
12080	OVR_REFERENCE_IS_PROG_FEED	Override reference velocity (SW 2 and higher)	
12082	OVR_REFERENCE_IS_MIN_FEED	Defining the reference for the path override	
12100	OVR_FACTOR_LIMIT_BIN	Limit for binary-coded override switch	
12200	RUN_OVERRIDE_0	Traversing with override 0	FBMA
12202	PERMANENT_FEED[n]	Fixed feeds for linear axes	FBMA
12204	PERMANENT_ROT_AX_FEED[n]	Fixed feeds for rotary axes	FBMA
12240	SCALING_SYSTEM_IS_METRIC	Basic system metric	G2
Axis-specific ($MA_ ...)			
30300	IS_ROT_AX	Rotary axis	R2
32000	MAX_AX_VELO	Maximum axis velocity	G2
32060	POS_AX_VELO	Initial setting for positioning axis velocity	P2
32300	MAX_AX_ACCEL	Axis acceleration	B2
35100	SPIND_VELO_LIMIT	Maximum spindle speed	S1
35130	GEAR_STEP_MAX_VELO_LIMIT	Maximum speed of gear step	S1
35140	GEAR_STEP_MIN_VELO_LIMIT	Minimum speed of gear step	S1
35160	SPIND_EXTERN_VELO_LIMIT	Spindle speed limitation via PLC	S1
Channel-specific ($MC_ ...)			
20100	DIAMETER_AX_DEF	Geometry axes with facing axis functions	P1
20172	COMPRESS_VELO_TOL	Maximum permissible deviation from path feed for compression	
20150	GCODE_RESET_VALUES	Reset G groups	K1
20200	CHFRND_MAXNUM_DUMMY_BLOCKS	Maximal number of dummy blocks for chamfer/radii	V1
20201	CHFRND_MODE_MASK	Feed for chamfer/rounding	V1
20660	THREAD_AUTO_LIFTFASTANGLE	Determining the retraction angle (thread cutting); not included in SW 4.3 and higher	K1
20750	ALLOW_GO_IN_G96	G0 logic in G96	
21200	LIFTFAST_DIST	Traversing path for fast retraction from the contour	K1
21220	MULTFEED_ASSIGN_FASTIN	Assignment of the input bytes of the NCK I/Os for “Multiple feeds in one block” (SW 2 and higher)	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Reference
21230	MULTFEED_STORE_MASK	Store input signals for the function “Multiple feeds in one block” (SW 2 and higher)	
22240	AUXFU_F_SYNC_TYPE	Output timing of F functions	H2
22410	F_VALUES_ACTIVE_AFTER_RESET	F function active after reset	
Axis/channel-specific ($MA_ ...)			
34990	ENC_ACTIVAL_SMOOTH_TIME	Smoothing time constant for actual values (SW 6.3 and higher)	

7.3 Setting data

Number	Identifier	Name	Reference
Channel-specific ($SC_ ...)			
42000	THREAD_START_ANGLE	Start angle for thread	K1
42010	THREAD_RAMP_DISP	Run-in and run-out path of feed axis in thread cutting	
42100	DRY_RUN_FEED	Dry run feed	
42101	DRY_RUN_FEED_MODE	Dry run feedrate mode	
42110	DEFAULT_FEED	Default value for path feed	
42600	JOG_FEED_PER_REV_SOURCE	In the JOG mode, revolutional feedrate for geometry axes on which a frame with rotation acts	
43300	ASSIGN_FEED_PER_RES_SOURCE	Revolutional feedrate for positioning axes	
Axis-specific ($SA_ ...)			
43210	SPIND_MIN_VELO_G25	Minimum spindle speed	S1
43220	SPIND_MAX_VELO_G26	Maximum spindle speed	S1
43230	SPIND_MAX_VELO_LIMS	Spindle speed limitation with G96	S1

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, “Diagnostics Guide”
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

SINUMERIK 840D/840Di/810D Description of Functions Basic Machine (Part 1)

Tool Offset (W1)

1	**Brief Description**		**1/W1/1-3**
2	**Detailed Description**		**1/W1/2-5**
	2.1	Tool	1/W1/2-5
	2.1.1	Offset memory structure	1/W1/2-7
	2.1.2	Calculation of the tool offset	1/W1/2-8
	2.1.3	Address extension for NC addresses T and M (SW 5 and higher)	1/W1/2-9
	2.1.4	Free D number allocation (SW 5 and later)	1/W1/2-10
	2.1.5	Correction block for error during tool change (SW 5 and higher)	1/W1/2-16
	2.1.6	Definition of the effect of the tool parameters	1/W1/2-18
	2.2	Flat D number structure (SW 4 and higher)	1/W1/2-19
	2.2.1	Creating a new D number (correction block)	1/W1/2-19
	2.2.2	Read D number from the PLC	1/W1/2-20
	2.2.3	D number programming	1/W1/2-21
	2.2.4	Programming the T number	1/W1/2-24
	2.2.5	Programming M6	1/W1/2-24
	2.2.6	Program test	1/W1/2-25
	2.2.7	Tool management or “Flat D numbers”	1/W1/2-25
	2.3	Tool cutting edge	1/W1/2-26
	2.3.1	Tool type (tool parameters)	1/W1/2-27
	2.3.2	Tooledge position (tool parameter 2)	1/W1/2-29
	2.3.3	Geometry tool length compensation (tool parameters 3 to 5)	1/W1/2-30
	2.3.4	Geometry tool radius compensation (tool parameters 6 to 11)	1/W1/2-32
	2.3.5	Wear tool length compensation (tool parameters 12 to 14)	1/W1/2-33
	2.3.6	Wear tool length compensation (tool parameters 15 to 20)	1/W1/2-33
	2.3.7	Base dimension/adapter dimension tool length compensation (tool parameters 21 to 23)	1/W1/2-34
	2.3.8	Technology – tool clearance angle (tool parameter 24)	1/W1/2-35
	2.3.9	Tools with relevant tool point direction (SW 5 and higher)	1/W1/2-36
	2.4	Tool radius compensation: 2D (TRC)	1/W1/2-38
	2.4.1	Selecting the TRC (G41/G42)	1/W1/2-39
	2.4.2	Approach and retraction behavior (NORM/KONT)	1/W1/2-40
	2.4.3	Smooth approach and retraction (SW 4.3 and higher)	1/W1/2-41
	2.4.4	Deselecting the TRC (G40)	1/W1/2-54
	2.4.5	Compensation at outside corners	1/W1/2-54
	2.4.6	Compensation at inner corners	1/W1/2-58
	2.4.7	Collision detection and bottleneck detection	1/W1/2-60
	2.4.8	Blocks with variable compensation value	1/W1/2-61
	2.4.9	Keep tool radius compensation constant (SW 4 and higher)	1/W1/2-63
	2.4.10	Changed alarm response (SW 4 and higher)	1/W1/2-66
	2.4.11	Intersection procedure for polynomials (SW 4 and higher)	1/W1/2-67

© Siemens AG, 2003. All rights reserved

2.4.12 G461/G462: Approach/retract strategy extension (SW 5 and higher) . . . 1/W1/2-68

2.5 Orientational toolholder (SW 4 and higher) . . . 1/W1/2-72
2.5.1 General . . . 1/W1/2-72
2.5.2 Kinematic interaction and machine design (SW 4 and higher) . . . 1/W1/2-78
2.5.3 Inclined machining with 3 + 2 axes (SW 5.3 and higher) . . . 1/W1/2-85
2.5.4 Machine with rotary table (SW 5.3 and higher) . . . 1/W1/2-86
2.5.5 Procedure when using orientational toolholders . . . 1/W1/2-89
2.5.6 Programming (SW 4 and higher) . . . 1/W1/2-93
2.5.7 Suppl. conditions and control system response for orientation . . . 1/W1/2-94

2.6 Incrementally programmed compensation values . . . 1/W1/2-96
2.6.1 G91 extension (SW 4.3 and higher) . . . 1/W1/2-96
2.6.2 Machining in direction of tool orientation (SW 5.3 and higher) . . . 1/W1/2-97

2.7 Basic tool orientation (SW 6.1 and higher) . . . 1/W1/2-99

2.8 Tool offset special features (SW 5 and higher) . . . 1/W1/2-102
2.8.1 Tool lengths in the WCS under consideration of the (SW 5.2 and later) . . . 1/W1/2-106
2.8.2 Tool length offsets in tool direction (SW 6.1 and later) . . . 1/W1/2-107
2.8.3 Extension of the tool length determination (SW 6.1 and higher) . . 1/W1/2-111

2.9 Sum and setup offsets (SW 5 and higher) . . . 1/W1/2-116
2.9.1 General . . . 1/W1/2-116
2.9.2 Description of function . . . 1/W1/2-117
2.9.3 Activation . . . 1/W1/2-120
2.9.4 Examples . . . 1/W1/2-126

3 Supplementary Conditions . . . 1/W1/3-127

3.1 Flat D number structure (SW 4.1 and higher) . . . 1/W1/3-127

4 Data Descriptions (MD, SD) . . . 1/W1/4-129

4.1 General machine data . . . 1/W1/4-129

4.2 Channelspecific machine data . . . 1/W1/4-133

4.3 Setting data . . . 1/W1/4-146

5 Signal Descriptions . . . 1/W1/6-153

6 Examples . . . 1/W1/6-153

6.1 Example of orientational toolholders . . . 1/W1/6-153
6.1.1 Example of orientational toolholders with rotary table (SW 5.3 and higher) . . . 1/W1/6-154
6.1.2 Example basic tool orientation (SW 6.1 and higher) . . . 1/W1/6-156
6.1.3 Calculation of offsets according to location of use or workpiece . . 1/W1/6-157

7 Data Fields, Lists . . . 1/W1/7-159

7.1 Interface signals . . . 1/W1/7-159

7.2 Machine data . . . 1/W1/7-159

7.3 Setting data . . . 1/W1/7-161

7.4 Alarms . . . 1/W1/7-162

■

© Siemens AG, 2003. All rights reserved

1 Brief Description

The SINUMERIK 840D/810D and FMNC controls allow calculation of tool compensation data.

1. Length compensation
2. Radius compensation
3. Storage of tool data in a flexible tool offset memory
   - Tool identification with T numbers from 0 to 32000
   - Definition of a tool with a maximum of 9 cutting edges
   - Cutting edge described by up to 25 tool parameters
     Type
     Geometry: Length Wear: Length Basic/adapter dimension
     Geometry: Radius Wear: Radius
     Technology
4. Tool selection selectable: Immediate or via selectable M function
5. Tool radius compensation
   - Selection and deselection strategy configurable: Normal or contourrelated
   - Compensation active for all interpolation types:
     Linear
     Circle
     Helical
     Spline
     Polynomial
   - Compensation at outer corners selectable: Transition circle/ellipse (G450) or equidistant intersection (G451)
   - Parameterdriven adaptation of G450/G451 functions to the contour
   - Free traversing on outer corners with G450 and DISC parameter
   - Number of dummy blocks without axis motion selectable in the compensation plane
   - Selectable collision monitoring:
     Possible contour violations are predicted if
     the path distance is shorter than the tool radius
     the width of an inner corner is smaller than the tool diameter.
   - Keep tool radius compensation constant (from SW 4)
   - Intersection procedure for polynomials (from SW 4)

© Siemens AG, 2003. All rights reserved

Orientational toolholder (SW 4 and higher)

This function, which is available as of SW 4, permits the machining of inclined surfaces with allowance for tool length compensation, provided that the kinematics of the toolholder (without NC axes) permits a static orientation of the tool. The more complex 5-axis transformation is not required for this case.

References: /FB/, F2, "3 – 5-axis transformation"

Appropriate selection of the tool data and toolholder data describes the kinematics such that it can make allowance for the tool length compensation. The control can take some of the description data direct from the current frame.

Note

Please refer to the following documentation for further information on tools and tool compensations and a full technical description of the general and specific programming features for tool compensation (TLC and TRC):
References: /PG/, "Programming Guide: Fundamentals"

Flat/unique D number structure

With SW 4 and higher, simple offset selection using D numbers is available in the basic version without management functions.

With SW 5 and higher, offset selection via unique D numbers is available with management function (see FB Tool Management).

Tool compensation, special handling (SW 5 and higher)

Setting data SD 42900 to SD 42950 can be used to control the evaluation of the sign for the tool length and tool wear.

The same applies to the response of the wear components when mirroring geometry axes or changing the machining plane via setting data.

References: /PG/ Programming Guide Fundamentals,
"Tool offsets"

G461/G462 (SW 5 and higher)

In order to enable the solid machining of internal corners in certain situations with the activation and deactivation of tool radius compensation, commands G461 and G462 have been introduced and the approach/retraction strategy has thus been extended for tool radius compensation.

- G461: If no intersection is possible between the last TRC block and a previous block, the control calculates an intersection by extending the offset curve of this block with a circle whose center point coincides with the end point of the noncorrected block, and whose radius is equal to the tool radius.
- G462: If no intersection is possible between the last TRC block and a previous block, the control calculates an intersection by inserting a straight line at the end point of the last block with tool radius compensation (the block is extended by its end tangent).

Changing from G40 to G41/42 (SW 5 and higher)

The change from G40 to G41/42 and vice versa is no longer treated as a tool change for tools with relevant tool point direction (turning and grinding tools).

■

© Siemens AG, 2003. All rights reserved

2 Detailed Description

2.1 Tool

Tool selection

The T function is used to select a tool in a program. The setting in MD 22550: TOOL_CHANGE_MODE (new tool offset with M function) determines whether the new tool is loaded immediately on execution of the T function.

Toolchange immediate

TOOL_CHANGE_MODE = 0
The new tool is loaded immediately on execution of the T function. This setting is used mainly on turning machines with tool turrets.

Change tool with "M06"

TOOL_CHANGE_MODE = 1

The new tool is prepared for loading on execution of the T function. This setting is used mainly on milling machines with a tool magazine, in order to bring the new tool into the tool change position without interrupting the machining process. The MD 22560: TOOL_CHANGE_M_CODE (M function for tool change) is used to remove the old tool from the spindle and load the new tool onto the spindle. This tool change is required to be programmed with the M function M06, in accordance with DIN 66025.

The next tool is preselected with MD 20121: Whose assigned tool length compensation values must be taken into account on RESET and power up according to MD 20110: RESET_MODE_MASK are to be taken into account.

Value range of T

The T function accepts whole numbers

- from T0 (no tool)
- to T32000 (tool number 32000).

Tool cutting edge

Each tool can have up to 9 cutting edges. The 9 tool cutting edges are assigned to the D functions D1 to D9.

© Siemens AG, 2003. All rights reserved

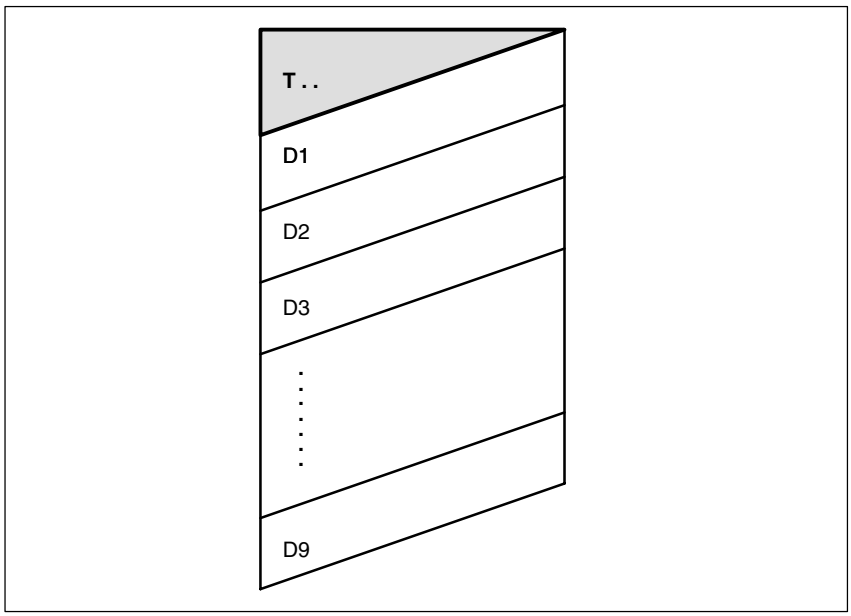


Fig. 2-1 Example of a tool T... with 9 cutting edges (D1 to D9)

D function

The tool cutting edge is programmed with D1 (edge 1) to D9 (edge 9). The tool cutting edge always refers to the currently active tool. An active tool cutting edge (D1 to D9) without an active tool (T0) is inactive. Tool cutting edge D0 deselects all tool offsets of the active tool.

Selecting of the cutting edge when changing tool

When a new tool (new T number) has been programmed and the old one replaced, the following options are available for selecting the cutting edge:

1. The cutting edge number is programmed
2. The cutting edge number is specified by MD 20270:
   CUTTING_EDGE_DEFAULT
   = 0 no automatic cutting edge selection after M06
   < > 0 number of the cutting edge selected after M06
   = –1 the cutting edge no. of the old tool is retained and is selected for new tool also after M06

Activating the tool offsets

D1 to D9 activates tool compensation for a cutting edge on the active tool. Tool length compensation and tool radius compensation can be activated at different times:

- **Tool length compensation** (TLC) is performed on the first traversing motion of the axis on which the TLC is to act. This traversing motion must be a linear interpolation (G0, G1, POS, POSA) or polynomial interpolation (POLY). If the POS/POSA axis is one of the active geometry axes, the tool length compensation is applied with the first axis motion in which it is operative.
- **Tool radius compensation** (TRC) becomes active when G41/G42 is programmed in the active plane (G17, G18 or G19). The selection of tool radius compensation with G41/G42 is only permitted in a program block with G0 (rapid traverse) or G1 (linear interpolation).

© Siemens AG, 2003. All rights reserved

2.1.1 Offset memory structure

Tool offset memory size

Each channel can have a dedicated tool offset memory (TO unit). MD 28085: MM_LINK_TOA_UNIT (assignment of TO unit to a channel) is set to define which TO memory exists for the appropriate channel. The maximum number of tool cutting edges for all tools managed by the NCK is set by means of MD 18100: MM_NUM_CUTTING_EDGES_IN_TOA (number of tool cutting edges in NCK).

Tools

The tool offset memory consists of tools with the numbers T1 to T32000. Each tool can be set up via TOA files or individually using the “New tool” soft key. Offsets which are not required should be assigned the value zero (this is the default setting when the offset memory is created): The individual values in the offset memory (tool parameters) can be read and written from the program using system variables.

Note

The tools (T1 to T32000) do not have to be stored in ascending order or contiguously in the tool offset memory, and the first tool does not have to be assigned number T1.

Tool cutting edges

Each tool can have up to 9 cutting edges (D1 to D9). The first cutting edge (D1) is automatically set up when a new tool is loaded in the tool offset memory. Other cutting edges (up to 8) are set up consecutively and contiguously using the “New cutting edge” soft key. A different number of tool cutting edges can assigned to each tool in this way.

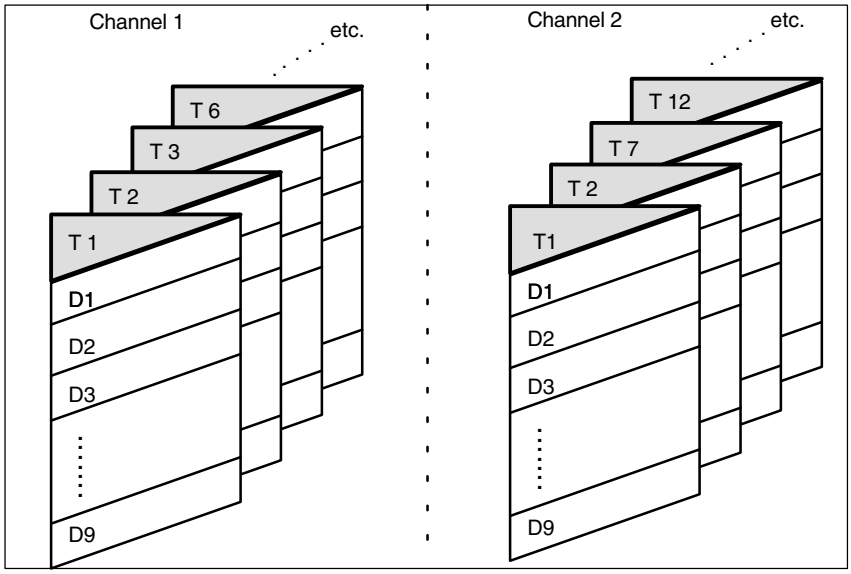


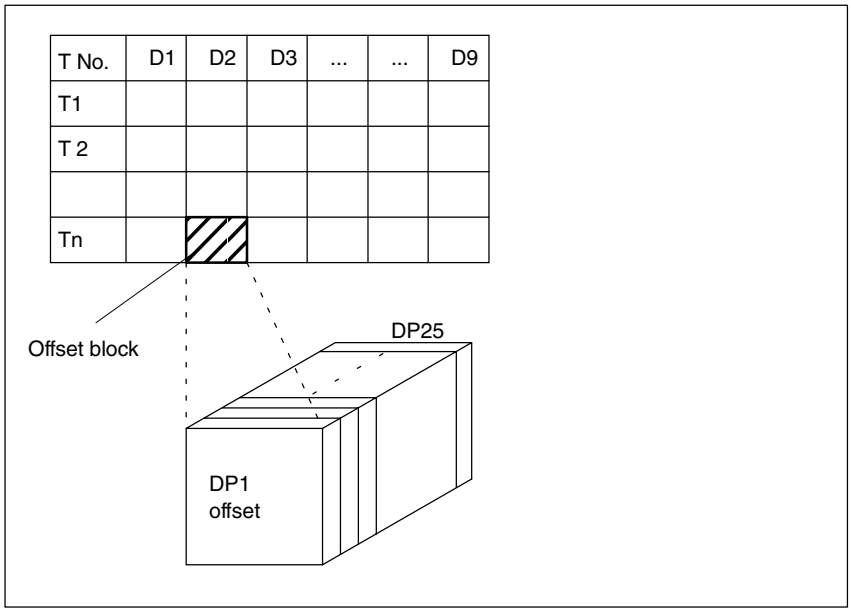
Fig. 2-2 Example of a tool offset memory structure for 2 channels

© Siemens AG, 2003. All rights reserved

2.1.2 Calculation of the tool offset

Up to SW 3.x

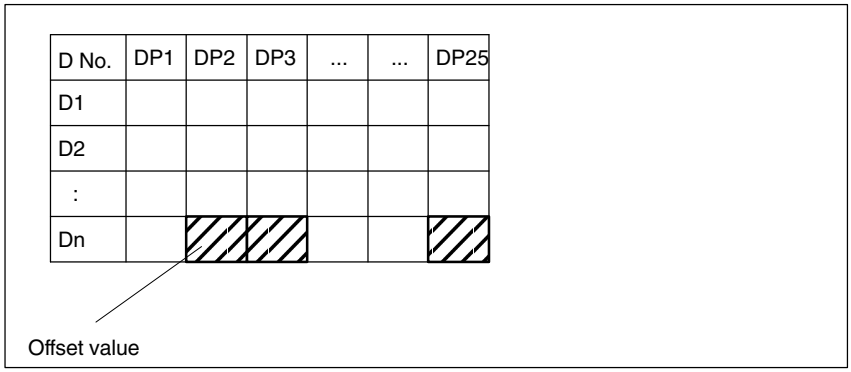
The **D No.** and the **T No.** are required for calculation of the tool offset.



Example:
The above offset is to be calculated in the NC:
Call in the parts program
...
Tn D2
...

SW 4 and higher (also possible)

The **D No.** is sufficient for calculating the tool compensation values (can be set per MD).



Example:
The above offset is to be calculated in the NC:
Call in the parts program
...
D_n

© Siemens AG, 2003. All rights reserved

2.1.3 Address extension for NC addresses T and M (SW 5 and higher)

In SW 5 and higher, you can use machine data **T_M_ADDRESS_EXT_IS_SPINO** to define whether the address extension of T and M is to be interpreted as a spindle number, even when tool management is **not** activated. The same rules then apply to the reference between the D number and T number as when the 'tool management' function is active.

Effect on the D number

An offset data set is determined by the D number.
The **D address cannot be programmed with an address extension**.
The evaluation of the D address always refers to the tool which is currently active. If T_M_ADDRESS_EXT_IS_SPINO = TRUE, the programmed D address refers to the active tool in relation to the master spindle (same as for tool management function).

Effect on the T number

If the 'tool management' function is active, the values programmed with reference to the master spindle (or master toolholder) are displayed as programmed/active T numbers.

If tool management is not active, **all** programmed T values are displayed as programmed/active, regardless of which address extension was programmed.

If T_M_ADDRESS_EXT_IS_SPINO = TRUE
(spindle number as address extension), only the T value programmed with reference to the master spindle is displayed as programmed/active.

Example

The following example illustrates the effect of the machine data.
Two spindles are used. Spindle 1 is the master spindle.
M6 was defined as the tool change signal.

T1 = 5
M1 = 6
T2 = 50
M2 = 6
D4

- When tool management is active, D4 refers to tool '5'.
  T2=50 assigns the tool to the auxiliary spindle, whose tool does not affect the path compensation. The path is determined exclusively by the tool programmed for the master spindle.
- If tool management is not active and T_M_ADDRESS_EXT_IS_SPINO = **FALSE**, D4 refers to tool '50'. The address extension of T and M are not evaluated in the NCK. Each tool change command defines a new path correction.
- If tool management is not active and T_M_ADDRESS_EXT_IS_SPINO = **TRUE**, D4 refers to tool '5' (same as if tool management active).
  Address extension 1 (T1= ..., M1= ...) addresses the master spindle.

© Siemens AG, 2003. All rights reserved

Note

Previously, when tool management was not activated, each tool change command (programmed with T or M) caused the tool offset to be recalculated in the path. The address extension is not defined further by this operation.
The meaning of the extension is defined by the user (in the PLC user program).

2.1.4 Free D number allocation (SW 5 and later)

In the NCK, it is possible to manage the D numbers as 'relative' D numbers for the tool offset data sets. The corresponding D numbers are assigned to each T number.
The maximum number of D numbers was previously limited to 9.

Functions (SW 5 and higher)

In SW 5 and higher, the functions for allocating D numbers have been extended significantly:

- The maximum number of D numbers allowed is defined by a machine data (MD 18105: MM_MAX_CUTTING_EDGE_NO). The default value is 9, in order to maintain compatibility with existing applications.
- The number of cutting edges (or offset data sets) **per tool** can also be defined by machine data
  (MD 18106: MM_MAX_CUTTING_EDGE_PERTOOL).

  This allows you to customize the number of cutting edges to be configured for each tool to the actual number of real cutting edges for monitoring purposes.
- It is also possible to rename D numbers in the NCK and thus to allocate any D numbers to the cutting edges.

Note

In addition to relative D number allocation, the D numbers can also be assigned as 'flat' or 'absolute' D numbers (1–32000) without a reference to a T number (within the 'flat D number structure' function).

Cutting edge number CE

When you rename D numbers, the information in the tool catalog detailing the numbers defined for these cutting edges is lost.
It is therefore impossible to determine, after the renaming operation, which cutting edge of the catalog is referenced.

Since this information is required for retooling procedures, a **cutting edge number CE** has been introduced for each cutting edge. This number remains stored when the D number is renamed.

© Siemens AG, 2003. All rights reserved

The D number identifies the cutting edge offset in the parts program. In SW 5 and higher, this **offset number D** is administered separately from the **cutting edge number CE** (the number in the tool catalog). Any number can be used. The number is used to identify an offset in the parts program and on the display.

The CE number identifies the actual physical cutting edge during retooling. The cutting edge number CE is not evaluated by the NCK on offset selection during a tool change (only available via the OPI).

The cutting edge number is defined with system variable **$TC_DPCE[t,d]**.

- t stands for the internal T number
- d stands for the D number.

Write accesses are monitored for collisions, i.e. all cutting edge numbers of a tool must be different. The variable $TC_DPCE is a component of the cutting edge parameter data set $TC_DP1,..., $TC_DP25.

It is only practical to parameterize $TC_DPCE if the maximum cutting edge number (MD 18105) is greater than the maximum number of cutting edges per tool (MD 18106).

In this case, the default cutting edge number is the same as the classification number of the cutting edge. Offsets of a tool are created starting at number 1 and are incremented up to the maximum number of cutting edges per tool (MD 18106: MM_MAX_CUTTING_EDGE_PER_TOOL).

If
MM_MAX_CUTTING_EDGE_NO <=
MM_MAX_CUTTING_EDGE_PER_TOOL, the CE cutting edge number is equal to the D number (in compatibility with the previous response). A read operation returns CE=D. A write operation is ignored without an alarm message.

Note

The offset values $TC_DP1,..., $TC_DP25 of the active tool offset can be read with system variable $P_AD[n]; where n=1,...,25. The CE cutting edge number of the active offset is returned with n=26.

Commands

If the maximum cutting edge number (MD 18105) is greater than the maximum number of cutting edges per tool (MD 18106), the following commands are available:

- CHKDNO – Checks whether the existing D numbers are unique;
  The D numbers of all tools defined within a TO unit may not occur more than once.
  No allowance is made for replacement tools.
- GETDNO – Gets the D number for the cutting edge of a tool. If no D number matching the input parameters exists, d=0. If the D number is invalid, a value greater than 32000 is returned.

© Siemens AG, 2003. All rights reserved

- SETDNO – Sets or changes the D number of the cutting edge CE of a tool T; if no data set matching the input parameters exists, FALSE is returned. Syntax errors generate an alarm. The D number cannot be set explicitly to 0.
- GETACTTD – Gets the T number associated with an absolute D number; the system does not check whether the numbers are unique. If several D numbers within a TO unit are the same, the T number of the first tool found in the search is returned. This command is not suitable for use with 'flat' D numbers, because the value 1 is always returned in this case (no T numbers in database).
- DZERO – Tags all D numbers of the TO unit as invalid; This command is used for support during retooling. Offset data sets tagged with this command are no longer verified by the CHKDNO language command. These data sets can be accessed again by setting the D number again with SETDNO.

Note

If the maximum cutting edge number is less than the maximum number of cutting edges per tool (MM_MAX_CUTTING_EDGE_NO < MM_MAX_CUTTING_EDGE_PERTOOL), the defined language commands have no effect on the system.
So that existing applications can still run, this relation is therefore preset in the NCK.

The individual commands are described in detail in the Programming Guide.
References: /PG/, "Programming Guide: Fundamentals"

Activation

In order to work with unique D numbers and thus with the defined language commands, it must be possible to name D numbers freely for the tools. The following conditions must be fulfilled for this purpose:

- The machine data MD 18105: MM_MAX_CUTTING_EDGE_NO must be greater than MD 18106: MM_MAX_CUTTING_EDGE_PERTOOL.
- The 'flat D number' function is not activated. (–> MD 18102: MM_TYPE_OF_CUTTING_EDGE).

Examples

MM_MAX_CUTTING_EDGE_NO = 1

A maximum of one offset can be defined per tool (with D number = 1).

© Siemens AG, 2003. All rights reserved

Note

When the 'flat D numbers' function is active, only one D offset can be defined in the TO unit.

MM_MAX_CUTTING_EDGE_NO = 9999

Tools can be assigned unique D numbers.
For example:

- D numbers 1, 2, 3 are assigned to T number 1
- D numbers 10, 20, 30, 40, 50 are assigned to T number 2
- D numbers 100, 200 are assigned to T number 3
- etc.

CHKDNO; MAX_CUTTING_EDGE_NO = 9999

The following data are to be checked for unique D numbers:

- T number 1 with D numbers 1, 2, 3
- T number 2 with D numbers 10, 20, 30, 40, 50
- T number 3 with D numbers 100, 200, 30
  (typographic error on definition: 30 was entered instead of 300)

CHKDNO	The FALSE state is returned when the above constellation is checked because D=30 exists twice.
CHKDNO (2, 3, 30)	The FALSE state is returned when the specified D number 30 is checked because D=30 exists twice
CHKDNO (2, 3, 100)	The TRUE state is returned because D=100 only exists once.
CHKDNO (1, 3)	The TRUE state is returned although there is a conflict between the D=30 of the third tool and D=30 of the second tool.

MM_MAX_CUTTING_EDGE_PERTOOL = 1

Only tools which have one cutting edge are used. The value 1 of the machine data inhibits the definition of a second cutting edge for a tool.

MM_MAX_CUTTING_EDGE_PERTOOL = 12

Up to 12 cutting edges can be defined for a tool (9 cutting edges per tool were allowed up to SW 4).

© Siemens AG, 2003. All rights reserved

Sample programs

Renaming a D number

The D number of cutting edge CE = 3 is to be renamed from 2 to 17. The following specifications apply:

- Internal T number T = 1
- D number = 2
- Tool with one cutting edge with

  $TC_DP2[1, 2] = 120

  $TC_DP3[1, 2] = 5.5

  $TC_DPCE[1, 2] = 3 ; Cutting edge number CE
- MM_MAX_CUTTING_EDGE_NO = 20

Within the parts program, this offset is programmed as standard with T1, ...D2.

You assign the current D number of cutting edge 3 to a variable (DNoOld) and define the variable DNoNew for the new D number:

```
def int DNoOld, DNoNew = 17

DNoOld = GETDNO( 1, 3 )

SETDNO( 1, 3, DNoNew )
```

The new D value 17 is then assigned to cutting edge CE=3.
The data of this cutting edge are now addressed with D number 17, both via the system variables and in programs with NC address D.

This offset is now programmed in the parts program with T1, ...D17 and the data are addressed as follows:

```
$TC_DP2[ 1, 17 ] = 120

$TC_DP3[ 1, 17 ] = 5.5

$TC_DPCE[ 1, 17 ] = 3  ; Cutting edge number CE
```

Note

If another cutting edge has been defined for the tool; e.g.
$TC_DPCE[1, 2] = 1 ; = CE,

D number 2 of cutting edge 1 cannot be addressed with the same name as the D number of cutting edge 3; i.e.
SETDNO(1, 1, 17)

returns FALSE.

© Siemens AG, 2003. All rights reserved

DZERO – Invalidate D numbers

The activation of this command invalidates all D numbers of the tools in the TO unit. It is no longer possible to activate an offset until valid D numbers are again available in the NCK. The D numbers must be assigned again with the command SETDNO.

The following tools must be defined (all with cutting edge number 1):

T1, D1	D no. of cutting edge CE=1
T2, D10	D no. of cutting edge CE=1
T3, D100	D no. of cutting edge CE=1

The following command is then programmed:

```
DZERO
```

If one of the offsets is now activated (e.g. with T3 D100), an alarm is generated, because D100 is not currently defined.

The D numbers are redefined with:

```
SETDNO( 1, 1, 100 )     ;T=1, cutting edge 1 is assigned the (new) D number 100
SETDNO( 2, 1, 10 )      ;T=2, cutting edge 1 is assigned the (old) D number 10
SETDNO( 3, 1, 1 )       ;T=3, cutting edge 1 is assigned the (new) D number 1
```

Note

In the event of a power failure, the DZERO command can leave the NCK in an undefined state with reference to the D numbers. If this happens, repeat the DZERO command when the power is returned.

Operating principle of a retooling program

Let us assume you want to ensure that the required tools and cutting edges are available. The status of the toolholding magazine of the NCK is arbitrary. The D numbers in the parts programs for the new machining operation generally do not match the D numbers of the actual cutting edges. The retooling program can have the following appearance:

```
DZERO       ; All D numbers of the TO unit are tagged as invalid
....        ; One or more loops over the locations of the magazine(s) to check
              the tools and their cutting edge numbers.
              If a tool is found which is not yet disabled ($TC_TP8) and has the
              desired cutting edge number CE (GETDNO), the new D number is
              allocated to the cutting edge (SETDNO).
....        ; Loading and unloading operations are performed.
              It is possible to work with the tool status 'to be unloaded' and 'to be
              loaded'.
CHKDNO      ; Loading/unloading and the operation for renaming D numbers are
              complete.
              Individual tools and/or D numbers can be checked, and collisions
              can be handled automatically according to the return value.
```

© Siemens AG, 2003. All rights reserved

2.1.5 Correction block for error during tool change (SW 5 and higher)

If a tool preparation has been programmed in the parts program and the NCK detects an error (e.g. the data set for the programmed T number does not exist in the NCK), the machining program is terminated with an alarm message (up to SW 4).

In SW 5 and higher, the user can assess the error situation (programming error in the parts program, tool data set not in NCK) and initiate appropriate measures before continuing with machining.

The tool change can be programmed in various ways depending on machine data MD 22550: TOOL_CHANGE_MODE:

TOOL_CHANGE_MODE = 0

```
T= 'T no.'  ; Tool preparation and tool change in one NC block;
            ; i.e. when T is programmed, a new D offset is activated
            ; in the NCK (see MD 20270: CUTTING_EDGE_DEFAULT)
```

TOOL_CHANGE_MODE = 1

```
T= 'T no.'  ; Tool preparation

M06         ; Tool change
            ; (the number of the tool change M code can be set)
            ; i.e. when M06 is programmed, a new D offset is activated
            ; in the NCK (see
            ; MD 20270: CUTTING_EDGE_DEFAULT)
```

The following problems can occur if tool management is not active:

- D offset data set missing
- Error in parts program

Note

The 'tool not in magazine' problem cannot be detected since the NCK did not have access to any magazine information during the tool compensation.

D offset data set missing

Program execution is interrupted at the block containing the invalid D value (independent of the value of machine data MD 22550). The operator must either correct the program or reload the missing data set.

To do this, he needs the D number for the flat D number function, or otherwise the T number as well. These parameters are passed together with the activated alarm 17181 (same function as 17180).

Error in parts program

The options for intervention in the event of an error depend on how the tool change was programmed (defined via machine data MD 22550: TOOL_CHANGE_MODE).

© Siemens AG, 2003. All rights reserved

Tool change with T programming (TOOL_CHANGE_MODE = 0)

In this case, the ‘correction block’ function available in the NCK is used. The NC program stops at the NC block in which a programmed T value error was detected. The ’correction block’ is executed again when the program is resumed.

The operator can intervene as follows:

- Correct the parts program.
- Reload the missing cutting edge offset data from the MMC.
- Include the missing cutting edge offset data in the NCK by ’Overstore’.

After the operator action, the START key is pressed and the block which caused the error is executed again. If the error was corrected, the program is continued. Otherwise, an alarm is output again.

Tool change with T and M06 programming (TOOL_CHANGE_MODE = 1)

In this case, an error is detected in the NC block containing the tool preparation (T programming), however this error is to be ignored initially. Processing continues until the tool change request (usually M06) is executed. The program is to stop at this point.

The programmed T address can contain any number of program lines ahead of the M06 command, or the two instructions can appear in different (sub)programs. For this reason, it is not possible to modify a block or a correction block which has already been executed.

The operator has the same intervention options as for TOOL_CHANGE_MODE = 0. It is possible to reload missing data.
In this case, however, T must be programmed with ’Overstore’.

If a program error has occurred, the line with the error cannot be corrected (Txx); only the line at which the program stopped and which generated the alarm can be edited (only if TOOL_CHANGE_ERROR_MODE Bit 0 = 1).

The sequence is as follows:

```
Txx     ; Error! Data set with xx does not exist
        ; Store status; store xx;
        ; Continue program

....

M06     ; Detect ’xx missing’ –> output alarm,
        ; Stop program
        ; Correct block, e.g. with Tyy M06, start,
        ; Block Tyy M06 is interpreted and is O.K.
        ; Continue processing.
```

© Siemens AG, 2003. All rights reserved

The following occurs when this part of the program is executed again:

```
Txx        ; Error! Data set with xx does not exist,
           ; store status; store xx;
           ; continue program
....
Tyy M06    ; Detect flag 'xx missing' –> reject without further reaction,
           ; because Tyy M06 is correct –> program does not stop (correct).
```

If necessary, the original point of the T call can be corrected after the end of the program. If the tool change logic on the machine cannot process this, the program must be aborted and the point of the error corrected.

If only one data set is missing, it is transferred to the NCK, Txx is programmed in 'Overstore' and the program is subsequently resumed.

As in the case of 'missing D number', the required parameter (T number) can be accessed by the user for 'missing T number' via the appropriate alarm (17191).

Note

During program testing, the program also stops immediately at the Txx block containing the error if TOOL_CHANGE_ERROR_MODE Bit 0=1, to allow the program to be corrected.

2.1.6 Definition of the effect of the tool parameters

MD 20360: TOOL_PARAMETER_DEF_MASK can control the effect of the tool parameters on the transverse axis in connection with diameter programming. Details are described with the mentioned MD (extended in SW 5.3 and higher).

DRF handwheel traversal with half distance (SW 6 and higher)

During DRF handwheel traversal, it is possible to move a transverse axis through only half the distance of the specified increment as follows:

Specify the distance with handwheel via
MD 11346: HANDWHEEL_TRUE_DISTANCE = 1

Define the DRF offset in the transverse axis as a diameter offset with bit 9 = 1 of MD 20360: TOOL_PARAMETER_DEF_MASK

When an axial DRF offset is deselected, an existing tool offset (handwheel override in tool direction) is also deleted.

Note

For further information about superimposed movements with the handwheel, please refer to:

References: /FB/ H1, "Jogging with the handwheel"
/PG/, "Programming Guide: Fundamentals"
The Programming Guide contains a complete technical description of axisspecific deselection of the DRF offset.

© Siemens AG, 2003. All rights reserved

2.2 Flat D number structure (SW 4 and higher)

With SW 4 and higher, simple tool management using D numbers is possible (no replacement tools, no magazines).

The function is available in the basic level of tool management (without tool management function activated) and is intended for turning machines. Grinding tools cannot be defined using this function.

Activation

The MD18102: **MM_TYPE_OF_CUTTING_EDGE** can be used to define the type of D number that is active:

- Value = 0 = as before = default setting
- Value = 1 = flat D number structure with absolute **direct D programming**

Cutting edges can be deleted individually via PI command or NC programming command.
Cutting edges with a specific number can also be created selectively using the MMC.

2.2.1 Creating a new D number (correction block)

The system variables $TC_DP1... $TC_DP25 can be used to program tool offsets. The contents have the same meaning as before.
The syntax changes: No T number is specified.

- Function 'flat D number' **active**:
  $TC_DPx[d] = value ; where x=parameter no., d=D number
  i.e., data with this syntax can only be loaded to NCK if the 'flat D number' function is activated.
- Function 'flat D number' **inactive**:
  $TC_DPx[t][d] = value ; with t=T number, d=D number

A D number can only be assigned once for each tool; i.e. each D number stands for exactly one offset data block.
A new data block is stored in the NCK memory when a D number that does not exist is created for the first time.

MD 18100 MM_NUM_CUTTING_EDGES_IN_TOA is used to set the maximum number of D or offset data blocks (max. 600).

Data backup

Data backup is carried out in the same format, i.e. a backup file with which the 'flat D number' function is created cannot be input to the NCK of a control that has not activated the function. The same also applies in reverse for transfer.

D number range

1 – 99 999 999

© Siemens AG, 2003. All rights reserved

2.2.2 Read D number from the PLC

The programmer has various options for specifying the D number in the parts program.

One option is to read it from the system parameter
$A_DNO[n] (n=1,...9)
("DNO" stands for D number.)

Example

Offset selection in the parts program
D9 is written in the program. With $A_DND[9], absolute D number stored in the 9th table location at the time of the call is read.

Table 2-1 System variable **$A_DNO[n]**; (n=1,...9)

Meaning	The variable reads the value of the nth position from a table containing the D numbers. The parameter is typically used for indirect D programming (via parameter): D=$A_DNO[1] The variable reads a field element (=the concrete D number) in the D number table of VDI. The PLC writes the matching D number for this T number to the field after the NCK has output the auxiliary function T to the PLC. Field elements have the value **zero**, if no D number has been defined for them. The PLC can store a maximum of 9 different D numbers for 1 T number in the VDI interface by the PLC. The D number themselves can take values between 0 and 3200, i.e. the table may also have gaps.
Data type	INT
Value range	1–99 999 999
Indices	n=1–9. Index n indicates the position in the D number table that is to be read.
Access	Read in parts program
Preprocessing stop	Yes
Synchronized action	Read in synchronized actions
Validity	Only relevant in conjunction with the 'flat D numbers' or tool management on PLC.

© Siemens AG, 2003. All rights reserved

2.2.3 D number programming

MD 18102 can be used to define how the D number is programmed.
D0 retains the previous significance 'Deselection of active offset in NCK'.

Extending the address of D

It is not possible to extend the address of D. Only one active offset data block is possible for the tool path at a given time.

Direct, absolute programming

Programming in the parts program is carried out as before. Only the value range of the programmed D number is increased.

Example 1:
MD 22550: TOOL_CHANGE_MODE = 0
MD 18102: MM_TYPE_OF_CUTTING_EDGE = 1
MD 20270: CUTTING_EDGE_DEFAULT = -1

```
...
D92
X0          Traverse with the offset values of D92
T17         Outputs T=17 to PLC
X1          Traverse with the offset values of D92
D16
X2          Traverse with the offset values of D16
D32000
X3          Traverse with the offset values of D32000
T29000      Outputs T=29000 to PLC
X4          Traverse with the offset values of D32000
D1
X5          Traverse with the offset values of D1
...
```

Example 2:

```
MD 22550 = 0
T1
T2
T3
D777        No waiting, D777 is activated,
            T3 = programmed and active tool in the display
            D777 = programmed and active offset
```

Note

- Tool change and
- the assignment of a D offset for a concrete tool are not the responsibility of NCK.

© Siemens AG, 2003. All rights reserved

Indirect parameterized program

D=$A_DNO[n]
means: Select the D number entered at position n (=1 to 9) in the VDI D number table.
The variable $A_DNO is read during run-up with a forced synchronization beforehand in the main run.

The D numbers available in the VDI interface can be read with $A_DND.

Note

It must be guaranteed beyond doubt that the PLC has written the D numbers that match the previously programmed T value, before the NCK main run reads the D number. This mechanism (in the NCK) initiates a general wait for PLC in the NCK.

Indirect indexed programming

Dn or **D=n**
means: Select the D number entered at position n (=1 to 9) in the VDI D number table. This syntax is identical to the conventional one, but it internally activates the offset block for the D number determined for index n.

Example for the indirect D programming sequence:
Let MD 20270 be 0
Let MD 22550 be 0
Let spindle no. 2 not be the master spindle

Parts programs	Action
T 4	Tool change command: NCK outputs the value 4 as T auxiliary function. PLC evaluates it and provides the associated D numbers in the VDI. The PLC only does this if the address extension received is the number of the master spindle. A simple communication protocol allows the NCK to detect whether the associated D numbers are available for the subsequent programmed D.
X1	
...	
D1	The NCK feed is synchronized with the main run, checks whether the new D numbers are present in the VDI (and waits if necessary) and accepts the desired (absolute) D number at position 1 (D1); e.g. the number 4711. NCK now determines the offset block 4711 and calculates the geometry. The NC transfers the complete contents of the D number table to the VDI interface (this always happens when a tool change command is detected).
D=$P_DNO[2]	Contents correspond to the programming of D2 (for indirect programming)
D3	The NC accesses the absolute D number of location 3; a synchronized action is not required, because a new T has not been programmed.
T2 = 12000000	The NC does not detect a tool change, because spindle no. 2 is not the number of the master spindle.
D4	The NC accesses the absolute D number of location 4; a synchronized action is not required, because a new T has not been programmed.

© Siemens AG, 2003. All rights reserved

Before outputting the T number to the PLC, the NCK stores the status of the VDI with reference to the queued D number(s). A counter is contained in the VDI for this purpose. The PLC increments the counter each time the D number is refreshed. If necessary the next D number request from the NCK waits until the counter value has changed. The VDI then contains the new D number table which belongs to the value previously programmed.

!

Important

D can only be programmed without an address extension. D always refers to the master spindle. T can be programmed with an address extension.
The PLC must only write the D offsets to the VDI with T values or M6 commands with reference to the master spindle. The NCK assumes that this is the case when performing synchronization actions during reading of the D numbers from the VDI.

Example

```
MD 22550: TOOL_CHANGE_MODE = 1
MD 22560: TOOL_CHANGE_M_CODE = 6
                    ; i.e. tool change with M6 programming
                      let spindle number 3 not be master spindle
T1
M6                  ; Tool change command
T3
T3 = 11
D = 2               ; Wait until the counter in the VDI has changed; this
                      indicates that the PLC has written the D values for T1 into
                      the D number table (e.g. at location 2 = 4711;
                      then activate offset 4711 in the NCK)
                      T3 = programmed T,
                      T1 = active T in the display
```

Delete D no. via parts program

- **With** flat D number
  $TC_DP1[d] = 0
  Offset block is deleted with the number d in NCK.
  The memory is then free for the definition of another D number.
- **Without** flat D number
  $TC_DP1[t][d] = 0 =
  Cutting edge of tool t is deleted.
- $TC_DP1[0] = 0
  Delete all D offsets in NCK.

Active offset data blocks (D numbers) cannot be deleted. It may therefore be necessary to program D0 before deleting.

© Siemens AG, 2003. All rights reserved

Tool MDs

The following machine data affect the way tools and cutting edges (D numbers) work in the NCK:

- MD20270: CUTTING_EDGE_DEFAULT
- MD20130: CUTTING_EDGE_RESET_VALUE
- MD20120: TOOL_RESET_VALUE
- MD20121: TOOL_PRESEL_RESET_VALUE
- MD22550: TOOL_CHANGE_MODE
- MD22560: TOOL_CHANGE_M_CODE
- MD 20110: RESET_MODE_MASK
- MD20112: START_MODE_MASK

2.2.4 Programming the T number

When the function “Flat D number structure” is active, NC address T continues to be evaluated, i.e. the programmed T number and the active T number are displayed. However, the NC determines the D number without reference to the programmed T value.

The NC recognizes 1 master spindle per channel (spindle number settable via MD).
Offsets and the M6 command (tool change) are only calculated with reference to the master spindle.

An address extension T is interpreted as a spindle number (e.g. T2 = 1; tool 1 to be selected on spindle 2); a tool change is only detected if spindle 2 is the master spindle.

2.2.5 Programming M6

The NC recognizes 1 master spindle per channel (spindle number settable via MD).
Offsets and the M6 command (tool change) are only calculated with reference to the master spindle.

MD 22550 defines whether the tool change command is performed with an M function. T is used as the tool preparation command. MD 22560 specifies the name of the M function for the tool change. The default is M6. An address extension of M6 is interpreted as a spindle number.

Example

Two spindles are defined, spindle 1 and spindle 2 and the following applies:

```
MD 20090 = 2    ; Spindle 2 is the master spindle.
M6              ; Tool change desired, command refers implicitly to
                  master spindle
M1 = 6          ; No tool change because spindle no. 2 is the master spindle
M2 = 6          ; Tool change performed because spindle no. 2 is the master
                  spindle
```

© Siemens AG, 2003. All rights reserved

2.2.6 Program test

MD 20110, bit 3 can be used to define that the active tool and the tool offset are transferred as follows:

- Bit 3=1 from the last test program to finish in test mode, or
- Bit 3=0 from the last program to finish before activation of the program test

Precondition Bits 0 and 6 of MD 20110 must be enabled.

2.2.7 Tool management or “Flat D numbers”

Toolrelated management

The active tool management of the NCK works on the basis of the following assumptions:

1. Tools are managed in magazines.
2. Cutting edges are monitored; limits reached cause the tool to be disabled.
3. Idea behind replacement tools: Tools are programmed for selection only the on the basis of their identifier. NCK then selects the concrete tool according to the strategy.

This means that the tool management only makes sense when concrete tools have been defined and these are to be utilized by the NCK.

Flat D number

Flat D number means that the tool management is carried out outside the NCK and there is no reference made to T numbers.

No mixture
of tool management and flat D No.

It does not make sense to mix or distribute the tool management functions over the NCK and PLC, since the main reason for tool management on the NCK is **to save time**.
This only works if the tasks that are timecritical are carried out on the NCK. This is not the case for ’flat D number’ however.

Note

Activation of both of the functions “Flat D number structure” and “Tool management” is monitored. If both are activated at the same time, “Tool management” takes priority.

© Siemens AG, 2003. All rights reserved

2.3 Tool cutting edge

Tool cutting edge

The following data are used to describe a tool cutting edge uniquely:

- Tool type (end mill, drill, etc.)
- Geometrical description
- Technological description

Tool parameter

The geometrical description, the technological description and the tool type are mapped to tool parameters for each type of tool.
The following tool parameters are provided for this purpose:

Tool parameter	Meaning	Notes	Reserved for extensions
1	Tool type		
2	Cutting edge position	Only for turning tools	
Geometry tool length compensation			
3	Length 1		
4	Length 2		
5	Length 3		
Geometry tool radius compensation			
6	Radius 1/Length 1	For 3D face milling see Subsection 2.3.4	
7	Length 2	For 3D face milling	
8	Radius 1	For 3D face milling	
9	Radius 2	For 3D face milling	
10	Angle 1	For 3D face milling	
11	Angle 2	For 3D face milling	
Wear tool length compensation			
12	Length 1		
13	Length 2		
14	Length 3		
Wear tool radius compensation			
15	Radius 1/Length 1	For 3D face milling see Subsection 2.3.4	
16	Length 2	For 3D face milling	
17	Radius 1	For 3D face milling	
18	Radius 2	For 3D face milling	
19	Angle 1	For 3D face milling	
20	Angle 2	For 3D face milling	
Base dimension/adapter dimension tool length compensation			
21	Basic length 1		
22	Basic length 2		
23	Basic length 3		
Technology			
24	Tool clearance angle	Only for turning tools	
25			Reserved

Reserved ... means that this tool parameter of the 840D/810D and FM-NC is not used (reserved for extensions).

Milling cutter types 111, 120, 121, 130, 155, 156 and 157 3D are given special treatment for face milling by evaluating tool parameters (1 – 23). For further information about different tool types see Subsection 2.3.1 Tool type (tool parameter) and

References /FB/, W5, “3D tool radius compensation”

© Siemens AG, 2003. All rights reserved

2.3.1 Tool type (tool parameters)

Meaning

A 3-digit number is used to define the tool type. The operator/machine setter/programmer selects the tool type. This determines further components such as geometry, wear and tool base dimensions in advance. The tool type has no meaning in the grinding and turning tool groups. Non-listed numbers are also permitted, in particular with grinding tools (400–499).

Table 2-2 Table of the main tool parameters available (only the tool type is necessary)

Tool type	CLDATA*1 "cutter Location Data" Cutter location data to DIN 66215, Bl 1 SUG*2 Grinding wheel peripheral speed	Cutting edge position	Geometry length 1	length 2	length 3	Geometry radius length	–Length	–Radius	–Wheel radius min.	–Wheel width min/curr.	–Angle of inclined wheel	Wear length	Wear radius	Tool base dimension, length 1 – 3	–Peripheral speed	Maximum speed	Tool clearance angle
Milling tools and drilling tools (all other)																	
100	Milling tool according to CLDATA*1		x	x	x			x				x	x	x			
110	Ball end mill		x	x	x			x				x	x	x			
120	End mill		x					x				x	x	x			
121	End mill with corner rounding		x					x				x	x	x			
130	Angle head mill		x	x	x			x				x	x	x			
131	Angle head mill with corner rounding		x	x	x			x				x	x	x			
140	Face mill		x					x				x	x	x			
145	Thread mill		x					x				x	x	x			
150	Side mill		x					x				x	x	x			
155	Truncated cone mill		x	x	x			x				x	x	x			
200	Twist drill		x									x		x			
205	Rev. tip drill		x									x		x			
210	Boring rod		x									x		x			
220	Center drill		x									x		x			
230	Countersink		x									x		x			
231	Counterbore		x									x		x			
240	Tap/rough thread		x									x		x			
241	Tap/fine thread		x									x		x			
242	Tap/Whitworth thread		x									x		x			
250	Reamer		x									x		x			
Grinding tools and turning tools (400 – 599)																	
400	Surface grinding wheel		x	x	x		x	x			x	x	x	x	x		
401	Surface grinding wheel with monitoring		x	x	x		x	x	x	x	x	x	x	x	x	x	
403	Same as 401 but without tool base dimension for GWPS*2		x	x	x		x	x	x	x	x	x	x	x		x	
410	Face grinding wheel		x	x	x		x	x				x	x	x	x		
411	Face grinding wheel with monitoring		x	x	x		x	x	x	x		x	x	x	x	x	
413	Same as 411 but without tool base dimension for GWPS*2		x	x	x		x	x	x	x		x	x	x		x	
490	Dresser		x	x	x		x	x				x	x	x			
500	Roughing tool	x	x	x	x			x				x	x	x			x
510	Finish cutting tool	x	x	x	x			x				x	x	x			x
520	Recessing tool	x	x	x	x			x				x	x	x			x
530	Cutoff tool	x	x	x	x			x				x	x	x			x
540	Thread cutting tool	x	x	x	x			x				x	x	x			x
Special tools (700)																	
700	Slotting saw		x	x	x							x		x			

© Siemens AG, 2003. All rights reserved

Special points to be noted

- The tool type must be specified for each tool cutting edge.
- The tool type is only permitted to accept the defined values (see Table 2–1)
- Tool type 0 (zero) means that no valid tool is defined.

Tool offset data for slotting saw type

The following offset data (TOA data) can be specified for slotting saw tool type (type 700):

	Geometry	Wear	Base	
Length compensation Length 1	$TC_DP3	$TC_DP12	$TC_DP21	mm
Length 2	$TC_DP4	$TC_DP13	$TC_DP22	mm
Length 3	$TC_DP5	$TC_DP14	$TC_DP23	mm
Saw blade compensation Diameter d	$TC_DP6	$TC_DP15		mm
Slot width b	$TC_DP7	$TC_DP16		mm
Projection k	$TC_DP8	$TC_DP17		mm

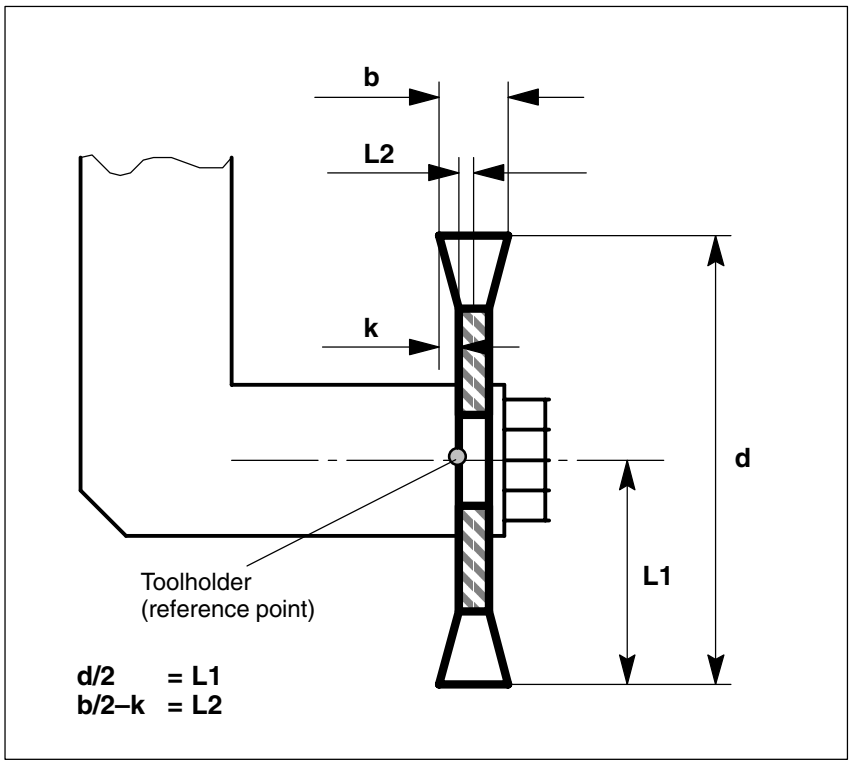


Fig. 2-3 Geometry of slotting saw (analogous to angle head cutter)

The width of the saw blade is taken into account with G40 to G42 (G40 no compensation for saw blade, G41 saw blade compensation left, G42 saw blade compensation right).

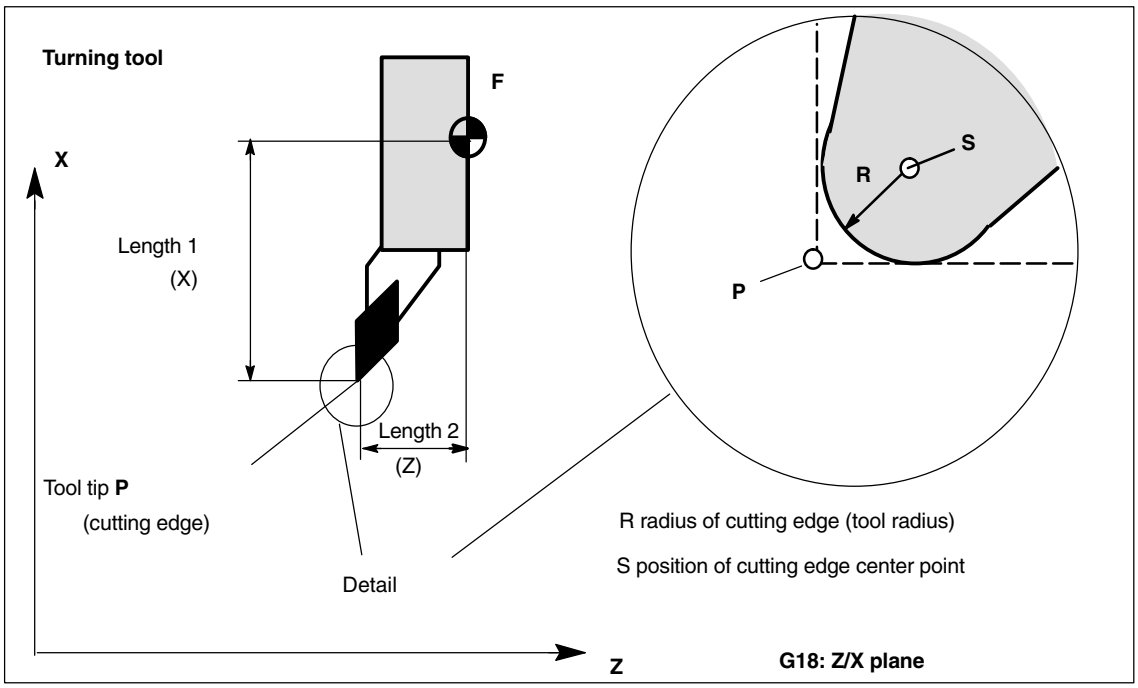
© Siemens AG, 2003. All rights reserved

2.3.2 Tooledge position (tool parameter 2)

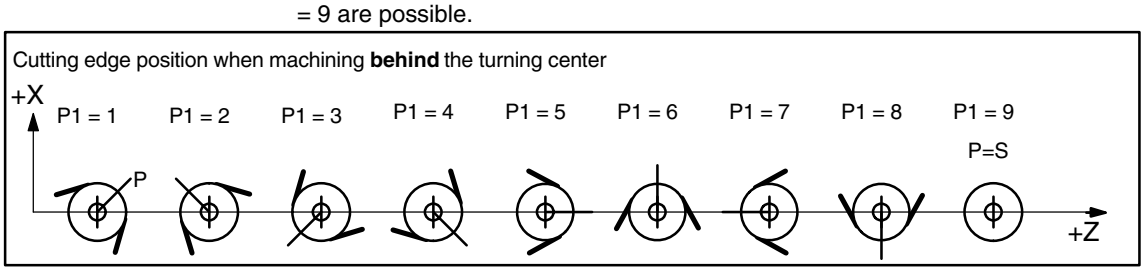
Meaning

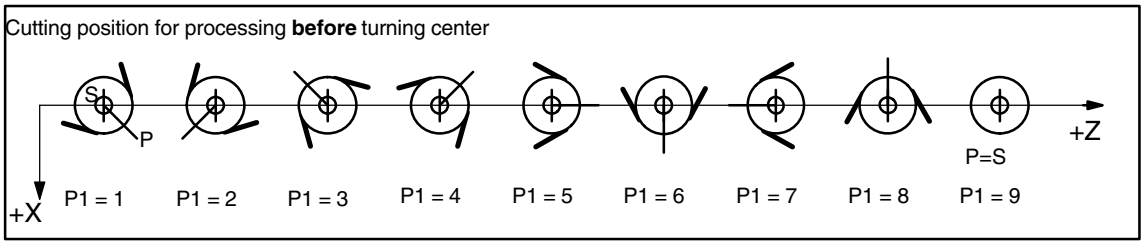
On turning tools (tool type 5xx), the control requires the cutting edge radius (tool parameter 8) and a cutting edge position. The cutting edge radius describes how the turning tool is fixed in the toolholder. As illustrated in Fig. 2-4), the position of point P relative to the center point S of the cutting edge is entered as an identifier (value between 1 and 9) in the cutting edge position (tool parameter 2).

Fig. 2-4 Definition of the cutting edge position on turning tools



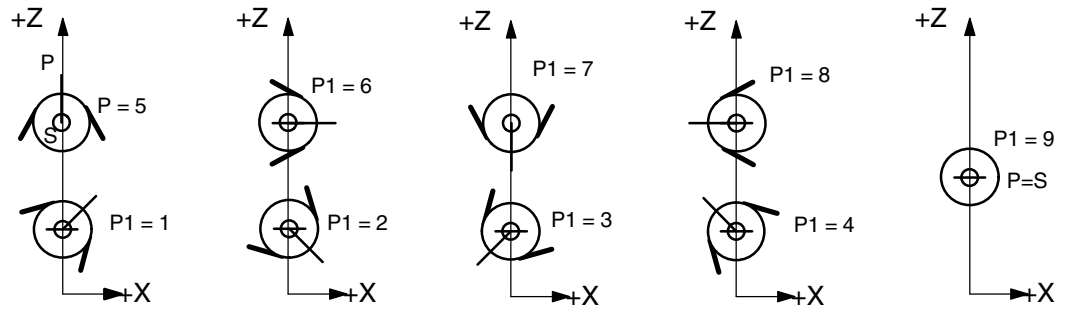
The tool parameter (P2) specifies the position of the cutting edge. P2 = 1 to P2 = 9 are possible.





© Siemens AG, 2003. All rights reserved

Cutting edge position on vertical turning machine



Special feature

- If the cutting edge center point S is used instead of point P as a reference point to calculate the tool length compensation, the identifier 9 must be entered for the cutting edge position.
- The identifier 0 (zero) is not permitted as a cutting edge position.

2.3.3 Geometry tool length compensation (tool parameters 3 to 5)

Tool length 1 to 3

The lengths of the tools for the tool length compensation are entered as tool lengths 1 to 3 (tool parameters 3 to 5). The following basic length specifications are required to be entered, according to the tool type:

- Tool type 12x, 140, 145, 150: Tool length 1
- Tool type 13x: Tool length 1–3 (dep. on plane G17–G19)
- Tool type 2xx: Tool length 1
- Tool type 5xx: Tool length 1–3

Important

Irrespective of the tool type, all three tool parameters 3 to 5 (tool lengths 1 to 3) are calculated into the three geometry axes.
If more tool lengths are entered in tool parameters 3 to 5 than are required for a particular tool type (tool length 1 is only required for tool type 2xx), these additional tool lengths are calculated into the geometry axes without an alarm being output.

© Siemens AG, 2003. All rights reserved

Note

Please refer to the following documentation for information concerning the entry of the tool size (lengths) in tool parameters 3 to 4 (tool lengths 1 to 3) and how they are calculated into the three geometry axes (on planes G17 to G19):
References: /BA/, Operator's Guide

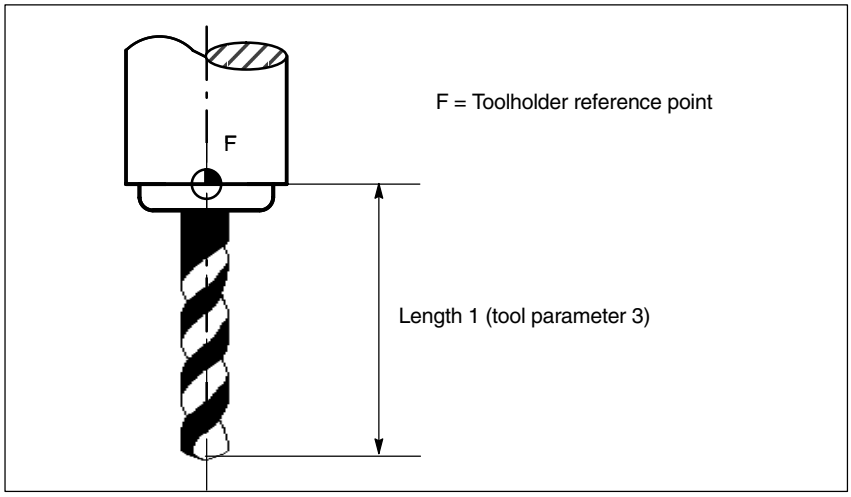


Fig. 2-5 Twist drill (tool type 200) with tool length (tool parameter 3)

Special feature

The active size of the tool is only defined when the geometry tool length compensation (tool parameters 3 to 5) and the wear tool length compensation (tool parameters 12 to 14) are added together (see Subsection 2.2.5). The base dimension/adapter dimension is also added (see Subsection 2.2.7) in order to calculate the complete tool length compensation into the geometry axes.

© Siemens AG, 2003. All rights reserved

2.3.4 Geometry tool radius compensation (tool parameters 6 to 11)

The geometry tool radius compensation defines the shape of the tool.

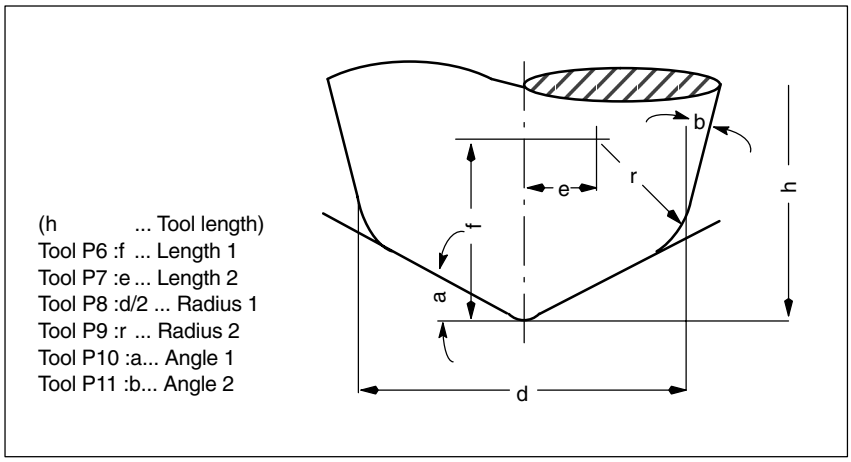


Fig. 2-6 Description of the tool geometry

Note

The tool description as given in Fig. 2-6 is required only for 3D face milling.
Otherwise:
Of the parameters from 6 to 11, SINUMERIK 840D/810D and FM-NC only uses tool parameter 6 (tool radius 1).
Please refer to the following documentation for information concerning the entry of the tool shape (radius for the tool radius compensation) in tool parameters 6 to 11 and how they are calculated into the three geometry axes:
References: /PG/, “Programming Guide: Fundamentals”
/FB/, W5, “3D Tool Radius Compensation”

Tool Length 1

Not used (see Fig. 2–5).

Tool Length 2

Not used (see Fig. 2–5).

Tool radius 1

The tool radius must be entered for the following tool types in tool parameter 6 (tool radius 1):

- Tool type 1xx Milling tools
- Tool type 5xx Turning tools

A tool radius is not required to be entered for drilling tools (tool type 2xx). The cutting edge position is required to be entered additionally for turning tools (tool type 5xx, see Subsection 2.2.2).

The tool radius compensation is used to calculate the tool radius into the current plane (G17 to G19), see Section 2.3.

Tool radius 2
Tool angle 1
Tool angle 2

Not used (see Note)

© Siemens AG, 2003. All rights reserved

2.3.5 Wear tool length compensation (tool parameters 12 to 14)

Meaning

While the geometry tool length compensation (tool parameters 3 to 5) is used to define the size of the tool, the wear tool length compensation can be used to correct the change in the active tool size. The active tool size can change due to:

- Differences between the tool fixture on the tool measurement machine and the tool fixture on the machine tool.
- The wear on the tool caused during the service life by machining.
- Definition of a finishing allowance.

Active tool size

The geometry tool compensation (tool parameters 3 to 5) and the wear tool length compensation (tool parameters 12 to 14) are added together (geometry tool length 1 is added to wear tool length 1, etc.) to arrive at the size of the active tool.

2.3.6 Wear tool length compensation (tool parameters 15 to 20)

Meaning

While the geometry tool radius compensation (tool parameters 6 to 11) is used to define the shape of the tool, the wear tool radius compensation can be used to correct the change in the active tool shape.

The active tool size can change due to:

- The wear on the tool caused during the service life by machining.
- Definition of a finishing allowance.

Active tool shape

The geometry tool radius compensation (tool parameters 6 to 11) and the wear tool radius compensation (tool parameters 15 to 20) are added together (geometry tool radius 1 is added to wear tool radius 1, etc.) to arrive at the shape of the active tool.

© Siemens AG, 2003. All rights reserved

2.3.7 Base dimension/adapter dimension tool length compensation (tool parameters 21 to 23)

Meaning

The base dimension/adapter dimension can be used when the reference point of the tool holder (tool size) is different from the reference point of the toolholder. This is the case when:

- The tool and the tool adapter are measured separately but are installed on the machine in one unit (the tool size and adapter size are entered separately in a cutting edge).
- The tool is used in a second tool fixture located at another position (e.g. vertical and horizontal spindle).
- The tool fixtures of a tool turret are located at different positions.

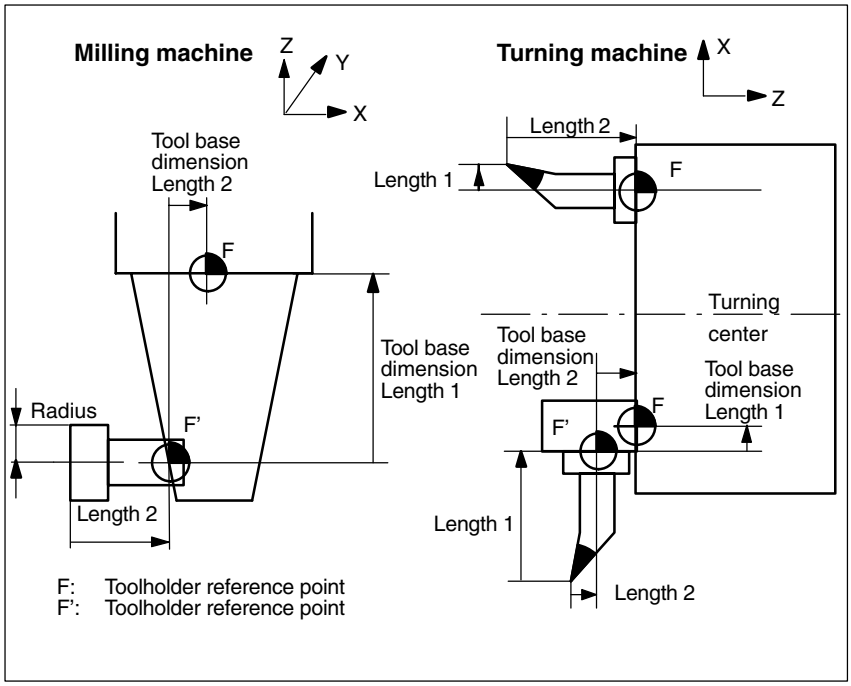


Fig. 2-7 Example applications for base dimension/adapter dimension TLC

Basic length 1 to 3

In order that the discrepancy between the toolholder reference point F and the toolholder reference point F' can be corrected on the three geometry axes (three dimensional), all 3 basic lengths are active irrespective of the tool type. In other words, a twist drill (tool type 200) with a tool length compensation (length 1) can also have a base dimension/adapter dimension in 3 axes.

Note

Please refer to the following documentation for more information on the base/adapter dimension tool length compensation.
References: /PG/, "Programming Guide: Fundamentals"

© Siemens AG, 2003. All rights reserved

2.3.8 Technology – tool clearance angle (tool parameter 24)

Meaning

Certain turning cycles, in which traversing motions with tool clearance are generated, monitor the tool clearance angle of the active tool for possible contour violations.

Value range

The angle (0 to 90 degrees with no leading sign) is entered in tool parameter 24 as the tool clearance angle.

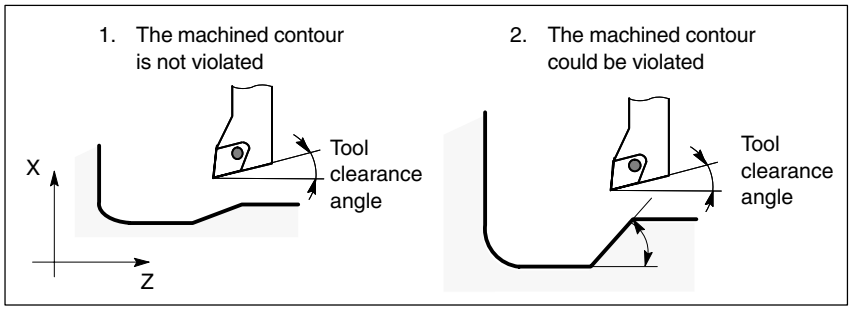


Fig. 2-8 Tool clearance angle of the turning tool

Longitudinal/face

The tool clearance angle is entered differently, according to the type of machining (longitudinal or face). If a tool is to be used for both longitudinal and face machining, two cutting edges must be entered for different tool clearance angles.

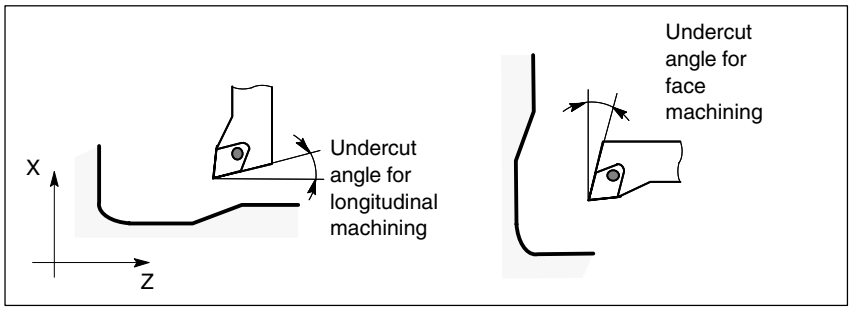


Fig. 2-9 Tool clearance angle for longitudinal and face machining

Note

If zero is entered for the tool clearance angle (tool parameter 24), there is no monitoring of relief cutting in the turning cycles.
Please refer to the following documentation for more information on the tool clearance angle.
References: /PAZ/, “Programming Guide: Cycles”

© Siemens AG, 2003. All rights reserved

2.3.9 Tools with relevant tool point direction (SW 5 and higher)

Up to SW 4.x

The change from G40 to G41/G42 or vice versa is treated as a tool change for tools with relevant tool point direction. This causes a preprocessor stop when a transformation (e.g. TRANSMIT) is active, and thus to deviations from the intended part contour.

SW 5 and higher

The following changes have been made:

1. The change from G40 to G41 / G42 or vice versa is no longer treated as a tool change. A preprocessor stop therefore no longer occurs with Transmit.
2. The straight line between the cutting edge center points at the start and end of the block is used to calculate intersections with the approach and retraction block. The difference between the tool reference point and the cutting edge center point is superimposed on this movement.
   For approach and/or retraction with KONT, the movement is superimposed in the linear subblock of the approach or retraction movement. Differences to the previous behavior occur only in relatively seldom circumstances, if the approach and/or retraction blocks do not intersect with an adjacent block (see figure below).

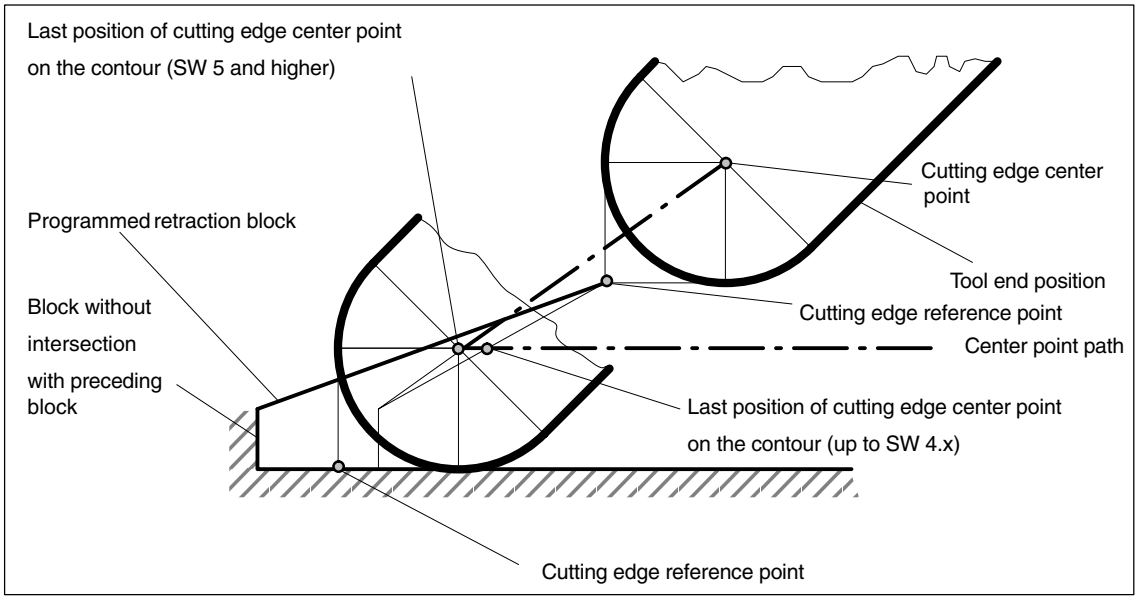


Fig. 2-10 Retraction behavior for tool with relevant tool point direction

© Siemens AG, 2003. All rights reserved

3. In circle blocks and in traversing blocks with rational polynomials with a denominator degree > 4, it is not permitted to perform a tool change with active tool radius compensation where the distance between the cutting edge center point and the cutting edge reference point changes. With other types of interpolation, a change is permitted even when a transformation (e.g. Transmit) is active, in contrast to previous versions.
4. In the case of tool radius compensation with variable tool orientation, the transformation from the cutting edge reference point to the cutting edge center point can no longer be implemented with a simple zero offset. Tools with relevant tool point direction are therefore prohibited for 3D circumferential milling (alarm).

Note

The issue is not relevant for face milling, since, as before, only defined tool types without relevant tool point direction are approved. (A tool with a type which has not been explicitly approved is treated as a ball end mill with the specified radius. The specification of a tool point direction is ignored.)

© Siemens AG, 2003. All rights reserved

2.4 Tool radius compensation: 2D (TRC)

Note

Please refer to the following documents for information on tool radius compensation (TRC):
References: /PG/, “Programming Guide: Fundamentals”
Only the Programming Guide contains a complete technical description of the tool radius compensation (TRC).

Why TRC?

The contour (geometry) of the workpiece programmed in the parts program should be independent of the tools used in production. This makes it necessary to draw the values for the tool length and tool radius from a current offset memory. Tool radius compensation can be used to calculate the equidistant path to the programmed contour from the tool radius.

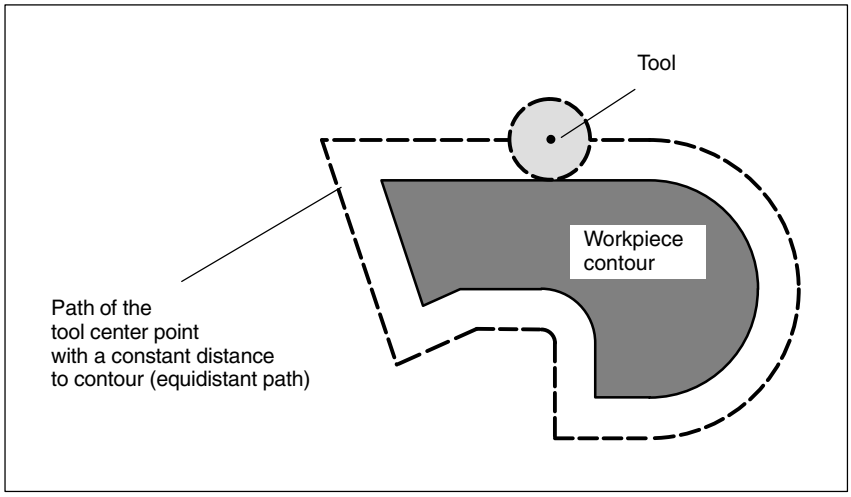


Fig. 2-11 Workpiece contour (geometry) with equidistant path

TRC on the plane

Tool radius compensation is active on the current plane (G17 to G19) for the following types of interpolation:

- Linear interpolation ... G0, G1
- Circle interpolation ... G2, G3, CIP
- Helical interpolation ... G2, G3
- Spline interpolation ... ASPLINE, BSPLINE, CSPLINE
- Polynomial interpolation ... POLY

© Siemens AG, 2003. All rights reserved

2.4.1 Selecting the TRC (G41/G42)

Tool radius compensation (TRC) calculates a path which is equidistant to the programmed contour. The compensation can be performed on the left or righthand side of the contour in the direction of motion.

- G41 ... TRC on the left-hand side of the contour in the direction of
- G42 ... TRC on the right-hand side of the contour in the direction of
- G40 ... Deselection of TRC (see Subsection 2.3.3)

Intermediate blocks

In general, only program blocks with positions on geometry axes in the current plane are programmed when tool radius compensation (TRC) is active. However, dummy blocks can still also be programmed with active TRC. Dummy blocks are program blocks which do not contain any positions on a geometry axis in the current plane:

- Positions on the infeed axis
- Auxiliary functions
- etc.

The maximum number of dummy blocks can be defined in MD 20250: CUTCOM_MAXNUM_DUMMY_BLOCKS (number of blocks with no traversing motion with TRC).

Special points to be noted

- Tool radius compensation can only be selected in a program block with G0 (rapid traverse) or G1 (linear interpolation).
- A tool must be loaded (T function) at the latest in the program block with the tool radius compensation selection and the tool cutting edge (tool offset) (D1 to D9) must be activated.
- Tool radius compensation is not selected with a tool cutting edge/tool offset of D0.
- If only one geometry axis is programmed on the plane when the tool radius compensation is selected, the second axis is automatically added on the plane (last programmed position).
- If no geometry axis is programmed for the current plane in the block with the tool radius compensation selection, no selection takes place.
- When tool radius compensation is deselected (G40) in the block following the tool radius compensation selection, no selection takes place.
- When tool radius compensation is selected, the approach response is determined by the NORM/KONT (see Subsection 2.3.2).

© Siemens AG, 2003. All rights reserved

2.4.2 Approach and retraction behavior (NORM/KONT)

The instructions NORM and KONT can be used to control the approach response (selection of tool radius compensation with G41/42) and the retraction response (deselection of tool radius compensation with G40):

NORM ... Normal setting at start point/end point (initial setting)
KONT ... Follow contour at start point/end point

Special points to be noted

- KONT only differs from NORM when the tool start position is behind the contour.

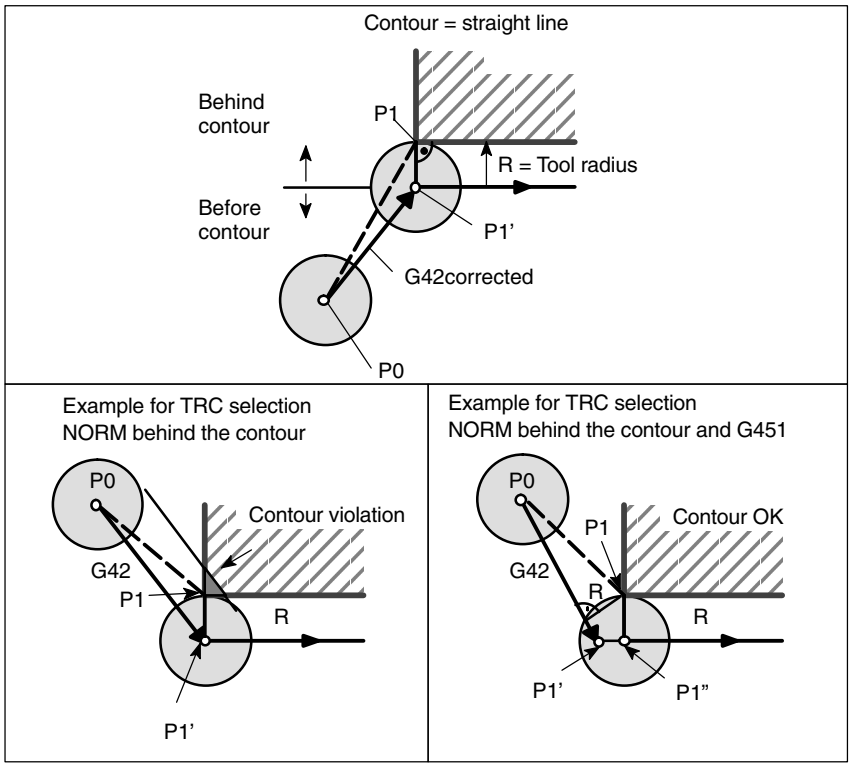


Fig. 2-12 Example for selecting TRC with KONT or NORM in front of and behind the contour

- KONT and G450/G451 (corner response at outer corners) has a general effect and determines the approach and retraction response with TRC.
- When tool radius compensation is deselected, the retraction response is determined by the KONT/NORM instructions (see Subsection 2.3.2).

© Siemens AG, 2003. All rights reserved

2.4.3 Smooth approach and retraction (SW 4.3 and higher)

Meaning

The soft approach and retraction function (SAR) is used to achieve a tangential approach to the start point of a contour, regardless of the position of the start point.

The approach response can be varied and adapted to special needs using a range of additional parameters.

The two functions, soft approach and soft retraction, are largely symmetrical. The following Section is therefore restricted to a detailed description of approach; special reference is made to differences that occur with retraction.

Approach movement

A maximum of 4 component movements:

- Start point of the movement P_0
- Intermediate points P_1, P_2, and P_3
- End point P_4

Points P_0, P_3 and P_4 are always defined. Intermediate points P_1 and P_2 can be omitted, according to the parameters defined and the geometrical conditions.

Retraction movement

On retraction, the points are crossed in the reverse direction, i.e. starting at P_4 and ending at P_0.

Parameters

The response of the soft approach and retraction function is determined by up to 9 parameters:

- **Nonmodal G code for defining the approach and retraction contour.**
  This G code may not be omitted.
  - **G147:** Approach on straight line
  - **G148:** Retraction on straight line
  - **G247:** Approach on quadrant
  - **G248:** Retraction on quadrant
  - **G347:** Approach on semicircle
  - **G348:** Retraction on semicircle

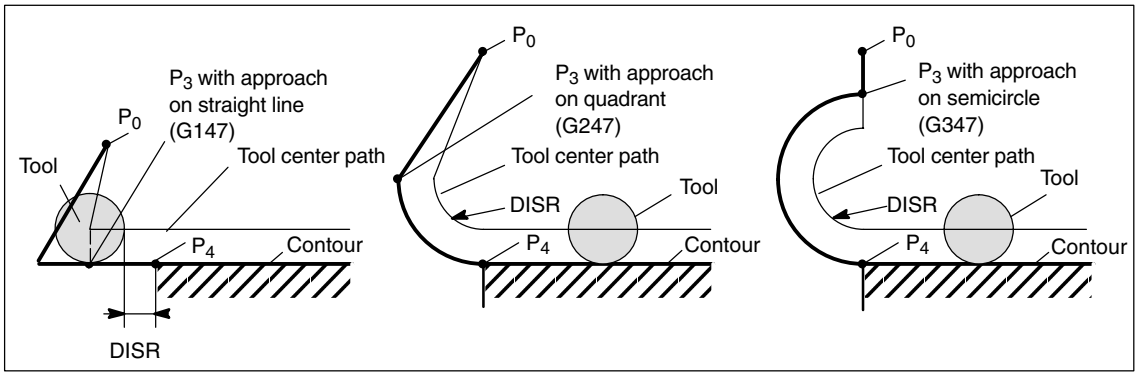


Fig. 2-13 Approach response depending on G147 to G347 and DISR (on simultaneous activation of tool radius compensation)

© Siemens AG, 2003. All rights reserved

- **Modal G code for defining the approach and retraction contour.**

  This G code is only relevant if the approach contour is a quadrant or semicircle. The approach and retraction direction can be determined as follows:

  – **G140:** Defining the approach and retraction direction using active tool radius compensation. (G140 is the initial setting)

    With positive tool radius: G41 active → approach from left
    G42 active → approach from right

    If no tool radius compensation is active (G40), the response is identical to G143. In this case, an alarm is not output. If the radius of the active tool is 0, the approach and retraction side is determined as if the tool radius were positive.

  – **G141:** Approach contour from left, or retract towards the left.

  – **G142:** Approach contour from right, or retract towards the right.

  – **G143:** Automatic determination of the approach direction, i.e. the con tour is approached from the side where the start point is located, relative to the tangent at the start point of the following block (P_4).

Note

The tangent at the end point of the preceding block is used accordingly on **retraction**. If the end point is not programmed explicitly on retraction, i.e. if it is to be determined implicitly, G143 is not permitted on retraction, since there is a mutual dependency between the approach side and the position of the end point. If G143 is programmed in this case, an alarm is output. The same applies if, when G140 is active, an automatic switchover to G143 takes place as a result of an inactive tool radius compensation.

- **Modal G code (G340, G341) which defines the subdivision of the movement into individual blocks from the start point to the end point.**

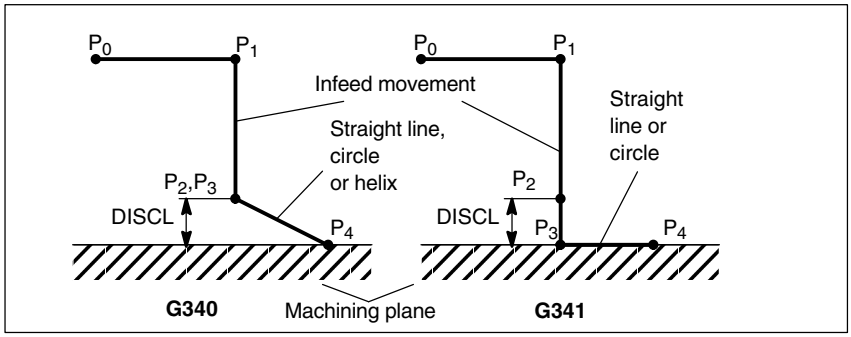


Fig. 2-14 Sequence of the approach movement depending on G340/G341

© Siemens AG, 2003. All rights reserved

– **G340:** The approach characteristic from P_0 to P_4 is shown in Fig. 2-14.

  If G247 or G347 is active (quadrant or semicircle) and start point P_3 is outside the machining plane defined by the end point P_4, a helix is inserted instead of a circle. Point P_2 is not defined or coincides with P_3.

  The circle plane or the helix axis is determined by the plane which is active in the SAR block (G17 – G19), i.e. the projection of the start tangent is used by the following block, instead of the tangent itself, to define the circle.

  The movement from point P_0 to point P_3 takes place along two straight lines at the velocity which was valid before the SAR block.

– **G341:** The approach characteristic from P_0 to P_4 is shown in Fig. 2-14.

  P_3 and P_4 are located within the machining planes, with the result that a circle is always inserted instead of a helix with G247 or G347.

Note

Active, rotating frames are included in all cases where the position of the active plane G17 – G19 (circle plane, helix axis, infeed movements perpendicular to the active plane) is relevant.

© Siemens AG, 2003. All rights reserved

- **DISR: Specifies the length of a straight approach line or the radius of an approach arc.** (See Fig. 2-13)

  On approach/retraction along a straight line, DISR specifies the distance from the cutter edge to the start point of the contour, i.e. the length of the straight line with active TRC is calculated as the total of the tool radius and the programmed value of DISR.

  An alarm is output on approach and retraction with straight lines:

  – If DISR is negative and the amount is greater than the tool radius (the length of the resulting approach line is less than or equal to zero).

  With circles, DISR always specifies the radius of the tool center path. If tool radius compensation is activated, a circle is generated internally, the radius of which is dimensioned such that the tool center path is derived, in this case also, from the programmed radius.

  An alarm is output on approach and retraction with circles:

  – If the radius of the circle generated internally is zero or negative,

  – if DISR is not programmed, or

  – if it has a value ≤ 0.

- **DISCL: Specifies the distance from point P_2 to the machining plane.** (See Fig. 2-14)

  If the position of point P_2 is to be specified by an absolute reference on the axis perpendicular to the circle plane, the value must be programmed in the form DISCL = AC(....).

  If DISCL is not programmed, points P_1, P_2 and P_3 are identical with G340 and the approach contour is generated from P_1 to P_4.

  The system monitors that the point defined by DISCL lies between P_1 and P_3, i.e. in all movements which have a component perpendicular to the machining plane (e.g. infeed movements, approach movements from P_3 to P_4), this component must have the same leading sign. It is not permitted to change direction. An alarm is output if this condition is violated.

  On detection of the direction reversal, a defined tolerance in MD 20204: WAB_CLEARANCE_TOLERANCE is allowed. However, if P_2 is outside the range defined by P_1 and P_3 and the deviation is less than or equal to this tolerance, it is assumed that P_2 is in the plane defined by P_1 and/or P_3.

  Example:
  An approach is made with G17 starting at position Z=20 of point P_1. The SAR plane defined by P_3 is at Z=0. The point defined by DISCL must therefore lie between these two points. MD 20204=0.010. If P_2 is between 20.000 and 20.010 or between 0 and –0.010, it is assumed that the value 20.0 or 0.0 is programmed. The alarm is output if the Z position of P_2 is greater than 20.010 or less than –0.010.

  Depending on the relative position of start point P_0 and end point P_4 with reference to the machining plane, the infeed movements are performed in the negative (normal for approach) or positive (normal for retraction) direction, i.e. with G17 it is possible for the Z component of end point P_4 to be greater than that of the start point P_0.

- **Programming the end point P_4 (or P_0 for retraction) generally with X... Y... Z....**

  – Possible ways of programming the end point P_4 for approach:
  End point P_4 can be programmed in the actual SAR block.

© Siemens AG, 2003. All rights reserved

P_4 can be determined by the end point of the next traversing block.

Further blocks (dummy blocks) can be inserted between the SAR block and the next traversing block without moving the geometry axes.

The end point is deemed to have been programmed in the actual SAR block for approach if at least one geometry axis is programmed on the machining plane (X or Y with G17). If only the position of the axis perpendicular to the machining plane (Z with G17) is programmed in the SAR block, this component is taken from the SAR block, but the position in the plane is taken from the following block. In this case, an alarm is output if the axis perpendicular to the machining plane is also programmed in the following block.

Example:

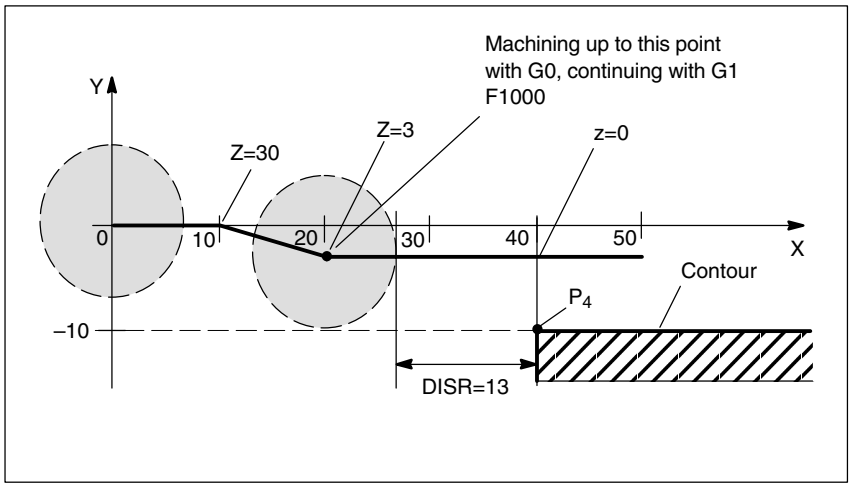


Fig. 2-15 Example

```
$TC_DP1[1,1]=120                ; Milling tool T1/D1
$TC_DP6[1,1]=7                  ; Tool with 7 mm radius

N10 G90 G0 X0 Y0 Z30 D1 T1
N20 X10
N30 G41 G147 DISCL=3 DISR=13 Z=0 F1000
N40 G1 X40 Y–10
N50 G1 X50
...
...
N30/N40 can be replaced by:
1.
N30 G41 G147 DISCL=3 DISR=13 X40 Y–10 Z0 F1000
or
2.
N30 G41 G147 DISCL=3 DISR=13 F1000
N40 G1 X40 Y–10 Z0
```

© Siemens AG, 2003. All rights reserved

– Possible ways of programming the end point P_0 for retraction:

The end position is always taken from the SAR block, no matter how many axes have been programmed. We distinguish between the following situations:

1. No geometry axis is programmed in the SAR block.
   In this case, the contour ends at point P_2 (or at point P_1 where P_1 and P_2 coincide). The position in the axes which describe the machining plane is determined by the retraction contour (end point of the straight line or arc). The axis component perpendicular to this is defined by DISCL. If, in this case, DISCL = 0, the movement takes place completely within the plane.
2. Only the axis perpendicular to the machining plane is programmed in the SAR block. In this case, the contour ends at point P_1. The position of the two other axes is determined in the same way as in 1.

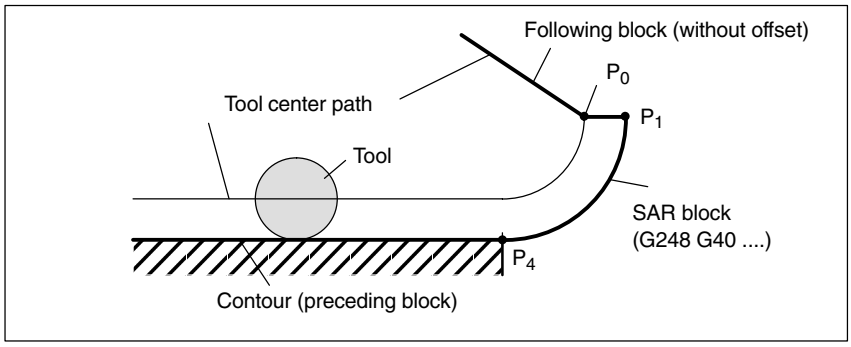


Fig. 2-16 Retraction with SAR with simultaneous deactivation of TRC

If the SAR retraction block is also used to deactivate tool radius compensation in the case of 1. and 2. an additional path from P_1 to P_0 is inserted such that no movement is produced when the tool radius compensation is deactivated at the end of the retraction contour, i.e. this point defines the tool center point and not a position on a contour to be corrected.

3. At least one axis of the machining plane is programmed. The second axis of the machining plane can be determined modally from its last position in the preceding block. The position of the axis perpendicular to the machining plane is generated as described in 1. or 2., depending on whether this axis is programmed or not. The position generated in this way defines the end point P_0.

   No special measures are required for deselection of tool radius compensation, because the programmed point P_0 already immediately defines the position of the tool center point at the end of the complete contour.

   The start and end points of the SAR contour (P_0 and P_4) can coincide on approach and retraction.

- **Velocity of the preceding block (typically G0).**

  All movements from point P_0 to point P_2 are performed at this velocity, i.e. the movement parallel to the machining plane and the part of the infeed movement up to the safety clearance.

© Siemens AG, 2003. All rights reserved

- **Programming the feed with FAD.**
  - FAD programmed with G340: Feedrate from P_2 or P_3 to P_4.
  - FAD programmed with G341: Feedrate of the infeed movement perpendicular to the machining plane from P_2 to P_3.

  If FAD is not programmed, this part of the contour is traversed at the speed which is active modally from the preceding block, in the event that no F command defining the speed is programmed in the SAR block.

  – Programmed response:

  FAD=0 or negative → Alarm output

  FAD=... → Programmed value acts in accordance with the active G code of group 15 (feed type; G93, G94, etc.)

  FAD=PM(...) → Programmed value is interpreted as linear feed (like G94), regardless of the active G code of group 15

  FAD=PR(...) → Programmed value is interpreted as linear feed (like G94), regardless of the active G code of group 15

  Example:

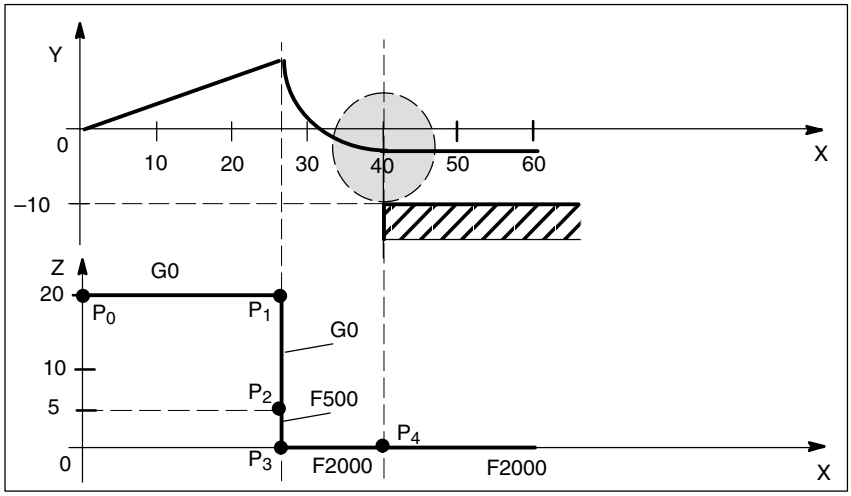


Fig. 2-17 Example

```
$TC_DP1[1,1]=120                    ; Milling tool T1/D1
$TC_DP6[1,1]=7                      ; Tool with 7 mm radius

N10 G90 G0 X0 Y0 Z20 D1 T1
N20 G41 G341 G247 DISCL=AC(5) DISR=13 FAD 500 X40 Y–10 Z=0 F2000
N30 X50
N40  X60
...
```

- **Programmed feed F.**

  This feed value is effective from point P_3 (or from point P_2, if FAD is not programmed). If no F command is programmed in the SAR block, the speed of the preceding block is valid. The speed defined by FAD is not used for following blocks.

© Siemens AG, 2003. All rights reserved

Velocities

In the approach display shown in Figures 2-18 and 2-19 it is assumed that no new velocity is programmed in the block following the SAR block. If this is not the case, the new velocity comes into effect after point P_4.

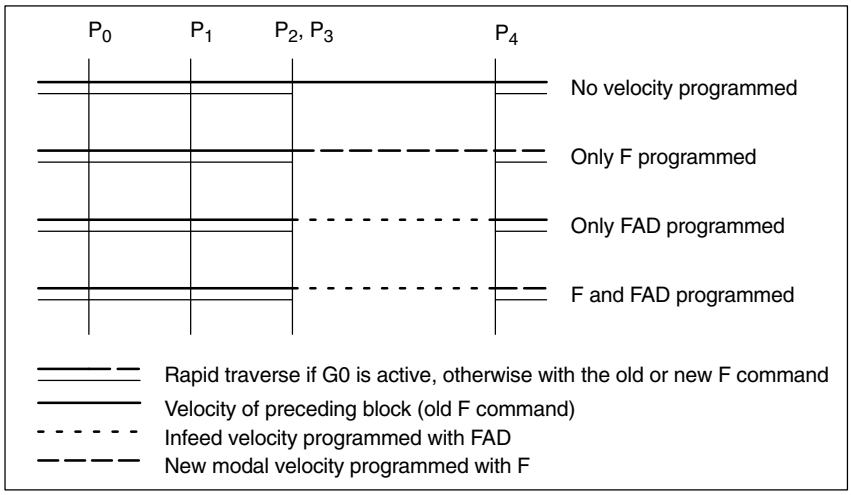


Fig. 2-18 Velocities in the SAR subblocks on approach with G340

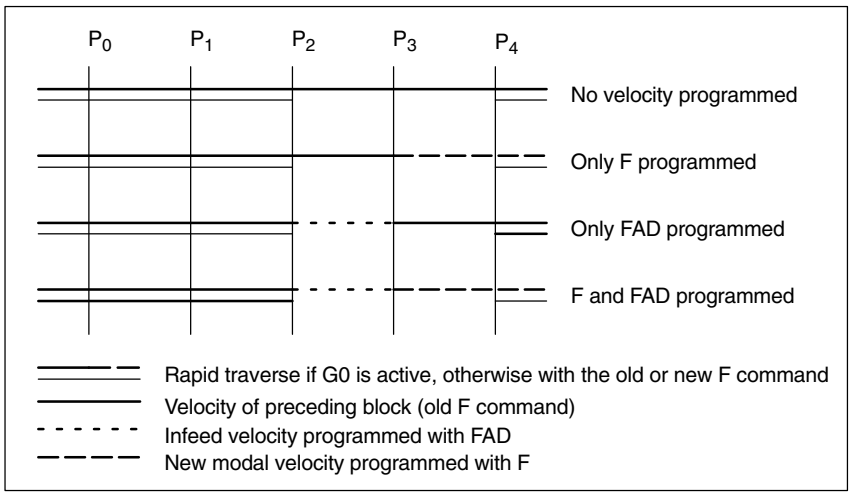


Fig. 2-19 Velocities in the SAR subblocks on approach with G341

On retraction, the roles of the modal feed from the preceding block and the feed value programmed in the SAR block are swapped, i.e. the actual retraction contour (straight line, circle, helix) is traversed at the old feedrate; a new velocity programmed with the F command is valid from point P_2 to point P_0. If level retraction is active and FAD is programmed, the path from P_3 to P_2 is traversed with FAD, otherwise the old velocity applies. The last F command programmed in a preceding block always applies for the path from P_4 to P_2. G0 has no effect in these blocks.

Traversing from P_2 to P_0 takes place with the F command programmed in the SAR block or, if no F command is programmed, with the modal F command from a preceding block. This applies on condition that G0 is not active.

© Siemens AG, 2003. All rights reserved

If rapid traverse is to be used on retraction in the blocks from P_2 to P_0, G0 must be activated before the SAR block or in the SAR block itself. If an additional F command is programmed in the actual SAR blocks, it is then ineffective. However, it remains modally active for following blocks.

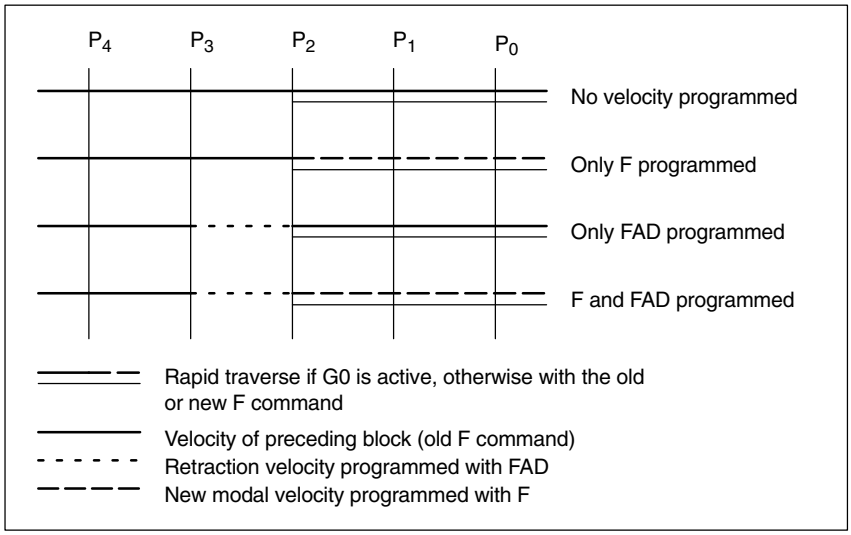


Fig. 2-20 Velocities in the SAR subblocks on retraction

System variables

Points P_3 and P_4 can be read in the WCS as a system variable during approach.

- $P_APR: Read P_3 (start point) in WCS
- $P_AEP: Read P_4 (contour start point) in WCS
- $P_APDV=1 if the contents of $P_APR and $P_AEP are valid, i.e. if they contain the position values belonging to the last SAR approach block programmed.

  With $P_APDV=0, the position of an older SAR approach block are read.

Changing the WCS between the SAR block and the read operation has no effect on the position values.

Supplementary conditions

- Any further NC commands (e.g. auxiliary function outputs, synchronous axis movements, positioning axis movements, etc.) can be programmed in a SAR block. These are executed in the first subblock on approach and in the last subblock on retraction.

© Siemens AG, 2003. All rights reserved

- If the end point P_4 is taken not from the SAR block but from a subsequent traversing block, the actual SAR contour (straight line, quadrant or semicircle) is traversed in this block. The last subblock of the original SAR block does not then contain traversing information for geometry axes. It is always output, however, because further actions (e.g. single axes) may have to be executed in this block.
- At least two blocks must always be taken into consideration:
  – The SAR block itself and
  – the block which defines the approach or retraction direction

  Further blocks can be programmed between these two blocks.

  With MD 20202: WAB_MAXNUM_DUMMY_BLOCKS limits the maximum number of intermediate blocks.
- If tool radius compensation is activated simultaneously in an approach block, the first linear block of the SAR contour is the block in which activation takes place. The complete contour generated by the SAR function is handled by the tool radius compensation as if it has been programmed explicitly (collision monitoring, calculation of intersection, approach response NORM/KONT).
- The direction of the infeed movement and the position of the circle plane or the helix axis are defined exclusively by the active plane (G17 – G19) – rotated with an active frame where appropriate.
- On approach, a preprocessor stop must not be inserted between the SAR block and the following block which defines the direction of the tangent. Whether programmed explicitly or inserted automatically by the controller, a preprocessor stop results, in this case, in an alarm.

Alarms

The following alarms are output when programming is incorrect:

Alarm 10740	“Too many empty blocks on SAR programming”
Alarm 10741	“Direction reversal on SAR infeed movement”
Alarm 10742	“Invalid SAR distance”
Alarm 10743	“SAR programmed more than once”
Alarm 10744	“SAR direction is not defined”
Alarm 10745	“SAR end position not unique”
Alarm 10746	“Preprocessor stop on SAR”
Alarm 10747	“Retraction direction not defined on SAR”

Response with REPOS

If a SAR cycle is interrupted and repositioned, it resumes at the point of interruption on RMI. The contact point on RME is the end point of the last SAR block; with RMB it is the start point of the first SAR block.

If RMI is programmed together with DISPR (reapproach at distance DISPR in front of interruption point), the reapproach point can appear in a subblock of the SAR cycle before the interruption subblock.

© Siemens AG, 2003. All rights reserved

Example 1

The conditions are as follows:

- Soft approach is activated in block N20
- X=40 (end point); Y=0; Z=0
- Approach movement performed with quadrant (G247)
- Approach direction not programmed, G140 is valid, i.e. because TRC active (G42) and offset value is positive (10), the contour is approached from the right
- Approach circle generated internally (SAR contour) has radius 20, so that the radius of the tool center path is equal to the programmed value DISR=10
- Because of G341, the approach movement takes place with a circle in the plane, resulting in a start point at (20, –20, 0)
- Because DISCL=5, point P2 is at position (20, –20, 5) and, because of Z30, point P1 is in N10 at (20, –20, 30)

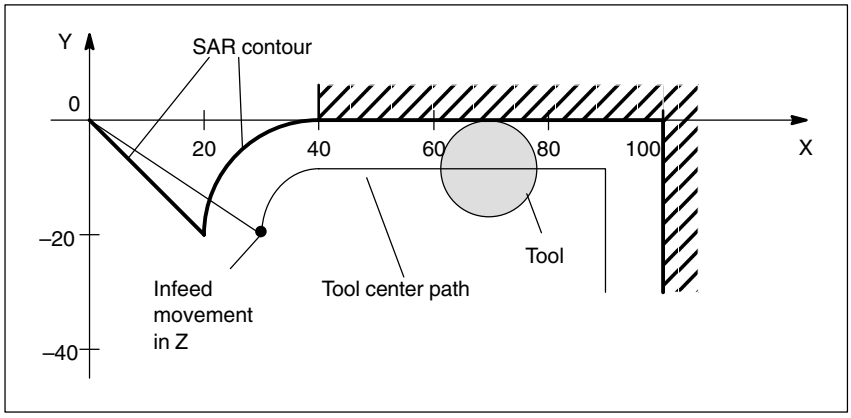


Fig. 2-21 Contour example 1

Parts program:

```
$TC_DP1[1,1]=120                    ; Tool definition T1/D1
$TC_DP6[1,1]=10                     ; Radius
N10 G0 X0 Y0 Z30
N20 G247 G341 G42 NORM D1 T1 Z0 FAD=1000 F=2000 DISCL=5 DISR=10
N30 X40
N40 X100
N50 Y–30
...
```

© Siemens AG, 2003. All rights reserved

Example 2

The conditions are as follows for approach:

- Soft approach is activated in block N20
- Approach movement performed with quadrant (G247)
- Approach direction not programmed, G140 is valid, i.e. because TRC active (G41), the contour is approached from the left
- Contour offset OFFN=5 (N10)
- Current tool radius=10, and so the effective compensation radius for TRC=15; the radius of the SAR contour is thus equal to 25, with the result that the radius of the tool center path is equal to DISR=10
- The end point of the circle results from N30, since only a Z position is programmed in N20
- Infeed movement
  - From Z20 to Z7 (DISCL=AC(7)) with rapid traverse
  - Then on to Z0 with FAD=200
  - Approach circle in XY plane and following blocks with F1500 (for this velocity to be active in the following blocks, the active G code G0 must be overwritten in N30 with G1. Otherwise the contour would continue to be machined with G0.)

The following conditions apply to retraction:

- Soft approach is activated in block N60
- Retraction movement performed with quadrant (G248) and helix (G340)
- FAD not programmed, since no effect with G340
- Z=2 at start point; Z=8 at end point, because DISCL=6
- With DISR=5, the radius of the SAR contour=20, the radius of the tool center path=5
- After the circle block, the retraction movement leads from Z8 to Z20 and the movement is parallel to the XY plane up to the end point at X70 Y0.

© Siemens AG, 2003. All rights reserved

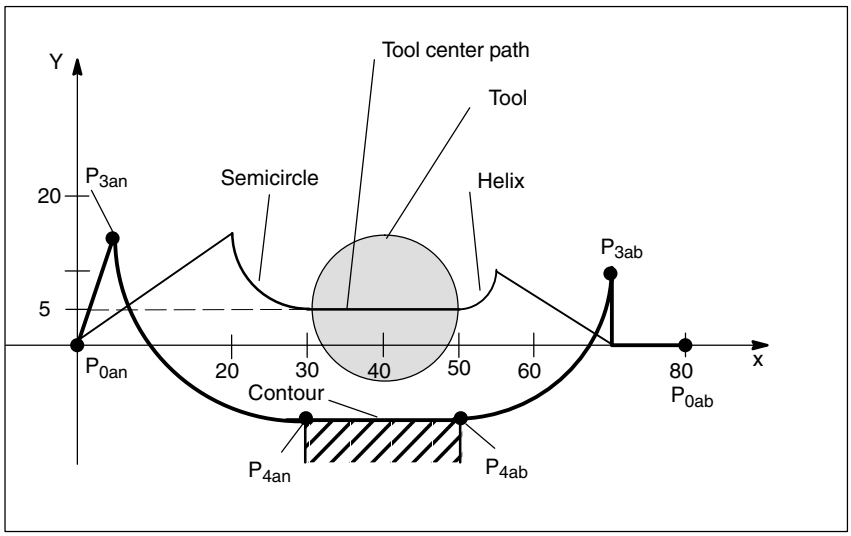


Fig. 2-22 Contour example 2

Parts program:

```
$TC_DP1[1,1]=120                    ; Tool definition T1/D1
$TC_DP6[1,1]=10                     ; Radius

N10 G0 X0 Y0 Z20 G64 D1 T1 OFFN = 5                                  (P0an)
N20 G41 G247 G341 Z0 DISCL = AC(7) DISR = 10 F1500 FAD=200           (P3an)
N30 G1 X30 Y–10                                                      (P4an)
N40 X40 Z2
N50 X50                                                              (P4ab)
N60 G248 G340 X70 Y0 Z20 DISCL = 6 DISR = 5 G40 F10000               (P3ab)
N70 X80 Y0                                                           (P0ab)
N80 M 30
```

Note

The contour generated in this way is modified by the tool radius compensation which is activated in the SAR approach block and deactivated in the SAR retraction block. The tool radius compensation allows for an effective radius of 15, which is the sum of the tool radius (10) and the contour offset (5). The resulting radius of the tool center path in the approach block is therefore 10, and 5 in the retraction block.

© Siemens AG, 2003. All rights reserved

2.4.4 Deselecting the TRC (G40)

The deselection of tool radius compensation is performed with the G40 instruction.

Special points to be noted

- The deselection of tool radius compensation can only be performed in a program block with G0 (rapid traverse) or G1 (linear interpolation).
- If D0 is programmed when the tool radius compensation is active, the compensation is not deselected and error message 10750 is output.
- If a geometry axis is programmed in the block with the tool radius compensation deselection, then the compensation is deselected even if it is not on the current plane.

2.4.5 Compensation at outside corners

The functions G450/G451 can be used to control the response with discontinuous block transitions at outside corners:

- G450 ... discontinuous block transitions with transition circle
- G451 ... discontinuous block transitions with intersection of equidistant paths

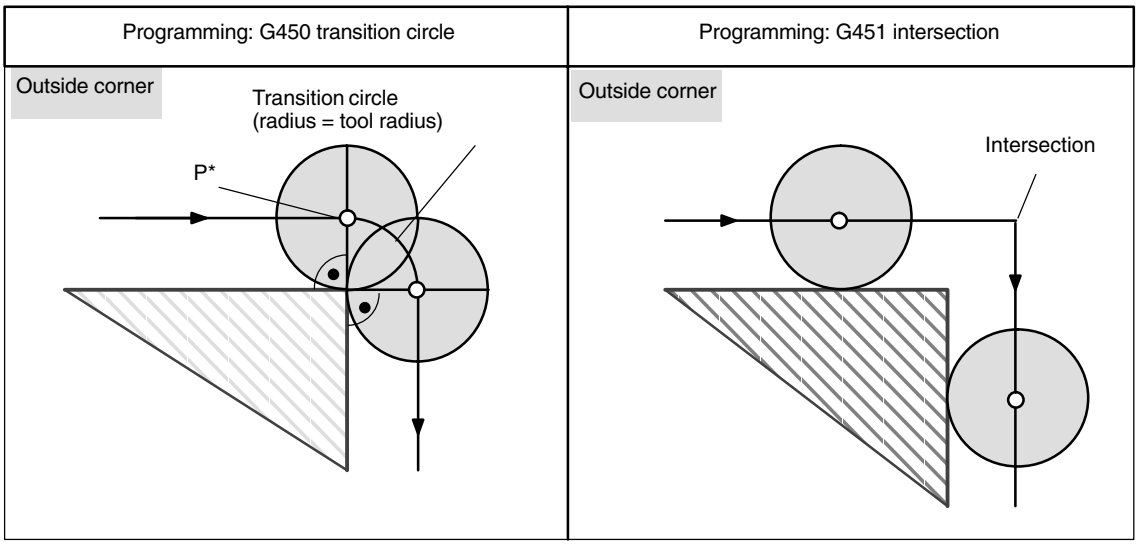


Fig. 2-23 Example of a 90 degree outside corner with G450 and G451

© Siemens AG, 2003. All rights reserved

G450 transition circle

With the G function G450 active, the center point of the tool with outside corners performs a circular path along the tool radius. The circular path starts with the normal position (perpendicular to the path tangent) at the end point of the previous path section (program block) and finishes in the normal position at the start point of the new path section (program block).
Where the outside corners are very flat, the different responses with G450 (transition circle) and G451 (intersection) become increasingly similar ——> see very flat outside corners.
If the outside contour corners are to be pointed, the tool must be lifted off the contour ——> see DISC.

DISC

The G450 transition circle does not produce sharp outside contour corners because the path of the tool center point through the transition circle is controlled so that the cutting edge stops at the outside corner (programmed position). When sharp outside corners are to be machined with G450, the DISC instruction can be used to program an overshoot. This transforms the transition circle into a conic section and the cutting edge lifts off from the outside corner.
The value range of the DISC instruction extends from 0 to 100 in steps of 1.
DISC = 0 ... Overshoot disabled, transition circle active
DISC = 100 ... Overshoot large enough to theoretically produce a response similar to intersection (G451).
The MD 20220: CUTCOM_MAX_DISC (maximum value for DISC) can be used to define the maximum value that can be programmed with DISC. Values over 50 are generally not advisable with DISC.

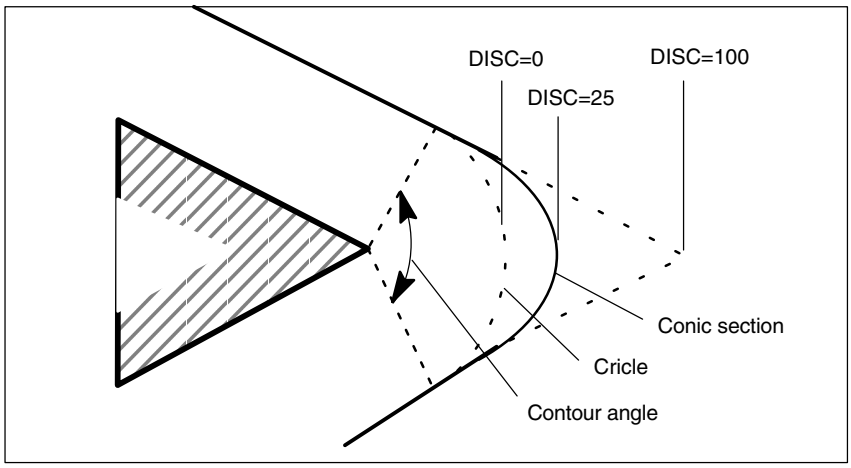


Fig. 2-24 Example: Overshoot with DISC = 25

© Siemens AG, 2003. All rights reserved

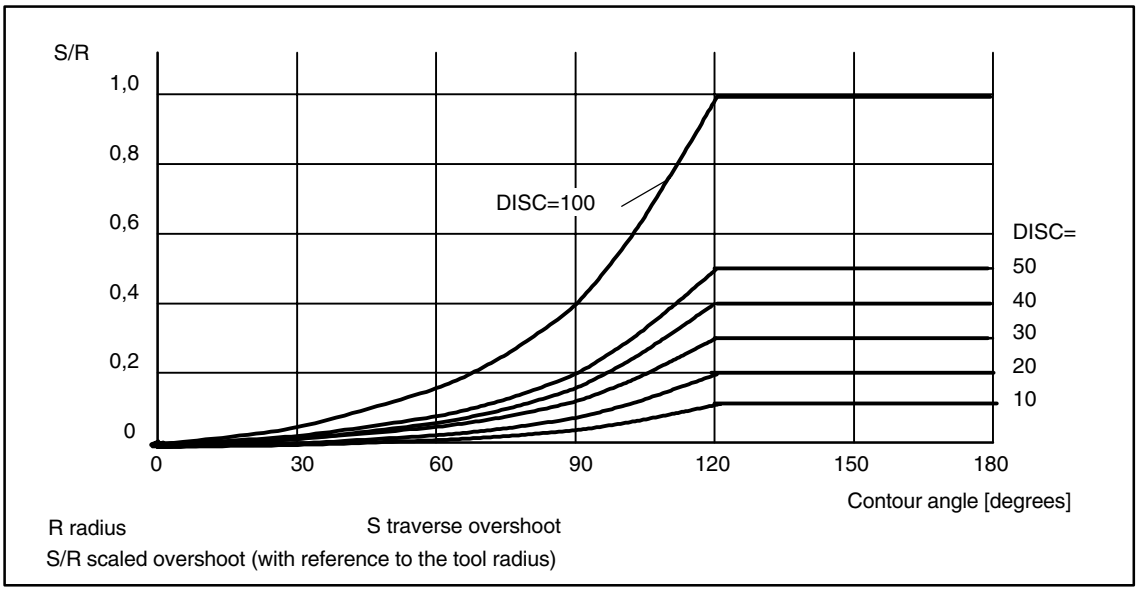


Fig. 2-25 Overshoot with DISC depending on contour angle

G451 intersection

With the G function G451 active, the position (the intersection point) results from the path lines (with straight line, circle or helix only) is approached, which are at the distance of the tool radius to the programmed contour (centerpoint path of the tool); splines and polynomials are basically not extended.
Where outside corners are very pointed, G451 can result in excessive idle paths ——> see Very pointed outside corners.

Very pointed outside corners

Where outside corners are very pointed, G451 can result in excessive idle paths (see Fig. 2-26). The system therefore switches automatically from G451 (intersection) to G450 (transition circle, with DISC where appropriate) when the outside corners are very pointed. The contour angle which can be traversed following this automatic switchover (intersection ——> transition circle) can be defined in MD 20210: CUTCOM_CORNER_LIMIT (maximum angle for compensation blocks with tool radius compensation).

© Siemens AG, 2003. All rights reserved

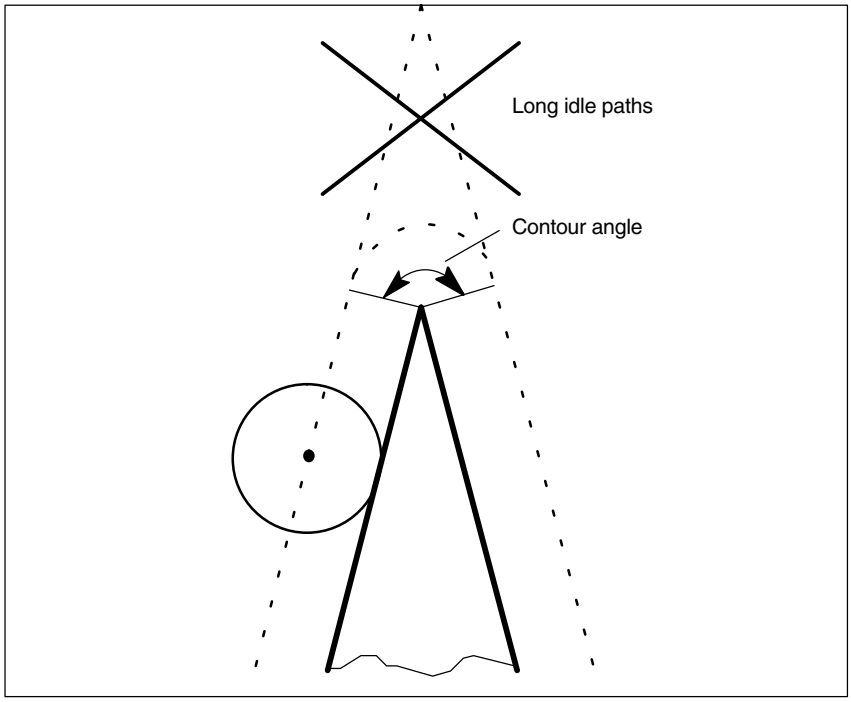


Fig. 2-26 Example of automatic switchover to transition circle

Very flat outside corners

Where the outside corners are very flat, the response with G450 (transition circle) and G451 (intersection) becomes increasingly similar. In this case, it is no longer advisable to insert a transition circle. One reason why it is not permitted to insert a transition circle at these outside corners with 5axis machining, is that this would impose restrictions on the speed in contouring mode (G64). Where the outside corners are very flat, the system therefore switches automatically from G450 (transition circle, with DISC where appropriate) to G451 (intersection). The contour angle which can be traversed following this automatic switchover (transition circle ——> intersection) can be defined in MD 20230: CUTCOM_CURVE_INSERT_LIMIT (maximum angle for intersection calculation with tool radius compensation).

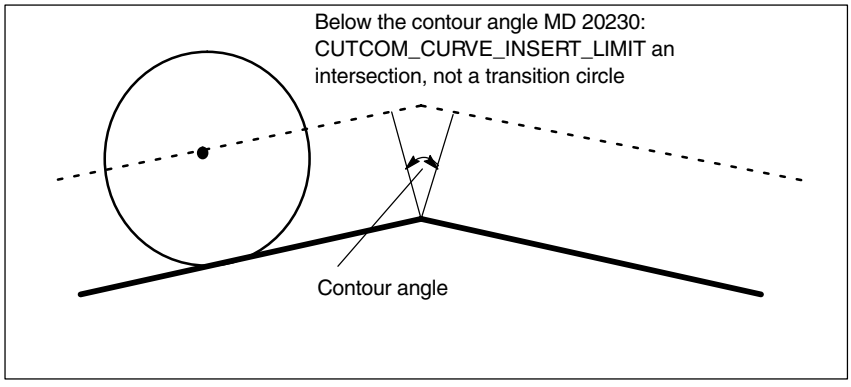


Fig. 2-27 Example of automatic switchover to intersection

© Siemens AG, 2003. All rights reserved

2.4.6 Compensation at inner corners

Intersection

If two consecutive blocks form an inside corner, an attempt is made to find an intersection of the two equidistant paths. If an intersection is found, the programmed contour is shortened to the intersection (first block shortened at end, second block shortened at beginning).

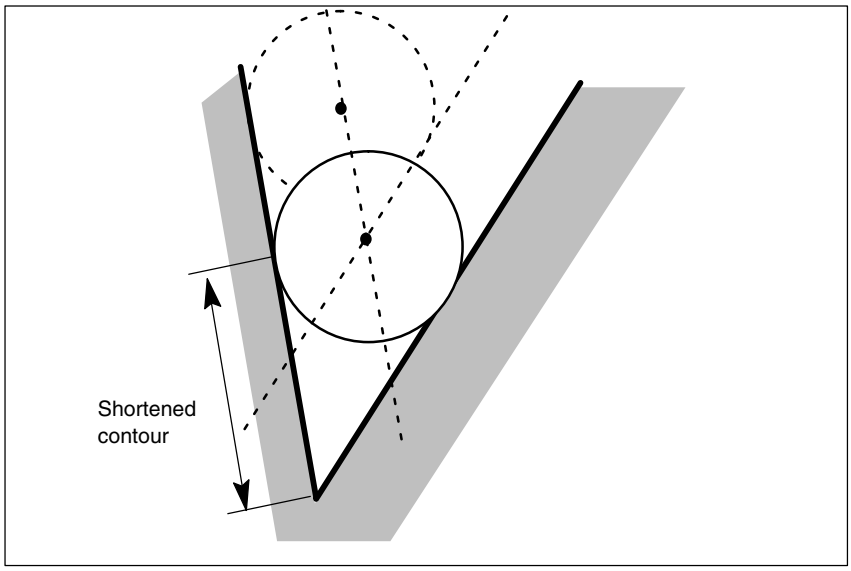


Fig. 2-28 Example of a shortened contour

No intersection

In certain cases, no intersection is found between two consecutive blocks for inside corners (see Fig. 2-29).

Predictive contour calculation

If no intersection is found between two consecutive blocks, the control automatically checks the next block and attempts to find an intersection with the equidistant paths of this block (see Fig. 2-29: Intersection S). This automatic check of the next block (predictive contour calculation) is always performed until a number of blocks defined by machine data has been reached. This maximum number of blocks used for the predictive check can be entered in MD 20240: CUTCOM_MAXNUM_CHECK_BLOCKS. If no intersection is found within the number of blocks defined for the check, program execution is interrupted and alarm 10751 is output.

© Siemens AG, 2003. All rights reserved

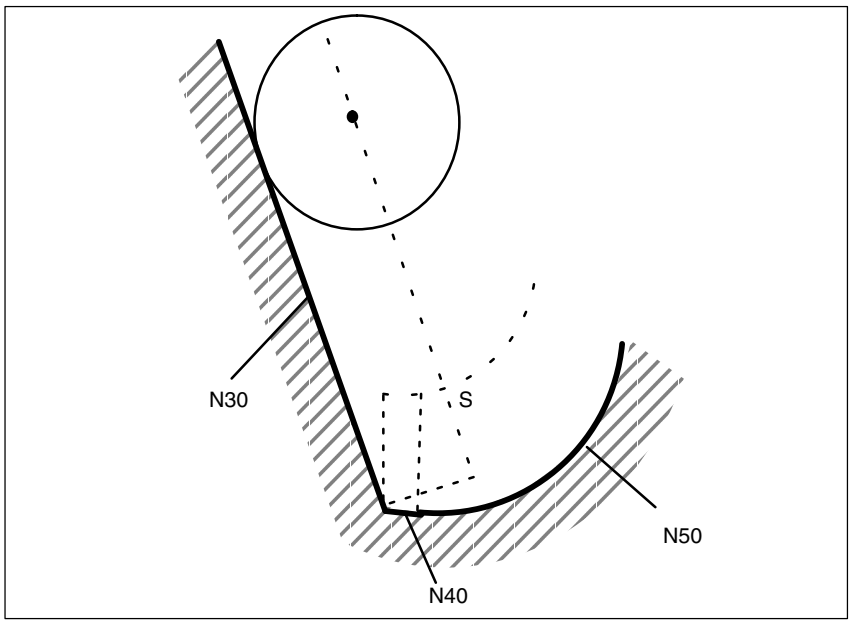


Fig. 2-29 If there is no intersection between N30 and block N40, the intersection between block N30 and block N50 is calculated.

Multiple intersections

——> see also collision monitoring (Subsection 2.4.7)
In certain cases with inside corners, multiple intersections of the equidistant paths can be found in several consecutive blocks. In these cases, the last intersection is always used as the valid intersection (see Fig. 2-30). The maximum number of blocks used for the predictive check can be entered in MD 20240: CUTCOM_MAXNUM_CHECK_BLOCKS (blocks for predictive contour calculation for TRC).

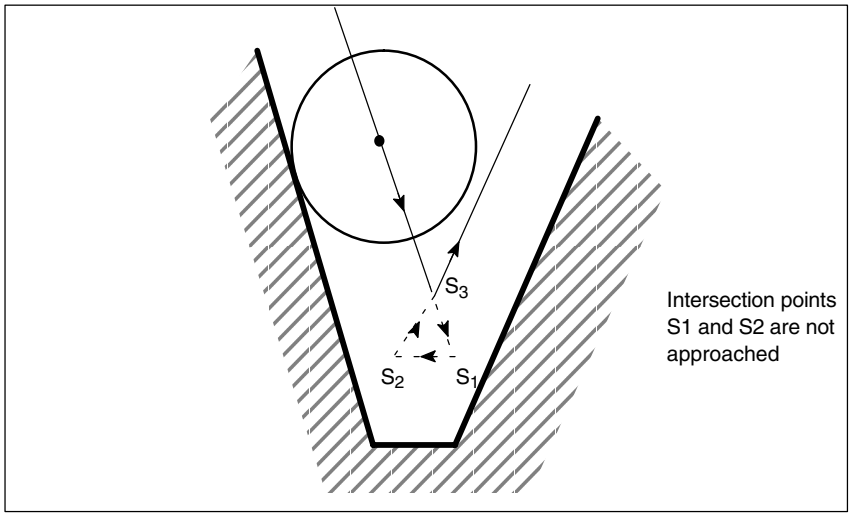


Fig. 2-30 Example: Inside corner with TRC without contour violation (predicting 3 blocks)

Special points to be noted

Where multiple intersections with the next block are found, the intersection nearest the start of the next block applies.

© Siemens AG, 2003. All rights reserved

2.4.7 Collision detection and bottleneck detection

The collision monitoring (bottleneck detection) examines whether the equidistant paths of non-consecutive blocks intersect. If an intersection is found, the response is the same as for inside corners with multiple intersections: The intersection found last is valid (see also Fig. 2-30).
This maximum number of blocks used for the predictive check can be entered in MD 20240: CUTCOM_MAXNUM_CHECK_BLOCKS (blocks for predictive contour calculation for TRC).

Programming

The collision monitoring can be activated or deactivated in the program:

- CDON ... Collision monitoring on
- CDOF ... Collision monitoring off

With CDOF, the search for an intersection initially examines two consecutive blocks. Other blocks are not included in the search. If an intersection is found between adjacent blocks, no further blocks are examined. With outside corners, an intersection is always found between two consecutive blocks.

Predictive examination of more than two adjacent blocks is thus possible with CDON and CDOF.

Omission of block

If an intersection is detected between two blocks which are not consecutive, none of the motions programmed between these blocks on the compensation plane are executed. All other motions and executable instructions (M commands, traversal of positioning axes, etc.) contained in the omitted blocks are executed at the intersection position in the sequence in which they are programmed in the NC program.

Warning 10763

If a block has been omitted as a result of the collision or bottleneck detection functions, the warning 107663 is output; the program is not interrupted. The output of this warning is suppressed if bit 1 in machine data

MD 11410: SUPPRES_ALARM_MASK is set.

Special points to be noted

When the intersections of nonconsecutive blocks are checked, it is not the programmed original contours which are examined but the associated calculated equidistant paths. This may result in a “bottleneck” being falsely detected at outside corners. The reason for this is that the calculated tool path does not run equidistant to the programmed original contour when DISC>0.

© Siemens AG, 2003. All rights reserved

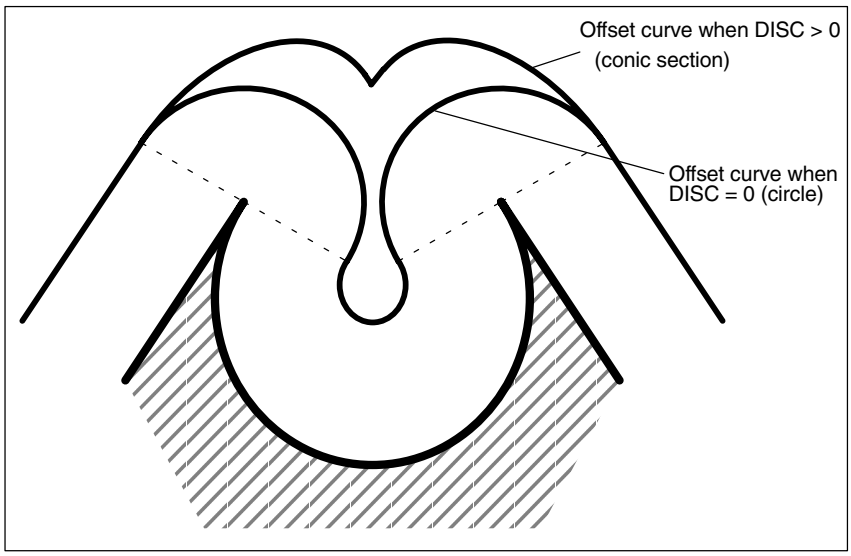


Fig. 2-31 Bottleneck detection and outside corners

2.4.8 Blocks with variable compensation value

Supplementary conditions

A variable compensation value may be programmed for all interpolation types (even circle and spline interpolation).
Sign changes (and thus also a change in compensation side!) are also permitted.

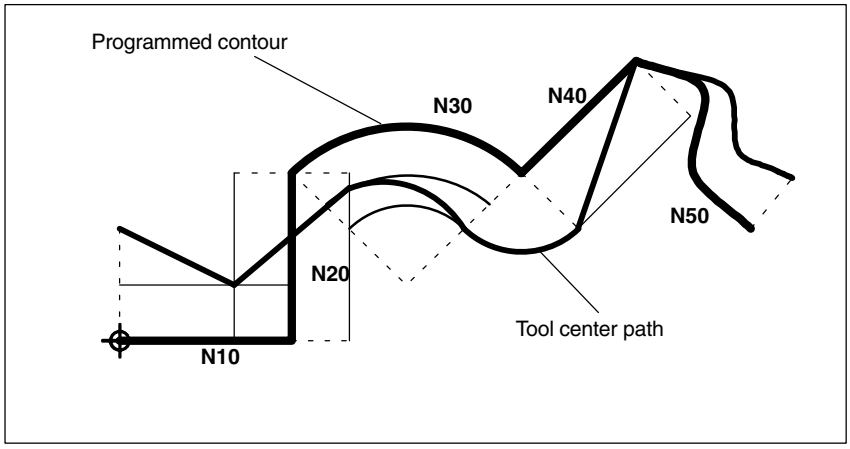


Fig. 2-32 Tool radius compensation with variable compensation value

© Siemens AG, 2003. All rights reserved

Intersection calculation

When the intersections in blocks with variable compensation value are calculated, the intersection of the offset curves (tool paths) is always calculated based on the assumption that the compensation value is constant.
If the block with the variable compensation value is the first of the two blocks to be examined in the direction of travel, then the compensation value at the block end is used for the calculation; the compensation value at the block start is used otherwise.

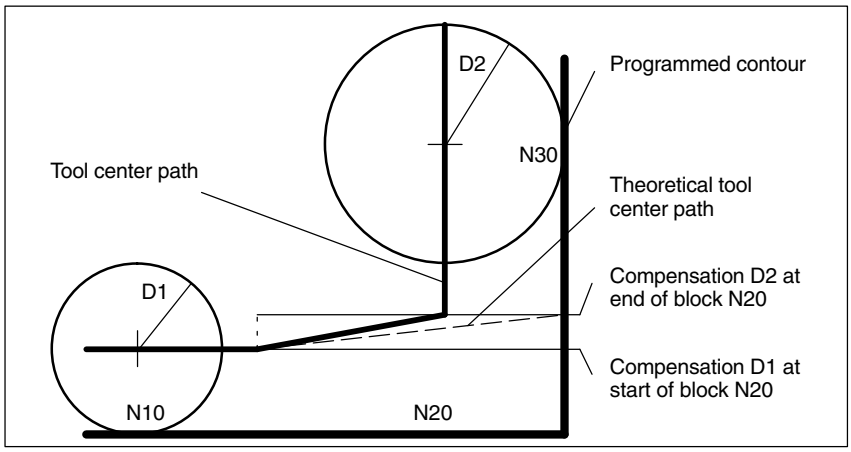


Fig. 2-33 Intersection calculation with variable compensation value

Restriction

If the compensation radius is programmed as being larger than the programmed circle radius for machining on the inside of the circle, the machining operation is rejected with the alarm 10758 “Curvature radius with variable compensation value too small”.

© Siemens AG, 2003. All rights reserved

2.4.9 Keep tool radius compensation constant (SW 4 and higher)

Meaning

The “Keep tool radius compensation constant” function is used to suppress the tool radius compensation for a number of blocks, whereby a difference between the programmed and the actual tool center path traveled set up by the tool radius compensation in the previous blocks is retained as the offset.
It can be an advantage to use this method when several traversing blocks are required during line milling in the reversal points, but the contours produced by the tool radius compensation (follow strategies) are not wanted.

Activation

The “Keep tool radius compensation constant” function is activated with the G code

CUTCONON (CUTter compensation CONstant ON)

and deactivated with the G code

CUTCONOF (CUTter compensation CONstant OFF).

CUTCONON and CUTCONOF form a modal G code group. The basic setting is CUTCONOF.
It can be used independently of the type of tool radius compensation ($2^1/_2$D, 3D face milling, 3D circumferential milling).

Normal case

Normally, the tool radius compensation is already active prior to activation of the offset suppression, and is still active when offset suppression is deactivated again.

In the last traversing block before CUTCONON, the offset point in the block end point is approached.
All following blocks in which offset suppression is active are traversed without offset.
However, they are offset by the vector from the end point of the last offset block to its offset point.
These blocks can have any type of interpolation (linear, circular, polynomial).

The deactivation block of the offset suppression, i.e. the block that contains CUTCONOF, is offset normally; it starts in the offset point of the start point.
One linear block is inserted between the end point of the previous block, i.e. the last programmed traversing block with active CUTCONON, and this point.

Circular blocks for which the circle plane is perpendicular to the offset plane (vertical circles), are treated as though they had CUTCONON programmed.
This implicit activation of the offset suppression is automatically canceled in the first traversing block that contains a traversing motion in the offset plane and is not such a circle.
Vertical circle in this sense can only occur during circumferential milling.

© Siemens AG, 2003. All rights reserved

Example:

```
N10                          ; Definition of tool d1
N20 $TC_DP1[1,1]= 110        ; Type
N30 $TC_DP6[1,1]= 10.        ; Radius
N40
N50 X0 Y0 Z0 G1 G17 T1 D1 F10000
N60
N70 X20 G42 NORM
N80 X30
N90 Y20
N100 X10 CUTCONON; Activate offset suppression
N110 Y30 KONT  ; During deactivation of offset suppression
               ; insert follow cycle if necessary
N120 X-10 CUTCONOF
N130 Y20 NORM  ; No follow cycle on deactivation of the TRC
N140 X0 Y0 G40
N150 M30
```

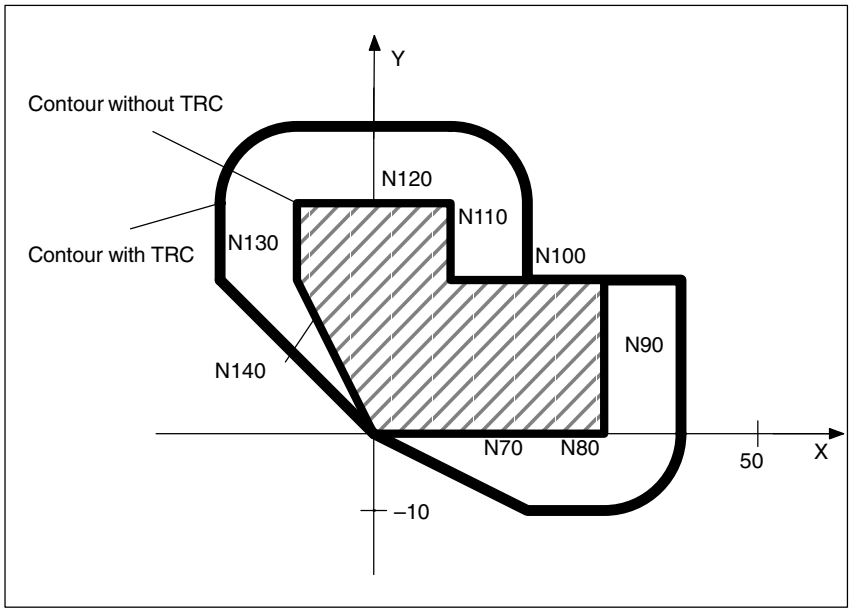


Fig. 2-34 Sample program for offset suppression

Special cases

1. If tool radius compensation active (G40), CUTCONON has no effect. No alarm is produced.
   The G code remains active, however. This is significant when tool radius compensation is to be activated in a later block with G41 or G42.
2. It is permissible to change the G code in the 7th G code group (tool radius compensation; G40 / G41 / G42) with active CUTCONON. A change to G40 is active immediately.
   The offset used for traversing the previous blocks is traveled.
3. If CUTCONON or CUTCONOF is programmed without traversing in the active compensation plane, the effect is delayed until the next block that has such a traversing motion.

© Siemens AG, 2003. All rights reserved

4. If CUTCONON is programmed with active tool radius compensation and not canceled before the end of the program, the traversing blocks are traversed with the last valid offset. The same allies for reprogramming of G41 or G42 in the last traversing block of a program.
5. If the tool radius compensation is activated with G41 or G42 and CUTCONON is also already active, activation of the offset is delayed until the next traversing block with CUTCONOF.
6. When reapproaching the contour with CUTCONOF, the 17th G code group (approach and retraction response with tool compensation; NORM / KONT) is evaluated, i.e. a follow cycle is inserted if necessary for KONT. A follow cycle is inserted under the same conditions as for activation of the tool radius compensation with G41 or G42.
7. The number of blocks with suppressed tool radius compensation is restricted (MD: CUTCOM_MAXNUM_SUPPR_BLOCKS). If it is exceeded, machining is aborted and an error message issued. The restriction is necessary because the internal block processing in the last block before CUTCONON must be resumed again when repositioning.
8. The response after reprogramming G41 or G42 when tool radius compensation is already active is similar to offset suppression. The following deviations apply:
   - Only linear blocks are permissible
   - A single traversing block that contains G41 or G42 is modified so that it ends at the offset point of the start point in the following block. Thus it is not necessary to insert a dummy block. The same applies for the last block in a sequence of traversing blocks where each contains G41 or G42.
   - The contour is always reapproached with NORM, independent of the G code of the 17th group (approach and retraction response with tool compensation; NORM/KONT).
9. If G41/G42 is programmed several times in consecutive traversing blocks, all blocks are machined as for CUTCONON, except for the last one.
10. The type of contour suppression is evaluated only in the first traversing block of a sequence of consecutive traversing blocks.
    If both CUTCONON and G41 or G42 are programmed in the first block, the response to deactivating the contour suppression is determined by CUTCONON.
    Changing from G41 to G42 or vice versa makes sense in this case as a means of changing the offset side (left or right of the contour) when restarting.
    A change of offset side (G41/G42) can also be programmed later even if contour suppression is active.
11. Collision monitoring and bottleneck detection is deactivated for all blocks with active contour suppression.

© Siemens AG, 2003. All rights reserved

2.4.10 Changed alarm response (SW 4 and higher)

Response up to SW 3

If an alarm occurs (which can be deleted with RESET), block processing in the main run and the blocks that have already been processed are not affected.

Response from SW 4 and higher

If a tool radius compensation alarm occurs,

- processing in the main run is stopped at the next end of block that can be reached, i.e. generally at the end of the current interpolated block,
- with active Look Ahead, when the axis have come to a standstill.

Alarms for preprocessing stop and active tool radius compensation (SW 4 and higher)

Response up to SW 3

The tool radius compensation generally requires at least one of the following traversing blocks (even more for bottlenecks) to determine the end point of a block. Since the preprocessing stop of such a block is not available, traversing is made to the offset point in the last block. Correspondingly, the offset point in the start point is approached in the first block after a preprocessing stop.
The contour obtained may deviate considerably from the one that would result without preprocessing stop. In particular, contour violations are possible.

Response from SW 4 and higher

A setting data ($SC_STOP_CUTCOM_STOPRE) is therefore inserted so that the response of the tool radius compensation remains unchanged compared to the previous status, and/or an alarm is output for preprocessing stop during active tool radius compensation and the program is halted, depending on the value.
The user can acknowledge this alarm and continue the NC program with NC start or abort it with reset.

© Siemens AG, 2003. All rights reserved

2.4.11 Intersection procedure for polynomials (SW 4 and higher)

Response up to SW 3

For polynomials (splines), up to SW 3, the intersection procedure at outside corners was not implemented; i.e. even if the transition were almost at a tangent, a block (possible very short) was always inserted. These short blocks always produce unwanted drops in speed during G64 operation.

Response from SW 4 and higher

If two curves with active tool radius compensation form an outside corner, Group (corner response with tool offset; G450/G451) regardless of the type of curves involved (straight lines, circles, polynomials)

- either a conical cut is inserted to follow the corner
- or the curves involved are extrapolated to form an intersection.

If no intersection is found with G451 activated, or if the angle formed by the two curves is too steep, switchover to insert mode is automatic.

The MD: CUTCOM_INTERS_POLY_ENABLE enables the intersection procedure for polynomials. If this MD is set to inactive, the response is identical to that in SW 3 and lower.

© Siemens AG, 2003. All rights reserved

2.4.12 G461/G462: Approach/retract strategy extension (SW 5 and higher)

In certain special geometrical situations, extended approach and retraction strategies are required, in comparison with the previous implementation, for activation/deactivation of tool radius compensation (see the following figure).

Note

The following Section refers exclusively to the situation for deactivation of tool radius compensation. The response for approach is virtually identical.

Example

```
G42 D1 T1 ; Tool radius 20mm
...
G1 X110 Y0
N10 X0
N20 Y10
N30 G40 X50 Y50
```

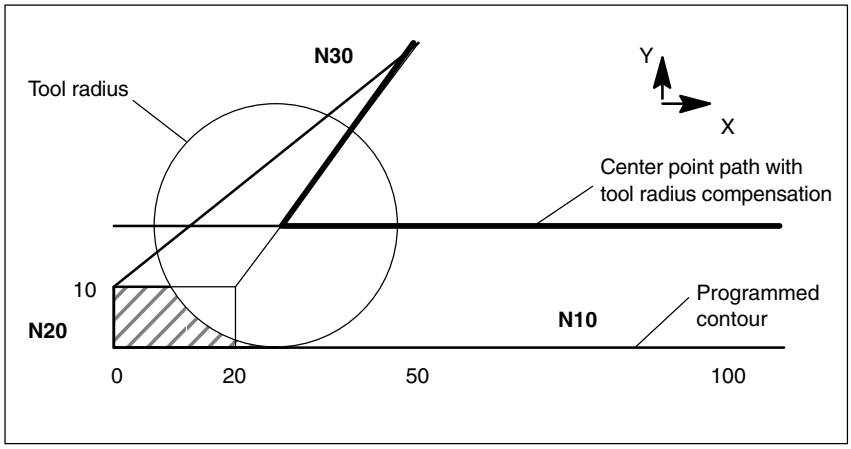


Fig. 2-35 Retraction behavior with G460 (identical to behavior up to SW 4.x

The last block with active tool radius compensation (N20) is so short that an intersection between the offset curve and the preceding block (or an earlier block) would no longer exist with the current tool radius. An attempt is therefore made to find an intersection between the offset curves of the following block and the preceding block, i.e. in the example between N10 and N30. The curve used for the retraction block is not a genuine offset curve, but is a straight line from the offset point at the end point of block N20 to the programmed end point of N30. If an intersection is found, it is approached. In this case, the shaded area in the figure is not machined, although this would be possible with the tool used.

G460 With G460, the approach/retraction strategy is the same as before.

© Siemens AG, 2003. All rights reserved

G461

If no intersection is possible between the last TRC block and a previous block, the offset curve of this block is extended with a circle whose center point coincides with the end point of the noncorrected block, and whose radius is equal to the tool radius.

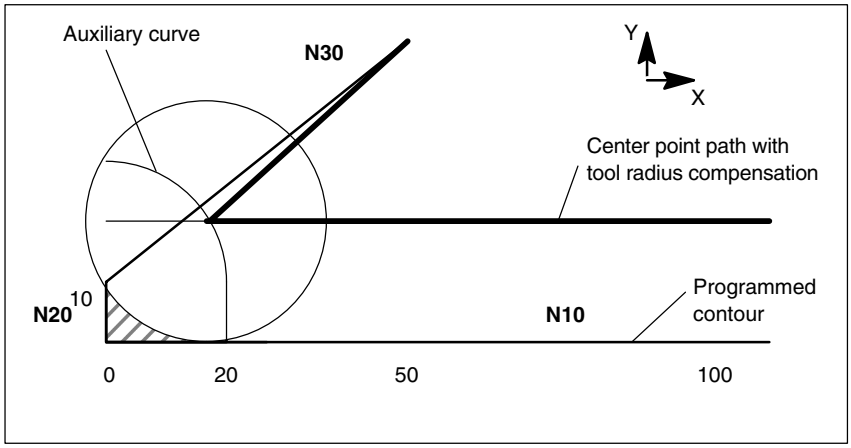


Fig. 2-36 Retraction behavior for G461

The control attempts to intersect this circle with one of the preceding blocks. If CDOF is active, the search is terminated when an intersection is found, i.e. the system does not check whether further intersections with still earlier blocks exist.

If CDON is active, the search continues for further intersections after the first intersection is found.

An intersection point which is found in this way is the new end point of a preceding block and the start point of the deactivation block. The inserted circle is used exclusively to calculate the intersection and does not produce a traversing movement.

Note

If no intersection is found, alarm 10751 (collision danger) is output.

G462

If no intersection is possible between the last TRC block and a preceding block, a straight line is inserted, on retraction with G462 (initial setting), at the end point of the last block with tool radius compensation (the block is extended by its end tangent), see figure below.

© Siemens AG, 2003. All rights reserved

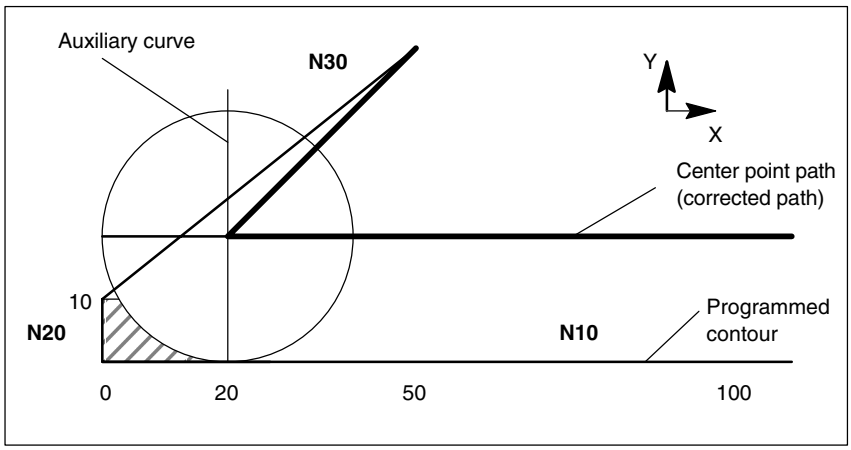


Fig. 2-37 Retraction behavior for G462

The search for the intersection is then identical to the procedure for G461.

With G462, the corner generated by N10 and N20 in the example program is not cleared to the full extent actually possible with the tool used. However, this behavior may be necessary if the part contour (as distinct from the programmed contour), to the left of N20 in the example, is not permitted to be violated even with y values greater than 10 mm.

If KONT is active (travel round contour at start or end point), the behavior differs according to whether the end point is in front of or behind the contour.

End point in front of the contour

If the end point is in front of the contour, the retraction behavior is the same as with NORM. This property does not change even if the last contour block for G451 is extended with a straight line or a circle. Additional bypass strategies to avoid a contour violation in the vicinity of the contour end point are therefore not required.

End point behind the contour

If the end point is behind the contour, a circle or straight line is always inserted depending on G450/G451. In this case, G460 – G462 has no effect.

If, in this situation, the last traversing block has no intersection with a preceding block, an intersection with the inserted contour element or with the linear section from the end point of the bypass circle to the programmed end point can result.

If the inserted contour element is a circle (G450), and it intersects with the preceding block, this is the same as the intersection which would be produced with NORM and G461. In general, however, a remaining section of the circle still has to be traversed. An intersection calculation is no longer required for the linear section of the retraction block.

In the second case (if no intersection is found between the inserted contour element and the preceding blocks), the intersection between the retraction straight line and a preceding block is traversed.

Therefore, when G461 or G462 is active, a behavior different to G460 can only arise if NORM is active or if the behavior with KONT is identical to NORM due to the geometrical conditions.

© Siemens AG, 2003. All rights reserved

Note

The approach behavior is symmetrical to the retraction behavior.

The approach/retraction behavior is determined by the state of the G command in the approach/retraction block. The approach behavior can therefore be set independent of the retraction behavior.

Example:
Program for using G461 for approach:

```
N10 $TC_DP1[1,1]=120 ; Tool type mill
N20 $TC_DP6[1,1]=10 ; Radius

N30 X0 Y0 F10000 T1 D1
N40 Y20
N50 G42 X50 Y5 G461
N60 Y0 F600
N70 X30
N80 X20 Y–5
N90 X0 Y0 G40
N100 M30
```

© Siemens AG, 2003. All rights reserved

2.5 Orientational toolholder (SW 4 and higher)

2.5.1 General

Introduction

The orientation of the tool can be changed (e.g. **when loading a different tool**) for one class of machine tools. When the machine is operating, the orientation that has been set is **permanent**, however, and cannot be changed during traversing. For this reason, kinematic orientation transformation (3, 4 or 5-axis transformations, TRAORI) is neither necessary nor does it make sense for such machines. However, it is necessary to take account of the changes in the tool length components caused by changing the orientation, without having to trouble the user with mathematics involved. The control performs these calculations.

Required data

If the control tool compensations for orientational toolholders are to be taken into account, the following must be present:

- **Tool** data (geometry, wear ...)
- Tool**holder** data (data for the geometry of the orientational toolholder)

Tool holder selection

A toolholder defined in the control must be specified for the “Orientational toolholder” function. This is done using the NC program command:

TCARR = m

where **m** is the number of the toolholder. The toolholder has an associated toolholder data block, which describes its geometry.

Activating the toolholder and its block has an immediate effect, i.e. from the next traversing block.

Assignment tool/toolholder

The tool that was active beforehand is assigned to the new toolholder.

From the point of view of the control, toolholder number m and tool numbers T can be combined freely. In the real application, however, certain combinations can be ruled out for machining or mechanical reasons. The control does not check whether the combinations make sense.

Description of the toolholder kinematics (SW 5.3 and higher)

The kinematic of the orientational toolholder is described:
In SW 5.2 and lower by a data block of 17 REAL values.

In SW 5.3 and higher
by a data block of 20 REAL values, 2 AXIS identifiers and 1 CHAR kinematic type which is stored in the offset memory.

In SW 6.1 and higher with a total of 33 parameter sets.

The data of the data block can be edited by the user.

© Siemens AG, 2003. All rights reserved

Orientational toolholder

Example: Cardan toolholder with two axes for the tool orientation

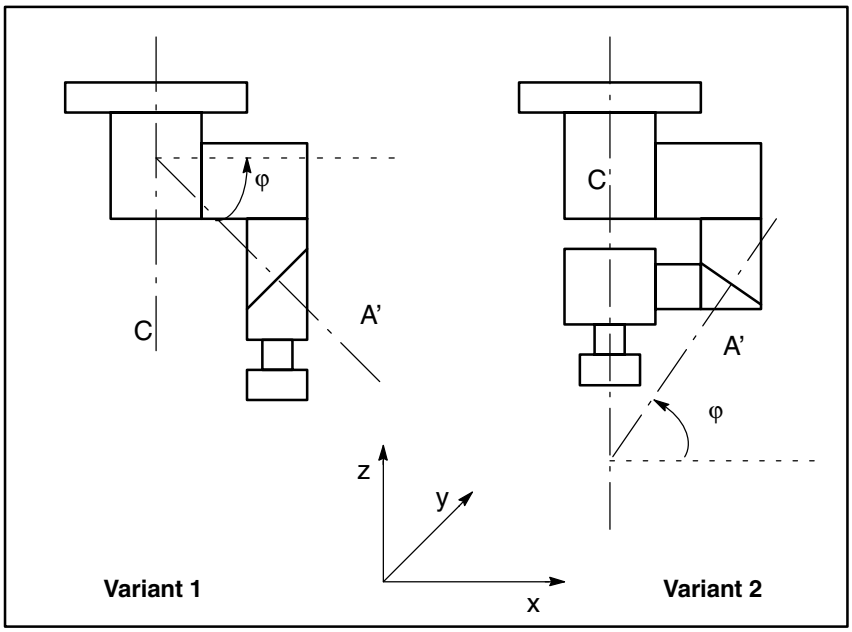


Fig. 2-38 Cardan toolholder with two axes

Processing the toolholder data blocks

Two options are available:

1. Explicit entry in the toolholder data block from the parts program
2. Automatic transfer of some of the values (angles) from a frame.
   This requires that
   TCOFR, (Tool Carrier Orientation FRame) is also specified for toolholder selection.

   The tool orientation used to calculate the tool length is determined again from the frame active at this time when a toolholder is changed.

Orientation in Z direction

The G function **TOFRAME** defines a frame such that the Z direction in this frame is the same as the current tool orientation.

If	Then
No toolholder is active	The Z direction is in the new frame: With G17 same as the old Z direction With G18 same as the old Y direction With G19 same as the old X direction
Toolholder without orientation change is active	

TCOABS for active frame

The absolute toolholder orientation is set by:
TCOABS, (Tool Carrier Orientation ABSolute).
The orientation taken into account for the tool length compensation is **independent** of the orientation of the active frame.

Only one of the instructions TCOABS or TCOFR can be valid.

© Siemens AG, 2003. All rights reserved

Change of frame

The user can change the frame after selection of the tool. This does not have any effect on the components of the tool length compensation. Angle in the toolholder data:
The angles of rotation stored in the toolholder data are not affected by the angle of rotation defined by the frames. When changing from TCOFR to TCOABS, the original (programmed) angles or rotation in the toolholder data are activated again.

Tool compensation types

The TRC takes account of the current tool orientation when CUT2D or CUT3DFS is active.

All other tool compensation types
These are all compensation types of G code group 22, except for CUT3DC and CUT3DF. The response remains the same with respect to the plane used for compensation. This is determined independent of the tool orientation from the active frame.

For CUT2DF and CUT3DFF, the compensation plane used for TRC is determined from the frame **independent** of the current tool orientation. The active plane (G17/G18/G19) is considered.

CUT3DC and CUT3DF
3D tool compensation for circumferencial milling
3D tool compensation for face milling with active 5-axis transformation is not affected by the “Orientational toolholder” function.
The orientation information is determined by the active kinematic 5-axis transformation.

Limited toolholder orientation

An alarm is output if an orientation that cannot be reached with the defined toolholder kinematic is defined by the frame. The following kinematics cannot achieve any orientation:

- if the two rotary axes which are necessary to define the kinematic are not perpendicular to each other and
  if the tool axis which defines the tool direction does not stand perpendicular on the second rotary axis or
- less than two axes have been defined.

Non-rotary toolholder
The tool orientation used internally is dependent only on the basic orientation of the tool (see Subsection 2.8.3) and the active plane (G17–G19).

Ambiguities

A particular frame-selected tool orientation can with two axes generally be set with **two** different rotary angle pairs. The control selects the one setting for which the rotary angle is as close as possible to the programmed rotary angle.

Storing angles in the toolholder data
In virtually any case where ambiguities may arise, it is necessary to store the approximate angle expected from the frame in the toolholder data.

© Siemens AG, 2003. All rights reserved

Parameter sets

A complete set of parameters for an orientational toolholder consists of 33 values. The individual system variables are available as follows for the various software versions:

- SW 4 and higher — $TC_CARR1 to $TC_CARR17
- SW 5.3 and higher — $TC_CARR1 to $TC_CARR23
- SW 6.1 and higher — $TC_CARR1 to $TC_CARR33
  SW 6.4 and higher — $TC_CARR34 to $TC_CARR65 also freely available to user and for fine offsets.

The meaning of the individual parameters is distinguished as follows:

Machine kinematics:
$TC_CARR1 to $TC_CARR20 and $TC_CARR23

$TC_CARR18 to $TC_CARR20 hereby define a further vector l_4, which is needed to describe the machine with extended kinematics (both tool and table can be rotated), see Fig. 2-41.

$TC_CARR21 and $TC_CARR22 contain the channel axis identifiers of the rotary axes, the positions of which can if necessary to used to determine the orientation of the orientational toolholder.

Kinematic type:
$TC_CARR23 using a letter T, P or M
The following three options are available for the kinematic type; lower and upper case notation is supported:

- T: Only the **T**ool can be rotated (basic value).
- P: Only the workpiece (**P**art) can be rotated.
- M: Both tool and workpiece can be rotated (**M**ixed mode).

Any character other than the three mentioned here will result in the alarm 14153 if it is tried to activate the orientational toolholder.

Rotary axis parameters:
$TC_CARR24 to $TC_CARR33

The system variables in $TC_CARR24 to $TC_CARR33 can be used to define offsets, angle offsets, Hirth tooth system and axis limits.

Note

The system variables are available with and without active tool management.

Components and presetting of the chain/data block

The values $TC_CARR1 to $TC_CARR20 and $TC_CARR24 – $TC_CARR33 in the toolholder data block are of NC language format type REAL.

The values $TC_CARR21 and $TC_CARR22 for the axis identifier of the first rotary axis (v_1) and the second rotary axis (v_2) are of NC language format type AXIS. They are all preset to zero.

The value $TC_CARR23 is initialized with the uppercase letter “T” (only tool can be rotated).

$TC_CARRn[m]
$TC_CARR[0]= 0 has a special meaning, see Subsection 2.5.6.

© Siemens AG, 2003. All rights reserved

System variables for orientational toolholders

$TC_CARRn[m]

n: = Parameters 1...33
m:= Number orientational foodholder 1... value of MD 18088: MM_NUM_TOOL_CARRIER.

Description	NCK variables	Language format	Default
x component of the offset vector l_1	$TC_CARR1	REAL	0
y component of the offset vector l_1	$TC_CARR2	REAL	0
z component of the offset vector l_1	$TC_CARR3	REAL	0
x component of the offset vector l_2	$TC_CARR4	REAL	0
y component of the offset vector l_2	$TC_CARR5	REAL	0
z component of the offset vector l_2	$TC_CARR6	REAL	0
x component of rotary axis v_1	$TC_CARR7	REAL	0
y component of rotary axis v_1	$TC_CARR8	REAL	0
z component of rotary axis v_1	$TC_CARR9	REAL	0
x component of rotary axis v_2	$TC_CARR10	REAL	0
y component of rotary axis v_2	$TC_CARR11	REAL	0
z component of rotary axis v_2	$TC_CARR12	REAL	0
Angle of rotation α_1 (in degrees)	$TC_CARR13	REAL	0
Angle of rotation α_2 (in degrees)	$TC_CARR14	REAL	0
x component of the offset vector l_3	$TC_CARR15	REAL	0
y component of the offset vector l_3	$TC_CARR16	REAL	0
z component of the offset vector l_3	$TC_CARR17	REAL	0
x component of the offset vector l_4	$TC_CARR18	REAL	0
y component of the offset vector l_4	$TC_CARR19	REAL	0
z component of the offset vector l_4	$TC_CARR20	REAL	0
Axis identifier of the rotary axis v_1	$TC_CARR21	AXIS	0
Axis identifier of the rotary axis v_2	$TC_CARR22	AXIS	0
Kinematic type	$TC_CARR23	CHAR	T
Offset of rotary axis v_1	$TC_CARR24	REAL	0
Offset of rotary axis v_2	$TC_CARR25	REAL	0
Angle offset of rotary axis v_1 (Hirth tooth)	$TC_CARR26	REAL	0
Angle offset of rotary axis v_2 (Hirth tooth)	$TC_CARR27	REAL	0
Angle increment of rotary axis v_1 (Hirth tooth)	$TC_CARR28	REAL	0
Angle increment of rotary axis v_2 (Hirth tooth)	$TC_CARR29	REAL	0
Minimum position of rotary axis v_1 (SW limit)	$TC_CARR30	REAL	0
Minimum position of rotary axis v_2 (SW limit)	$TC_CARR31	REAL	0
Maximum position of rotary axis v_1 (SW limit)	$TC_CARR32	REAL	0
Maximum position of rotary axis v_2 (SW limit)	$TC_CARR33	REAL	0

© Siemens AG, 2003. All rights reserved

SW 6.4 and higher

System variables for the user and for fine offsets

$TC_CARR34 to $TC_CARR40
contains parameters that are freely available to the user and were not interpreted further up to software version 6.4 within the NCK or have no meaning.

$TC_CARR41 to $TC_CARR65
contain fine offset parameters that can be added to the values in the basic parameters. The fine offset value assigned to a basic parameter is obtained when the value 40 is added to the parameter number.

$TC_CARR47 to $TC_CARR54 and $TC_CARR61 to $TC_CARR63
are not defined and produce an alarm is read or write access is attempted.

Description	NCK variables	Lan-guage format	Default
Tool holder name *	$TC_CARR34	String[32]	""
Axis name 1 **	$TC_CARR35	String[32]	""
Axis name 2 **	$TC_CARR36	String[32]	""
Identifier **	$TC_CARR37	INT	0
Position component X **	$TC_CARR38	REAL	0
Position component Y **	$TC_CARR39	REAL	0
Position component Z **	$TC_CARR40	REAL	0
x comp. fine offset of the offset vector l_1	$TC_CARR41	REAL	0
y comp. fine offset of the offset vector l_1	$TC_CARR42	REAL	0
z comp. fine offset of the offset vector l_1	$TC_CARR43	REAL	0
x comp. fine offset of the offset vector l_2	$TC_CARR44	REAL	0
y comp. fine offset of the offset vector l_2	$TC_CARR45	REAL	0
z comp. fine offset of the offset vector l_2	$TC_CARR46	REAL	0
x comp. fine offset of the offset vector l_3	$TC_CARR55	REAL	0
y comp. fine offset of the offset vector l_3	$TC_CARR56	REAL	0
z comp. fine offset of the offset vector l_3	$TC_CARR57	REAL	0
x comp. fine offset of the offset vector l_4	$TC_CARR58	REAL	0
y comp. fine offset of the offset vector l_4	$TC_CARR59	REAL	0
z comp. fine offset of the offset vector l_4	$TC_CARR60	REAL	0
Offset of fine offset of rotary axis v_1	$TC_CARR64	REAL	0
Offset of fine offset of rotary axis v_2	$TC_CARR65	REAL	0

Remarks:
* System variable $TC_CARR34 cannot be assigned a number later, only a number.
** System variables $TC_CARR35 to $TC_CARR40 refer to the intended use of the orientable toolholder within the measuring cycles and can also be used for different purposes.

© Siemens AG, 2003. All rights reserved

2.5.2 Kinematic interaction and machine design (SW 4 and higher)

Representation of the kinematic chain (SW 4 and higher)

The concept of the kinematic chain is used to describe the kinematic interaction between a reference point and the tool tip.

The chain specifies all the data required for the toolholder data block in a schematic. To describe the concrete case with a particular kinematic, the relevant components of the chain must be assigned real vectors, lengths and angles. The chain represents the maximum constellation. In simple applications, individual components can be zero, e.g. kinematics with one or no rotary axis.

The machine does not have to have axes that rotate the tool and/or workpiece table. The function can be used even if the orientations are set manually by handwheel or reconfiguration.

The machine design is described by the following parameters:

- Two rotary axes (v_1 and v_2) each with
  one angle of rotation (α_1 or α_2) which counts positively for clockwise rotation facing the direction of the rotation vector.
- Up to four offset vectors (l_1 to l_4) for relevant machine dimensions (axis distances, distances to machine or tool reference points).

Zero vectors

Vectors v_1 and v_2 can be zero. The associated angle of rotation (explicitly programmed or calculated from the active frame) must then also be zero, since the direction of the rotating axis is not defined. If this condition is not satisfied, an alarm is produced when the toolholder is activated.

Less than two rotating axes

The option not to define a rotating axis makes sense when the toolholder to be described can only rotate the tool in one plane. A sensible minimum data block may therefore contain only one single entry different from 0 in the toolholder data; namely, a value in one of the components of v_1 or v_2 to describe a rotating axis parallel to the axis where the angle of rotation α_1 or α_2 is determined from one frame.

Further special cases

The two vectors v_1 and v_2 can be colinear. However, the degree of freedom for orientation is lost, i.e. such a kinematic is the same as one where only one rotary axis defined. All possible orientations lie on one cone sheath. The conical sheath deforms to a straight line if tool orientation t and v_1 or v_2 become colinear. Change of orientation is therefore not possible in this special case. The cone sheath deforms to a circular surface (i.e. all orientations are possible in one plane), if tool orientation t and v_1 or v_2 are perpendicular to each other.

It is permissible for the two vectors v_1 and v_2 to be zero. A change in orientation is then no longer possible. In this special case, the various lengths l_1 and l_2, which are not equal to zero, act as additional tool length compensations, in which the components in the individual axes are not affected by changing the plane (G17 – G19).

© Siemens AG, 2003. All rights reserved

Extension of the kinematics data (SW 5.3 and higher)

In software versions up to and including SW 5.2, system variable sets $TC_CARRR1 to $TC_CARR17 were available for “orientational toolholders” with the scope described at the start of Subsection 2.5.2.

The functionality described there is extended as follows in SW 5.3 and higher:

- Possibility of immediate access to existing machine axes in order to define the toolholder setting via the rotary axis positions.
- Extension of the kinematics with rotary workpiece and on kinematics with rotary tool and rotary workpiece.
- Possibility to permit only discrete values in a grid for the rotary axis positions (Hirth tooth system).

The extensions are compatible with earlier software versions and encompass the kinematic data blocks from $TC_CARR18 to $TC_CARR23.

Machines with rotary tool (SW 5.3 and higher)

On machines with rotary tool there is no change in the definition of the kinematics compared to older software versions. The newly introduced vector l_4, in particular, has no significance. Should the contents of l_4 not be zero, this is ignored.

The concept “Orientational toolholders” is actually no longer really appropriate for the new kinematic types with which the table can also be rotated, either alone or additionally to the tool. However, it has been kept for reasons of compatibility.

The kinematic chains used to describe the machine with rotary tool (general case) is shown in the diagram below in Fig. 2-39:

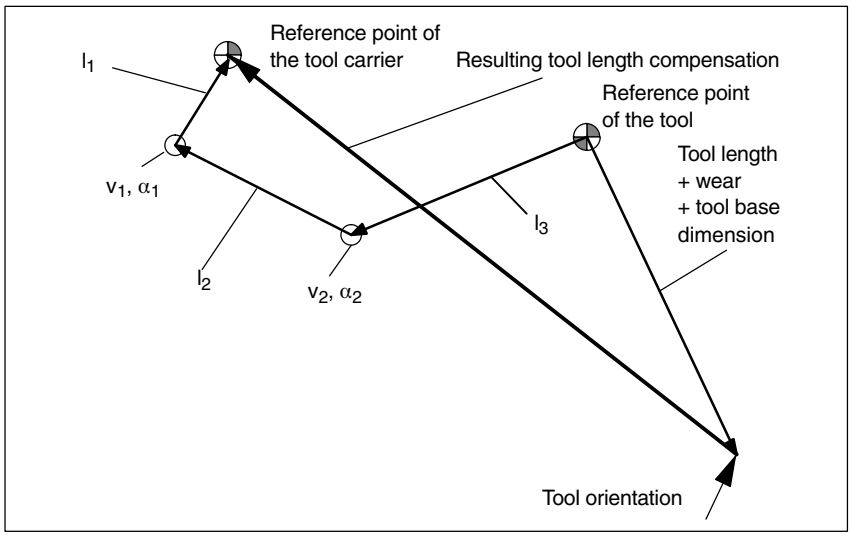


Fig. 2-39 Kinematic chain to describe a tool with orientation

Vectors which describe offsets in the rotary head are positive in the direction from the tool tip to the reference point of the toolholder.

The following kinematic type is defined for machines with a rotary tool:

Kinematic type:
$TC_CARR23 using letter T

© Siemens AG, 2003. All rights reserved

Machines with rotary workpiece (SW 5.3 and higher)

On machines with rotary workpiece the vector l_1 has no significance. If it contains a value other than zero, this is ignored. The kinematic chain for machines with rotary workpiece is shown in the diagram below in Fig. 2-40.

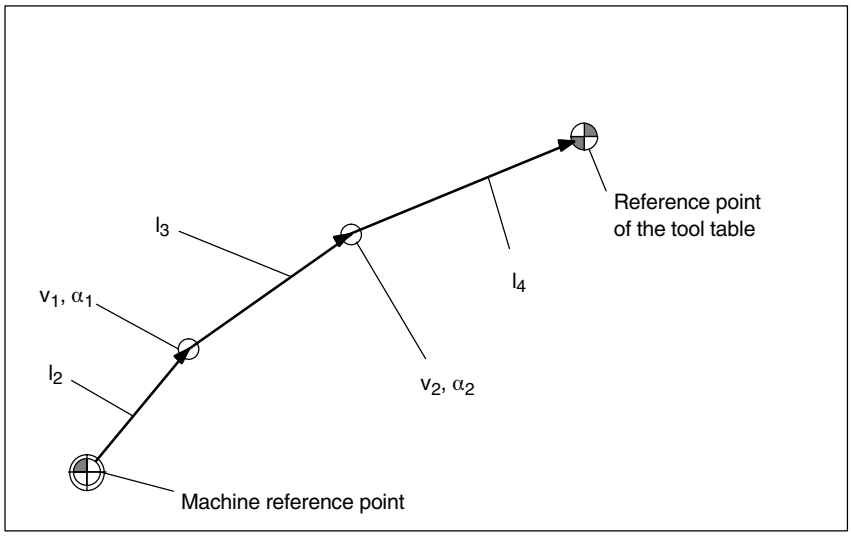


Fig. 2-40 Kinematic chain to describe a rotary table

Vectors which describe offsets in the rotary table are positive in the direction from the machine reference point to the table.

The following kinematic type is defined for machines with a rotary workpiece:

Kinematic type:
$TC_CARR23 using letter P

Note

On machines with rotary workpiece it is generally useful if the selected machine reference point and the reference point of the table are identical. Selecting the reference points in this way brings the advantage that the position of the workpiece zero in the basic setting (i.e. with rotary axes not turned) does not change when the rotary table is activated. The (open) kinematic chain in Fig. 2-40 is then closed. In this special case, therefore, the following formula applies:

Formula

$$\mathbf{l}_2 = -(\mathbf{l}_3 + \mathbf{l}_4) \quad (10)$$

© Siemens AG, 2003. All rights reserved

Maschinen mit extended kinematics (SW 5.3 and higher)

On machines with extended kinematics (both tool and workpiece are rotary) it is only possible to turn each of the components with one axis.

The kinematic of the rotary tool is described with the first rotary axis (v_1) and the two vectors l_1 and l_2, that of the rotary table with the second rotary axis (v_2) and the two vectors l_3 and l_4. The two chain components for machines with rotary tool and rotary workpiece are shown in the diagram below in Fig. 2-41.

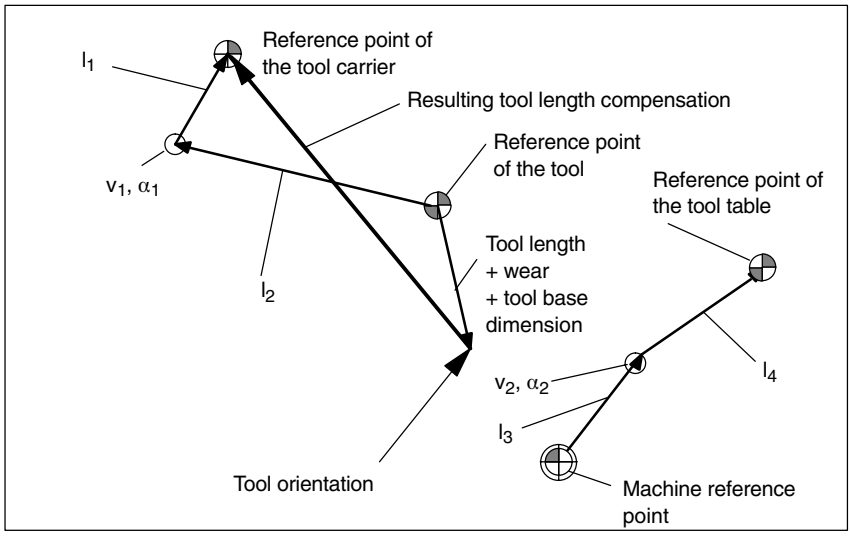


Fig. 2-41 Kinematic sequence with extended kinematics

The following kinematic type is defined for machines with a rotary tool and rotary workpiece:

Kinematic type:
$TC_CARR23 using letter M (extended kinematics)

Note

On machines with extended kinematics it is generally useful, as with machines where only the table can be rotated, for the machine reference point and the reference point of the table to be identical. The (open) partial chain to describe the table, as in Fig. 2-41, is then closed. In this special case the formula applies

---

Formula

$$l_3 = -l_4 \quad (11)$$

Rotary tool types T and M

For machine kinematics with a rotary tool (types T and M), the orientational toolholder component which describes the tool or head component (as opposed to the table component) acts, in conjunction with the active tool, as a new overall tool.

© Siemens AG, 2003. All rights reserved

Fine offset (SW 6.4 and higher)

The offset vectors l_1 to l_4 and the offsets of the rotary axes v_1 and v_2 can be represented as the sum of a basic value **and** a fine offset. The fine offset parameters assigned to the basic values are obtained by adding the

to the index of the basic value of **value 40**.

Example:
The parameter $TC_CARR5 is assigned to the fine offset $TC_CARR45.

Note

The meaning of the system variables $TC_CARR41 to $TC_CARR65 available for the fine offset are described in the table in Subsection 2.5.1 and also
Reference: /PGA/, Programming Guide Production Planning “Tool Offsets”

Activation
The fine offsets are added to the basic values when setting data SD 42974: TOCARR_FINE_CORRECTION = 1 is set.

Supplementary conditions
The amount is limited to the permissible fine offset. The maximum permissible value is defined for the

- components of vectors l_1 to l_4
  Machine data MD 20188: TOCARR_FINE_LIM_LIN
- offsets of the two rotary axes v_1 and v_2
  Machine data MD 20190: TOCARR_FINE_LIM_ROT

A permissible fine offse value is not detected unless an orientable toolholder that contains such a value is activated and setting data SD 42974: TOCARR_FINE_CORRECTION is set simultaneously.

Description of a rotation

A data block for describing a rotation comprises one vector v_1/v_2 to describe the direction of rotation of the rotary axis in its initial state and an angle α_1/α_2. The angle of rotation is counted positively in clockwise rotation and in the viewing direction.
The two toolholder angles α_1 and α_2 are determined using a frame, independent of the active plane currently selected (G17 – G19).

The tool orientation in the basic setting (both angles α_1 and α_2 equal to zero) is (as for the standard case) as follows:
For G17: Parallel to Z,
For G18: Parallel to Y and
For G19: Parallel to X.

© Siemens AG, 2003. All rights reserved

Assigning data to the toolholder

Machine with rotary toolholder (SW 5 and higher)

The following settings are obtained at the mill head show for a machine with orientable toolholder of kinematic type T:

Component of the offset vector l_1 = (–200, 0, 0)
Component of the offset vector l_2 = (0, 0, 0)
Component of the offset vector l_3 = (–100, 0, 0)
Component of the rotary axis v_1 = (1, 0, 0)
Component of the rotary axis v_2 = (–1, 0, 1)
Base dimension reference point tool (0, 0, 250)

Note

The tool reference point for the basic dimension is defined by the reference point at the machine.

For further information about reference points in the work area refer to:
References: /FB/, K1, “Axes. Coordinate Syst., Frames”

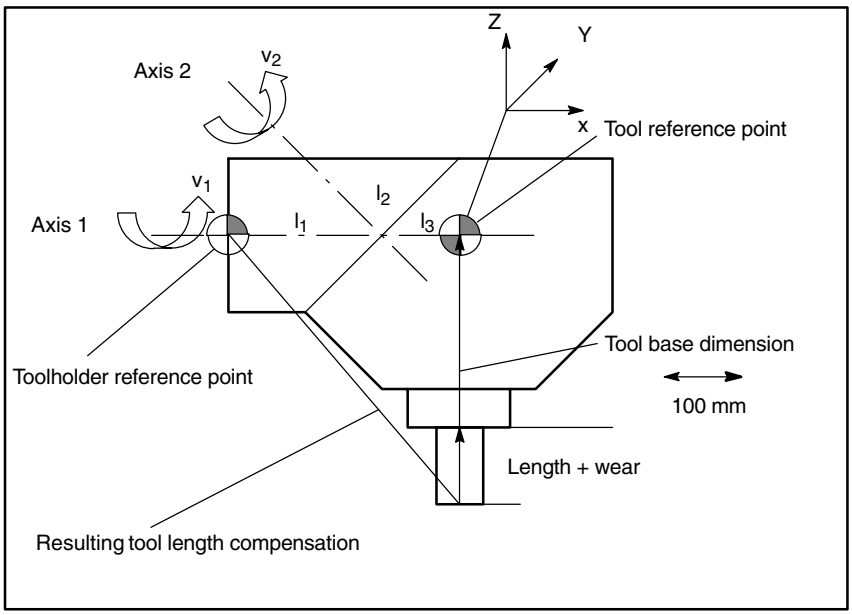


Fig. 2-42 Assignment of the toolholder data

Suitable assumptions were made for the following values in the data block:

- The two rotary axes intersect at one point. All components of l_2 are therefore zero.
- The first rotary axis lies in the xz plane, the second rotary axis is parallel to the x axis. These conditions define the directions of v_1 and v_2 (the lengths are irrelevant, provided that they are no zero).
- The reference point of the toolholder lies 200 mm in the negative x direction viewed from the intersection of the two rotary axes. This condition defines l_1.

© Siemens AG, 2003. All rights reserved

Specify associated data block values

The following associated data block values are specified for the toolholder shown on a machine with rotatble toolholder:

Description	NCK variables	Value
x component of the offset vector l_1	$TC_CARR1	– 200
y component of the offset vector l_1	$TC_CARR2	0
z component of the offset vector l_1	$TC_CARR3	0
x component of the offset vector l_2	$TC_CARR4	0
y component of the offset vector l_2	$TC_CARR5	0
z component of the offset vector l_2	$TC_CARR6	0
x component of rotary axis v_1	$TC_CARR7	1
y component of rotary axis v_1	$TC_CARR8	0
z component of rotary axis v_1	$TC_CARR9	0
x component of rotary axis v_2	$TC_CARR10	–1
y component of rotary axis v_2	$TC_CARR11	0
z component of rotary axis v_2	$TC_CARR12	1
Angle of rotation α_1 (in degrees)	$TC_CARR13	0
Angle of rotation α_2 (in degrees)	$TC_CARR14	0
x component of the offset vector l_3	$TC_CARR15	–100
y component of the offset vector l_3	$TC_CARR16	0
z component of the offset vector l_3	$TC_CARR17	0

Explanations

The toolholder kinematic chosen in the example is such that the two rotary axes form an angle of 45 degrees, which means that the orientation cannot take just any value. In concrete terms, this example does not permit the display of orientations with negative X components.

x component of the base dimension: 0
y component of the base dimension: 0
z component of the base dimension: 250

Note

The required data cannot be determined unequivocally from the geometry of the toolholder, i.e. the user is free to a certain extent to decide which data are to be stored. Thus, for the example, it is possible to specify only one z component for the base dimension up to the second axis. In this case, l_2 would no longer be zero, but would contain the components of the distance between this point on the second axis and a further point on the first axis. The point on the first axis can also be selected freely. Depending on which point is selected, l_1 must be selected such that the reference point (which can also be selected freely) is reached.

It is generally the case that vector components that are not changed by rotation of an axis can be distributed over any vectors “before” and “after” rotation.

© Siemens AG, 2003. All rights reserved

2.5.3 Inclined machining with 3 + 2 axes (SW 5.3 and higher)

Funktionale function

Inclined machining with 3 + 2 axes describes an extension of the concept of orientational toolholders and applies this concept to machines with a rotary table on which the orientation of the tool and table can be changed simultaneously.

The “Inclined machining with 3 + 2 axes” function is used to machine surfaces with any rotation with reference to the main planes XY (G17), ZX (G18) and YZ (G19).

It is possible to produce any orientation of the tool relative to the workpiece by rotating either the tool, the workpiece or both the tool and the workpiece.

The software automatically calculates the necessary compensating movements resulting from the tool lengths, lever arms and the angle of the rotary axis. It is always assumed that the required orientation is set first and not modified during a machining process such as pocket milling on an inclined plane.

Furthermore, the following 3 functions are described which are required for oblique machining:

1. **Path programming** in the tool orientation direction independent of an active frame.
2. Definition of a **frame rotation** by specifying the solid angle.
3. Definition of **rotation component in tool direction** in the programmed frame while maintaining the remaining frame components.

Demarcation to 5-axis tranformation

If the required functionality specifies that the TCP (tool center point) does not vary in the event of reorientation with reference to the workpiece, even during interpolation, the 5-axis software is required.

Note

For further information about 5-axis transformation, please refer to:
References: /FB/, F2, “3 to 5-axis transformation”

Specification of the orientational toolholder

The orientational toolholder is represented by a general 5-axis kinematic sequence described by a data block in the tool offset memory with a total of 33 REAL values. 31 of these values are constant for a toolholder (e.g. a millhead) which uses two rotary axes to set the orientation.
In SW 6.4 and higher, a data block is written in the tool correction memory with a total of 47 real values. For toolholders that have two rotary axes for setting the orientation, 45 of these value sare constant.

The remaining two values are variable and are used to specify the orientation. The constant values describe offsets and directions and setting options for the rotary axes; the variable values describe the angles of the rotary axes.

© Siemens AG, 2003. All rights reserved

2.5.4 Machine with rotary table (SW 5.3 and higher)

System variables

To date, the angles stored in $TC_CARR13 and $TC_CARR14 were used for the calculation of the active tool length with TCOABS. This still applies if $TC_CARR21 and $TC_CARR22 do not refer to rotary axes. If $TC_CARR21 or $TC_CARR22 contains a reference to a rotary axis in the channel, the axis position of the axis concerned at the start of the current block is taken as the angle instead of the entry in $TC_CARR13 or $TC_CARR14.
Mixed operation is permissible, i.e. the angle can be taken from the entry in the system variable $TC_CARR13 or $TC_CARR14 for the angle of one axis and from the position of a channel axis for the other axis.

This makes it possible for machines on which the axes which serve to set the orientational tool holder are known within the NC to access their position directly, whereas it was previously necessary for example to read the system variable $AA_IM[axis] and to write the result of the read operation into $TC_CARR13/14. This saves especially the implicit preprocessing stop for reading the axis positions.

MD 20180

If MD 20180: TOCARR_ROT_ANGLE_INCR[i] is zero, the rotary axis position is used with its programmed or calculated value. If the machine data is not zero however, the position used is the nearest grid point obtained for a suitable integer value n from the equation

Formula

$$\varphi = \$MC_TOCARR_ROT_ANGLE_OFFSET[i] + n * \$MC_TOCARR_ROT_ANGLE_INCR[i] \quad (12)$$

when n is an integer. This functionality is required if the rotary axes need to be indexed and thus cannot assume freely defined positions (e.g. with Hirth tooth systems). The system variable $P_TCANG[i] delivers the approximated valued, the system variable $P_TCDIFF[i] the difference between the exact and the approximated value.

Frame orientation TCOFR

With TCOFR (determination of the angle from the orientation defined by an active frame), the increments are scaled after determination of the angle from the active frame rotation. If the requested orientation is not possible due to the machine kinematic, the machining is aborted with an alarm. This also applies if the target orientation is very close to an achievable orientation. In particular the alarm in such situations cannot be prevented through the angle approximation.

TCARR Frame offset

A frame offset as a result of a toolholder change becomes effective immediately on selection of TCARR=... A change in the tool length, on the other hand, only becomes effective immediately if a tool is active.

TCOFR / TCOABS Frame rotation

A frame rotation does not take place on activation and a rotation which is already active is not changed. As in case T (only the tool can be rotated), the position of the rotary axes used for the calculation is dependent on the G code TCOFR/TCOABS and determined from the rotation component of an active frame or from the entries $TC_CARRn.

© Siemens AG, 2003. All rights reserved

Activation of a frame changes the position in the workpiece coordinate system accordingly, without compensating movement by the machine itself. The conditions are shown in the figure below.

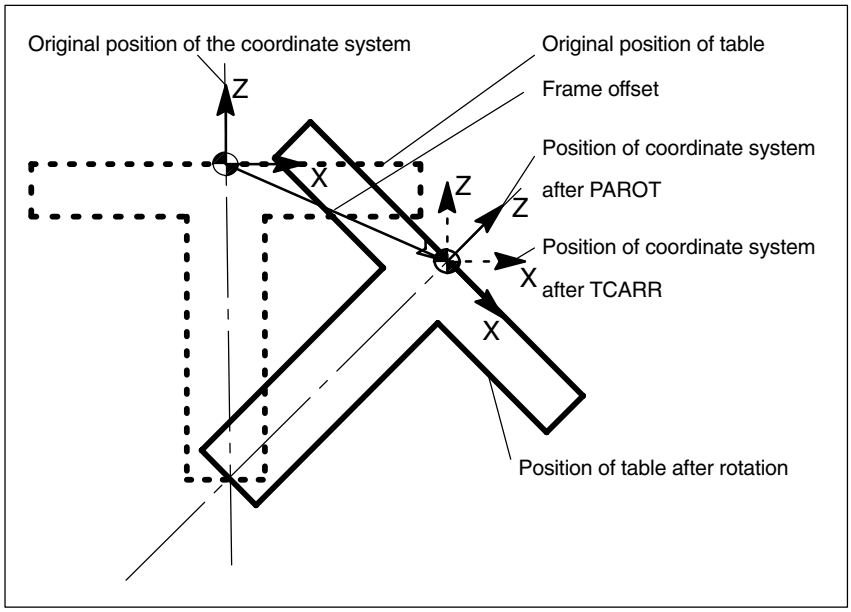


Fig. 2-43 Zero offset on activation of a rotary table with TCARR

Example

On the machine in the upper figure the rotary axis of the table is pointing in the positive Y direction. The table is rotated by +45 degrees. PAROT defines a frame which similarly describes a rotation of 45 degrees around the Y axis. The coordinate system is not rotated relative to the actual environment (marked in Fig. 2-43 with “Position of the coordinate system after TCARR”), but is rotated by –45 degrees relative to the defined coordinate system (position after PAROT). If this coordinate system is defined with ROT Y–45 for example, and if the toolholder is then selected with active TCOFR, an angle of +45 degrees will be determined for the rotary axis of the toolholder.

Rotary table

With rotary tables (kinematic types P and M) the activation with TCARR similarly does not lead to an immediate rotation of the coordinate system, (see Fig. 2-43), i.e. even though the zero point of the coordinate system is offset relative to the machine, while remaining fixed relative to the zero point of the workpiece, the orientation remains unchanged in space.

Activation kinematic types P and M

With kinematics of type P and M the selection of a toolholder activates an additive frame (table offset of the orientational toolholder), which takes into account the zero point offset as a result of the rotation of the table. The zero offset is entered

up to SW 5.3 in the base frame defined by MD 20184: TOCARR_BASE_FRAME_NUMBER. The translational part of this frame is then overwritten. Other contents of the frame are left unchanged.

© Siemens AG, 2003. All rights reserved

In SW 6.1 and higher, the zero offset can be written to a dedicated system frame $P_PARTFR. Bit 3 in

MD 28082: MM_SYSTEM_FRAME_MASK must be set for this purpose.
The basic frame specified by MD 20184: TOCARR_BASE_FRAME_NUMBER is then no longer required for the zero offset.

Activation kinematic type M

With kinematic type M (tool and table are each rotary around one axis), the activation of a toolholder with TCARR simultaneously produces a corresponding change in the effective tool length (if a tool is active) and the zero offset.

Rotations

Depending on the machining task it is necessary to take into account not only a zero offset (whether as frame or as tool length) when using a rotary toolholder or table, but also a rotation. However, the activation of an orientational toolholder never leads directly to a rotation of the coordinate system.

TOROT

If only the tool can be rotated, a frame whose Z axis points in the direction of the tool can be defined with TOFRAME or TOROT.

PAROT

If the coordinate system needs to be fixed relative to the workpiece, i.e. not only offset relative to the original position but also rotated according to the rotation of the table, then PAROT can be used to activate such a rotation in similar manner to the situation with rotary tool.

With PAROT the translations, scalings and mirroring in the active frame are kept, while the rotation component is rotated by the rotation component of an orientational toolholder corresponding to the table.

PAROT and TOROT take into account the overall orientation change in cases where the table or the tool are oriented with two rotary axes. With mixed kinematics only the corresponding component caused by a rotary axis is considered. It is thus possible, for example, when using TOROT, to rotate a workpiece such that an oblique plane lies parallel to the fixed XY plane of the room, whereby rotation of the tool must be taken into account in machining where any holes to be drilled, for example, are not perpendicular to this plane.

The language command PAROT is not rejected if no orientational toolholder is active. It causes no changes in the programmed frame.

Note

For further details of the TCARR, TOROT and PAROT functions with regard to channel-specific system frames, please refer to:

References: /FB1/, K2, “Axes, Coordinate Systems, Frames, Reset”

© Siemens AG, 2003. All rights reserved

2.5.5 Procedure when using orientational toolholders

Creating a toolholder

The number of available toolholder data blocks in the NCK is defined in machine data MD 18088: MM_NUM_TOOL_CARRIER.

1. The value is calculated as follows:

   MD 18088: MM_NUM_TOOL_CARRIER =
   'number of TO units' * 'number of toolholder data blocks of a TO unit'

   MD 18088: MM_NUM_TOOL_CARRIER **/** 'number of TO units'
   are allocated permanently to each TO unit.

Note

For further information about the definition and assignment of a TO unit using MD 28085: MM_LINK_TOA_UNIT, please refer to:

References: /FB/, S7, "Memory configuration"

---

2. Zero setting of toolholder data:

   To zero all values of all data blocks, you can use the command
   $TC_CARR1[0] = 0 **Zero all values in all data blocks**

   Individual toolholder data blocks can be deleted selectively with the NC command DELTC or the PI service _N_DELTCAR.

3. Accessing the data of a toolholder:

   a) Parts program

   $TC_CARRn[m] = value
   This writes the new 'value' of the system variable to parameter n for toolholder m.

   Value = $TC_CARRn[m]
   with 'def real value' – the parameters of a toolholder m can be read if they have already been defined
   (MD 18088: MM_NUM_TOOL_CARRIER).
   Otherwise, an alarm is output.

   b) Operator panel interface

   The parameters of an orientational toolholder can be read and written with the NCKMMC (OPI) variable services using the system variable $P_TCANG[<n>].

4. Data backup:

   The system variables specified above are saved as part of the general NCK data backup.

© Siemens AG, 2003. All rights reserved

Selecting the toolholder

A toolholder with number m is selected with the NC program command TCARR = m (TCARR **T**ool **Carr**ier). TCARR = 0 deselects an active toolholder.

New tool or new toolholder
When a new tool is activated, it is always treated as if it was mounted on the active toolholder.

A new toolholder is activated immediately when it is programmed. It is not necessary to change tools or reprogram the active tool. The toolholder (number) and tool (number) are independent and can be used in any combination.

Toolholder from G code of group 42

TCOABS (**T**ool **C**arrier **O**rientation **ABS**olute): The tool orientation is determined explicitly if the corresponding values are entered in system variable $TC_CARR13 or $TC_CARR14 and G code TCOABS is activated in G code group 42.

TCOFR (**T**ool **C**arrier **O**rientation **FR**ame):
The tool orientation can also be determined automatically from the current orientation of an active frame on tool selection if one of the following G codes is active in G code group 42 when the toolholder is selected:

- TCOFR or TCOFRZ (SW 6.1 and higher)
  The orientational toolholder is set such that the tool points in the Z direction.
- TCOFRX (SW 6.1 and higher)
  The orientational toolholder is set such that the tool points in the X direction.
- TCOFRY (SW 6.1 and higher)
  The orientational toolholder is set such that the tool points in the Y direction.

The effect of TCOFR is such that, when machining on an inclined surface, the tool offsets are considered implicitly as if the tool were standing vertically on the surface.

Note

The tool orientation is not bound strictly to the frame orientation. When a frame is active and G code TCOABS is active, you can select a tool on which the orientation of the tool is independent of the orientation of the active frame.

After the tool selection, you can change the frame which does not affect the components of the tool length compensation. It is then no longer certain that the tool is positioned perpendicular to the machining plane. You should therefore first check that the intended tool orientation is maintained on an inclined surface.

When TCOFR etc. is active, the tool orientation used in the tool length calculation is always determined from the active frame each time the toolholder is changed.

© Siemens AG, 2003. All rights reserved

Toolholder from G code of group 53

The G codes of group 53 (TOFRAME, TOROT, etc.) can be used to define a frame such that an axis direction (Z, Y or X) in this frame is equal to the current tool orientation.

The G code of group 6 (G17 – G19) which is active at the time TOFRAME is called determines the tool orientation.

If no toolholder is active, or if a toolholder is active but does not cause the tool orientation to change, the Z direction in the new frame is

- the same as the old Z direction with G17,
- the same as the old Y direction with G18,
- the same as the old X direction with G19.

These directions are modified accordingly for rotating toolholders. The same applies to the new X and Y directions.

In SW 6.1 and higher, any of the G codes TOFRAMEX, TOFRAMEY, TOROTX or TOROTY can be used instead of TOFRAME or TOROT. The meanings of the axes are interchanged accordingly.

Group change

Changing the G code from group 42 (TCOABS,TCOFR, etc.) causes recalculation of the tool length components.

The (programmed) angles of rotation stored in the toolholder data are not affected, with the result that the angles originally stored in the tool data are reactivated on a change from TCOFR to TCOABS.

Read angle of rotation (α_1 or α_2):
The angles currently used to calculate the orientation can be read from system variable $P_TCANG[n] with n = 1 or n = 2.

If two permissible solutions (i.e. a second valid pair of angles) are available for a particular orientation, the values can be accessed with $P_TCANG[3] or $P_TCANG[4]. The number of valid solutions 0 to 2 can be read with $P_TCSOL.

Tool radius compensation with CUT2D or CUT3DFS:
The current tool orientation is included in the tool radius compensation if either CUT2D or CUT3DFS is active in G code group 22 (tool compensation type).

For non-rotating toolholders, the behavior depends solely on the active plane of G code group 6 (G17 – G19) and is thus identical to the previous behavior.

All other tool compensation types:
The behavior is unchanged for all other tool compensation types.

For CUT2DF and CUT3DFF in particular, the compensation plane used for TRC is determined from the frame independent of the current tool orientation. Allowance is made for the active plane (G17 – G19) and the behavior is thus the same as before.

The two remaining G codes of group 22, CUT3DC and CUT3DF, are not affected by the toolholder functionality because the tool orientation information in these cases is prepared by the active kinematic transformation.

© Siemens AG, 2003. All rights reserved

Two rotary axes

Two general solutions exist for two rotary axes. The control itself chooses these two solution pairs such that the orientation angles resulting from the frame are as close as possible to the specified angles.

The two following options are available for specifying the angles:

1. If $TC_CARR21 or $TC_CARR22 contains a reference to a rotary axis, the position of this axis at the start of the block in which the toolholder is activated is used to specify the angle.
2. If $TC_CARR21 or $TC_CARR22 does not contain a reference to a rotary axis, the values contained in $TC_CARR13 or $TC_CARR14 are used.

Example

The control first calculates an angle of 10 degrees for one axis. The specified angle is 750 degrees. 720 degrees (= 2 * 360 degrees) are then added to the initial angle, yielding a final angle of 730 degrees.

Rotary axis offset (SW 6.1 and higher)

Rotary axis offsets can be specified with system variables $TC_CARR24 and $TC_CARR25. A value not equal to zero in one of these parameters means that the associated rotary axis has the position specified by the parameter (and not position zero) in the initial setting. All angle specifications then refer to the coordinate system displaced by this value.

When the machining plane is changed (G17 – G19)
only the tool length components of the active tool are interchanged. The components of the toolholder are not interchanged. The resulting tool length vector is then rotated in accordance with the current toolholder and, if necessary, modified by the offsets belonging to the toolholder.

The two toolholder angles α_1 and α_2 are determined using a frame, independent of the active plane currently selected (G17 – G19)

Limits (SW 6.1 and higher)

Limit angles (software limits) can be specified for each rotary axis in the system variable set ($TC_CARR30 to $TC_CARR33) used to describe the orientational toolholder. These limits are not evaluated if both the minimum and maximum value is zero.

If at least one of the two limits is not equal to zero, the system checks whether the previously calculated solution is within the permissible limits. If this is not the case, an initial attempt is made to reach a valid setting by adding or subtracting multiples of 360 degrees to or from the invalid axis position. If this is impossible and two different solutions exist, the first solution is discarded and the second solution is used. The second solution is treated the same as the first with reference to the axis limits.

If the first solution is discarded and the second used instead, the contents of $P_TCANG[1/2] and $P_TCANG[3/4] are swapped, hence the solution actually used is also stored in $P_TCANG[1/2] in this case.

The axis limits are monitored even if the axis angle is specified instead of being calculated. This is the case if TCOABS is active when an orientational toolholder is activated.

© Siemens AG, 2003. All rights reserved

2.5.6 Programming (SW 4 and higher)

Selecting the toolholder

A toolholder with number m is selected with the number **m** of the toolholder with: **TCARR = m**

Access to toolholder data blocks

The following access is possible from the parts program:

The new 'value' is **written** to parameter **n** for toolholder **m** with:
$TC_CARRn[m] = value

The parameters of a toolholder m can be
read with: **value = $TC_CARRn[m]**,

if the toolholder data block has already been defined.
value must be a REAL variable.

The toolholder data block no. must be within the range defined by machine data MD 18088: MM_NUM_TOOL_CARRIER (total number of definable toolholder data blocks).
For an NC channel, this number can be defined as divided by the number of active toolholder data block channels. Exception: If MD 28085: MM_LINK_TOA_UNIT is used to select values that deviate from standard.

Canceling all toolholder data blocks

From within the parts program, it is possible to deleted all the values of all the toolholder data blocks using one command.

$**TC_CARR1[0] = 0**

Values not set by the user are preset to 0.

Activation

A toolholder becomes active when both a tool**holder** and a **tool** have been activated. The selection of the toolholder alone has no effect. The effect of selecting a toolholder depends on the G code **TCOABS/TCOFR** (modal G code group for toolholders).

Changing the G code in the TCOABS/TCOFR group causes recalculation of the tool length components when the toolholder is active. With TCOABS, the values stored in the toolholder data for both angles of rotation α_1 and α_2 are used to determine the tool orientation.

With TCOFR, the two angles are determined from the current frame. The values stored in the toolholder data are not changed, however. These are also used to resolve the ambiguity that can result when the angle of rotation is calculated from one frame. Here, the angle that deviates least from the programmed angle is selected from the various possible angles.

Note

For further information about programming tool offsets with toolholder kinematics and system variables, please refer to:

Reference: /PGA/, "Programming Guide, Advanced"

© Siemens AG, 2003. All rights reserved

2.5.7 Supplementary conditions and control system response for orientation

Full orientation

For a given data block that describes a certain kinematic, all the conceivable special orientations can only be displayed when the following conditions are satisfied:

- The two vectors v_1 and $v_{2,}$ that describe the rotary axes, must be defined, i.e. must not be equal to zero.
- The two vectors v_1 and v_2 must be perpendicular to each other.
- The tool orientation must be perpendicular to the second rotary axis.

Non-defined orientation

If these conditions are not satisfied and an orientation that cannot be achieved by an active frame is requested with TCOFR, an alarm is output.

Dependencies between vector/angle of rotation
If a vector v_1 or v_2 that describes the direction of rotation is equal to zero, the associated angle of rotation α_1 or α_2 must also be zero. Otherwise, an alarm is produced. The alarm is not output until the toolholder is activated, i.e. when the toolholder is being changed.

Tool fine offset combined with orientation
The combination of tool fine offset and toolholders is **not** permissible. The activation of tool fine offset for an active toolholder, and vice versa the activation of the tool holder with an active tool fine offset, produces an alarm.

Automatic toolholder selection, RESET

The machine data MD 20126: TOOL_CARRIER_RESET_VALUE can be used to select a toolholder automatically on Reset or program start and is controlled via MD 20120: TOOL_RESET_VALUE (analogous to controlled tool selection). The response to Reset or program start is controlled by the same bit 6 MD 20110: RESET_MODE_MASK or in MD 20112: START_MODE_MASK as for tool selection.

References: /FB/, K1, “Mode Group, Channel, Program Operation, Reset Response”

SW 6.3 and higher
If TCOABS was active for the last selection before reset, the behavior is unchanged compared to previous versions. A differnet active G–Code causes the orientable toolholder to be activated with the frame that was active before the last reset. Modified toolholder data ($TC_CARR...) are also considered. If these data are unchanged, the toolholder is activated in exactly the same state as before reset. If the toolholder data were changed after the toolholder selection before reset, selection corresponding to the last frame is not always possible. In this case, the orientable toolholder is selected according to the G–Code (group 42) values valid at this time and the active frame.

MD 22530 Auxiliary function output to PLC

Machine data MD 22530: $MC_TOCARR_CHANGE_M_CODE can be set so that, optionally, a constant or an M code is output when the toolholder is selected. The number of the code is derived from the toolholder number.

References: /FB/, H2, “Output of Auxiliary Functions to PLC”

© Siemens AG, 2003. All rights reserved

Toolholder kinematics

The following supplementary conditions must be met for the toolholder kinematics:

- The tool orientation in the initial setting (both angles α_1 and α_2 are zero) is as standard, even when
  - – G17: Parallel to Z
  - – G18: Parallel to Y
  - – G19: Parallel to X
- A permissible position in terms of the axis limits must be reachable.
- For any possible orientation to be set, the two rotary axes must be perpendicular to each other.

  Furthermore, on machines on which both axes rotate the table, the tool orientation must stand perpendicular to the first rotary axis.
  On machines with mixed kinematics the tool orientation must stand perpendicular on the axis which rotates the tool, i.e. similarly on the first rotary axis.

The following applies for orientations **specified in a frame**:

- The orientation specified in a frame must be achievable with the defined toolholder kinematics otherwise an alarm is output.

  This situation can occur if the two rotary axes required to define the kinematics are not perpendicular to each other. This applies if less than two rotary axes are defined and

  the case for **kinematic type T with rotary tool** if
  the tool axis which defines the tool direction does not stand perpendicular on the **second** rotary axis.
  the case for **kinematic types M and P with rotary workpiece** if the tool axis which defines the tool direction is not perpendicular to the **first** axis.
- Rotary axes which require a frame with a defined tool orientation in order to reach a specific position are only determined unambiguously in the case of one rotary axis. Two general solutions exist for two rotary axes.
- In all cases where ambiguities may arise, it is particularly important that the approximate angles expected from the frame are stored in the tool data, and that the rotary axes are in the vicinity of the expected positions.

Response with ASUB, REPOS

The toolholder can be changed in an asynchronous subprogram (ASUB). When the interrupted program is continued with REPOS, the approach motion of the new toolholder is taken into account and the program continued with this motion. The treatment here is analogous to tool change in an ASUB.

References: /FB/, K1, “Mode Group, Channel, Program Operation, Reset Response”

© Siemens AG, 2003. All rights reserved

2.6 Incrementally programmed compensation values

2.6.1 G91 extension (SW 4.3 and higher)

Preconditions

Incremental programming with G91 is defined in SW 4.3 and higher such that the compensation value is traversed additively to the incrementally programmed value when a tool offset is selected.

Applications

For applications such as scratching it is important to traverse only the programmed path when traversing incrementally. The activated tool offset is not traversed.

Sequence

Selection of a tool offset with incremental programming

- Scratch workpiece with tool tip
- Save the actual position in the basic frame (set actual value) after reducing it by the tool offset
- Traverse incrementally from the zero position

Activation

With the setting data SD 42442: TOOL_OFFSET_INCR_PROG it is possible to define whether a changed tool length is traversed with FRAME and incremental programming of an axis, or whether only the programmed path is traversed.

Zero offset/frames G91

With the setting data SD 42440: With FRAME_OFFSET_INCR_PROG it is possible to set whether zero offset is traversed as standard with FRAME and incremental programming (value = 1) or whether only the programmed value is traversed (value = 0), see /FB/, K2, “Frames, Frames with G91”.

Condition

If the response is set such that the offset remains active beyond program end and RESET (MD 20110: RESET_MODE_MASK, Bit 6=1), and if an incremental path is programmed in the first parts program block, the offset is always traversed additively to the programmed path.

Note

With this configuration, parts programs must always begin with absolute programming.

© Siemens AG, 2003. All rights reserved

2.6.2 Machining in direction of tool orientation (SW 5.3 and higher)

Typical application

On machines with orientational toolholder it is possible to traverse in tool direction (typically when drilling),

- without activating a frame (e.g. with TOFRAME or TOROT) in which one of the axes points in the direction of the tool.

Similarly on machines on which, during oblique machining

- a frame defining the oblique plane is active, but the tool cannot be set exactly perpendicularly because an indexed toolholder (Hirth tooth system) prevents free setting of the tool orientation.

In these cases it is then necessary – contrary to the actually requested motion perpendicular to the plane – to drill in tool direction, as the drill would otherwise not be guided in the direction of its longitudinal axis, which among other things would lead to breaking of the drill.

MOVT

The end point of such a motion is programmed with MOVT=.... The programmed value is effective incrementally in the tool direction as standard. The positive direction is defined from the tool tip to the toolholder. The contents of MOVT is thus generally negative for the infeed motion (when drilling), and positive for the retraction motion. This corresponds to the situation with normal paraxial machining, e.g. with G91 Z...

If the motion is programmed in the form MOVT=AC(...), MOVT functions absolutely. In this case a plane is defined which runs through the current zero point, and whose surface normal vector is parallel to the tool orientation. MOVT then gives the position relative to this plane, see figure below.

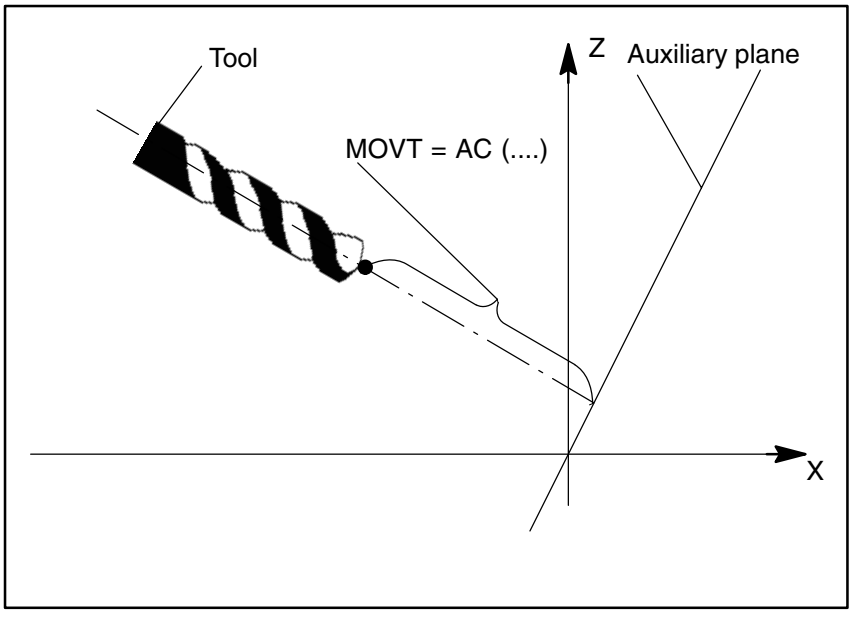


Fig. 2-44 Definition of the position for absolute programming of a motion in tool direction

© Siemens AG, 2003. All rights reserved

The reference to this auxiliary plane serves only for calculation of the end position. Active frames are not affected by this internal calculation.

Instead of MOVT=... it is also possible to write MOVT=IC(...) if it is to be plainly visible that MOVT is to function incrementally. There is no functional difference between the two forms.

Supplementary conditions

The following supplementary conditions apply for programming with MOVT:

- It is independent of the existence of an orientational toolholder. The direction of the motion is dependent on the active plane. It runs in the directions of the vertical axes, i.e. with G17 in Z direction, with G18 in Y direction and with G19 in X direction. This applies both where no orientational toolholder is active and for the case of an orientational toolholder without rotary tool or with a rotary tool in its basis setting.
- MOVT acts similarly for active orientation transformation (3–4–5-axis transformation).
- If in a block with MOVT the tool orientation is changed simultaneously (e.g. active 5-axis transformation by simultaneous interpolation of the rotary axes), the orientation at the start of the block is decisive for the direction of movement of MOVT. The path of the tool tip (tool center point – TCP) is not affected by the orientation change.
- Linear or spline interpolation (G0, G1, ASPLINE, BSPLINE, CSPLINE) must be active. Otherwise, an alarm is produced. If a spline interpolation is active, the resultant path is generally not a straight line, since the end point calculated by MOVT is treated as if it had been programmed explicitly with X, Y, Z.
- A block with MOVT must not contain any programming of geometry axes (Alarm 14157).

© Siemens AG, 2003. All rights reserved

2.7 Basic tool orientation (SW 6.1 and higher)

Application

Normally, the orientation assigned to the tool itself depends exclusively on the active machining plane. For example, the tool orientation is parallel to Z with G17, parallel to Y with G18 and parallel to X with G19.

Different tool orientations can only be programmed by activating a 5-axis transformation. The following system variables have been introduced in order to assign a separate orientation to each tool cutting edge:

System variables	Description of tool orientation	Format	Default
$TC_DPV[t, d]	Tool cutting edge orientation	INT	0
$TC_DPV3[t, d]	L1 component of tool orientation	REAL	0
$TC_DPV4[t, d]	L2 component of tool orientation	REAL	0
$TC_DPV5[t, d]	L3 component of tool orientation	REAL	0

Index coding: Same as tool system variable $TC_DPx[t, d]

t: T number of cutting edge
d: D number of cutting edge

Identifiers
$TC_DPV3 to $TC_DPV5 are analogous to identifiers
$TC_DP3 to $TC_DP5 of the tool length components.

MD 18114

The system variables for describing the tool orientation are only available if machine data

MD18114: MM_ENABLE_TOOL_ORIENT is not equal to zero.

If this machine data MD18114: MM_ENABLE_TOOL_ORIENT contains
value 1: Only parameter $TC_DPV[t, d] is available
value 2: All four system variables are available.

Define direction vector

If the contents of all four system variables is 0, the orientation is only defined by the active plane, as before.

If system variable $TC_DPV[t, d] is equal to zero,
the other three parameters – if available – define a direction vector. The amount of the vector is insignificant.

Example:
$TC_DPV[1, 1] = 0
$TC_DPV3[1, 1] = 1.0
$TC_DPV4[1, 1] = 0.0
$TC_DPV5[1, 1] = 1.0

In this example, the basic orientation points in the direction of the bisector in the L1-L3 plane, i.e. the basic orientation in the bisector for a

milling tool and active plane G17 in the ZX plane

© Siemens AG, 2003. All rights reserved

Basic orientation of tools

The basic orientation of

- turning and grinding tools is assumed with G18
- milling tools is assumed with G17

The active tool orientation is unchanged in these cases and is equivalent to the original settings in $TC_DPVx[t, d].

The basic orientation is always the direction perpendicular to the plane in which tool radius compensation is performed. With turning tools, in particular, the tool orientation generally coincides with the longitudinal tool axis.

The setting data specified below are operative only
if the basic orientation of the tool is defined by an entry in at least one of the system variables $TC_DPVx[t, d].

They have no effect
if the tool orientation is only determined by the plane selection G17 – G19 and is compatible with previous behavior.

If setting data SD 42950: TOOL_LENGTH_TYPE is not equal to zero, the plane of the basic orientation for a cutting edge is treated either like a milling tool or like a turning tool, irrespective of the entry in $TC_DP1.

Plane change

A change of plane causes a change of orientation. The following rotations are initiated:

On a change from

G17 ==> G18:
G18 ==> G19:
G19 ==> G17:

Rotation through –90 degrees about the Z axis and then
rotation through –90 degrees about the X axis.

On a change from

G17 ==> G19:
G18 ==> G17:
G19 ==> G18:

Rotation through 90 degrees about the X axis and then
rotation through 90 degrees about the Z axis.

These rotations are the same as those that have to be performed in order to interchange the components of the tool length vector on a plane change.

The basic orientation is also rotated when an adapter transformation is active.

If setting data SD 42940: TOOL_LENGTH_CONST is not equal to zero, the tool orientation is not rotated on a plane change.

Tool length components

The tool orientation components are handled in in terms of their setting data
SD 42910: MIRROR_TOOL_LENGTH and
SD 42950: TOOL_LENGTH_TYPE

in exactly the same way as tool length components, i.e. the components are exchanged accordingly and assigned to geometry axes.

© Siemens AG, 2003. All rights reserved

System variable $TC_DPV[t, d]

The purpose of system variable $TC_DPV[t, d] is to allow the simple specification of certain frequently needed basic orientations (parallel to coordinate axes). The permissible values are shown in the table below. The values in the first and second/third columns are equivalent.

$TC_DPV[t, d]	Basic orientation	
	Milling tools *	Turning tools *
<= 0 or > 6	($TC_DPV5[t, d], $TC_DPV4[t, d], $TC_DPV3[t, d],) **	($TC_DPV3[t, d], $TC_DPV5[t, d], $TC_DPV4[t, d],) **
1	(0, 0, V)	(0, V, 0)
2	(0, V, 0)	(0, 0, V)
3	(V, 0, 0)	(V, 0, 0)
4	(0, 0, –V)	(0, –V, 0)
5	(0, –V, 0)	(0, 0, –V)
6	(–V, 0, 0)	(–V, 0, 0)

*) Turning tools in this context are any tools whose tool type ($TC_DP1[t, d]) is between 400 and 599.
All other tool types identify milling tools.

**) If all three values $TC_DPV3[t, d], $TC_DPV4[t, d], $TC_DPV5[t, d] are equal to zero in this case, the tool orientation is determined by the active machining plane (default).

In the table above, V
stands for a positive value in the system variable.

Example:
For milling tools,
$TC_DPV[t, d] = 2 is equivalent to
$TC_DPV3[t, d] = 0, $TC_DPV4[t, d] = 0, $TC_DPV5[t, d] = V.

Supplementary conditions

If the “Scratching” function is used in the RESET state, the following must be noted with respect to the initial setting value:

- The wear components are evaluated depending on the initial setting values of the G code groups TOWSTD, TOWMCS and TOWWCS.
- If a value other than the initial setting value is needed to ensure correct calculation, scratching may be performed only in the Stop state.

Note

The following Section 2.8 “Special handling of tool offsets”
is paid to tool offsets with evaluation of sign for tool length with wear and temperature fluctuations.

The following factors are taken into account:
Tool type
Transformations for tool components
Assignment of tool length components
to geometry axes and independently of tool type

© Siemens AG, 2003. All rights reserved

2.8 Tool offset special features (SW 5 and higher)

SD 42900–42950

The setting data SD 42900 – SD 42940 permit control of the evaluation of the signs for tool length and wear. In SW 5.2 and higher, this can be performed with SD 42950 independent of the actual tool type.

The same applies to the response of the wear components when mirroring geometry axes or changing the machining plane via setting data.

References: /PG/ Programming Guide Fundamentals, "Tool offsets"

Note

Where reference is made below to wear values, this refers to the sum of the actual wear values ($TC_DP12 to $TC_DP20) and the sum offsets with the wear values ($SCPX3 to $SCPX11) and setup values ($ECPX3 to $ECPX11). For more information on sum offsets, please refer to /FBW/, Description of Functions, Tool Management.

Required setting data

- SD 42900: MIRROR_TOOL_LENGTH (mirroring of tool length components and components of the tool base dimension)
- SD 42910: MIRROR_TOOL_WEAR (mirroring of wear values of tool length components)
- SD 42920: WEAR_SIGN_CUTPOS (sign evaluation of the wear components)
- SD 42930: WEAR_SIGN (inverts the sign of the wear dimensions)
- SD 42935: WEAR_TRANSFORM (transformation of wear values)
- SD 42940: TOOL_LENGTH_CONST (assign tool length components to geometry axes)
- SD 42950: TOOL_LENGTH_TYPE (assign tool length components independent of tool type)
- SD 42960: TOOL_TEMP_COMP (tool length offsets in Subsection 2.8.2)

a) Mirroring of tool lengths (SD 42900: MIRROR_TOOL_LENGTH)

Setting data **not equal to** zero:
The tool length components ($TC_DP3, $TC_DP4 and $TC_DP5) and the components of the tool base dimensions ($TC_DP21, $TC_DP22 and $TC_DP23), whose associated axes are

mirrored, are also mirrored – by sign inversion.

The wear values are **not** mirrored. If these must be mirrored as well, setting data SD 42910: MIRROR_TOOL_WEAR must also be set.

© Siemens AG, 2003. All rights reserved

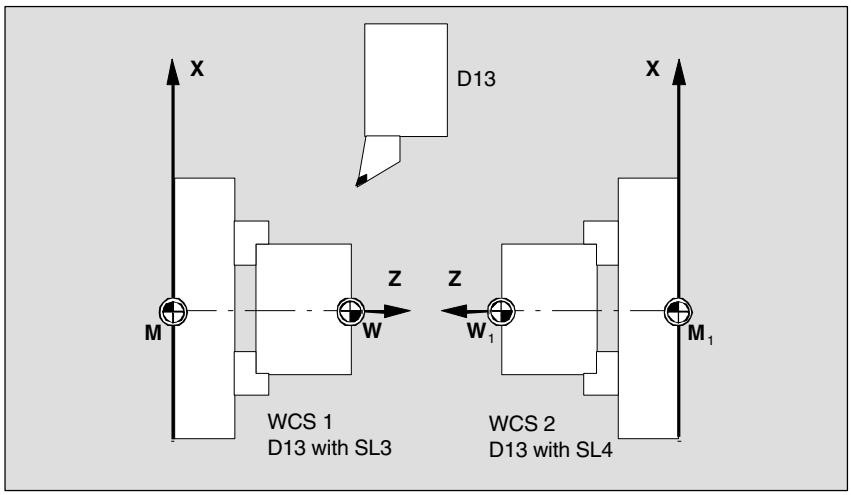


Fig. 2-45 Example for mirroring tool lengths

SD 42910: MIRROR_TOOL_WEAR

Setting data **not equal to** zero:
The wear values of the tool length components, whose associated axes are mirrored, are also mirrored – by sign inversion.

b) Wear sign evaluation (SD 42920: WEAR_SIGN_CUTPOS)

Setting data **not equal to** zero:
In the case of tools with a relevant tool point direction (turning and grinding tools – tool types 400–599), the sign evaluation of the wear components depends on the tool point direction in the machining plane. This setting data has no effect with tool types which do not have a relevant tool point direction.

In the following table, the dimensions whose sign is inverted by SD 42920 (not equal to 0) are marked with an X:

Cutting edge position	Length 1	Length 2
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

Note

The sign settings of SD 42920 and 42910 are independent. For example, if the sign of a measurement parameter is changed by both setting data, the resulting sign is unchanged.

© Siemens AG, 2003. All rights reserved

SD 42930: WEAR_SIGN

Setting data **not equal to** zero:
Inverts the sign of all wear dimensions. This affects both the tool length and other variables such as tool radius, rounding radius, etc.
Entering a positive wear dimension makes the tool “shorter” and “thinner”.

Activation of modified setting data

When the setting data described above are modified, the tool components are not reevaluated until the next time a tool edge is selected. If a tool is already active and the data of this tool are to be reevaluated, the tool must be selected again.

Example:

N10 $SC_WEAR_SIGN = 0	No sign inversion of the wear values
N20 $TC_DP1[1,1] = 120	End mill
N30 $TC_DP6[1,1] = 100	Tool radius 100 mm
N40 $TC_DP15[1,1] = 1	Wear dimension of tool radius 1 mm, resulting tool radius 101 mm
N100 T1 D1 G41 X150 Y20	
....	
N150 G40 X300N10	
....	
N200 $SC_WEAR_SIGN = 1	Sign inversion for all wear values; the new radius of 99 mm is activated on a new selection (D1). Without D1, the radius would continue to be 101 mm.
N300 D1 G41 X350 Y–20	
N310	

The same applies in the event that the resulting tool length is modified due to a change in the mirroring status of an axis. The tool must be selected again after the mirror command, in order to activate the modified tool length components.

c) Tool length and plane change (SD 42940: TOOL_LENGTH_CONST)

Plane change

The assignment between the tool length components (length, wear and tool base dimension) and the geometry axes is not modified when the machining plane is changed (G17–G19).

© Siemens AG, 2003. All rights reserved

Assignment of tools

The assignment of tool length components to geometry axes for turning and grinding tools (tool types 400 to 599) is generated from the value of setting data SD 42940 in accordance with the following table:

Plane	Length 1	Length 2	Length 3
17	Y	X	Z
18*)	X	Z	Y
19	Z	Y	X
–17	X	Y	Z
–18	Z	X	Y
–19	Y	Z	X

*) Each value not equal to 0 which is not equal to one of the six listed values is evaluated as the value 18.

The following table shows the assignment between the tool length components and the geometry axes for all other tools (tool types < 400 and > 599):

Plane	Length 1	Length 2	Length 3
17***)**	Z	Y	X
18	Y	X	Z
19	X	Z	Y
–17	Z	X	Y
–18	Y	Z	X
–19	X	Y	Z

*) Each value not equal to 0 which is not equal to one of the six listed values is evaluated as the value 17.

Note

For representation in tables, it is assumed that geometry axes 1 to 3 are named X, Y, Z. The axis order and not the axis identifier determines the assignment between an offset and an axis.

Three tool length components can be arranged on the 6 different types above.

d) Tool type (SD 42950: TOOL_LENGTH_TYPE)

Definition of the assignment between tool length components (length, wear and tool base dimension) and geometry axes independent of tool type.

Setting data **not equal to** zero: (the default definition is applied)
A distinction is made between turning and grinding tools (tool types 400 to 599) and other tools (milling tools).

The value range is from 0 to 2. Any other value is interpreted as 0. The assignment of tool length components is always independent of the actual tool type:

Value 1: As for milling tools.

Value 2: As for turning tools.

© Siemens AG, 2003. All rights reserved

Orientational toolholder

Setting data SD 42900 – SD 42950 (SW 5.1 and higher)

Setting data SD 42900 – SD 42950 have no effect on the components of an active orientational toolholder. However, the calculation with an orientational toolholder always allows for a tool with its total resulting length (tool length + wear + tool base dimension). The calculation of the resulting total length allows for all modifications caused by the setting data.

Note

When orientational toolholders are used, it is frequently practical to define all tools for a non-mirrored basic system, even those which are only used for mirrored machining. When machining with mirrored axes, the toolholder is then rotated such that the actual position of the tool is described correctly. All tool length components then automatically act in the correct direction, dispensing with the need for control of individual component evaluation via setting data, depending on the mirroring status of individual axes.

It can also be practical to use the orientational toolholder functionality if the machine provides no physical means of rotating tools although tools with different orientations are permanently installed. The standard tool dimensions can then be specified in a basic orientation and the dimensions relevant for machining can be achieved by rotating a virtual toolholder.

2.8.1 Tool lengths in the WCS under consideration of the (SW 5.2 and later)

Change tool or working plane

The values displayed for the tool correspond to the expansion in the WCS. If a toolholder with an inclined clamping position is to be used, you should make sure that the transformation used supports the toolholder. If this is not the case, incorrect tool dimensions will be displayed. When changing the working plane from G17 to G18 or G19, you should also ensure that the transformation can also be used for these planes. If the transformation is only available for G17 machining, the dimensions continue to be displayed for a tool in the Z direction after the plane change.

When transformation is deactivated, the basic tool is displayed in the x, y or z direction, according to the working plane. Allowance is made for a programmed toolholder. These tool dimensions are not altered when traversing without a transformation.

Note

Further information about handling the wear values in the WCS or MCS when tool orientation is active can be found in Subsection 2.8.3.

© Siemens AG, 2003. All rights reserved

2.8.2 Tool length offsets in tool direction (SW 6.1 and later)

Temperature compensation in real time

On 5-axis machines with moving tool, temperature fluctuations can occur in the machining heads. These can result directly in expansion fluctuations which are transmitted to the tool spindle in the form of linear expansion. A typical case on 5-axis heads, for example, is thermal expansion in the direction of the longitudinal spindle axis.

It is possible to compensate this thermal expansion even when the tool is orientated by assigning the temperature compensation values to the tool rather than to the machine axes. In this way, linear expansion fluctuations can be compensated even when the tool orientation changes.

Using the orientation transformation whose direction is determined by the current tool orientation, it is possible to overlay motions in real time and rotate them simultaneously. At the same time, the compensation values are adjusted continuously in the tool coordinate system.

In the current version, compensation applies to the “Temperature compensation” option and is active only when the axis to be compensated is really referenced.

Applicability

Temperature compensation in the tool direction is operative only with generic 5-axis transformations where

- transformation type 24 two axes rotate the tool
- transformation type 56 one axis rotates the tool,
  the other axis rotates the workpiece without temperature compensation

In generic 5-axis transformation with

- transformation type 40 rotary workpiece,
  the tool orientation is constant which means that the movement of the rotary axes on the machine does not affect the temperature compensation direction.

Temperature compensation in the tool direction also works in conjunction with the orientation transformations (not generic 5–axis transformations) with

- transformation types 64 to 69 rotary linear axis

Note

Temperature compensation can be activated with all other types of transformation. It is not affected by a change in the tool orientation. The axis move as if no orientation transformation with temperature compensation were active.

Limit values

The compensation values are limited to maximum settings by machine data

MD 20392: TOOL_TEMP_COMP_LIMIT[0] to
MD 20392 TOOL_TEMP_COMP_LIMIT[2]

limited to maximum values. The limit value default setting is 1mm. If a temperature compensation value higher than this limit is specified, it will be limited without an alarm.

© Siemens AG, 2003. All rights reserved

SD 42960

The three temperature compensation values together form a compensation vector and are contained in setting data

SD 42960: TOOL_TEMP_COMP[0] to SD 42960: TOOL_TEMP_COMP[2].

The setting data are userdefined, e.g. using synchronized actions or from the PLC. The compensation values can therefore be applied for other compensation purposes.

All three compensation values act in the direction of the three geometry axes (typically in the order X, Y Z) in their initial setting or when the orientation transformation is deactivated.

The assignment between the components and geometry axes is independent of the tool type (turning, grinding or milling tools) and the selected machining plane (G17 to G19). Changes to the setting data values take effect immediately.

Activation

Temperature compensation in the tool direction is an option which must enabled beforehand. It is activated by setting machine data

MD 20390: TOOL_TEMP_COMP_ON to a value **other than zero**.

In addition, bit 2 must be set for each of the relevant channel axes in machine data

MD 32750: TEMP_COMP_TYPE[<axis index>].

This can be more than three axes in cases where more than three channel axes in succession can be temporarily assigned to geometry axes as a result of geometry axis replacement of transformation switchover. If this bit is not set for a particular channel axis, the compensation value cannot be applied in the axis. This does not have any affect on other axes. In this case, an alarm is not output.

Orientational toolholder

If an orientational toolholder is active, the temperature compensation vector is rotated simultaneously to any change in orientation. This applies independently of any active orientation transformation.

If an orientational toolholder is active in conjunction with a generic 5-axis transformation or a transformation with rotating linear axis, the temperature compensation vector is subjected to both rotations.

Note

While transformations with rotating linear axes take changes in the tool vector (length) into account, they **ignore** its change in orientation which can be effected by an orientational toolholder.

Temperature compensation values immediately follow any applied change in orientation. This applies in particular when an orientation transformation is activated or deactivated.

The same is true when the assignment between geometry axes and channel axes is changed. The temperature compensation value for an axis is reduced to zero (interpolatively), for example, when it ceases to be a geometry axis after a transformation change. Conversely, any temperature compensation value for an axis which changes over to geometry axis status is applied immediately.

© Siemens AG, 2003. All rights reserved

Examples

Temperature compensation in tool direction

Example of a 5-axis machine with rotating tool on which the tool can be rotated about the C and B axes.

In its initial setting, the tool is parallel to the Z axis. If the B axis is rotated through 90 degrees, the tool points in the X direction. For this reason, a temperature compensation value programmed in SD 43960: TOOL_TEMP_COMP[2] also points in the direction of the machine X axis when a transformation is active.

If the transformation is deactivated with the tool in this direction, the tool orientation is, by definition, parallel again to the Z axis and thus different to its actual orientation. The temperature offset in the X axis direction is therefore reduced to zero and reapplied simultaneously in the Z direction.

Example of a 5-axis machine with rotating tool (transformation type 24). The relevant machine data are listed below:

The first rotary axis rotates about Z	C axis
The second rotary axis rotates about Y	B axis

The relevant machine data are listed below:

```
MD 20390: TOOL_TEMP_COMP_ON = TRUE          ; Temperature compen. active
Option                                      ; Enable option
MD 32750: TEMP_COMP_TYPE[ AX1 ] = 4         ; Offset in tool direction
MD 32750: TEMP_COMP_TYPE[ AX2 ] = 4         ; Offset in tool direction
MD 32750: TEMP_COMP_TYPE[ AX3 ] = 4         ; Offset in tool direction

                                            ; Assignment:

MD 24100: TRAFO_TYPE_1 = 24                 ; Transf. type 24 in 1st channel
MD 24110: TRAFO_AXES_IN_1[0] = 1            ; 1st axis in transformation
MD 24110: TRAFO_AXES_IN_1[1] = 2            ; 2nd axis in transformation
MD 24110: TRAFO_AXES_IN_1[2] = 3            ; 3rd axis in transformation
MD 24110: TRAFO_AXES_IN_1[3] = 5            ; 5th axis in transformation
MD 24110: TRAFO_AXES_IN_1[4] = 4            ; 4th axis in transformation

MD 24120: TRAFO_GEOAX_ASSIGN_TAB_1[0] = 1   ; Geo axis to channel axis 1
MD 24120: TRAFO_GEOAX_ASSIGN_TAB_1[1] = 2   ; Geo axis to channel axis 2
MD 24120: TRAFO_GEOAX_ASSIGN_TAB_1[2] = 3   ; Geo axis to channel axis 3

MD 24570: TRAFO5_AXIS1_1[0] = 0.0           ;
MD 24570: TRAFO5_AXIS1_1[1] = 0.0           ; Direction
MD 24570: TRAFO5_AXIS1_1[2] = 1.0           ; 1st rotary axis is parallel to Z

MD 24572: TRAFO5_AXIS1_2[0] = 0.0           ; Direction
MD 24572: TRAFO5_AXIS1_2[1] = 1.0           ; 2nd rotary axis is parallel to Y
MD 24572: TRAFO5_AXIS1_2[2] = 0.0           ;

MD 25574: TRAFO5_BASE_ORIENT_1[0] = 0.0     ;
MD 25574: TRAFO5_BASE_ORIENT_1[1] = 0.0     ; Basic tool orientation
MD 25574: TRAFO5_BASE_ORIENT_1[2] = 1.0     ; in Z direction
```

Supplementary conditions

The “temperature compensation” in the tool direction

- is an option which must be enabled for the user concerned.
- available for the generic 5-axis transformation.
- available for transformations with rotating linear axis for transformation types 64 to 69.

© Siemens AG, 2003. All rights reserved

NC program

Temperature compensation values in the NC program.

The compensation values assigned to axes X and Z are not zero and are applied for temperature compensation with respect to tool length. The machine axis positions reached in each case are specified as comments in the program lines.

```
SD 42960: TOOL_TEMP_COMP[0] = –0.3          : 1st compensation value
SD 42960: TOOL_TEMP_COMP[1] = 0.0           ;
SD 42960: TOOL_TEMP_COMP[2] = –1.0          ; 2nd compensation value

                                            ; Setpoint positions
                                            ; of machine axes
N10 g74 x0 y0 z0 a0 b0                      ;  X      Y      Z
N20 x20 y20 z20 f10000                      ; 20.30  20.00  21.00
N30 traori()                                ; 20.30  20.00  21.00
N40 x10 y10 z10 b90                         ; 11.00  10.00   9.70
N50 trafoof                                 ; 10.30  10.00  11.00
N60 x0 y0 z0 b0 c0                          ;  0.30   0.00   1.00
N70 m30
```

With the exception of block N40, the temperature compensation always acts in the original directions as the tool is pointing in the basic orientation direction. This applies particularly in block N50. The tool is actually still pointing in the direction of the X axis because the B axis is still at 90 degrees. However, because the transformation is already deactivated, the applied orientation is parallel to the Z axis again.

```
MD 20390: TOOL_TEMP_COMP_ON = TRUE          ; Temperature compen. active
Option                                      ; Enable option
MD 32750: TEMP_COMP_TYPE[ AX1 ] = 4         ; Offset in tool direction
MD 32750: TEMP_COMP_TYPE[ AX2 ] = 4         ; Offset in tool direction
MD 32750: TEMP_COMP_TYPE[ AX3 ] = 4         ; Offset in tool direction
```

Note

For further information about “Temperature compensation”, please see:
References: /FB/, K3, “Compensation” Section 2.1

For information about “Generic 5-Axis Transformation”, please see:
References: /FB/, F2, “Generic 5-axis transformation” Section 2.6

© Siemens AG, 2003. All rights reserved

2.8.3 Extension of the tool length determination (SW 6.1 and higher)

Taking into account the compensation values location and workpiecespecifically

Composition of the effective tool length

The tool length effective in an NC program without active kinematic transformation consists of a maximum of 8 vectors with three tool parameters each as follows:

1. Tool length (geometry)	($TC_DP3 – $TC_DP5)
2. Wear	($TC_DP12 – $TC_DP14)
3. Base dimension*	($TC_DP21 – $TC_DP23)
4. Adapter dimension*	($TC_ADPT1 – $TC_ADPT3)
5. Sum offsets fine	($TC_SCPx3 – $TC_SCPx5)
6. Sum offsets coarse or settingup offsets	($TC_ECPx3 – $TC_ECPx5)
7. Offset vector $\mathbf{l}_1$ of the orientational toolholder	($TC_CARR1 – $TC_CARR3)
8. Offset vector $\mathbf{l}_2$ of the orientational toolholder	($TC_CARR4 – $TC_CARR6)
9. Offset vector $\mathbf{l}_3$ of the orientational toolholder	($TC_CARR15 – $TC_CARR17)

Note

Out of the 9 values specified above base dimension* and adapter dimension* can **only alternatively** be active.

Type of action of the individual vectors

The type of action of the individual vectors or groups of vectors depends on the following further quantities:

Influencing quantity	Scope
G codes	Active processing level
Tool type	Milling tool or turning/grinding tools
Machine data	Tool management active / not active, orientational toolholder exists / does not exist.
Setting data	Behavior of tool length components when mirroring or when changing the plane
Orientational toolholder	Set values of the orientational toolholder
Adapter transformations	Transformed tool compensation values

© Siemens AG, 2003. All rights reserved

Distribution over the geometry axis components

How the three vector components of partial totals of the vectors involved are distributed over the three geometry axis components, is determined by the following quantities:

Influencing quantity	Dependencies
Active processing level: G17 X/Y direction G18 Z/X direction G19 Y/Z direction	Infeed direction: Z Y X
Tool type: Milling tools, drilling tools, grinding tools, turning tools	See Subsection 2.3.1 “Tool type” Table 2-2 Minimum number of required tool parameters.
SD 42900: MIRROW_TOOL_LENGTH SD 42910: MIRROW_TOOL_WEAR SD 42920: WEAR_SIGN_CUTPOS	See Section 2.8 “Tool offset special features” and Section 4.3 “Setting data”
Adapter transformations	See “FB tool management” Section 3.9

The resulting tool orientation always remains parallel to one of the three axis directions X, Y or Z and exclusively depends on the active machining plane G17–G19, since it has not yet been possible to date to assign the tool an orientation.

Stepless variation with tool orientation

In addition to further offsets or changes in the length, the orientational toolholder offers the option of

varying the tool orientation steplessly using the offset vectors $\mathbf{I}_1 - \mathbf{I}_3$.

For further explanations, please refer to Section 2.5 “Orientational toolholders”.

Minor operator compensations

In the typical case of operation, except for the compensations, all tool parameters are already determined in the production planning phase and are no longer changed by the machine operator.

Minor compensations, however, must also be modified during the normal production mode. Possible reasons are, e.g.

- tool wear, clamping errors or sensitivity of the machine to temperature.

These compensations are defined as follows:

Definition	Wear components
The total of wear	$TC_DP12 – $TC_DP14,
Sum offsets fine	$TC_SCPx3 – $TC_SCPx5,
Sum offsets coarse or setting-up offsets	$TC_ECPx3 – $TC_ECPx5

The machine should enter offsets which affect the tool length calculation in the coordinates used for measurement.

These workpiece offsets can be set more easily in setting data
SD 42935: WEAR_TRANSFORM and G code group 56 with the three values TOWSTD, TOWMCS and TOWWCS.

© Siemens AG, 2003. All rights reserved

SD 42935

Setting data SD 42935: WEAR_TRANSFORM defines which of the three wear components

($TC_DP12 – $TC_DP14),	wear
($TC_SCPx3 – $TC_SCPx5),	additive offset fine
($TC_ECPx3 – $TC_ECPx5)	additive offset coarse

of a rotation should be subjected to an adapter transformation or an orientational toolholder if one of the G codes TOWMCS or TOWWCS is active. Each of the three wear components can be set separately. You can control whether

- only the adapter transformation,
- only the transformation by way of the orientational toolholder or
- both transformations

are to be suppressed.

Top half of the diagram:
None of the three wear components suppresses the transformation

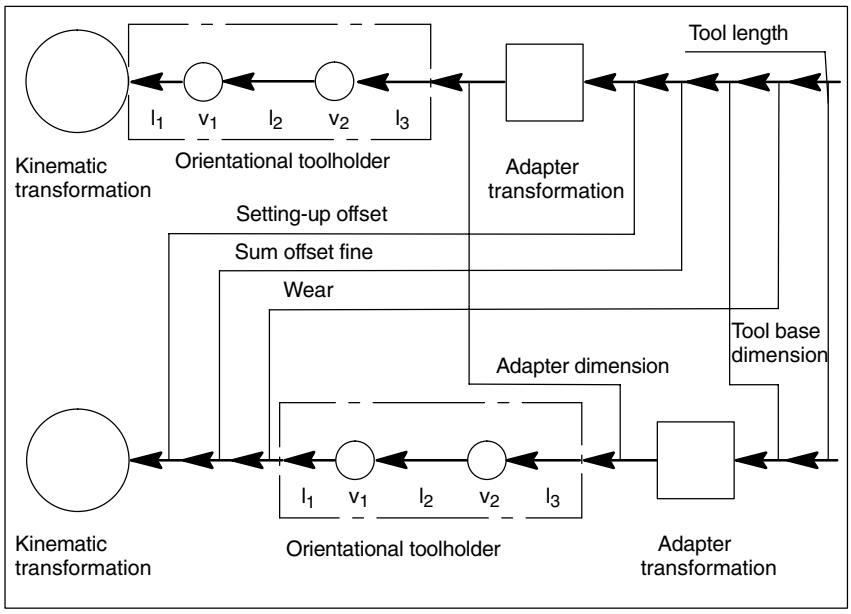


Fig. 2-46 Composition of the tool length

Bottom half of the diagram:
All three wear components suppress the transformation

Components which modify the tool orientation

- are displayed as circles. These are
  rotary axis v_1 and v_2 of the orientational toolholder and
  kinematic transformation.

Squares are used to represent

- the adapter transformation which modifies only length assignments,
  but not the orientation.

The transformed components each affect the sum of all length components represented to the right of this.

© Siemens AG, 2003. All rights reserved

Programming

The G code group 56 can be used to define the following three values:

Syntax	Offsets
TOWSTD	Initial setting value for offsets in the tool length
TOWMCS	Wear data in the machine coordinate system (MCS)
TOWWCS	Wear data in the workpiece coordinate system (WCS)

Functionality of the individual wear values

TOWSTD

Initial setting value (default behavior):

- The wear values are added to the other tool length components.

The resulting total tool length is then used in further calculations.

In the case of an active orientational toolholder:

- The wear values are subjected to the appropriate rotation.

TOWMCS

Wear data in the machine coordinate system (MCS):
In the case of an active rotation by an orientational toolholder: The toolholder rotates only the vector of the resulting tool length

- without allowing for the wear.

Then the tool length vector rotated in this way and the wear are added. The wear is not subjected to the rotation.

If no orientational toolholder is active or this does not result in a rotation, TOWMCS and TOWSTD are identical.

TOWWCS

Wear data in the workpiece coordinate system (WCS):
In the case of an active orientational toolholder, the tool vector

- is calculated as with TOWMCS without allowing for the wear.

The parameters

- of the wear are interpreted in the workpiece coordinate system.

The wear vector in the workpiece coordinate system is converted to the machine coordinate system and added to the tool vector.

Changing the G code

Changing the G code of the TOWSTD, TOWMCS, TOWWCS group does not affect a tool that is already active and does not apply until the next tool is selected. A new G code of this group will also come into effect if it is programmed in the same block in which a tool is selected.

© Siemens AG, 2003. All rights reserved

Evaluation of individual wear components

The evaluation of individual wear components (assignment to geometry axes, sign evaluation) is affected by

- the active plane,
- the adapter transformation and
- the five setting data shown in the table below:

Setting data	Wear components		
SD 42910: MIRROW_TOOL_WEAR	TOWSTD	TOWMCS	TOWWCS
SD 42920: WEAR_SIGN_CUTPOS	X	X	—
SD 42930: WEAR_SIGN	X	—	—
SD 42940: TOOL_LENGTH_CONST	X	X	X
SD 42950: TOOL_LENGTH_TYPE	X	X	X

Note

Wear components which are subjected to an active rotation by an adapter transformation or an orientational toolholder are referred to as nontransformed wear components.

Special points to be noted

If TOWMCS or TOWWCS is active,

- SD 42920: WEAR_SIGN_CUTPOS

has no effect on the non-transformed wear components.

In addition, with TOWWCS

- setting data SD 42910: MIRROR_TOOL_WEAR

has no effect on the non-transformed wear components because, in this case, an active mirror operation is already contained in the frame used to evaluate the wear components.

On a plane change, the assignment between the non-transformed wear components and the geometry axes is retained, i.e. these are not interchanged as with other length components. The assignment of components depends on the active plane for tool selection.

Examples

Let's assume a milling tool is used where only the wear value $TC_DP12 assigned to length L1 is not equal to zero. If G17 is active,

- this length is effective in the direction of the Z axis.

If TOWMCS or TOWWCS active and bit 1 is set in SD 42935: WEAR_TRANSFORM,

- this dimension is also effective in the Z direction in the event of a plane change after tool selection.

If G18 is active on tool selection, e.g., the component is always effective in the Y direction instead.

© Siemens AG, 2003. All rights reserved

2.9 Sum and setup offsets (SW 5 and higher)

2.9.1 General

Sum offsets can be treated as **programmable process offsets** during machining and are composed of all the error sizes (including the wear) which cause the workpiece to deviate from the specified dimensions.

Sum offsets are a generalized form of wear.
They are a component of the cutting edge data. The parameters of the sum offset refer to the geometrical data of a cutting edge.

The offset data of a sum offset are addressed by a **DL** number (**DL:** Location-dependent; offsets with reference to the location of use).

In contrast, the wear values of a D number describe the physical wear of the cutting edge, i.e. in special situations, the sum offset can match the wear of the cutting edge.

Sum offsets are intended for general use, i.e. with active or inactive tool management or with the flat D number function.

Machine data are used to classify the sum offsets into:

- **Sum offset fine**
- **Sum offset coarse (setup offset)**.

The setup offset is the offset to be entered by the setup engineer before machining. These values are stored separately in the NCK. The operator subsequently only has access to the 'sum offsets fine' via MMC.
The 'sum offset fine' and 'sum offset coarse' are added internally in the NCK. This value is referred to in the following as sum offset.

Note

The function is enabled by setting MD 18080 MM_TOOL_MANAGEMENT_MASK, Bit 8=1.

If kinematic transformations (e.g. 5-axis transformations) are active, the tool length is calculated first after allowing for the various wear components. The total tool length is then used in the transformation. Unlike the case of an orientational toolholder, the wear values are thus always included in the transformation irrespective of the G code of group 56.

© Siemens AG, 2003. All rights reserved

2.9.2 Description of function

Several sum offsets (DL numbers) can be defined per D number. This allows you to determine, e.g. **workpiece location-dependent** offset values and assign them to a cutting edge. Sum offsets have the same effect as wear, i.e. they are added to the offset values of the D number.

The data are permanently assigned to a D number.

You can define the following settings in machine data:

- Activate sum offset
- Define maximum quantity of DL data sets to be created in NCK memory
- Define maximum quantity of DL numbers to be assigned to a D number
- Define whether the sum offsets (fine/coarse) are to be saved during a data backup
- Define which sum offsets are to be activated when:
  – A new cutting edge offset is activated
  – An operator panel RESET is performed
  – An operator panel START is performed, or
  – The end of program has been reached.

The name is oriented to the logic of the corresponding machine data for tools and cutting edges.
The setup offset and sum offset 'fine' can be read and written via system variables and OPI services.

Note

When tool management is active, a machine data can be used to define whether the sum offset of a tool activated during a programmed tool change remains unchanged or is set to zero.

Summary of offset parameters $TC_DPx

The following general system variables were previously defined for describing a cutting edge:

$TC_DP1	Tool type
$TC_DP2	Cutting edge position

Parameters for geometry and wear

The offsets of the tool geometry are assigned to system variables $TC_DP3 to $TC_DP11. System variables $TC_DP12 to $TC_DP20 allow you to name a wear for each of these parameters.

© Siemens AG, 2003. All rights reserved

Geometry	Wear	Length offsets
$TC_DP3	$TC_DP12	Length 1
$TC_DP4	$TC_DP13	Length 2
$TC_DP5	$TC_DP14	Length 3
Geometry	**Wear**	**Radius compensation**
$TC_DP6	$TC_DP15	Radius
$TC_DP7	$TC_DP16	Corner radius (tool type 700; slotting saw)
Geometry	**Wear**	**Further offsets**
$TC_DP8	$TC_DP17	Length 4 (tool type 700; slotting saw)
$TC_DP9	$TC_DP18	Length 5
$TC_DP10	$TC_DP19	Angle 1 (angle between face of tool and torus surface)
$TC_DP11	$TC_DP20	Angle 2 (angle between tool longitudinal axis and upper end of torus surface)

Base dimension/ adapter dimension

$TC_DP21	Adapter length 1
$TC_DP22	Adapter length 2
$TC_DP23	Adapter length 3

Technology

$TC_DP24	Clearance angle 1.) The clearance angle is stored here for ManualTurn; tool type 5xx. Same meaning as in standard cycles for turning tools. 2.) The tip angle of the drill is stored here for ShopMill; tool type 2xx. 3.) Used in standard cycles for turning tools; tool type 5xx. This is the angle at the secondary cutting edge for these tools.
$TC_DP25	1.) The value for the cutting rate is stored here for ManualTurn. 2.) A bitcoded value for various states of tool types 1xx and 2xx is stored here for ShopMill.

Parameters of the sum and setup offsets ($TC_SCPxy, $TC_ECPxy)

The numbering of the parameters is oriented to the numbering of system variables $TC_DP3 to $TC_DP11.

The effect of the parameters is similar to the wear (additive to the tool geometry). Up to six sum/setup parameters can be defined per cutting edge parameter.

© Siemens AG, 2003. All rights reserved

Tool geometry parameter to which the offset is added.	Sum/setup parameters length offsets	Tool wear parameters
$TC_DP3	Length 1 $TC_SCP**1**3, $TC_SCP**2**3,$TC_SCP**3**3, $TC_SCP**4**3,$TC_SCP**5**3,$TC_SCP**6**3 $TC_ECP**1**3, $TC_ECP**2**3,$TC_ECP**3**3, $TC_ECP**4**3,$TC_ECP**5**3,$TC_ECP**6**3 The numbers in bold type 1, 2, ... 6 denote paarameters of the maximum of six (location/independent) contours that are programmable with DL =1, ... 6 for the parameter in column one.	$TC_DP12
$TC_DP4	Length 2 $TC_SCP**1**4, $TC_SCP**2**4,$TC_SCP**3**4, $TC_SCP**4**4,$TC_SCP**5**4,$TC_SCP**6**4 $TC_ECP**1**4, $TC_ECP**2**4,$TC_ECP**3**4, $TC_ECP**4**4,$TC_ECP**5**4,$TC_ECP**6**4	$TC_DP13
$TC_DP5	Length 3 etc. ...	$TC_DP14
	Radius compensation	
$TC_DP6	Radius	$TC_DP15
$TC_DP7	Corner radius	$TC_DP16
	Further offsets	
$TC_DP8	Length 4	$TC_DP17
$TC_DP9	Length 5	$TC_DP18
$TC_DP10	Angle 1 ...etc.	$TC_DP19
$TC_DP11	Angle 2 $TC_SCP**2**1, $TC_SCP**3**1,$TC_SCP**4**1, $TC_SCP**5**1,$TC_SCP**6**1,$TC_SCP**7**1 $TC_ECP**2**1, $TC_ECP**3**1,$TC_ECP**4**1, $TC_ECP**5**1,$TC_ECP**6**1,$TC_ECP**7**1 The numbers in bold type 3, 2, ... 7 denote paarameters of the maximum of six (location-independent) contours that are programmable with DL =1, ... 6 for the parameter in column one.	$TC_DP20

Supplementary conditions

The maximum number of DL data sets of a cutting edge and the total number of sum offsets in the NCK are defined by machine data. The default value is zero; i.e. no sum offsets can be programmed.

© Siemens AG, 2003. All rights reserved

When the 'monitoring function' is activated, it is possible to monitor a tool for wear or for 'sum offset'.

The additional sum/setup data sets use additional buffered memory. 8 bytes are required per parameter. A sum offset data set requires 8 bytes * 9 parameters = 72 bytes. A setup data set requires an equal amount of memory. A certain number of bytes is also required for internal administration data.

2.9.3 Activation

The function must be activated in a machine data.

The system variable $TC_ECPx and $TC_SCPx and setup and sum offsets ('fine') defined via the OPI interface can be activated in the parts program.

This is done by programming the language command DL='nr'. When a new D number is activated, either a new DL is programmed or the DL number defined in machine data MD 20272: SUMCORR_DEFAULT is activated.

DL-Programming

The sum offset is always programmed relative to the active D number with the command:

DL = 'n'.

The sum offset 'n' is added to the wear of the active D number.

Note

If you use setup offset **and** sum offset 'fine', both offsets are combined and added to the tool wear.

The sum offset is deselected with the command

DL = 0.

Note

DL0 is not allowed. When the offset is deselected (D0 and T0), the sum offset is also deactivated.

If you program a sum offset which does not exist, an alarm is output, as in the case of a nonexistent D offset.

The defined wear is then the only component of the offset (defined in system variables $TC_DP12 to $TC_DP20).

© Siemens AG, 2003. All rights reserved

!

Important

If a sum offset is programmed when a D offset is active (also applies to deselection), this has the same effect on the path as when you program a D command. An active radius compensation thus loses its reference to adjacent blocks.

Configuration

MM_KIND_OF_SUMCORR, bit 4=0: Standard setting:

Only **one** set of sum offsets exists per DL number.
We refer in general to the sum offset.
This refers to the data represented by $TC_SCPx.

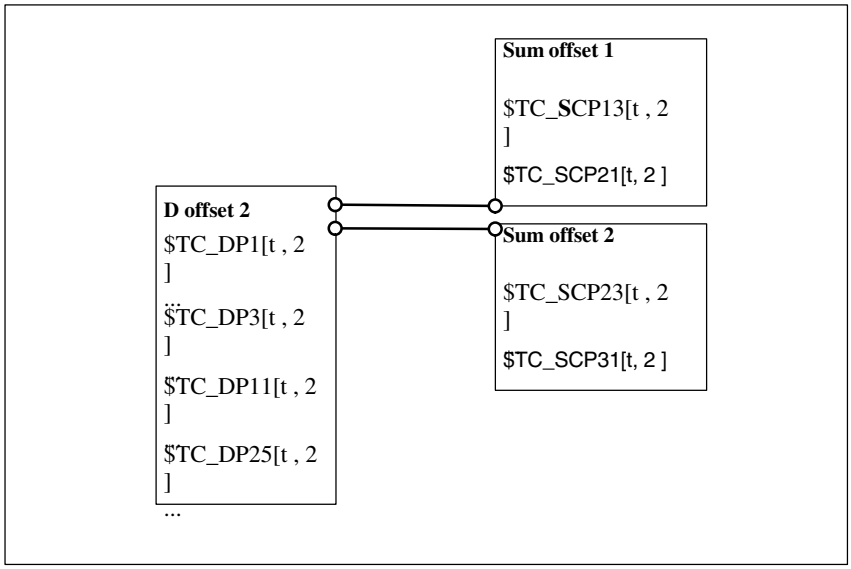


Fig. 2-47 MD 18112 MM_KIND_OF_SUMCORR, bit 4 = 0

Tool T = t is active. With the data in the figure, the following is programmed:

D2 ; Cutting edge offsets; i.e. $TC_DP3,... $TC_DP11 + wear ($TC_DP12,...$TC_DP20) + adapter dimension

...

DL=1 ; Sum offset 1 is added in addition to the previous offsets of D2, i.e. $TC_SCP**1**3,...$TC_SCP21

...

DL=2 ; Sum offset 2 is added to offset D2 instead of sum offset 1, i.e. $TC_SCP**2**3,...$TC_SCP31

...

DL=0 ; Deselect sum offset;
Only the data of D2 are now effective.

© Siemens AG, 2003. All rights reserved

MM_KIND_OF_SUMCORR, bit 4=1:
Setup offsets are available.

The sum offset is now composed of the sum offset 'fine' (represented by $TC_SCPx) and the setup offset (represented by $TC_ECPx). Two data sets therefore exist for one DL number. The sum offset is calculated by adding the corresponding components ($TC_ECPx + $TC_SCPx).

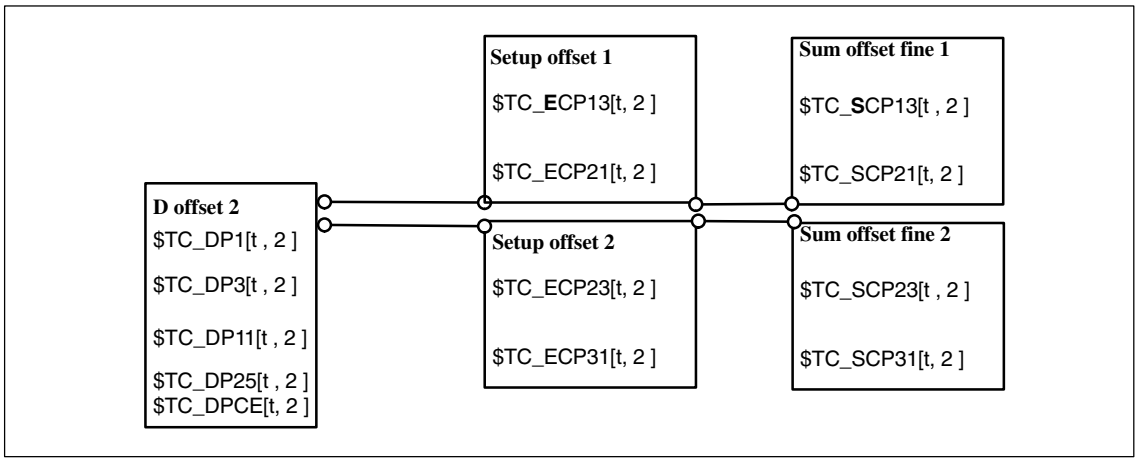


Fig. 2-48 MD 18112 MM_KIND_OF_SUMCORR, Bit 4 = 1 setup offsets + sum offsets 'fine

Tool T = t is active. With the data in the figure, the following is programmed:

D2 ; Cutting edge offsets; i.e. $TC_DP3,... $TC_DP11 + wear ($TC_DP12,...$TC_DP20) + adapter dimension

...

DL=1 ; Sum offset 1 is added in addition to the previous offsets of D2; i.e. $TC_ECP**1**3 + $TC_SCP**1**3,...$TC_ECP21 + $TC_SCP21

...

DL=2 ; Sum offset 2 is added to offset D2 instead of sum offset 1; i.e. $TC_ECP**2**3 + $TC_SCP**2**3,...$TC_ECP31 + $TC_SCP31

...

DL=0 ; Deselect sum offset;
Only the data of D2 are now effective.

© Siemens AG, 2003. All rights reserved

Reading/writing in the parts program

The individual sets of sum offset parameters are differentiated according to the number ranges of system variable $TC_SCP.

The meaning of the individual variables is similar to geometry variables $TC_DP3 to $TC_DP11. Only length1, length2 and length3 are enabled for the basic functionality (variables $TC_SCP13 – $TC_SCP15 for the first sum offset of the cutting edge).

R5 = **$TC_SCP13**[t, d] ; Sets the value of the R parameter to the value of the **first** component of **sum offset 1** of the cutting edge (d) of the tool (t).

R6 = **$TC_SCP21**[t, d] ; Sets the value of the R parameter to the value of the **last** component of **sum offset 1** of the cutting edge (d) of the tool (t).

R50 = **$TC_SCP23**[t, d] ; Sets the value of the R parameter to the value of the **first** component of **sum offset 2** of cutting edge (d) of the tool (t).

$TC_SCP43[t, d] = 1.234 ; Sets the value of the **first** component of **sum offset 4** of the cutting edge (d) of the tool (t) to the value 1.234.

The above statements also apply to the setup offsets (if the NCK is configured with this option), i.e.

R5 = **$TC_ECP13**[t, d] ; Sets the value of the R parameter to the value of the **first** component of **setup offset 1** of the cutting edge (d) of the tool (t).

R6 = **$TC_ECP21**[t, d] ; Sets the value of the R parameter to the value of the **last** component of **setup offset 1** of the cutting edge (d) of the tool (t).

etc.

When working with setup offsets, the sum offsets 'fine' are written with the $TC_SCPx system variables.

Creating a new sum offset

If the offset data set (x) does not yet exist, it is created on the first write operation to one of its parameters (y).

$TC_SCPxy[t, d] = r.r ; Parameter y of sum offset x is assigned the value 'r.r.' The other parameters of x are zero.

When working with setup offsets, the sum offsets 'fine' are written with the $TC_SCPx system variables.

© Siemens AG, 2003. All rights reserved

Note

When working with setup offsets, the data set for the setup offset is created when a data set is created for sum offset 'fine' if a data set did not already exist for [t, d].

Creating a new setup offset

If the offset data set (x) does not yet exist, it is created on the first write operation to one of its parameters (y).

$TC_ECPxy[t, d] = r.r ; Parameter y of setup offset x is assigned the value 'r.r.' The other parameters of x are zero.

Note

When working with setup offsets, the data set for the setup offset is created when a data set is created for sum offset 'fine' if a data set did not already exist for [t, d].

DELDL – Delete sum offset

Sum offsets are generally only relevant when machining with a cutting edge at a certain time at a certain location of the workpiece. You can use the NC language command **DELDL** to delete the sum offsets from cutting edges (in order to release memory).

status = DELDL(t, d) ; Deletes all sum offsets of cutting edge d of tool t; t, d are optional parameters:

If d is not specified, all sum offsets of all cutting edges of tool t are deleted.
If d and t are not specified, all sum offsets of the cutting edges of all tools of the TO units are deleted (for the channel in which the command is programmed).

When working with setup offsets, the DELDL command deletes both the setup offset and the sum offsets 'fine' of the specified cutting edge(s).

Note

The memory used for the data sets is released after the deletion.
The deleted sum offsets can subsequently no longer be activated or programmed.

Sum offsets and setup offsets of active tools cannot be deleted (similar to the deletion of D offsets or tool data).

© Siemens AG, 2003. All rights reserved

The 'status' return value indicates the result of the deletion command:

0	Deletion was successful
–1	Deletion was not (one cutting edge) or not completely (several cutting edges) successful.

Data backup

The data are saved during a general tool data backup (as a component of the D number data sets).

It is advisable to save the sum offsets, in order to allow the current status to be restored in the event of an acute problem. The sum offsets can be excluded from a data backup by machine data (can be set separately for setup offsets and sum offsets 'fine').

Note

Sum offsets behave in the same way as D offsets with reference to block search and REPOS. The behavior on Reset and PowerOn can be defined by machine data.

If machine data MD 20110: RESET_MODE_MASK defines that the last active tool offset number (D) is to be activated after PowerOn, the last active DL number is then no longer active.

© Siemens AG, 2003. All rights reserved

2.9.4 Examples

Example 1

Machine data are to be used to define that no offsets and no sum offsets are active on a tool change ($MC_CUTTING_EDGE_DEFAULT=0, $MC_SUMCORR_DEFAULT=0):

```
T5 M06          ; Tool number 5 is loaded – no offset active
D1 DL=3         ; Offset D1 + sum offset 3 of D1 are activated
X10
DL=2            ; Offset D1 + sum offset 2 are activated
X20
DL=0            ; Sum offset deselection, only offset D1 is now active
D2              ; Offset D2 is activated – the sum offset
                ; is not included in the offset
X1
DL=1            ; Offset D2 + sum offset 1 are activated
X2
D0              ; Deselect offset
X3
DL=2            ; No effect – DL2 of D0 is zero (same as
                ; programming T0 D2)
```

Example 2

Machine data are to be used to define that offset D2 and sum offset DL=1 are activated on a tool change ($MC_CUTTING_EDGE_DEFAULT=2, $MC_SUMCORR_DEFAULT=1):

```
T5 M06          ; Tool number 5 is loaded – D2 + DL=1
                ; are active (= values of MDs)
D1 DL=3         ; Offset D1 + sum offset 3 of D1 are activated
X10
DL=2            ; Offset D1 + sum offset 2 are activated
X20
DL=0            ; Sum offset deselection, only offset D1 is now active
D2              ; Offset D2 is activated – sum offset DL=1 is
                ; activated
X1
DL=2            ; Offset D2 + sum offset 2 are activated
D1              ; Offset D1 + sum offset 1 are activated
```

■

© Siemens AG, 2003. All rights reserved

3 Supplementary Conditions

3.1 Flat D number structure (SW 4.1 and higher)

Grinding tools

Grinding tools (tool types 400–499) cannot be defined using the simple tool management structure (flat D numbers).

Power ON, block search

T number output to PLC triggers a synchronization process in the NCK: With absolute, indirect D programming, the PLC returns the D values via VDI. The NCK waits until the output of a T number is followed by a response from the PLC: “I have written the D number”. With block search without calculation, this process of synchronization must be deactivated until the first valid T number has been output again.

That means that the NCK must not wait on D programming.

Note

MD $MC_AUXFU_AT_BLOCK_SEARCH_END can be used to control the timing of output of auxiliary functions to PLC at the end of a block search – automatically on End, or on NC Start.

REORG

The (only) writable variable $A_MONIFACT which is defined here is stored by a main run data. Since the write process takes place synchronously to the main run, no special measures are required for Reorg.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

4 Data Descriptions (MD, SD)

4.1 General machine data

18088 MD number	**MM_NUM_TOOL_CARRIER**		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 99999999	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	The maximum number of toolholders that can be defined for orientational tools in the TO range. The value is divided by the number of active TO units. The whole number indicates how many toolholders can be defined per TO unit. The data for definition of a toolholder are set using the system variables $TC_CARR1, ... $TC_CARR17.		
Application example(s)	2 channels are active and there is one TO unit on each channel (=preset). 3 carriers are to be defined in channel 1, one in channel 2. The value to be set is 6, since 6 / 2 = 3, i.e. maximum 3 carrier definitions in each TO unit.		

18102 MD number	**MM_TYPE_OF_CUTTING_EDGE** Activation of flat D number management		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 4		
Significance:	This MD activates the 'flat D number management' function. The individual values determine the type of D programming: direct or indirect programming. MD= 0 The standard value is zero. This means that the NCK manages the T and D numbers. 1 The value 1 means that D numbers are programmed direct and as absolute values. Activation (change value from 0 to > 0) or deactivation (change value from >0 to 0) causes a reconfiguration of the buffered memory; i.e. the data are deleted. A value >0 is only accepted by the NCK if bit 0 is not enabled in MD 18080: MM_TOOL_MANAGEMENT_MASK, i.e. the tool management function must not be active at the same time.		
Related to			

© Siemens AG, 2003. All rights reserved

18105 MD number	**MM_MAX_CUTTING_EDGE_NO** Maximum value of D number		
Default setting: 9	Minimum input limit: 1	Maximum input limit: 32000	
Changes effective after POWER ON	Protection level: 7/2	Unit: Number	
Data type: DWORD	Applies from SW 5.1		
Significance:	Definition of maximum D number value The maximum number of D numbers per tool is not affected (defined in MD 18106: MM_MAX_CUTTING_EDGE_PERTOOL). D number assignment monitoring is only effective on the definition of new D numbers. Existing data sets are not verified automatically if this machine data is changed. The machine data is not evaluated with the 'flat D number' function. When the relation between the maximum D number value and the maximum cutting edge number per tool is changed, the buffered memory requirements also change (both default values are identical).		
Related to	MD 18106: MM_MAX_CUTTING_EDGE_PERTOOL		

18106 MD number	**MM_MAX_CUTTING_EDGE_PERTOOL** Maximum number of cutting edges per tool		
Default setting: 9	Minimum input limit: 1	Maximum input limit: 12	
Changes effective after POWER ON	Protection level: 7/2	Unit: Number of	
Data type: DWORD	Applies from SW 5.1		
Significance:	Definition of the maximum number of cutting edges (D offsets) per tool (T number). This machine data provides a high level of security during data definition by allocating the maximum number of cutting edges actually available for a tool (e.g. for a tool with only one cutting edge). Up to 12 offsets can be allocated per tool. The machine data is not evaluated with the 'flat D number' function. When the relation between the maximum D number value and the maximum cutting edge number per tool is changed, the buffered memory requirements also change (both default values are identical).		
Related to	MD 18105: MM_MAX_CUTTING_EDGE_NO		

© Siemens AG, 2003. All rights reserved

18108 MD number	**MM_NUM_SUMCORR** Number of all sum offsets in NCK		
Default setting: –1	Minimum input limit: –1	Maximum input limit: *)	
Changes effective after POWER ON	Protection level: 7/2	Unit: Number of	
Data type: DWORD	Applies from SW 5.1		
Significance:	Value Meaning –1 The number of sum offsets is equal to the product of the number of cutting edges and the number of sum offsets per cutting edge. > 0 You can select a value less than the above product so that the system only reserves buffered memory for the actual number of sum offsets. If setup compensation is active, the memory required for a sum offset is doubled. *) The maximum value is calculated from the product of the upper limit of the 'number of cutting edges' and 'number of sum offsets per cutting edge' machine data.		
Related to	MD 18100: MM_NUM_CUTTING_EDGES_IN_TOA MD 18110: MM_MAX_SUMCORR_PERCUTTING_EDGE MD 18112: MM_KIND_OF_SUMCORR		

18110 MD number	**MM_MAX_SUMCORR_PER_CUTTEDGE** Maximum number of sum offsets per cutting edge		
Default setting: 1	Minimum input limit: 1	Maximum input limit: 6	
Changes effective after POWER ON	Protection level: 7/2	Unit: Number of	
Data type: DWORD	Applies from SW 5.1		
Significance:	Defines the maximum number of sum offsets per cutting edge The effect of this machine data depends on the value of MD 18108. MM_NUM_SUMCORR > 1 This data does not reserve memory and is used exclusively for monitoring purposes. MM_NUM_SUMCORR = –1 The machine data reserves memory.		
Related to	MD 18108: MM_NUM_SUMCORR MD 18100: MM_NUM_CUTTING_EDGES_IN_TOA		

© Siemens AG, 2003. All rights reserved

18112 MD number	MM_KIND_OF_SUMCORR Properties of sum offsets in the NCK		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0x1F	
Changes effective after POWER ON	Protection level: 7/2	Unit: Bit mask	
Data type: INT	Applies from SW 5.1		
Significance:	Defines the properties of sum offsets in the NCK: Bit no. Value Meaning 0 0 The sum offsets are saved with the tool data during data backup. 1 0x1 The sum offsets are not saved with the tool data during data backup. 1 0 The setup offsets are saved with the tool data during data backup. 1 0x2 The setup offsets are not saved with the tool data during data backup 2 0 If you use the 'tool management' function: The existing sum/setup offsets are not changed when you activate a tool. The existing sum/setup offsets are not changed when you activate a tool. 1 0x4 The existing sum offsets are set to zero when you activate a tool. The setup offsets are not changed. 3 0 If you use the 'tool management' plus 'adapter' function: A transformation of the sum offsets is performed. 1 0x8 There is no transformation of sum offsets. 4 0 There are no setup offset data sets. 1 0x10 The setup offset data sets are created. The sum offset is then calculated from the sum of the setup offset and sum offset 'fine' Changes to the state of bits 0, 1, 2, 3 do not affect the memory configuration. A change to bit 4 reconfigures the buffered memory on the next POWER ON.		
Related to	MD 18080: MM_TOOL_MANAGEMENT_MASK MD 18086: MM_NUM_MAGAZINE_LOCATION MD 18104: MM_NUM_TOOL_ADAPTER MD 20310: TOOL_MANAGEMENT_MASK		

18114 MD number	MM_ENABLE_TOOL_ORIENT Assign orientation to cutting edges		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2	
Changes effective after POWER ON	Protection level: 7/2	Unit: –	
Data type: DWORD	Applies from SW 5.3		
Significance:	Assigns an orientation that differs from the default to each tool cutting edge. Value 1: System variable $TC_DPV1[n, m] is assigned to each cutting edge. This parameter can be used to define one of 6 possible tool orientations in the positive or negative coordinate direction. Value 2: In addition to system variable $TC_DPV1[n, m], a further three system variables $TC_DPV2[n, m], $TC_DPV3[n, m] and $TC_DPV4[n, m] are assigned and can be used to define any define any tool orientation. The identifiers n and m stand for the T and D number. This machine data does not affect the amount of batterybacked memory required.		
Related to	–		

© Siemens AG, 2003. All rights reserved

4.2 Channelspecific machine data

20096 MD number	**T_M_ADDRESS_EXT_IS_SPINO** Address extension interpreted as spindle number		
Default setting: FALSE	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 7/2	Unit: Bool	
Data type: BOOLEAN	Applies from SW 5.1		
Significance:	Only relevant when 'tool management' function inactive: Value Meaning FALSE The contents of the address extensions of NC addresses T and M are not evaluated by the NCK. The PLC determines the meaning of the programmed extension TRUE The address extensions of NC addresses T and M are interpreted as a spindle number. The NCK handles the extension in the same way as when tool management is active. The programmed D number therefore always refers to the programmed main spindle number ('tool change command number'= TOOL_CHANGE_M_CODE; the default value is 6).		
Related to	MD 20090: SPIND_DEF_MASTER_SPIND MD 22550: TOOL_CHANGE_MODE MD 22560: TOOL_CHANGE_M_CODE		

20126 MD number	**TOOL_CARRIER_RESET_VALUE** Active toolholder on RESET		
Default setting: 0	Minimum input limit: 0	Maximum input limit: plus	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	To specify the toolholder that is used to the select the tool length compensation during power-up and reset or parts program end, depending on the MD: $MC_RESET_MODE_MASK and for parts program start according to MD: $MC_START_MODE_MASK. This data is only valid without tool management.		
References	FBW		

20132 MD number	**SUMCORR_RESET_VALUE** Number for selecting sum offset		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 6	
Changes effective after RESET	Protection level: 7/2	Unit: Number	
Data type: DWORD	Applies from SW 5.1		
Significance:	Defines the sum offset number used to select the sum offset on POWER ON and Reset or end of parts program (depending on RESET_MODE_MASK) and on parts program start (depending on START_MODE_MASK). The machine data MD 18108: MM_NUM_SUMCORR determines the maximum value for entry.		
Related to	MD 20110: RESET_MODE_MASK MD 20112: START_MODE_MASK MD 20130: CUTTING_EDGE_RESET_VALUE MD 18108: MM_NUM_SUMCORR		

© Siemens AG, 2003. All rights reserved

20180 MD number	**TOCARR_ROT_ANGLE_INCR[i]** Value of the minimum incremental step with orientational toolholder		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after NewConfig	Protection level: 3 / 7	Unit: degrees	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	With orientational toolholders this machine data provides the minimum incremental step with which the first or the second orientation axis can be changed (e.g. with Hirth tooth system). A programmed or calculated angle value is rounded up or down to the closest value calculated from $\varphi = s + n * d$ when n is an integer. s = $MC_TOCARR_ROT_ANGLE_OFFSET[i] and d = $MC_TOCARR_ROT_ANGLE_INCR[i] If this machine data is zero, no rounding takes place.		

20182 MD number	**TOCARR_ROT_ANGLE_OFFSET[i]** Offset of the rotary axis with orientational toolholder		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after NewConfig	Protection level: 3 / 7	Unit: degrees	
Data type: DOUBLE	Applies from SW 5.3		
Significance:	For orientational toolholders this machine data provides the offset of the rotary axis if its position cannot be changed continuously. It is only evaluated if $MC_TOCARR_ROT_ANGLE_INCR is not zero. For the detailed meaning of this machine data, please see the description of $MC_TOCARR_ROT_ANGLE_INCR.		

20184 MD number	**TOCARR_BASE_FRAME_NUMBER** Basis frame of the table offset for orientational toolholder with rotary table		
Default setting: –1	Minimum input limit: –1	Maximum input limit: FRAMES–1	
Changes effective after NewConfig	Protection level: 3 / 7	Unit: –	
Data type: DWORD	Applies from SW 5.3		
Significance:	This machine data the channel-specific basis frame to which the table offset of an orientational toolholder with rotary table is written. To avoid unintentional shifting of the coordinate system, it is urgently recommended not to use the specified basis frame additionally for other purposes. This machine data must refer to a valid basis frame. If its contents are less than zero or greater than or equal to the maximum number of basis frames set in $MC_MM_NUM_BASE_FRAMES, selection of a corresponding toolholder produces an alarm.		

20188 MD number	**TOCARR_BASE_FINE_LIM_LIN** Limit linear fine offset TCARR		
Default setting: 1,0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3 / 7	Unit: POSN_LIN	
Data type: DOUBLE	Applies from SW 6.4		
Significance:	This machine data defines the input limit for the **linear** fine offset values or an orientable toolholder for each channel.		
Related to	SD 42974: TOCARR_FINE_CORRECTION		

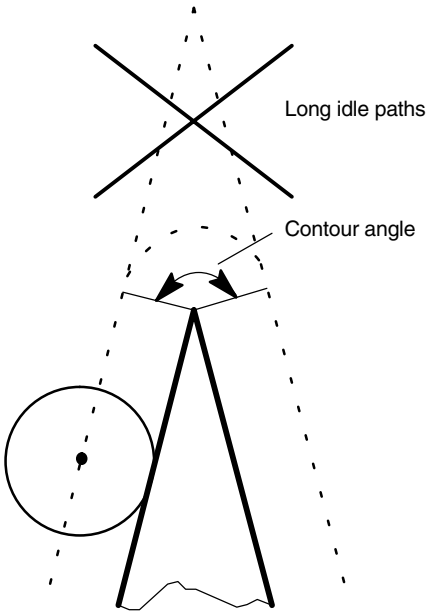
© Siemens AG, 2003. All rights reserved

20190 MD number	**TOCARR_BASE_FINE_LIM_ROT** Limit rotary fine offset TCARR		
Default setting: 1,0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 3 / 7	Unit: POSN_ROT	
Data type: DOUBLE	Applies from SW 6.4		
Significance:	This machine data defines the input limit for the **linear** fine offset values or an orientable toolholder for each channel.		
Related to	SD 42974: TOCARR_FINE_CORRECTION		

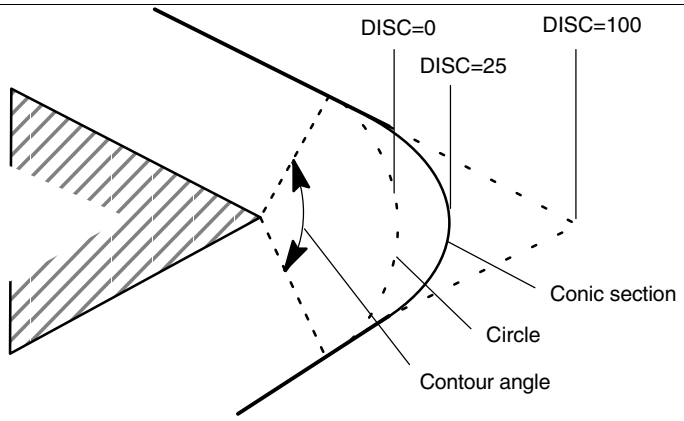
20202	**WAB_MAXNUM_DUMMY_BLOCKS**		
MD number	Maximum number of blocks with no traversing motions with SAR		
Default setting: 5, 5, 5, 5, 5, ...,,	Minimum input limit: 1	Maximum input limit: 10	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 4.3		
Significance:	Maximum number of blocks which can appear between the SAR block and the traversing block which determines the direction of the approach or retraction tangent.		

20204 MD number	**WAB_CLEARANCE_TOLERANCE** Defined tolerance for the point defined by DISCL for SAR		
Default setting: 0.001	Minimum input limit: 0	Maximum input limit:	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: mm	
Data type: REAL	Applies from SW 4.4		
Significance:	During soft approach and retraction, the point defined by DISCL from which the infeed traverse from the initial plane is performed at lower velocity (G341) or the point at which the actual approach movement starts (G340) must lie between the initial plane and the approach plane. If the point is outside this interval and the deviation is less than or equal to this machine data, it is assumed that the point is within the approach or retraction plane. If the deviation is greater, alarm 10741 (PRAL_WAB_DIRECTION_CHANGE) is output.		

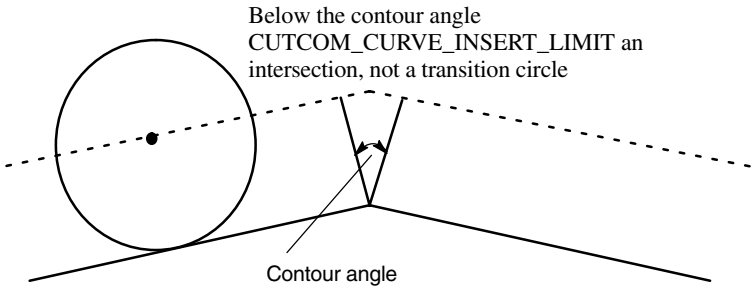
© Siemens AG, 2003. All rights reserved

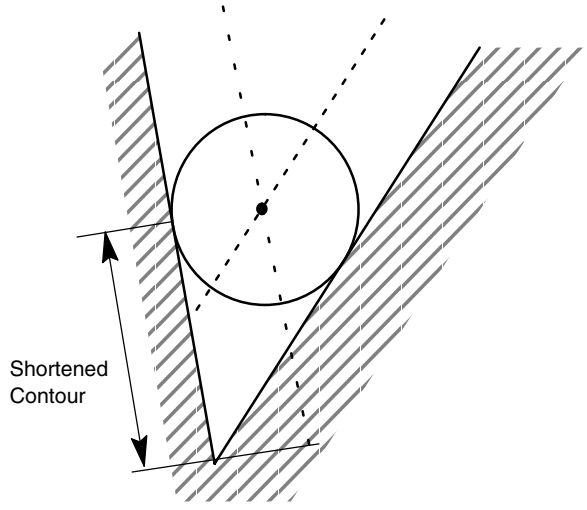
20210 MD number	**CUTCOM_CORNER_LIMIT** Max. angle for intersection calculation with tool radius compensation		
Default setting: 100.0	Minimum input limit: 0.0	Maximum input limit: 150.0	
Changes effective after RESET	Protection level: 2 / 7	Unit: degrees	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Where outside corners are very pointed, G451 can result in excessive idle paths (see Fig. 2-14). The system therefore switches automatically from G451 (intersection) to G450 (transition circle, with DISC where appropriate) when the outside corners are very pointed. The contour angle which can be traversed following this automatic switchover (intersection ——> transition circle) can be defined in CUTCOM_CORNER_LIMIT.		
			

© Siemens AG, 2003. All rights reserved

20220 MD number	**CUTCOM_MAX_DISC** Maximum value for DISC		
Default setting: 50.0	Minimum input limit: 0.0	Maximum input limit: 75.0	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	The G450 transition circle does not produce sharp outside contour corners because the path of the tool center point through the transition circle is controlled so that the cutting edge stops at the outside corner (programmed position). When sharp outside corners are to be machined with G450, the DISC instruction can be used to program an overshoot. This transforms the transition circle into a conic section and the cutting edge lifts off from the outside corner. The value range of the DISC instruction extends from 0 to theoretically 100 in steps of 1. DISC = 0 ... Overshoot disabled, transition circle active DISC = 100 ... Overshoot large enough to theoretically produce a response similar to intersection (G451). Programmed values of DISC which are higher than those stored in CUTCOM_MAX_DISC are limited to this maximum value without output of a message. A severely non-linear alteration in the path speed can thus be avoided.		
			
Special cases:	Values over 50 are generally not advisable with DISC. It is therefore not possible to enter values >75.		

© Siemens AG, 2003. All rights reserved

20230 MD number	**CUTCOM_CURVE_INSERT_LIMIT** Maximum angle for intersection calculation with tool radius compensation		
Default setting: 10.0	Minimum input limit: 0.0	Maximum input limit: 150.0	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: DOUBLE	Applies from SW 1.1		
Significance:	Where the outside corners are very flat, the response with G450 (transition circle) and G451 (intersection) becomes increasingly similar. In this case, it is no longer advisable to insert a transition circle. One reason why it is not permitted to insert a transition circle at these outside corners with 5-axis machining, is that this would impose restrictions on the speed in contouring mode (G64). Where the outside corners are very flat, the system therefore switches automatically from G450 (transition circle, with DISC where appropriate) to G451 (intersection). The contour angle which can be traversed following this automatic switchover (transition circle ——> intersection) can be defined in CUTCOM_CURVE_INSERT_LIMIT.		
			

20240 MD number	**CUTCOM_MAXNUM_CHECK_BLOCKS** Blocks for predictive contour calculation with tool radius compensation		
Default setting: 3	Minimum input limit: 2	Maximum input limit: 10	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	In certain cases with inside corners and “bottlenecks”, multiple intersections of the equidistant paths are found in several consecutive blocks. In these cases, the last intersection is always used as the valid intersection (see figure). This maximum number of blocks used for the predictive check can be entered in CUTCOM_MAXNUM_CHECK_BLOCKS. It is only possible to check two or more consecutive blocks if the collision monitoring has been activated with CDON (see Subsection 2.3.6).		
			

© Siemens AG, 2003. All rights reserved

20250 MD number	CUTCOM_MAXNUM_DUMMY_BLOCKS Max. no. of dummy blocks with no traversing movements		
Default setting: 3	Minimum input limit: 0	Maximum input limit: 5	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	In general, only program blocks with positions on geometry axes in the current plane are programmed when tool radius compensation (TRC) is active. When TRC is active, however, it is still possible to program individual intermediate blocks containing no positional data for the current plane, e.g.: • Positions in infeed axis • Auxiliary functions • General: Blocks that find their way into the main run and are executed there The maximum number of such dummy blocks is specified by this MD. If this is exceeded, alarm 10762 “Tool many blank blocks between 2 traversing blocks with active tool radius compensation” is activated. Note: Comment blocks, arithmetic blocks and blank blocks are not dummy blocks as far as these MDs are concerned and can therefore be programmed in any quantity (without an alarm being activated).		

20252 MD number	CUTCOM_MAXNUM_SUPPR_BLOCKS Maximum number of blocks with offset suppression		
Default setting: 5	Minimum input limit: 0	Maximum input limit: 10	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: BYTE	Applies from SW 4		
Significance:	Indicates the maximum number of blocks for tool radius compensation active, in which the “keep radius offset constant” function (CUTCONON or reprogramming of G41/G42 during active TRC) may be active. Note: The restriction of the number of blocks with active CUTONON is necessary in order to carry out repositioning in this situation too. Increasing this value for the machine data can lead to an increased memory requirement for NC blocks.		
Related to			

20256 MD number	CUTCOM_INTERS_POLY_ENABLE Intersection process possible for polynomials		
Default setting: 1	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BOOLEAN	Applies from SW 4		
Significance:	If this machine data is TRUE, the transitions at outside edges corners where polynomials (splines) are involved can be handled using the intersection procedure while tool radius compensation is active. If the machine data is FALSE, conical (circular) cuts are always inserted in this case. If FALSE, the response is identical to that in software releases older than 4.0.		
Related to			

© Siemens AG, 2003. All rights reserved

20270 MD number	**CUTTING_EDGE_DEFAULT** Basic setting of tool cutting edge without programming		
Default setting: 1	Minimum input limit: – 2	Maximum input limit: MD_SLMAX-CUTTINGEDGENUMBER	
Changes effective after: POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 2		
Significance:	MD: CUTTING_EDGE_DEFAULT = 0 On tool change, the offset is deselected automatically. MD: CUTTING_EDGE_DEFAULT > 0 the cutting number set automatically becomes active after tool change. MD: CUTTING_EDGE_DEFAULT = –1; the cutting edge number of the old tool is retained and is selected for the new tool after tool change. MD: CUTTING_EDGE_DEFAULT = –2; after changing to the new tool, the 2nd cutting edge of this tool is used for machining if the 2nd cutting edge of the previous tool was used. The offset values of the last offset selected remain active until a D number is programmed. **Tool management:** This setting should be selected if axes still have to be travesred between tool change (e.g. M06) and offset selection **Example:** MD: CUTTING_EDGE_DEFAULT = 1; After tool change the first cutting edge is active, if no cutting edge is programmed.		

20272 MD number	**SUMCORR_DEFAULT** Number for activating a new cutting edge offset		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 6	
Changes effective after RESET	Protection level: 7 / 2	Unit: Number	
Data type: DWORD	Applies from SW 5.1		
Significance:	Defines the sum offset of the cutting edge which is activated when a new sum offset is activated.		
Related to	MD 20270: CUTTING_EDGE_DEFAULT		

© Siemens AG, 2003. All rights reserved

20360 MD number	**TOOL_PARAMETER_DEF_MASK** Defines the effect of tool parameters		
Default setting: 0x0, 0x0, 0x0, 0x0, 0x0, ...	Minimum input limit: 0	Maximum input limit: 0xFFFF	
Changes effective after: POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 4.1 (bit 9 in SW 6 and higher)		
Significance:	Bit 0(LSB): For turning and grinding tools, the wear parameter of the transverse axis is calculated as a diameter value. The function is independent of DIAMON /DIAMOF, SW 5.3 and higher: Bit 1: For turning and grinding tools the tool length component of the transverse axis is calculated as a diameter value. Bit 2: If a wear component or length component is calculated as a diameter value, the tool may only be used in the planes which were active at tool selection. A change of planes produces an alarm. Bit 3: Zero offsets in frames in the transverse axis are calculated as diameter values. Bit 4: PRESET value is calculated as diameter value Bit 5: Work offsets external in the transverse axis are calculated as diameter values. Bit 6: Read actual values of the transverse axis as diameter values, (System variables: $AA_IW, $AA_IEN, $AA_IBN, $AA_IB, **but not** $AA_IM) independent of G code of the group 29 (DIAMON/DIAMOF). Bit 7: Display of all actual values of the transverse axis as diameter values independent of the G code of group 29 (DIAMON/DIAMOF). Bit 8: Always display distancetogo as radius in WCS. Bit 9: **In SW 6** and higher, when a transverse axis is moved in DRF handwheel mode, only half the distance of the defined increment is traversed if machine data MD 11346: HANDWHEEL_TRUE_DISTANCE = 1 is set.		

20390 MD number	**TOOL_TEMP_COMP_ON** Activation of temperature compensation for tool length		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after RESET	Protection level: 2 / 7	Unit: Number	
Data type: BOOLEAN	Applies from SW 6.1		
Significance:	This machine data activates temperature compensation in the tool direction (see also setting data SD 42960: TOOL_TEMP_COMP).		
Related to	SD 42960: TOOL_TEMP_COMP		

20392 MD number	**TOOL_TEMP_COMP_LIMIT** Maximum temperature compensation for tool length		
Default setting: 1.0	Minimum input limit: 0.0	Maximum input limit: plus	
Changes effective after RESET	Protection level: 7 / 7	Unit: mm	
Data type: DOUBLE	Applies from SW 6.1		
Significance:	This machine data specifies the maximum permissible temperature compensation value for tool length for each geometry axis. If a temperature compensation value higher than this limit is specified, it will be limited without an alarm.		
Related to	SD 42960: TOOL_TEMP_COMP		

© Siemens AG, 2003. All rights reserved

20396 MD number	**TOOL_OFFSET_DRF_ON** Handwheel override in tool direction		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after RESET	Protection level: 2 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 6.1		
Significance:	This machine data activates the handwheel override in tool direction. If this machine data is set, an active handwheel override in the axis assigned to length L1 of the active tool is active in the direction defined by the tool orientation. Example: G17 is active, the tool is a milling tool, the tool length L1 is therefore assigned to the Z axis (of the 3rd geometry axis). If the tool (e.g. for active 5–axis transformation) is rotated around the Y axis so that it points in the X direction, a handwheel override is acitve in the 3rd axis of the X axis.		
Related to			

© Siemens AG, 2003. All rights reserved

22530 MD number	**TOCARR_CHANGE_M_CODE** M code for change of toolholder		
Default setting: 0	Minimum input limit: –99999999	Maximum input limit: 99999999	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: DWORD	Applies from SW 4.1		
Significance:	The value of this machine data determines the number of the M code output when a toolholder is activated at the VDI interface. – If the MD is positive, the unchanged M code is always output. – If the MD is negative, the number of the toolholder is added to the value of the machine data and the sum is output.		
Application example(s)	MD value = –200 When TOCARR = 7 is programmed, M207 is output.		
Special cases, errors, ...	If the number of the code to be output or the value of this MD itself is one of the values 0 to 6, 17 or 30, no M code is output. There is no monitoring for conflicts between such a generated M code and other functions.		
References	Description of Functions H2		

22550 MD number	**TOOL_CHANGE_MODE** New tool offset with M function		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: BYTE	Applies from SW 1.1		
Significance:	The T function is used to select a tool in a program. The setting in this machine data determines whether the new tool is loaded immediately on execution of the T function: MD: TOOL_CHANGE_MODE = 0 The new tool is loaded immediately on execution of the T function. This setting is used mainly on turning machines with tool turrets. MD: TOOL_CHANGE_MODE = 1 The new tool is prepared for loading on execution of the T function. When using milling machines with tool magazine, this setting is used mainly in order to bring the new tool into the tool change position without interrupting the machining process. The MD 22560: Remove the old tool from the spindle and load the new tool onto the spindle. In accordance with DIN 66025 his tool change is required to be programmed with the M function M06.		
Related to	MD 22560: TOOL_CHANGE_M_CODE		

22560 MD number	**TOOL_CHANGE_M_CODE** M function for tool change		
Default setting: 6	Minimum input limit: 0	Maximum input limit: 9999 9999	
Changes effective after POWER ON	Protection level: 2	Unit: –	
Data type: DWORD	Applies from SW 1.1		
Significance:	If the T function is only used to prepare a new tool for a tool change (this setting is used mainly on milling machines with a tool magazine, in order to bring the new tool into the tool change position without interrupting the machining process), another M function must be used to trigger the tool change. The M function entered in TOOL_CHANGE_M_CODE triggers the tool change (remove old tool from spindle and insert new tool in spindle). This tool change is required to be programmed with M function M06, in accordance with DIN66025.		
Related to	MD 22550: TOOL_CHANGE_MODE		

© Siemens AG, 2003. All rights reserved

22562 MD number	**TOOL_CHANGE_ERROR_MODE** Error response on programmed tool change		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 7/2	Unit: –	
Data type: DWORD	Applies from SW 5.1		
Significance:	The default value = 0 of the machine data should not be changed when working with the setting TOOL_CHANGE_MODE = 0 (a tool change is only programmed with a T command). In this case, the value of the machine data has no effect. Bit no. Value Meaning 0 0 Default response The program stops at the NC block in which the error occurred. 1 0x1 If an error occurs in the block with the tool change preparation, the alarm referring to the preparation command (T) is ignored until the associated tool change command (M06) is interpreted in the program execution. Only then does the system output the alarm which was triggered by the preparation command. It is therefore not possible for the user to make corrections until this block is reached. The value = 1 is only significant if the setting $MC_TOOL_CHANGE_MODE = 1 is used. 1 0 Only when tool management is active (default response in this case). During tool change preparation, the NCK only detects tools whose data are assigned to a magazine. 1 0x2 Only when tool management is active. The NCK loads a tool even if its data are known in the NCK but are not assigned to any magazine. In this case, the NCK attempts to assign the tool data automatically to the programmed spindle location. The prerequisite is that there is exactly one tool in the programmed tool group which could be loaded but which is not assigned to a magazine location. If several tools can be used, the NCK searches for an active tool again. If none is available, the tool with the lowest duplo number is selected. 2 0 Default response Active D no. > 0 and active T no. = 0 produces an offset of zero Active DL no. > 0 and active D no. = 0 produces a total offset of zero 1 0x4 Active D no. > 0 and active T no. = 0 triggers an alarm message Active DL no. > 0 and active D no. = 0 triggers an alarm message Bits 3 and 4: Relevant only when tool management is active. Function: Control of behavior of Init block generation on program START if disabled tool is in the spindle and needs to be activated. For further information, see especially: $MC_START_MODE_MASK, $MC_RESET_MODE_MASK. At RESET in particular, the response 'leave disabled tool' active on the spindle is not affected.		

© Siemens AG, 2003. All rights reserved

22562 MD number	**TOOL_CHANGE_ERROR_MODE** Error response on programmed tool change		
Significance:	3	0	Default settings: If the tool on the spindle is disabled: Generate tool change command which requests a replacement tool. If there is no such tool, an alarm is generated.
		1 0x8	The disabled status of the spindle tool is ignored. The tool becomes active. The following parts program should be formulated such that no parts are machined with the disabled tool.
	4	0	Default settings: It is attempted to activate the spindle tool or its replacement tool.
		1 0x10	If the tool on the spindle is disabled, then T0 is programmed in the START init block.
Related to	MD 22550: TOOL_CHANGE_MODE		

© Siemens AG, 2003. All rights reserved

4.3 Setting data

42442 SD number	**TOOL_OFFSET_INCR_PROG** Retraction of tool offsets on incremental programming		
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 4.3		
Significance:	0: When incremental programming is used on an axis, only the programmed position delta is traversed. Tool length offsets are only traversed with absolute position specification. 1: When incremental programming is used on an axis, changes to tool length offsets are retracted after a tool change. (standard response up to SW 3)		
Related to	SD 42440: FRAME_OFFSET_INCR_PROG		

42470 SD number	**CRIT_SPLINE_ANGLE** Limit angle for spline and polynomial interpolation and compressor		
Default setting: 0.0, ...	Minimum input limit: –	Maximum input limit: –	
Changes effective after Reset	Protection level: 2/7	Unit: degrees	
Data type: DOUBLE	Applies from SW 4.1		
Significance:	MD 42470 determines the maximum angle between two neighboring blocks that can still be deemed to be a tangent transition for polynomial interpolation (POLY). If the angles are smaller, the polynomials are adapted to obtain an exact tangential transition. For A or C spline interpolation, the block transitions are formed with a larger angle at the corner (the spline ends and starts again at a tangent to the programmed blocks). When the online compressor is active (COMPON), the compression does not cover block boundaries with large angles, i.e. large angles are retained.		
Related to			

42480 SD number	**STOP_CUTCOM_STOPRE** Alarm reaction for tool radius compensation and preprocessing stop		
Default setting: 1	Minimum input limit: –	Maximum input limit: –	
Changes effective after Reset	Protection level: 7	Unit: –	
Data type: BOOLEAN	Applies from SW 4.1		
Significance:	If this setting data is TRUE, block processing for preprocessing stop and active tool radius compensation is stopped and continued only after user confirmation. If the value is FALSE, processing is not interrupted at such a point in the program.		
Related to			

© Siemens AG, 2003. All rights reserved

42490 MD number	**CUTCOM_G40_STOPRE** Traversing response for tool offset on preprocessing stop		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Change effective after: immediately	Protection level: 7 / 7	Unit: –	
Data type: BOOLEAN	Applies from SW 4.3		
Significance:	FALSE: If a feed stop (programmed or created internally by the control) is present before the deselect block (G40) for active too radius correction, the start point of the deselect block is approached before the last end point in front of the feed stop. Then, the deselect block is processed, i.e. tw3o traversing blocks are usually produced from the deselect block. No tool radius correction is active in the blocks. The behavior is therefore identical to the one before this setting data was introduced. TRUE: If a feed stop (programmed or created internally by the control) is present before the deselect block (G40) for active too radius correction, the start point of the deselect block is approached before the last end point in front of the feed stop.		
Related to			

© Siemens AG, 2003. All rights reserved

42494 SD number	**CUTCOM_ACT_DEACT_CTRL** Approach/retraction behavior for tool radius compensation in blocks without travel information		
Default setting: 2222, ...	Minimum input limit: –	Maximum input limit: –	
Changes effective after POWER ON	Protection level: 7	Unit: –	
Data type: DWORD	Applies from SW 5.2		
Significance:	The SD controls the approach and retraction behavior for tool radius compensation in cases in which the activation or deactivation block does not contain travel information. It is only evaluated for 2–1/2 TRC (CUT2D or CUT2DF). The coding is decimal:		
	N N N N Approach behavior for tools with tool point direction (turning tools) Approach behavior for tools without tool point direction (milling tools) Retraction behavior for tools with tool point direction (turning tools) Retraction behavior for tools without tool point direction (milling tool)		
	If the appropriate position contains a **1**, then an approach or retraction is always performed, even if G41/G42 or G40 stands alone in the block, e.g. N100 X10 X0 N110 G41 N120 X20 If a tool radius of 10 mm is assumed in the above example, then block N110 traverses to the position X10 Y10. If the appropriate position contains a **2**, then an approach or retraction is only performed, if at least one axis of the compensation plane is programmed in the activation/deactivation block. If the same result as in the above example is to be achieved with this setting, then the program must be changed as follows: N100 X10 X0 N110 G41 X10 N120 X20 If the axis specification X10 is missing in block N110, then activation of the TRC is delayed by one block, i.e. the activation block would be N120. If the significant position contains a different value than 1 or 2 (e.g. **0**), there no approach or retraction in a block that does not contain traversing information. **Tools with tool point direction:** These are tools with tool numbers between 400 and 599 (turning/grinding tools) whose tool point direction has a value between 1 and 8. Turning and grinding tools with tool point direction 0 or 9, or with other undefined values, are treated like milling tools. **Note** If the value of this SD is changed within a program, it is recommended to program a preprocessing stop (STOPRE) before writing, as it may otherwise occur that the new value is used in preceding program sections.		
Related to			

© Siemens AG, 2003. All rights reserved

42900 SD number	MIRROR_TOOL_LENGTH Mirror tool length offset		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after Reset	Protection level: 7	Unit: –	
Data type: BOOLEAN	Applies from SW 5.1		
Significance:	If this setting data is not equal to zero, the tool length components ($TC_DP3[..., ...] to $TC_DP5[..., ...]) and the components of the tool base dimension ($TC_DP21[..., ...] to $TC_DP23[..., ...]), whose associated axes are mirrored, are also mirrored, i.e. their sign is inverted. The wear values are not mirrored. If these are to be mirrored as well, setting data $SC_MIRROR_TOOL_WEAR must be enabled.		
Related to	SD 42910: MIRROR_TOOL_WEAR SD 42920: WEAR_SIGN_CUTPOS SD 42930: WEAR_SIGN SD 42940: TOOL_LENGTH_CONST		

42910 SD number	MIRROR_TOOL_WEAR Mirror wear values of tool length compensation		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after Reset	Protection level: 7	Unit: –	
Data type: BOOLEAN	Applies from SW 5.1		
Significance:	If this setting data is not equal to zero, the wear values of the tool length components whose associated axes are mirrored are also mirrored, i.e. their sign is inverted. The wear values are the sum of the actual tool wear values, the sum offsets and the setup offsets.		
Related to	SD 42900: MIRROR_TOOL_LENGTH SD 42920: WEAR_SIGN_CUTPOS SD 42930: WEAR_SIGN SD 42940: TOOL_LENGTH_CONST		

42920 SD number	WEAR_SIGN_CUTPOS Mirror wear values of machining plane		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after Reset	Protection level: 7	Unit: –	
Data type: BOOLEAN	Applies from SW 5.1		
Significance:	If this setting data is not equal to zero, the sign evaluation of the wear components in the machining plane depend on the tool point direction in the case of tools with a relevant tool point direction (turning and grinding tools, tool types 400 to 599). This setting data is ignored with tool types without a relevant tool point direction. In the following table, components whose sign is inverted if $SC_WEAR_SIGN_CUTPOS is not equal to zero are marked with an X.		
Related to	SD 42910: MIRROR_TOOL_WEAR SD 42900: MIRROR_TOOL_LENGTH SD 42930: WEAR_SIGN SD 42940: TOOL_LENGTH_CONST		

Tool point direction	Length 1	Length 2
1		
2		X
3	X	X
4	X	
5		
6		
7		X
8	X	
9		

The sign evaluations controlled by $SC_WEAR_SIGN_CUTPOS and $SC_MIRROR_TOOL_WEAR are independent of each other, i.e. if the sign of a component is inverted by both setting data, the resulting sign is unchanged.

© Siemens AG, 2003. All rights reserved

42930 SD number	**WEAR_SIGN** Invert sign of all wear values		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after Reset	Protection level: 7	Unit: –	
Data type: BOOLEAN	Applies from SW 5.1		
Significance:	If this setting data is not equal to zero, the sign of all wear dimensions is inverted. This affects both the tool length and other variables such as tool radius, rounding radius, etc. Entering a positive wear dimension makes the tool “shorter” and “thinner”.		
Related to	SD 42910: MIRROR_TOOL_WEAR SD 42920: WEAR_SIGN_CUTPOS SD 42900: MIRROR_TOOL_LENGTH SD 42940: TOOL_LENGTH_CONST		

42940 SD number	**TOOL_LENGTH_CONST** Retain the assignment of tool length components when changing the machining plane (G17 to G19)		
Default setting: 0	Minimum input limit: –	Maximum input limit: –	
Changes effective after Reset	Protection level: 7	Unit: –	
Data type: DWORD	Applies from SW 5.1		
Significance:	If this setting is not equal to zero, the assignment between tool length components (length, wear and tool base dimension) and geometry axes is not changed when the machining plane is changed (G17 – G19). The assignment of tool length components to geometry axes is determined by the value of the setting data in accordance with the table below. A distinction is made between the assignment of turning and grinding tools (tool types 400 to 599) and other tools (typically milling tools). In the tables it is assumed that geometry axes 1 to 3 are named X, Y and Z. However, the axis sequence, and not the axis identifier, is the determining factor in the assignment of an offset to an axis. Assignment for turning and grinding tools (tool types 400 to 599): Plane Length 1 Length 2 Length 3 17 Y X Z 18* X Z Y 19 Z Y X –17 X Y Z –18 Z X Y –19 Y Z X * All values which are not equal to zero and which are not equal to one of the six values listed are handled in the same way as value 18. For values with the same figure but a different sign, the assignment of length 3 is identical, but lengths 1 and 2 are swapped. Assignment for all tools which are neither turning tools nor grinding tools (tool types < 400 or > 599): Plane Length 1 Length 2 Length 3 17* Z Y X 18 Y X Z 19 X Z Y –17 Z X Y –18 Y Z X –19 X Y Z * All values which are not equal to zero and which are not equal to one of the six values listed are handled in the same way as value 17. For values with the same figure but a different sign, the assignment of length 3 is identical, but lengths 1 and 2 are swapped.		
Related to	SD 42900: MIRROR_TOOL_LENGTH SD 42910: MIRROR_TOOL_WEAR SD 42920: WEAR_SIGN_CUTPOS SD 42930: WEAR_SIGN SD 42950: TOOL_LENGTH_TYPE		

© Siemens AG, 2003. All rights reserved

42935 SD number	**WEAR_TRANSFORM** Transformation of wear data		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 7	Unit: –	
Data type: DWORD	Applies from SW 5.3		
Significance:	This bit-coded setting data specifies which of the three wear components $TC_DP12 – $TC_DP14 wear, $TC_SCPx3 – $TC_SCPx5 additive offset fine, $TC_ECPx3 – $TC_ECPx5 additive offset coarse, are subjected to an adapter transformation and a transformation by an orientational tool-holder if one of the G codes TOWMCS or TOWWCS from G code group 56 is active. If the initial setting G code TOWSTD is active, this setting data is not effective. The following assignment should be used: Bit 0 = TRUE: Transformation not at $TC_DP12 – $TC_DP14 Bit 1 = TRUE: Transformation not at $TC_SCPx3 – $TC_SCPx5 Bit 2 = TRUE: Transformation not at $TC_ECPx3 – $TC_ECPx5 The unspecified bits are unassigned.		
Related to	–		

42950 SD number	**TOOL_LENGTH_TYPE** Assign tool length components independent of tool type		
Default setting: 0	Minimum input limit: 0	Maximum input limit: –	
Changes effective immediately	Protection level: 7 / 7	Unit: –	
Data type: DWORD	Applies from SW 5.3		
Significance:	This setting data specifies the assignment between the tool length components and the geometry axes, independent of the tool type. The value range is from 0 to 2. Any other value is interpreted as 0. Value 0: The assignment is as standard. A distinction is made between grinding tools (tool types 400 to 599) and other tools (milling tools). Value 1: The assignment of tool length components is always the same as for milling tools, independent of the actual tool type. Value 2: The assignment of tool length components is always the same as for turning tools, independent of the actual tool type. The setting data is also effective for wear values assigned to the length components. If setting data SD 42940: TOOL_LENTH_CONST is set, the table defined for milling or turning tools defined by SD 42950: TOOL_LENFTH_TYPE is accessed, independent of the actual tool type, if the value of the latter setting data is not equal to zero.		
Related to	SD 42940: TOOL_LENGTH_CONST.		

© Siemens AG, 2003. All rights reserved

42960 SD number	**TOOL_TEMP_COMP** Temperature compensation value in relation to tool		
Default setting: 0x0, 0x0, 0x0, 0x0, 0x0, ...	Minimum input limit: –	Maximum input limit: –	
Changes effective immediately	Protection level: 7 / 7	Unit: –	
Data type: DOUBLE	Applies from SW 6.1		
Significance:	Temperature compensation value in relation to tool. The compensation value works as a vector according to the current rotation of the tool direction. This setting data is evaluated only if temperature compensation has been activated for tools with MD 20390: TOOL_TEMP_COMP_ON. The temperature compensation type for “Offset in tool direction” must also be set via bit 2 in MD 32750: TEMP_COMP_TYPE. **Note** “Temperature compensation” is an option which must be enabled in advance.		
Related to	MD 20390: TOOL_TEMP_COMP_ON MD 32750: TEMP_COMP_TYPE		

42974 SD number	**TOCARR_FINE_CORRECTION** Fine offset TCARR on / off		
Default setting: FALS,E FALSE, FALSE, FALSE, FALS, ...	Minimum input limit: –	Maximum input limit: –	
Changes effective immediately	Protection level: 7 / 7	Unit: –	
Data type: BOOL	Applies from SW 6.4		
Significance:	This setting data can be used to consider the fine offset values for an orientable toolholder. TRUE: The fine offset values are considered on activation of an orientable toolholder. FALSE: The fine offset values are **not** considered on activation of an orientable toolholder not taken into account.		
Related to	MD 20188: TOCARR_FINE_LIM_LIN MD 20190: TOCARR_FINE_LIM_ROT		

■

© Siemens AG, 2003. All rights reserved

5 Signal Descriptions

None

6 Examples

6.1 Example of orientational toolholders

Request

The following example uses a toolholder which is described fully by a rotation around the Y axis. It is therefore sufficient to enter only one value to define the rotary axis (block N20).

Blocks N50 to N70 describe an end mill with radius 5 mm and length 20 mm.

Block N90 defines a rotation of 37 degrees around the Y axis.

Block N120 activates the tool radius compensation and all settings are made to describe the offset in the following blocks with a rotation of 37 degrees around the Y axis.

```
N10                                  ; Definition of toolholder 1
N20 $TC_CARR8[1] = 1                 ; Component of the first rotary axis in Y direction
N30
N40                                  ; Definition of the tool offset memory T1/D1
N50 $TC_DP1[1,1] = 120               ; Shafttype milling cutter
N60 $TC_DP3[1,1]= 20                 ; Length 1
N70 $TC_DP6[1,1] = 5                 ; Radius
N80
N90 ROT Y37                          ; 37° rotation about y axis
N100
N110 X0 Y0 Z0 F10000
N120 G42 CUT2DF TCOFR TCARR = 1 T1 D1 X10
N130 X40
N140 Y40
N150 X0
N160 Y0
N170 M30
```

© Siemens AG, 2003. All rights reserved

6.1.1 Example of orientational toolholders with rotary table (SW 5.3 and higher)

Use of the MOV command

For use of the MOV command it is assumed that the program is running on a 5-axis machine on which the tool rotates around the Y axis in case of a rotation of the B axis:

```
N10 TRAORI()
N20 X0 X0 Z0 B45 F2000     ; Setting of the tool orientation
N30 MOVT=–10               ; Infeed motion 10 mm in tool direction (under
                           45 degrees in the YZ plane)
N40 MOVT=AC(20)            ; Return in tool direction to distance 20mm from
                           ; zero point
```

Machine with rotary table

Complete definition for the use of an orientational toolholder with rotary table:

```
N10   $TC_DP1[1,1]=  120
N20   $TC_DP3[1,1]=   13                ; Tool length 13 mm

; Definition of toolholder 1:

N30   $TC_CARR1[1] =   0                ; X component of 1st offset vector
N40   $TC_CARR2[1] =   0                ; Y component of 1st offset vector
N50   $TC_CARR3[1] =   0                ; Z component of 1st offset vector

N60   $TC_CARR4[1] =   0                ; X component of 2nd offset vector
N70   $TC_CARR5[1] =   0                ; Y component of 2nd offset vector
N80   $TC_CARR6[1] = –15                ; Z component of 2nd offset vector

N90   $TC_CARR7[1] =   1                ; X component of 1st axis
N100  $TC_CARR8[1] =   0                ; Y component of 1st axis
N110  $TC_CARR9[1] =   0                ; Z component of 1st axis

N120  $TC_CARR10[1] =  0                ; X component of 2nd axis
N130  $TC_CARR11[1] =  1                ; Y component of 2nd axis
N140  $TC_CARR12[1] =  0                ; Z component of 2nd axis

N150  $TC_CARR13[1] = 30                ; Angle of rotation of 1st axis
N160  $TC_CARR14[1] =–30                ; Angle of rotation of 2nd axis

N170  $TC_CARR15[1] =  0                ; X component of 3rd offset vector
N180  $TC_CARR16[1] =  0                ; Y component of 3rd offset vector
N190  $TC_CARR17[1] =  0                ; Z component of 3rd offset vector

N200  $TC_CARR18[1] =  0                ; X component of 4th offset vector
N210  $TC_CARR19[1] =  0                ; Y component of 4th offset vector
N220  $TC_CARR20[1] = 15                ; Z component of 4th offset vector

N230  $TC_CARR21[1] = A                 ; Reference for 1st axis
N240  $TC_CARR22[1] = B                 ; Reference for 2nd axis
N250  $TC_CARR23[1] = “P”               ; Type of toolholder
```

© Siemens AG, 2003. All rights reserved

```
N260  X0 Y0 Z0 A0 B45 F2000
N270  TCARR=1 X0 Y10 Z0 T1 TCOABS
N280  PAROT
N290  X0 Y0 Z0
N300  G18 MOVT=AC(20)
N310  G17 X10 Y0 Z0
N320  MOVT=–10
N330  PAROTOF (SW 6 and higher), ROT (up to SW 5.3)
N340  TCOFR
N350  X10 Y10 Z–13 A0 B0
N360  ROTS X–45 Y45
N370  X20 Y0 Z0 D0
N380  Y20
N390  X0 Y0 Z20
N400  M30
```

Explanation of the example above

The definition of the orientational toolholder is given in full. The components which contain the value 0 need not actually be given, as they are preset to zero in any case.

The toolholder is activated in N270. As *$TC_CARR21* and *$TC_CARR22* refer to the machine axes A and B and TCOABS is active, the values in *$TC_CARR13* and *$TC_CARR14* are ignored, i.e. the axis position A0 B45 is used for the rotation.

The rotation of the 4th offset vector (length 15 mm in Z direction) around the B axis causes an offsetting of the zero point by X10.607 [= 15 * sin(45)] and Z–4.393 [= –15 * (1.–cos(45))]. This zero offset is taken into account by an automatically written basic or system frame so that the position X10.607 Y10.000 Z8.607 is approached. In the Z direction the tool selection leads to an additional offset of 13 mm; the Y components are not affected by the table rotation.

N280 defines a rotation in accordance with the rotation of the table of the orientational toolholder. The new X direction thus points in the direction of the bisecting line in the 4th quadrant, the new Z axis in the direction of the bisecting line in the 1st quadrant.

The zero point is approached in N290, i.e. the machine position X10.607 Y0 Z–4.393, since the position of the zero point is not changed by the rotation.

N300 traverses in Y to the position Y33.000, since G18 is active and the Y components are not affected by the active frame. The X and Z positions remain unchanged.

The position X17.678 Y0 Z1.536 is approached in N310.

N320 changes only the Z position to the value –8.464 as a result of the MOVT command. As only the table can be rotated, the tool orientation remains unchanged parallel to the machine Z direction, even if the Z direction of the active frame is rotated by 45 degrees.

N330 deletes the basic or system frame; the frame definition from N280 is thus undone. In previous software versions older than SW 6, in which the disable command PAROTOF is not yet provided, the basic frame must be deleted explicitly, which can be carried out using the command $P_CHBFRAME[$MC_TOCARR_BASE_FRAME_NUMBER] = CTRANS(X,0).

© Siemens AG, 2003. All rights reserved

In N340 TCOFR specifies that the orientational toolholder is to be aligned according to the active frame. Since a rotation is no longer active in N330 due to the PAROTOF command, the initial settings results. The frame offset becomes zero.

N350 thus approaches the position X10 X10 Z0 (= Z–13 + tool length). Caution: Through the simultaneous programming of both rotary axes A and B the actual position of the orientational toolholder is made to match that used in N340. The position approached by the three linear axes is dependent on this position, however.

In N360 solid angles are used to define a plane whose intersecting lines in the XZ and in the YZ plane each form an angle of +45 degrees or –45 degrees with the X or Y axis. A plane defined in this way is therefore positioned as shown in Fig. 2-29, i.e. the surface normal is pointing in the direction of the space diagonal.

N370 traverses to the position X20 Y0 Z0 in the new coordinate system. Since at the same time the tool is deselected with D0, there is no longer an additional offset in Z. Since the new X axis lies in the old XZ plane, this block reaches the machine position X14.142 Y0 Z–14.142.

N380 traverses on the Y axis in the rotated coordinate system. This leads to a motion of all three machine axes. The machine position is X5.977 Y16.330 Z–22.307.

N390 approaches a point on the new Z axis. Relative to the machine axes this is thus on the solid diagonal. All three axes thus reach the position 11.547.

6.1.2 Example basic tool orientation (SW 6.1 and higher)

Basic orientation in the bisector

A milling tool is defined with length L1=10 whose basic orientation is in the bisector of the XZ plane.

```
N10  $TC_DP1[1,1]=120
N20  $TC_DP3[1,1]=10

N30  $TC_DPV [1,1] = 0
N40  $TC_DPV3[1,1] = 1
N50  $TC_DPV4[1,1] = 0
N60  $TC_DPV5[1,1] = 1

N70  g17 f1000 x0 y0 z0 t1 d1
N80  movt=10
N80  m30
```

Explanation of example:
In N10 to N60, a milling tool is defined with length L1=10 (N20). The basic orientation is in the bisector of the XZ plane N40 to N60.

In N70, the tool is activated and the zero position is approached. Machine axis positions X0 Y0 Z10 result in this block from the tool length.
In N80, an incremental traversing movement of 10 is executed in the direction of the tool. The resulting axis positions are thus X7.071 Y0 Z17.071.

© Siemens AG, 2003. All rights reserved

6.1.3 Calculation of offsets according to location of use or workpiece

Tool with adapter

A tool with adapter and orientational toolholder is defined in the following program example. In order to simplify the overview, only length L1 is different to zero for the additive and insert offsets and for the adapter. The offset vectors of the orientational toolholder are all zero.

```
N10   $TC_TP2[1] = “MillingTool“      ; Identifier
N20   $TC_TP7[1] = 9                 ; Location type
N30   $TC_TP8[1] = 2                 ; Status : Enabled and not blocked

; D corr. D=1

N40   $TC_DP1[1,1]=120               ; Tool type – mill
N50   $TC_DP3[1,1]=10.               ; Length offset vector

N60   $TC_DP12[1,1]= 1.              ; Wear

N70   $TC_SCP13[1,1]=0.1             ; Sum offset DL=1

N80   $TC_ECP13[1,1]=0.01            ; Insert offset DL=1

N90   $TC_ADPTT[1]=5                 ; Adapter transformation
N100  $TC_ADPT1[1]=0.001             ; Adapter dimensions

                                     ; Magazine data
N110  $TC_MAP1[1]=3                  ; Type of magazine: Revolver
N120  $TC_MAP2[1]=“Turret”           ; Identifier of a magazine
N130  $TC_MAP3[1]=17                 ; Status of magazine
N140  $TC_MAP6[1]=1                  ; Dimension – row
N150  $TC_MAP7[1]=2                  ; Dimension – column –> 2 locations

N160  $TC_MPP1[1,1]=1                ; Location type
N170  $TC_MPP2[1,1]=9                ; Location type
N180  $TC_MPP4[1,1]=2                ; Location state

N190  $TC_MPP7[1,1]=1                ; Bring adapter to location

N200  $TC_MPP6[1,1]=1                ; T number “MillingTool”

N210  $TC_MAP1[9999]= 7              ; Type of magazine: Buffer
N220  $TC_MAP2[9999]=“Buffer”        ; Identifier of a magazine
N230  $TC_MAP3[9999]=17              ; Status of magazine
N240  $TC_MAP6[9999]=1               ; Dimension – row
N250  $TC_MAP7[9999]=1               ; Dimension – column –> 1 location

N260  $TC_MPP1[9999,1]=2             ; Location type
N270  $TC_MPP2[9999,1]=9             ; Location type
N280  $TC_MPP4[9999,1]=2             ; Location state
N290  $TC_MPP5[9999,1]=1             ; Spindle number 1
N300  $TC_MDP2[1,1]=0                ; Distance from spindle to Mag.1

                                     ; Definition of toolholder 1
N310  $TC_CARR10[1] =  1             ; Component for 2nd rotary axis in X direction
N320  $TC_CARR14[1] =  45            ; Angle of rotation of 2nd axis
N330  $TC_CARR23[1] = “T”            ; Tool Mode
```

© Siemens AG, 2003. All rights reserved

```
N340  stopre
N350  $SC_WEAR_TRANSFORM = 'B101'
N360  T0 D0 DL=0
N370  ROT X30
N380  G90 G1 G17 F10000 X0 Y0 Z0

N390  T="MillingTool" X0 Y0 Z0 TOWSTD          ; X 0.000  Y11.110  Z 0.001
N400  T="MillingTool" X0 Y0 Z0 TOWMCS          ; X 0.000  Y10.100  Z 1.011
N410  T="MillingTool" X0 Y0 Z0 TOWWCS          ; X 0.000  Y 9.595  Z 0.876
N420  TCARR=1 X0 Y0 Z0                         ; X 0.000  Y 6.636  Z 8.017
N430  G18 X0 Y0 Z0                             ; X10.100  Y–0.504  Z 0.876
N440  m30
```

Explanation of the example above

Starting at block N390, various methods are used to approach position X0 Y0 Z0. The machine positions reached are specified in the blocks in comments. After the program a description is given of how the positions were reached.

N390: Adapter transformation 5 (block N90) transforms length L1 into length L2. Only the actual adapter dimension is not subject to this transformation. The Y value (L2 with G17) results from the sum of the tool length (10), tool wear (1), sum offset (0.1) and insert offset (0.01). The adapter dimension (0.001) is in Z (L1).

N400: In block N350, bits 0 and 2 are enabled in setting data *$SC_WEAR_TRANSFORM*. This means that the tool wear and the insert offset are not subject to the transformation because of TOWMCS in block N400. The sum of these two offsets is 1.01. The Z position is therefore increased by this amount and the Y position is reduced by this amount compared with block N390.

TOWWCS is active in N410. The sum of the tool wear and the insert offset is thus effective in the active workpiece coordinate system. In block N370, a rotation through 30 degrees is activated about the X axis. The original offset value of 1.01 in the Z direction thus yields a new Z component of 0.875 (= 1.01 * cos(30)) and a new Y component of –0.505 (= 1.01 * sin(30)). This yields the dimension specified in the program comment when added to the sum of the tool length, sum offset and adapter dimension produced in block N390.

In addition, an orientational toolholder is activated in block N420. This executes a rotation through 45 degrees about the X axis (see N310 – N330). Since all offset vectors of the toolholder are zero, there is no additional zero offset. The orientational toolholder acts on the sum of the tool length, sum offset and adapter dimension. The resulting vector component is X0 Y7.141 Z7.142. The sum of the tool wear and insert offset evaluated in the WCS is then added as in block N410.

G18 is activated in N430. The components of the sum of the tool length, sum offset and adapter dimension are interchanged accordingly. The orientational toolholder continues to act on this new vector (rotation through 45 degrees about X axis). The resulting vector component is X10.100 Y0.0071 Z0.0071. The vector generated from the tool wear and insert offset (X0 Y–0.505 Z0.875) is not affected by the plane change. The sum of the two vectors yields the dimension specified in the comment in N430.

■

© Siemens AG, 2003. All rights reserved

Data Fields, Lists

7

7.1 Interface signals

DB number	Bit, byte	Name	Reference
Channel-specific			
21, ...	61.0	T function 1 change	S5
21, ...	62.0	D function 1 change	S5
21, ...	116–119	T function 1	S5
21, ...	128–129	D function 1	S5
21, ...	214	Active G function of group 7	K1
21, ...	223	Active G function of group 16	K1
21, ...	224	Active G function of group 17	K1
21, ...	225	Active G function of group 18	K1
21, ...	230	Active G function of group 23	K1

7.2 Machine data

Number	Identifier	Name	Reference
General ($MN_ ...)			
18082	MM_NUM_TOOL	Number of tools that the NCK can manage (SRAM)	S7
18088	MM_NUM_TOOL_CARRIER	Number of toolholders	
18094	MM_NUM_CC_TDA_PARAM	Number of TOA data	FBW, S7
18096	MM_NUM_CC_TOA_PARAM	Number of TOA data which can be set up per tool and evaluated by the CC	FBW, S7
18100	MM_NUM_CUTTING_EDGES_IN_TOA	Tool offsets per TOA module	S7
18102	MM_TYPE_OF_CUTTING_EDGE	Activate flat D number management (SW 4 and higher)	
18105	MM_MAX_CUTTING_EDGE_NO	Address extension interpreted as spindle number (SW 5 and higher)	
18106	MM_MAX_CUTTING_EDGE_PERTOOL	Maximum number of cutting edges per tool (SW 5 and higher)	
18108	MM_NUM_SUMCORR	Number of all sum offsets in NCK (SW 5 and higher)	
18110	MM_MAX_SUMCORR_PER_CUTTEDGE	Maximum number of sum offsets per cutting edge (SW 5 and higher)	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Reference
18112	MM_KIND_OF_SUMCORR	Properties of sum offsets in the NCK (SW 5 and higher)	
18114	MM_ENABLE_TOOL_ORIENT	Assign orientation to cutting edges	
Channel-specific ($MC_ ...)			
20096	T_M_ADDRESS_EXT_IS_SPINO	Spindle number as address extension (SW 5 and higher)	FBW, S7
20110	RESET_MODE_MASK	Definition of control basic setting after power-up and RESET parts program end	K2
20120	TOOL_RESET_VALUE	Definition of tool for which tool length compensation is selected during power-up or on reset or parts program end as a function of MD 20110	K2
20121	TOOL_PRESEL_RESET_VALUE	Definition of the preselected tool for which the tool length compensation is selected during powerup and on reset or parts program end as a function of MD 20110	K2
20126	TOOL_CARRIER_RESET_VALUE	Active toolholder on RESET	
20130	CUTTING_EDGE_RESET_VALUE	Definition of tool cutting edge for which tool length compensation is selected during powerup or on reset or parts program end as a function of MD 20110	K2
20132	SUMCORR_RESET_VALUE	Number for selecting sum offset (SW 5 and higher)	
20140	TRAFO_RESET_VALUE	Definition of transformation block which is selected during power-up and or RESET or parts program end as a function of MD 20110	K2
20180	TOCARR_ROT_ANGLE_INCR[i]	Minimum incremental step for orientational toolholder (SW 5.3 and higher)	
20182	TOCARR_ROT_ANGLE_OFFSET[i]	Offset of the axis of rotation for orientational toolholder (SW 5.3 and higher)	
20184	TOCARR_BASE_FRAME_NUMBER	Base frame of the table offset for orientational toolholder with rotary table (SW 5.3 and higher)	
20188	TOCARR_FINE_LIM_LIN	Limit linear fine offset TCARR (SW 6.4 and higher)	
20190	TOCARR_FINE_LIM_ROT	Limit rotary fine offset TCARR (SW 6.4 and higher)	
20202	WAB_MAXNUM_DUMMY_BLOCKS	Maximum number of blocks with no traversing motions with SAR	
20204	WAB_CLEARANCE_TOLERANCE		
20210	CUTCOM_CORNER_LIMIT	Max. angle for intersection calculation with tool radius compensation	
20220	CUTCOM_MAX_DISC	Maximum value for tool radius compensation	
20230	CUTCOM_CURVE_INSERT_LIMIT	Minimum value for intersection calculation with tool radius compensation	
20240	CUTCOM_MAXNUM_CHECK_BLOCKS	Blocks for predictive contour calculation with tool radius compensation	
20250	CUTCOM_MAXNUM_DUMMY_BLOCKS	Max. no. of dummy blocks with no traversing movements	
20252	CUTCOM_MAXNUM_SUPPR_BLOCKS	Maximum number of blocks with offset suppression (from SW 4)	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Reference
20256	CUTCOM_INTERS_POLY_ENABLE	Intersection process possible for polynomials (from SW 4)	
20270	CUTTING_EDGE_DEFAULT	Selected cutting edge after tool change	
20272	SUMCORR_DEFAULT	Number for activating a new cutting edge offset (SW 5 and higher)	
20360	TOOL_PARAMETER_DEF_MASK	Defines the effect of tool parameters	
20390	TOOL_TEMP_COMP_ON	Activation of temperature compensation for tool length (SW 6.1 and higher)	
20392	TOOL_TEMP_COMP_LIMIT	Maximum temperature compensation for tool length (SW 6.1 and higher)	
20610	ADD_MOVE_ACCEL_RESERVE	Acceleration reserve for overlaid movements	K1
21080	CUTCOM_PARALLEL_ORI_LIMIT	Limit angle between path tangent and tool orientation with 3D tool radius compensation	W5
22530	TOCARR_CHANGE_M_CODE	M code for change of toolholder	
22550	TOOL_CHANGE_MODE	New tool offsets with M function	
22560	TOOL_CHANGE_M_CODE	M function for tool change	
22562	TOOL_CHANGE_ERROR_MODE	Error response on programmed tool change (SW 5 and higher)	
28085	MM_LINK_TOA_UNIT	Assignment of TO unit to a channel	FBW, S7
32750	TEMP_COMP_TYPE	Temperature compensation type	K3

7.3 Setting data

Number	Identifier	Name	Reference
42442	TOOL_OFFSET_INCR_PROG	Retraction of tool offsets on incremental programming	
42470	CRIT_SPLINE_ANGLE	Limit angle for spline and polynomial interpolation and compressor	
42480	STOP_CUTCOM_STOPRE	Alarm reaction for tool radius compensation and preprocessing stop (from SW 4)	
42494	CUTCOM_ACT_DEACT_CTRL	Approach/retraction behavior for tool radius compensation in blocks without travel information (SW 5.2 and higher)	
42900	MIRROR_TOOL_LENGTH	Mirror tool length offset	
42910	MIRROR_TOOL_WEAR	Mirror wear values of tool length compensation	
42920	WEAR_SIGN_CUTPOS	Mirror wear values of machining plane	
42930	WEAR_SIGN	Invert sign of all wear values	
42935	WEAR_TRANSFORM	Transformation of wear data	
42940	TOOL_LENGTH_CONST	Retain the assignment of tool length components when changing the machining plane (G17 to G19)	

© Siemens AG, 2003. All rights reserved

Number	Identifier	Name	Refer-ence
42950	TOOL_LENGTH_TYPE	Assign tool length components independent of tool type (SW 5.3 and higher)	
42960	TOOL_TEMP_COMP	Temperature compensation value referred to tool (SW 6.1 and higher)	
42974	TCARR_FINE_CORRECTION	Fine offset TCARR on/off SW 6.4 and higher	

7.4 Alarms

Detailed explanations of the alarms which may occur are given in
References: /DA/, Diagnostics Guide
or in the online help in systems with MMC 101/102/103.

■

© Siemens AG, 2003. All rights reserved

Abbreviations

A

AC	Adaptive Control
ACKNLG	Acknowledge from printer
ADF	Autofeed printer
ASCII	American Standard Code for Information Interchange
ASUB	Asynchronous subprogram (see also interrupt routines)
AuxFu	Auxiliary function
Basic axis	An axis whose setpoint or actual value is used to calculate a compensation value.
BCD	Binary-coded decimal
BCS	Basic coordinate system
BIN	Binary Files
BOT	Boot files for SIMODRIVE 611D
BP	Basic program
BUSY	Busy from printer
CAM	Reference cam
CC	**C**ompile **C**ycles
Compensation axis	Axis whose setpoint or actual value is modified by the compensation value.
Compensation table	Table of interpolation points. It provides the compensation values on the compensation axis for selected positions on the basic axis.
Compensation value	The difference between the axis position measured by the encoder and the desired, programmmed position.

© Siemens AG, 2003. All rights reserved

COR	Coordinate rotation
CPA	**C**ompiler **P**rojecting Dat**a**
CPU	Central processing unit
CR	Carriage return
CRC	Cutter radius compensation
CTS	Clear to send for serial interfaces
CUTCOM	**Cut**ter radius **com**pensation
DAC	Digital-to-analog converter
DATA	Printer data bit x (x is an index from 0 to 7)
DB	Data block on the PLC
DBB	Data block byte on the PLC
DBW	Data block word on the PLC
DBX	Data block bit on the PLC
DC	Direct control: movement of the rotary axis across the shortest path to the absolute position within one revolution
DCD	Data carrier detect
DCE	Data communication equipment
DIO	Data input/output: data transfer display
DIR	Directory
DITE	Displacement thread end
DITS	Displacement thread start
DPM	Dual port memory
DPR	Dual port RAM
DRF	Differential resolver function

© Siemens AG, 2003. All rights reserved

DRY	Dry run feed(rate)
DSB	Decoding single block
DSC	Dynamic stiffness control
DSR	Data set ready
DSR	Data send ready on serial interfaces
DTE	Data terminal equipment
DTR	Data terminal ready
EIA code	Special punchtape code: number of punched holes per character always odd
ENC	Encoder (actual value encoder)
EPROM	Erasable programmable read-only memory with permanently stored program
ERROR	Error from printer
ESR	Extended stop and retract
FC	Function call on the PLC
FDD	Feed drive
FIFO	First in first out: memory which operates without addresses where the data are always read out in the same order in which they were stored.
FIPO	Fine interpolator
FOC	Force control
FST	Feed stop
GEO	Geometry
GIA	Gear interpolation data
GND	Signal ground
GRC	Grinding wheel radius compensation
GS	Gear stage

© Siemens AG, 2003. All rights reserved

GSC	Gear stage changeover
GUD	Global user data
GWPS	Grinding wheel peripheral (surface) speed
HASH	SW procedure for imaging a large quantity of names on a finite memory area
HEX	Hexadecimal number
HMI	Human Machine Interface: User interface on numerical control systems for operator control, programming and simulation
HW limit switch	Hardware limit switch
IC (GD)	Implicit communication (global data)
IKA	Interpolatory compensation
INC	Increment
INI	Initializing data
INIT	Initialize printer
Intermediate point	A position of the basic axis and the corresponding compensation value of the compensation axis
IPO	Interpolator
I/RF	Infeed/regenerative feedback unit of the SIMODRIVE 611(D)
IS	Interface signal
ISO code	Special punchtape code, number of punched holes per character always even
JobShop	Technology byte for ShopMill and ManualTurn controls
JOG	Jog mode
K_{Ue}	Transmission ratio
K_V	Servo gain factor
LEC	Leadscrew error compensation

© Siemens AG, 2003. All rights reserved

LED	Light emitting diode
LF	Line feed
LR	Position control (Lageregler)
LUD	Local user data
Machine axis	Physical axis on the machine tool
MCP	Machine control panel
MCS	Machine coordinate system
MD	Machine data
MDA	Manual data automatic: manual input
mm	Millimeter
MPF	Main program file: NC parts program (main program)
MSD	Main spindle drive
NC	Numerical control
NCK	Numerical Control Kernel: Numerical kernel with block preparation, traversing ranges, etc.
OB	Organization block on PLC
OP	Operator panel
OPI	Operator panel interface
OPT	Options
Path axes	In contrast to positioning axes, all path axes assigned to a channel are controlled by the interpolator such that they start, stop, accelerate and reach their end point simultaneously (one feedrate for all path axes).
PC	Personal computer
PE	Pulse enable for drive module
PE	Paper error
PG	Programming unit (Programmiergerät)

© Siemens AG, 2003. All rights reserved

PLC	Programmable logic controller
PMS1	Position measuring system 1
PMS2	Position measuring system 2
PRAL	Process alarm
PRT	Program test
PTP	Point-to-point
PUD	Program global user data
RAM	Random access memory, in which data can be read and written
REF	Reference point approach function
REPOS	Repositioning function
RI	Ring indicator
ROV	Rapid (traverse) override
RPA	R parameter active: memory area on the NCK for R parameter numbers
RPY	Roll pitch yaw (type of rotation of a coordinate system)
RTLI	Rapid traverse linear interpolation
RTS	Request to send (control signal on serial data interfaces)
RXD	Receive data
SBL	Single block
SEA	Setting data active: memory area for setting data on the NCK
SD	Setting data
SERUPRO	Search run via program test
SKP	Skip block
SLCT	Select from printer
SM	Stepper motor

© Siemens AG, 2003. All rights reserved

SPF	Subprogram file
STROBE	Data strobe to printer
SW limit switch	Software limit switch
SYF	System files
SYNACT	Synchronized action
TCP	Tool center position
TEA	Testing data active: refers to machine data
TLC	Tool length compensation
TO	Tool offset
TOA	Tool offset active: memory area for tool offsets
TOOLMAN	Tool management
TxD	Transmitted data
UFR	User frame: zero offset
V.24	Definition of the interchange circuit between DTE and DCE (RS 232-C)
V.28	Definition of the electrical response of the signals
VDI	Communication interface between PLC and NC
WCS	Workpiece coordinate system
WPD	Workpiece directory
Xy	Connector identifer (y is the index)
ZO	Zero offset
450	Type of modern interface controller
550	Type of modern interface controller
75188	Driver chip for serial interface

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Definition of Terms

B

A

Absolute coordinate
A coordinate specifying the destination of an axis movement relative to the origin of the currently active coordinate system. See also →Incremental coordinate.

Approach fixed machine point
Traversing motion towards one of the predefined →fixed machine points.

AUTOMATIC
Sequential block mode (DIN): A mode on NC systems in which a →parts program is executed continuously.

Auxiliary functions
Auxiliary functions can be used in parts programs to pass parameters to the PLC where they trigger reactions defined by the machine manufacturer.

Axis address
See →Axis identifier

Axis identifier
In compliance with DIN 66217, axes are identified with X, Y and Z for a right-handed, right-angled →coordinate system.
Rotary axes rotating around →X, Y and Z are identified with A, B and C. Additional axes operating in parallel to those specified can be identified with other letters.

Axis name
See →Axis identifier

B

Basic coordinate system
The programmer uses axis names of the basic coordinate system in the →parts program. The basic coordinate system exists in parallel to the →machine coordinate system when no →transformation is active. The two systems differ only in terms of the axis identifiers used.

Blank
The part used to start machining a workpiece.

© Siemens AG, 2003. All rights reserved

Block A section of a →parts program terminated with a line feed. Blocks are divided into →main blocks and →subblocks.

Block A block is a file which is required for program generation and program management.

C

C-axis An axis around which a controlled rotation and positioning movement is performed with the workpiece spindle.

Channel A channel can execute a →parts program independently of other channels. A channel controls only those axes and spindles which have been assigned to it. It is possible to coordinate execution of parts programs in different channels by means of →sychronization.

Circular interpolation The →tool is required to travel in a circle between defined points on the contour at a specified feedrate while machining the workpiece.

COM Component of the NC control for the implementation and coordination of communication.

Command channel The PLC program uses the command channel to transfer or trigger NC functions (e.g. external S, transformation).

Contour Outline of a →workpiece

Continuous-path mode The purpose of continuous-path mode is to prevent excessive braking of the →path axes at the parts program block limits and to allow the axes to change to the next block at as constant a velocity as possible.

Coordinate system See →Machine coordinate system, →Workpiece coordinate system

D

Data block A data unit on the →PLC which can be accessed by →HIGHSTEP programs. Data blocks contain data definitions. The data can be initialized directly during the definition.

Data word A data unit, two bytes in size, within a →data block.

© Siemens AG, 2003. All rights reserved

DRF

Differential resolver function: an NC function which generates an incremental zero offset in AUTOMATIC mode in conjunction with an electronic handwheel.

E

Exact stop

When an exact stop instruction is programmed, the position specified in a block is approached precisely and in some cases very slowly. In order to reduce the approach time, →exact stop limits are defined for rapid traverse and normal speed/velocity.

Exact stop limit

If all path axes have reached their exact stop limit, then the control responds as if it had reached a target point exactly. Processing of the next block in the →parts program commences.

F

Finished part contour

Contour of the finished workpiece. See also →Blank.

Fixed point approach

Machine tools can approach defined fixed points such as a tool change point, loading point, pallet change point, etc. The coordinates of these points are stored on the control. Where possible, the control system traverses these axes in →Rapid traverse.

Frame

A frame is a calculation rule that translates one Cartesian coordinate system into another Cartesian coordinate system. A frame contains the components →zero offset, →rotation, →scaling and →mirroring.

G

Gantry axes

Gantry axes consist of at least one machine axis pair, i.e. a→ master and a →synchronized (slave) axis which are mechanically coupled. Consequently, the NC always traverses them simultaneously. The difference in the actual positions is monitored at all times.

Gantry axis grouping

A gantry axis grouping is programmed (via machine data) to define which synchronized (slave) axes are to be controlled by which →master axis. The master axis and →synchronized axes cannot be traversed separately.

Geometry

Description of a →workpiece in the →workpiece coordinate system.

© Siemens AG, 2003. All rights reserved

Geometry axis Geometry axes are used to describe a 2 or 3-dimensional area in the workpiece coordinate system.

H

HIGHSTEP Combination of programming features for →PLCs of the AS300/AS400 system.

I

Imperial measurement system Measurement system which defines distances in “inches” and fractions of inches.

Increment Traversing distance calculated from the number of increments x increment length. The number of increments can be stored in a →setting data or selected with keys labelled for 10, 100, 1000, 10 000.

Incremental coordinate Specification of a destination of an axis movement through a distance and direction to be travelled in relation to a point already reached. See also →Absolute coordinate.

Initialization block Initialization blocks are special program blocks. They contain values which are assigned prior to execution of the program.
Initialization blocks are used primarily for initializing predefined data.

Interpolator Logical unit of the →NCK which determines intermediate values for the movements to be traversed on the individual axes on the basis of destination positions specified in the parts program.

J

JOG A mode on the control: a manual setup mode Manual operating mode that allows the operator to manually control the traversing movements of the axis in feed or in →rapid traverse.

K

Keywords Words written in a specific way which have a defined meaning in the programming language for →parts programs.

K_{Ue} Transmission ratio

© Siemens AG, 2003. All rights reserved

K_v Servo gain, magnitude in a closed-loop control system

L

Leadscrew error compensation Compensation for the mechanical inaccuracies of a leadscrew participating in the feed. The control uses stored deviation values for the compensation.

Limit speed Max. (spindle) speed: The maximum speed of a spindle can be limited by machine data settings, commands from the →PLC or by setting data.

Linear axis The linear axis is an axis which, in contrast to a rotary axis, describes a straight line.

Linear interpolation The tool must approach a target point along a straight line, machining the workpiece as it moves.

M

Machine axes Axes which exist physically on the machine tool.

Machine coordinate system Coordinate system referred to the axes of the →machine tool

Machine point fixed A point clearly defined by the machine tool; e.g. machine reference point.

Machine control panel Operator panel on a →machine tool with control elements such as keys, rotary switches, etc. and simple indicators such as LEDs. It is used to directly control the machine tool via the PLC.

Machine zero Fixed point on the machine tool which can be referenced by all (derived) measuring systems.

Macro A collection of instructions under a common identifier. The identifier in the program represents the collected sequence of instructions.

Main block A block prefixed by ":" containing all the parameters required to start execution of a →parts program.

Main program A parts program identified by a number or identifier in which other master routines, subroutines or cycles can be called.

© Siemens AG, 2003. All rights reserved

Master axis
The master axis is the →gantry axis which actually exists from the point of view of the operator and programmer and can therefore be controlled in the same ways as a normal NC axis.

MDA
A mode on the control: “manual data automatic”: manual input of blocks with execution.

Metric measurement system
Standardized system of units for lengths in millimeters (mm), meters (m), etc.

Mirroring
Mirroring exchanges the leading signs of the coordinate values of a contour in relation to an axis. Mirroring can be performed simultaneously in relation to several axes.

Mode group
Axes and spindles which are related in terms of the technical process can be combined to form a mode group. Axes/spindles belonging to one mode group can be controlled from one or several →channels. The same →operating mode is always assigned to the channels of the mode group.

N

NC
Numerical control. It incorporates all the components of the machine tool control system: →NCK, →PLC, →MMC, →COM.

NCK
Numerical control kernel: Component of the NC which processes →parts programs and basically coordinates the sequence of motions on the machine tool.

O

Offset memory
Data area in the control system containing the tool offset data.

Operating mode
An operating concept on a SINUMERIK control. The modes →JOG, →MDA, and →AUTOMATIC are defined.

Oriented spindle stop
Stops the workpiece spindle with a specified orientation angle, e.g. to perform an additional machining operation at a specific position. M function M19 is permanently assigned this function in compliance with DIN 66025.

Override
Manual control feature which enables the user to overstore programmed feedrates or speeds, in order to adapt them to a specific workpiece or material.

© Siemens AG, 2003. All rights reserved

P

Parts program

A sequence of instructions to the NC which combine to produce a specific →workpiece by performing machining operations on a →blank.

Path axis

Path axes are all the machining axes of the →channel which are controlled by the →interpolator such that they start, accelerate, stop and reach their end point simultaneously.

Path feed

The path feed is applied to →path axes. It represents the geometric total of the feedrates of the participating geometry axes.

PLC

Programmable logic controller: a component of the →NC: a control which can be programmed to control the logic on a machine tool.

Point-to-point time

The time taken to travel from the point of interruption on the contour with the spindle in motion back to the interruption point with the new tool and the spindle in motion.

Polar coordinates

A coordinate system which defines the position of a point in a plane in terms of its distance from the origin and the angle formed by the radius vector with a defined axis.

Positioning axis

An axis which performs an auxiliary movement on a machine tool (e.g. tool magazine, pallet transport). Positioning axes do not interpolate with the path axes.

Pre-coincidence

Block change occurs when the path has reached a defined delta distance from the end position.

Program block

Program blocks contain the main program and subprograms of the parts program.

Programmable frames

Programmable →frames can be used to define new coordinate system starting points dynamically during execution of the parts program. A distinction is made between absolute definition using a new frame and additive definition with reference to an existing starting point.

Programmable working area limitation

Limitation of the area in which the tool can move to a space defined by programmed limits.

© Siemens AG, 2003. All rights reserved

Programming key Characters and character sequences which have a defined meaning in the programming language for →parts programs.

Protection area Three-dimensional area within a →working area which the tool tip may not enter.

R

Rapid traverse Fastest traversing velocity of an axis. It is used, for example, to move the tool from its rest position to the →workpiece contour or to retract the tool from the contour. Rapid traverse is programmed via machine data on a machine-specific basis.

Reference point Point on the machine tool to which the measuring system of the →machine axes is referred.

Rotation Component of a →frame which defines a rotation of the coordinate system through a specific angle.

Rotary axis Rotary axes cause the workpiece or tool to rotate to a specified angle position.

Rounding axis Rounding axes cause the workpiece or tool to rotate to an angle position described on a graduated grid. When the grid position has been reached, the axis is “in position”.

R parameter Calculation parameter. The programmer of the parts programmer can assign or request the values of the R parameter as required.

S

Scaling Component of a →frame which causes axis-specific alterations in the scale.

Setting data Data which provide the control with information on properties of the machine tool in a way defined by the system software.

Soft key A key whose name appears on an area of the screen. The choice of soft keys displayed is adapted dynamically to the operating situation. The soft keys can be assigned to software functions by the user. The functions are displayed in the menu and change according to which menu is displayed.

© Siemens AG, 2003. All rights reserved

Subblock Block prefixed by “N” containing information for an operation such as, for example, a position parameter.

Subprogram A sequence of instructions of a →parts program which can be called repeatedly with different defining parameters. A →cycle is a type of subprogram.

Synchronized axis The synchronized axis is the →gantry axis whose setpoint position is always derived from the traversing motion of the →master axis, thus ensuring that they both traverse synchronously. From the point of view of the operator and programmer, the synchronized axis is “non-existent”.

Synchronous axes Synchronous axes require the same time for their distance as geometry axes for their path distance.

Synchronization Instructions in →parts programs for coordination of operations in different →channels at specific machining points.

System variable A variable which exists although it has not been programmed by the programmer of the →parts program. It is defined by the data type and the variable name, which is prefixed with $. See also →User-defined variable.

T

TOA area The TOA area contains all tool and magazine data. As standard, this area corresponds with the channel area with regards to the data range. However, it is possible to define via MD that several channels share a TOA unit so that these channels have available common tool management data.

TOA unit Each TOA area may contain several TOA units. The number of possible TOA units is limited by the maximum number of active channels. One TOA unit contains exactly one tool management data block and one magazine data block. From SW 4 and higher, an (optional) tool holder data block may be available.

Tool A part used on the machine tool for machining, e.g. cutting tools, mills, drills, laser beams, etc.

Tool nose radius compensation When a contour is programmed, it is assumed that a pointed tool is used. Since this is not feasible in practice, the control makes allowance for the curvature radius of the tool being used. The curvature centre point displaced by the curvature radius is guided equidistantly to the contour.

Tool offset Allowance for the tool dimensions when calculating the path.

© Siemens AG, 2003. All rights reserved

Tool radius compensation
In order to program a desired →workpiece contour directly, the control must traverse a path equidistant to the programmed contour with allowance for the radius.

Transformation
Additive or absolute zero offset on an axis.

U

User-defined variable
Users can define variables in the →parts program for their own use. A definition contains a data type specification and the variable name. See also →System variable.

V

Variable definition
A variable definition includes the specification of a data type and a variable name. The variable name can be used to address the value of the variable.

Velocity control
In order to achieve an acceptable traversing speed for traversing motions over very short distances, velocity control combined with Look Ahead over several blocks can be programmed.

W

Working area
Three-dimensional space into which the tool tip can travel on the basis of the machine tool design. See also →Protection area.

Workpiece
A part to be created/machined on the machine tool.

Workpiece contour
Setpoint contour of the →workpiece to be created/machined.

Workpiece coordinate system
The workpiece coordinate system has its origin in the →workpiece zero. When programming in the workpiece coordinate system, the dimensions and directions refer to this system.

Workpiece zero
The workpiece zero is the origin of the →workpiece coordinate system. It is defined by the distance to the machine zero.

© Siemens AG, 2003. All rights reserved

Z

Zero offset

Specification of a new reference point for a coordinate system by means of a reference to an existing zero point and a →frame.

■

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

References

C

General Documentation

/BU/ SINUMERIK & SIMODRIVE, Automation Systems for Machine Tools
Catalog NC 60
Order No.: E86060-K4460-A101-A9-7600

/IKPI/ Industrial Communication and Field Devices
Catalog IK PI
Order No.: E86060-K6710-A101-B2-7600

/ST7/ SIMATIC
Products for Totally Integrated Automation and Micro Automation
Catalog ST 70
Order No.: E86060-K4670-A111-A8-7600

/Z/ MOTION-CONNECT
Cable, Connectors & System Components for SIMATIC, SINUMERIK, Masterdrives and SIMOTION
Catalog NC Z
Order No.: E86060-K4490-A001-B1-7600

Electronic Documentation

/CD1/ The SINUMERIK System (11.03 Edition)
DOC ON CD
(includes all SINUMERIK 840D/840Di/810D/802D/802SC and SIMODRIVE publications)
Order No.: 6FC5298-6CA00-0BG4

© Siemens AG, 2003. All rights reserved

User Documentation

/AUK/ SINUMERIK 840D/810D (09.99 Edition)
Short Guide **AutoTurn Operation**
Order No.: 6FC5298-4AA30-0BP2

/AUP/ SINUMERIK 840D/810D (02.02 Edition)
Operator's Guide **AutoTurn Graphic Programming System**
Programming/Setup
Order No.: 6FC5298-4AA40-0BP3

/BA/ SINUMERIK 840D/810D (10.00 Edition)
Operator's Guide **MMC**
Order No.: 6FC5298-6AA00-0BP0

/BAD/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Operator's Guide **HMI Advanced**
Order No.: 6FC5298-6AF00-0BP2

/BAH/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Operator's Guide **HT 6**
Order No.: 6FC5298-0AD60-0BP2

/BAK/ SINUMERIK 840D/840Di/810D (02.01 Edition)
Short Guide **Operation**
Order No.: 6FC5298-6AA10-0BP0

/BAM/ SINUMERIK 840D/810D (08.02 Edition)
Operation/Programming ManualTurn
Order No.: 6FC5298-6AD00-0BP0

/BAS/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Operation/Programming ShopMill
Order No.: 6FC5298-6AD10-0BP1

/BAT/ SINUMERIK 840D/810D (06.03 Edition)
Operation/Programming ShopTurn
Order No.: 6FC5298-6AD50-0BP2

/BEM/ SINUMERIK 840D/810D (11.02 Edition)
Operator's Guide **HMI Embedded**
Order No.: 6FC5298-6AC00-0BP2

© Siemens AG, 2003. All rights reserved

/BNM/	SINUMERIK 840D/840Di/810D User's Guide **Measuring Cycles** Order No.: 6FC5298-6AA70-0BP2	(11.02 Edition)
/BTDI/	SINUMERIK 840D/840Di/810D Motion Control Information System (MCIS) User's Guide **Tool Data Information** Order No.: 6FC5297-6AE01-0BP0	(04.03 Edition)
/CAD/	SINUMERIK 840D/840Di/810D Operator's Guide **CAD Reader** Order No.: (included in online help)	(03.02 Edition)
/DA/	SINUMERIK 840D/840Di/810D **Diagnostics Guide** Order No.: 6FC5298-6AA20-0BP3	(11.02 Edition)
/KAM/	SINUMERIK 840D/810D Short Guide **ManualTurn** Order No.: 6FC5298-5AD40-0BP0	(04.01 Edition)
/KAS/	SINUMERIK 840D/810D Short Guide **ShopMill** Order No.: 6FC5298-5AD30-0BP0	(04.01 Edition)
/KAT/	SINUMERIK 840D/810D Short Guide **ShopTurn** Order No.: 6FC5298-6AF20-0BP0	(07.01 Edition)
/PG/	SINUMERIK 840D/840Di/810D Programming Guide **Fundamentals** Order No.: 6FC5298-6AB00-0BP2	(11.02 Edition)
/PGA/	SINUMERIK 840D/840Di/810D Programming Guide **Advanced** Order No.: 6FC5298-6AB10-0BP2	(11.02 Edition)
/PGK/	SINUMERIK 840D/840Di/810D Short Guide **Programming** Order No.: 6FC5298-6AB30-0BP0	(10.00 Edition)
/PGM/	SINUMERIK 840D/840Di/810D Programming Guide **ISO Milling** Order No.: 6FC5298-6AC20-0BP2	(11.02 Edition)

© Siemens AG, 2003. All rights reserved

/PGT/ SINUMERIK 840D/840Di/810D
Programming Guide **ISO Turning** (11.02 Edition)
Order No.: 6FC5298-6AC10-0BP2

/PGZ/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Programming Guide **Cycles**
Order No.: 6FC5298-6AB40-0BP2

/PI/ **PCIN 4.4**
Software for Data Transfer to/from MMC Module
Order No.: 6FX2060 4AA00-4XB0 (English, French, German)
Order from: WK Fürth

/SYI/ SINUMERIK 840Di (02.01 Edition)
System Overview
Order No.: 6FC5298-6AE40-0BP0

Manufacturer/Service Documentation

a) Lists

/LIS/ SINUMERIK 840D/840Di/810D (09.03 Edition)
SIMODRIVE 611D
Lists
Order No.: 6FC5297-6AB70-0BP4

b) Hardware

/ASAL/ SIMODRIVE (06.03 Edition)
Planning Guide **General Information for Asynchronous Motors**
Order No.: 6SN1197-0AC62-0BP0

/APH2/ SIMODRIVE (07.03 Edition)
Planning Guide **1PH2 Asynchronous Motors**
Order No.: 6SN1197-0AC63-0BP0

/APH4/ SIMODRIVE (07.03 Edition)
Planning Guide **1PH4 Asynchronous Motors**
Order No.: 6SN1197-0AC64-0BP0

/APH7/ SIMODRIVE (06.03 Edition)
Planning Guide **1PH7 Asynchronous Motors**
Order No.: 6SN1197-0AC65-0BP0

© Siemens AG, 2003. All rights reserved

/APL6/ SIMODRIVE (07.03 Edition)
Planning Guide **1PL6 Asynchronous Motors**
Order No.: 6SN1197-0AC66-0BP0

/BH/ SINUMERIK 840D/840Di/810D (09.03 Edition)
Operator Components Manual (HW)
Order No.: 6FC5297-6AA50-0BP3

/BHA/ SIMODRIVE **Sensor** (03.03 Edition)
User's Guide (HW) **Absolute Position Sensor with PROFIBUS DP**
Order No.: 6SN1197-0AB10-0YP2

/EMV/ SINUMERIK, SIROTEC, SIMODRIVE (06.99 Edition)
Planning Guide (HW) **EMC Installation Guide**
Order No.: 6FC5297-0AD30-0BP1

The current Declaration of Conformity is available under the following Internet address:
http://www4.ad.siemens.de

Please enter the ID No.: 15257461 in the 'Search' field (top right) and click on 'go'.

/GHA/ SINUMERIK/SIMOTION
ADI4 – Analog Drive Interface for 4 Axes (09.03 Edition)
Manual
Order No.: 6FC5297-0BA01-0BP1

/PFK6/ SIMODRIVE (05.03 Edition)
Planning Guide **1FK6 Three-Phase AC Servomotors**
Order No.: 6SN1197-0AD05-0BP0

/PFK7/ SIMODRIVE (01.03 Edition)
Planning Guide **1FK7 Three-Phase AC Servomotors**
Order No.: 6SN1197-0AD06-0BP0

/PFS6/ SIMODRIVE (07.03 Edition)
Planning Guide **1FS6 Three-Phase AC Servomotors**
Order No.: 6SN1197-0AD08-0BP0

/PFT5/ SIMODRIVE (05.03 Edition)
Planning Guide **1FT5 Three-Phase AC Servomotors**
Order No.: 6SN1197-0AD01-0BP0

/PFT6/ SIMODRIVE (05.03 Edition)
Planning Guide **1FT6 Three-Phase AC Servomotors**
Order No.: 6SN1197-0AD02-0BP0

© Siemens AG, 2003. All rights reserved

/PHC/ SINUMERIK 810D
Configuring Manual CCU (HW) (11.02 Edition)
Order No.: 6FC5297-6AD10-0BP1

/PHD/ SINUMERIK 840D (09.03 Edition)
Configuring Manual NCU (HW)
Order No.: 6FC5297-6AC10-0BP3

/PJAL/ SIMODRIVE 611 / Masterdrives MC (01.03 Edition)
Planning Guide **Three-Phase AC Servomotors**
General Part
Order No.: 6SN1197-0AD07-0BP0

/PJAS/ SIMODRIVE (07.03 Edition)
Planning Guide **Asynchronous Motors** (Compendium)
Order No.: 6SN1197-0AC61-0BP0

/PJFE/ SIMODRIVE (02.03 Edition)
Planning Guide **1FE1 Built-In Synchronous Motors**
Three-Phase AC Motors for Main Spindle Drives
Order No.: 6SN1197-0AC00-0BP4

/PJF1/ SIMODRIVE (12.02 Edition)
Installation Guide **1FE1 051.–1FE1 147. Built-In Synchronous Motors**
Three-Phase AC Motors for Main Spindle Drives
Order No.: 610.43000.02

/PJLM/ SIMODRIVE (06.02 Edition)
Planning Guide **1FN1, 1FN3 Linear Motors**
ALL General Information about Linear Motors
1FN1 1FN1 Three-Phase AC Linear Motor
1FN3 1FN3 Three-Phase AC Linear Motor
CON Connections
Order No.: 6SN1197-0AB70-0BP4

/PJM/ SIMODRIVE (11.00 Edition)
Planning Guide **Motors**
Three-Phase AC Servo Motors for Feed and Main Spindle Drives
Order No.: 6SN1197-0AC20-0BP0

/PJM2/ SIMODRIVE (07.03 Edition)
Planning Guide **Servomotors**
Three-Phase AC Motors for Feed and Main Spindle Drives
Order No.: 6SN1197-0AA20-0BP4

© Siemens AG, 2003. All rights reserved

/PJTM/ SIMODRIVE (05.03 Edition)
Planning Guide **1FW6 Integrated Torque Motors**
Order No.: 6SN1197-0AD00-0BP0

/PJU/ SIMODRIVE 611 (02.03 Edition)
Planning Guide **Inverters**
Order No.: 6SN1197-0AA00-0BP6

/PKTM/ SIMODRIVE (09.03 Edition)
Planning Guide **1FW3 Complete Torque Motors**
Order No.: 6SN1197-0AC70-0BP0

/PMH/ SIMODRIVE **Sensor** (07.02 Edition)
Configuring/Installation Guide (HW)
Hollow-Shaft Measuring System SIMAG H
Order No.: 6SN1197-0AB30-0BP1

/PMHS/ SIMODRIVE (12.00 Edition)
Installation Guide **Measuring System for Main Spindle Drives**
SIZAG2 Toothed-Wheel Encoder
Order No.: 6SN1197-0AB00-0YP3

/PMS/ SIMODRIVE (02.03 Edition)
Planning Guide **ECO Motor Spindle** for Main Spindle Drives
Order No.: 6SN1197-0AD04-0BP1

/PPH/ SIMODRIVE (12.01 Edition)
Planning Guide **1PH2, 1PH4, 1PH7 Motors**
AC Induction Motors for Main Spindle Drives
Order No.: 6SN1197-0AC60-0BP0

/PPM/ SIMODRIVE (11.01 Edition)
Planning Guide **Hollow-Shaft Motors**
for **1PM4** and **1PM6** Main Spindle Drives
Order No.: 6SN1197-0AD03-0BP0

c) Software

/FB1/ SINUMERIK 840D/840Di/810D (11.03 Edition)
Description of Functions **Basic Machine (Part 1)**
(the various sections are listed below)
Order No.: 6FC5297-6AC20-0BP3

A2	Various Interface Signals
A3	Axis Monitoring, Protection Zones
B1	Continuous-Path Mode, Exact Stop and Look Ahead
B2	Acceleration
D1	Diagnostic Tools

© Siemens AG, 2003. All rights reserved

D2	Interactive Programming
F1	Travel to Fixed Stop
G2	Velocities, Setpoint/Actual Value Systems, Closed-Loop Control
H2	Output of Auxiliary Functions to PLC
K1	Mode Group, Channel, Program Operation Mode
K2	Axes, Coordinate Systems, Frames, Actual Value System for Workpiece, External Zero Offset
K4	Communication
N2	EMERGENCY STOP
P1	Transverse Axes
P3	Basic PLC Program
R1	Reference Point Approach
S1	Spindles
V1	Feeds
W1	Tool Offset

/FB2/ SINUMERIK 840D/840Di/810D(CCU2) (11.02 Edition)
Description of Functions **Extended Functions (Part 2**)
including FM-NC: Turning, Stepper Motor
(the various sections are listed below)
Order No.: 6FC5297-6AC30-0BP2

A4	Digital and Analog NCK I/Os
B3	Several Operator Panels and NCUs
B4	Operation via PG/PC
F3	Remote Diagnostics
H1	JOG with/without Handwheel
K3	Compensations
K5	Mode Groups, Channels, Axis Replacement
L1	FM-NC Local Bus
M1	Kinematic Transformation
M5	Measurement
N3	Software Cams, Position Switching Signals
N4	Punching and Nibbling
P2	Positioning Axes
P5	Oscillation
R2	Rotary Axes
S3	Synchronous Spindles
S5	Synchronized Actions (up to and including SW 3)
S6	Stepper Motor Control
S7	Memory Configuration
T1	Indexing Axes
W3	Tool Change
W4	Grinding

/FB3/ SINUMERIK 840D/840Di/810D(CCU2) (11.02 Edition)
Description of Functions **Special Functions (Part 3)**
(the various sections are listed below)
Order No.: 6FC5297-6AC80-0BP2

F2	3-Axis to 5-Axis Transformation
G1	Gantry Axes
G3	Cycle Times
K6	Contour Tunnel Monitoring
M3	Coupled Motion and Leading Value Coupling
S8	Constant Workpiece Speed for Centerless Grinding
T3	Tangential Control

© Siemens AG, 2003. All rights reserved

TE0	Installation and Activation of Compile Cycles
TE1	Clearance Control
TE2	Analog Axis
TE3	Master-Slave for Drives
TE4	Transformation Package Handling
TE5	Setpoint Exchange
TE6	MCS Coupling
TE7	Retrace Support
TE8	Path-Synchronous Switch Signal
V2	Preprocessing
W5	3D Tool Radius Compensation

/FBA/ SIMODRIVE 611D/SINUMERIK 840D/810D (11.02 Edition)
Description of Functions **Drive Functions**
(the various sections are listed below)
Order No.: 6SN1197-0AA80-1BP0

DB1	Operational Messages/Alarm Reactions
DD1	Diagnostic Functions
DD2	Speed Control Loop
DE1	Extended Drive Functions
DF1	Enable Commands
DG1	Encoder Parameterization
DL1	Linear Motor MD
DM1	Calculation of Motor/Power Section Parameters and Controller Data
DS1	Current Control Loop
DÜ1	Monitors/Limitations

/FBAN/ SINUMERIK 840D/SIMODRIVE 611 digital (02.00 Edition)
Description of Functions **ANA MODULE**
Order No.: 6SN1197-0AB80-0BP0

/FBD/ SINUMERIK 840D (07.99 Edition)
Description of Functions **Digitizing**
Order No.: 6FC5297-4AC50-0BP0

DI1	Start-up
DI2	Scanning with Tactile Sensors (scancad scan)
DI3	Scanning with Lasers (scancad laser)
DI4	Milling Program Generation (scancad mill)

/FBDM/ SINUMERIK 840D/840Di/810D (09.03 Edition)
Motion Control Information System (MCIS)
Description of Functions **DNC NC Program Management**
Order No.: 6FC5297-1AE81-0BP0

/FBDN/ SINUMERIK 840D/840Di/810D (03.03 Edition)
Motion Control Information System (MCIS)
Description of Functions **DNC NC Program Management**
Order No.: 6FC5297-1AE80-0BP0

DN1	DNC Plant / DNC Cell
DN2	DNC IFC SINUMERIK, NC Data Transfer via Network

© Siemens AG, 2003. All rights reserved

/FBFA/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Description of Functions **ISO Dialects for SINUMERIK**
Order No.: 6FC5297-6AE10-0BP3

/FBFE/ SINUMERIK 840D/840Di/810D (04.03 Edition)
Description of Functions **Remote Diagnosis**
Order No.: 6FC5297-0AF00-0BP2

FE1	Remote Diagnosis ReachOut
FE3	Remote Diagnosis pcAnywhere

/FBH/ SINUMERIK 840D/840Di/810D (11.02 Edition)
HMI Configuring Package
Order No.: (supplied with the software)

Part 1	User's Guide
Part 2	Description of Functions

/FBH1/ SINUMERIK 840D/840Di/810D (03.03 Edition)
HMI Configuring Package
ProTool/Pro Option SINUMERIK
Order No.: (supplied with the software)

/FBHLA/ SINUMERIK 840D/SIMODRIVE 611 digital (09.03 Edition)
Description of Functions **HLA Module**
Order No.: 6SN1197-0AB60-0BP4

/FBIC/ SINUMERIK 840D/840Di/810D (06.03 Edition)
Motion Control Information System (MCIS)
Description of Functions **TDI Ident Connection**
Order No.: 6FC5297-1AE60-0BP0

/FBMA/ SINUMERIK 840D/810D (08.02 Edition)
Description of Functions **ManualTurn**
Order No.: 6FC5297-6AD50-0BP0

/FBO/ SINUMERIK 840D/810D (09.01 Edition)
Description of Functions **Configuring OP 030 Operator Interface**
Order No.: 6FC5297-6AC40-0BP0

BA	Operator's Guide
EU	Development Environment (Configuring Package)
PS	Online only: Configuring Syntax (Configuring Package)
PSE	Introduction to Configuring of Operator Interface
IK	Screen Kit: Software Update and Configuration

/FBP/ SINUMERIK 840D (03.96 Edition)
Description of Functions **C-PLC Programming**
Order No.: 6FC5297-3AB60-0BP0

© Siemens AG, 2003. All rights reserved

/FBR/ SINUMERIK 840D/810D (09.01 Edition)
IT Solutions
Description of Functions **Computer Link** (SinCOM)
Order No.: 6FC5297-6AD60-0BP0

NFL Host Computer Interface
NPL PLC/NCK Interface

/FBSI/ SINUMERIK 840D / SIMODRIVE 611 digital (09.03 Edition)
Description of Functions **SINUMERIK Safety Integrated**
Order No.: 6FC5297-6AB80-0BP2

/FBSP/ SINUMERIK 840D/840Di/810D (08.03 Edition)
Description of Functions **ShopMill**
Order No.: 6FC5297-6AD80-0BP1

/FBST/ SIMATIC (01.01 Edition)
Description of Functions **FM STEPDRIVE/SIMOSTEP**
Order No.: 6SN1197-0AA70-0YP4

/FBSY/ SINUMERIK 840D/840Di/810D (10.02 Edition)
Description of Functions **Synchronized Actions**
Order No.: 6FC5297-6AD40-0BP2

/FBT/ SINUMERIK 840D/810D (06.03 Edition)
Description of Functions **ShopTurn**
Order No.: 6FC5297-6AD70-0BP2

/FBTC/ SINUMERIK 840D/810D (01.02 Edition)
IT Solutions
SINUMERIK Tool Data Communication SinTDC
Description of Functions
Order No.: 6FC5297-5AF30-0BP0

/FBTD/ SINUMERIK 840D/810D (02.01 Edition)
IT Solutions
Tool Information System (SinTDI) with Online Help
Description of Functions
Order No.: 6FC5297-6AE00-0BP0

/FBTP/ SINUMERIK 840D/840Di/810D (01.03 Edition)
Motion Control Information System (MCIS)
Description of Functions **TPM Total Productive Maintenance**, Version 3.0
Order No.: Document is supplied with the software

© Siemens AG, 2003. All rights reserved

/FBU/ **SIMODRIVE 611 universal/universal E** (07.03 Edition)
Closed-Loop Control Component for Speed Control and Positioning
Description of Functions
Order No.: 6SN1197-0AB20-0BP7

/FBU2/ **SIMODRIVE 611 universal** (04.02 Edition)
Installation Guide (enclosed with SIMODRIVE 611 universal)

/FBW/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Description of Functions **Tool Management**
Order No.: 6FC5297-6AC60-0BP1

/HBA/ SINUMERIK 840D/840Di/810D (03.02 Edition)
Manual @Event
Order No.: 6AU1900-0CL20-0BA0

/HBI/ SINUMERIK 840Di (09.03 Edition)
Manual
Order No.: 6FC5297-6AE60-0BP2

/INC/ SINUMERIK 840D/840Di/810D (06.03 Edition)
System Description Commissioning Tool **SINUMERIK SinuCOM NC**
Order No.: (an integral part of the online help for the start-up tool)

/PJE/ SINUMERIK 840D/810D (08.01 Edition)
Description of Functions **Configuring Package HMI Embedded**
Software Update, Configuration Installation
Order No.: 6FC5297-6EA10-0BP0
(the document PS Configuring Syntax is supplied with the software and available as a pdf file)

/POS1/ SIMODRIVE **POSMO A** (05.03 Edition)
User's Guide **Distributed Positioning Motor on PROFIBUS DP**
Order No.: 6SN2197-0AA00-0BP5

/POS2/ SIMODRIVE **POSMO A** (04.02 Edition)
Installation Guide (enclosed with POSMO A)

/POS3/ SIMODRIVE **POSMO SI/CD/CA** (07.03 Edition)
User's Guide **Distributed Servo Drive Systems**
Order No.: 6SN2197-0AA20-0BP4

/POS4/ SIMODRIVE **POSMO SI** (04.02 Edition)
Installation Guide (enclosed with POSMO SI)

© Siemens AG, 2003. All rights reserved

/POS5/ SIMODRIVE **POSMO CD/CA** (04.02 Edition)
Installation Guide (enclosed with POSMO CD/CA)

/S7H/ SIMATIC S7–300 (2002 Edition)
– Manual: **CPU Data** (Hardware)
– Reference Manual: **Module Data**
– Manual: **Technological Functions**
– Installation Manual
Order No.: 6ES7398-8FA10-8BA0

/S7HT/ SIMATIC S7–300 (03.97 Edition)
Manual: STEP 7, **Fundamentals**, V. 3.1
Order No.: 6ES7 10-4CA02-8BA0

/S7HR/ SIMATIC S7–300 (03.97 Edition)
Manual: STEP 7, **Reference Manuals**, V. 3.1
Order No.: 6ES7810-4CA02-8BR0

/S7S/ SIMATIC S7–300 (04.02 Edition)
FM 353 Positioning Module for Stepper Drive
Order together with configuring package

/S7L/ SIMATIC S7–300 (04.02 Edition)
FM 354 Positioning Module for Servo Drive
Order together with configuring package

/S7M/ SIMATIC S7–300 (01.03 Edition)
FM 357-2 Multimodule for Servo and Stepper Drives
Order together with configuring package

/SP/ SIMODRIVE **611–A/611–D**
SimoPro 3.1
Program for Configuring Machine-Tool Drives
Order No.: 6SC6111-6PC00-0BA□, Order from: WK Fürth

© Siemens AG, 2003. All rights reserved

d) Installation and Start-Up

/BS/ SIMODRIVE 611 analog (10.00 Edition)
Description **Start-Up Software for Main Spindle and Asynchronous Motor Modules**
Version 3.20
Order No.: 6SN1197-0AA30-0BP1

/IAA/ SIMODRIVE 611A (10.00 Edition)
Installation and Start-Up Guide
Order No.: 6SN1197-0AA60-0BP6

/IAC/ SINUMERIK 810D (11.02 Edition)
Installation and Start-Up Guide
(incl. description of SIMODRIVE 611D start-up software)
Order No.: 6FC5297-6AD20-0BP0

/IAD/ SINUMERIK 840D/SIMODRIVE 611 digital (11.02 Edition)
Installation and Start-Up Guide
(incl. description of SIMODRIVE 611 digital start-up software)
Order No.: 6FC5297-6AB10-0BP2

/IAM/ SINUMERIK 840D/840Di/810D (09.03 Edition)
Installation and Start-Up Guide HMI/MMC
Order No.: 6FC5297-6AE20-0BP3

AE1	Updates/Supplements
BE1	Expanding the Operator Interface
HE1	Online Help
IM2	Starting up HMI Embedded
IM4	Starting up HMI Advanced
TX1	Creating Foreign Language Texts

© Siemens AG, 2003. All rights reserved

Index

Symbols

"Selfacting Serupro", 1/K1/2-46

$A_DNO[n], 1/W1/2-20

Numbers

34104|REFP_PERMITTED_IN_FOLLOWUP, 1/R1/4-57

611D Ready, 1/A2/2-7, 1/A2/5-69

A

A_COORDINATE_SYSTEM, 1/K2/4-86

Abbreviations, A-1

Abs. Absolute compensation value meas. system 1 or 2, 1/D1/2-9

ABSBLOCK_FUNCTION_MASK, 1/K1/4-164

Absolute dimension tool length compensation geometry axis 1, 1/K2/4-85

Absolute dimension tool length compensation geometry axis 2, 1/K2/4-85

Absolute dimension tool length compensation geometry axis 3, 1/K2/4-85

Acceleration, 1/B2/1-3
- Knee-shaped acceleration characteristic, 1/B2/2-15
- Programmable, 1/B2/2-17

Acceleration (B2), 1/B2/1-3
- Alarms, 1/B2/7-34
- Axis-specific machine data, 1/B2/4-23
- Channel-specific machine data, 1/B2/4-28
- Channel-specific setting data, 1/B2/4-29
- Machine data, 1/B2/7-31
- Setting data, 1/B2/7-33
- System variables, 1/B2/7-34

Acceleration limitation, 1/B2/2-11

Acceleration profiles, 1/B2/2-5
- Abrupt acceleration changes, 1/B2/2-5
- Acceleration with jerk limitation, 1/B2/2-5

Access authorization, 1/A2/2-10

Access features, 1/A2/2-10

Access protection, 1/A2/2-10

Action single block, 1/K1/2-22

Activation and priorities, 1/K1/2-17

Activation of block skip, 1/A2/4-58

Activation of protection zones, Example, 1/A3/6-80

Active, 1/A2/5-81

Active data (frames) are immediately operative after editing, 1/K2/4-86

Active drive parameter set A, B, C, 1/A2/5-101

Active meas. system, 1/D1/2-9

Active motor, 1/A2/2-34

Active motor A, B, 1/A2/5-102

Active or passive file system, 1/A2/5-76

ACTIVE_SEL_USER_DATA, 1/K2/4-86

Actual motor, 1/D1/2-21

Actual parameter set, 1/D1/2-20

Actual position value measuring system, 1/D1/2-9

Actual speed = set speed, 1/D1/2-23

Actual value synchronization, 1/A2/2-27

Actual value correction, 1/G2/2-24

Actual value in WCS, 1/A2/5-72

Actual value in workpiece coordinate system, 1/A2/2-16

Actual value processing, 1/G2/2-34, 1/G2/2-47, 1/G2/2-50, 1/G2/2-56, 1/G2/2-60, 1/G2/2-61

Actual value resolution, 1/G2/2-36

Actual value routing, 1/G2/2-25

Actual value sensing, 1/G2/2-23

Actual value system 12101,12102,12103,12104,12105,12106, 1/G2/2-23
- Workpiece-related, 1/K2/2-77

Actual values, 1/A3/2-20

Adaptation, Dynamic performance, 1/B1/1-3

Adapting active frames, 1/K2/2-56

Adapting the motor/load ratios, 1/G2/2-31

ADDFRAME, 1/K2/2-70

Additive frame in frame chain, 1/K2/2-70

Air temperature alarm, 1/A2/2-8, 1/A2/5-70

Alarm, Delete criterion, 1/D1/2-6

© Siemens AG, 2003. All rights reserved

Alarm area selection, 1/A2/5-73

Alarm for preprocessing stop, 1/W1/2-66

ALARM_PAR_DISPLAY_TEXT, 1/D1/4-37

ALARM_REACTION_CHAN_NOREADY, 1/D1/4-37

Alarms, 1/A2/7-121

Analog spindle 1, utilization in percent, 1/A2/5-73

Analog spindle 2, utilization in percent, 1/A2/5-73

Applicability of MD 9203 to edge data and location-dependent offsets, 1/K2/4-86

ASCII editor, 1/D2/2-6

Assigned timers, 1/P3/6-253

Assignment of DBs, 1/P3/6-251

Assignment of FBs and FCs, 1/P3/6-251

Assignment tool/toolholder, 1/W1/2-72

ASUBs, 1/P3/2-55

Asynchronous events, 1/A2/1-3

Asynchronous subroutines (ASUBs), 1/K1/2-86

ATRANS, 1/K2/1-9, 1/K2/2-43

AUTOMATIC, 1/K1/2-16

Automatic mode, 1/K1/2-128

Automatic start of an ASUB following block search (SW 6.1 and higher, 1/K1/2-31

Auxiliary function output, 1/B1/2-15
  Detailed Description, 1/H2/2-5
  Overstore, 1/H2/2-30
  POWER ON, block search, 1/H2/2-29

Auxiliary functions, 1/H2/1-3

Availability
  Helix interpolation, 1/K2/3-81
  Screw interpolation, 1/K2/3-81
  Workpiece-related actualvalue system, 1/K2/3-82
  Zero offset, 1/K2/3-82

Availability of basic offset in Machine operating area, 1/K2/4-84

Availability of basic offset in Parameters operating area, 1/K2/4-84

AX_JERK_MODE, 1/B2/4-24

Axis, 1/A2/2-19

Axis configuration, 1/K2/2-23

Axis monitoring functions, 1/A3/1-3
  Actual velocity, 1/A3/2-19
  Axis/spindle-specific interface signals, 1/A3/7-81
  Axis/spindle-specific machine data, 1/A3/3-58, 1/A3/7-83
  Axis/spindle-specific setting data, 1/A3/7-85
  Axis/spindle-specific signals, 1/A3/5-69
  Channel-specific machine data, 1/A3/7-85
  Clamping, 1/A3/2-7, 1/A3/2-11
  Contour, 1/A3/2-5
  Following error, 1/A3/2-6
  Positioning monitoring., 1/A3/2-7
  Setting data, 1/A3/3-66
  Speed setpoint, 1/A3/2-17
  Supplementary conditions, 1/A3/3-55
  Zero speed, 1/A3/2-7, 1/A3/2-10

Axis monitoring, protection zones (A3), Channel-specific machine data, 1/A3/3-57

Axis parameter sets, 1/G2/2-46

Axis types, 1/K2/2-13

Axis/spindle disable, 1/A2/5-85

Axis/spindle lock DB31, ... DBX1.3, 1/V1/2-27

Axis/spindle service display, 1/D1/2-8

Axis/spindle stops, 1/A2/2-30, 1/A2/5-98

Axis/spindle-specific signals, 1/A2/5-85

Axis-related jerk limitation, 1/B2/2-12
  Jerk limitation, 1/B2/2-12

Axis-specific machine data, 1/K2/4-99

Axis-specific signals, 1/K1/5-192

B

Basic block display, 1/K1/4-164

Basic coordinate system (BCS), 1/K2/1-7, 1/K2/2-35

Basic setting for Program directory, 1/A2/4-49

Basic system, 1/G2/2-13

Basic tool orientation, 1/W1/6-156

Baud rate, 1/K4/3-21

© Siemens AG, 2003. All rights reserved

Block, 1/K1/2-75
Program operation mode, 1/K1/2-22
Search, 1/K1/2-26
Single block suppression, 1/K1/2-98, 1/K1/2-100

Block change, 1/B1/2-8, 1/B1/2-10

Block change point, 1/B1/2-13

Block search, 1/K1/1-5, 1/K1/1-6

Block search via program test, 1/K1/2-36

Block SPARPI, 1/K1/2-44

Block-related limit (FOC), 1/F1/2-23

Blockwise feed FB, 1/V1/2-42

BRISK, 1/B2/2-8

Bus addresses, 1/K4/2-13

Bus addresses on SINUMERIK 840D, 1/P3/2-46

Bus addresses on SINUMERIK FM-NC, 1/P3/2-47

C

Calculation resolution, 1/G2/2-9

Calipers, 1/V1/2-38

Calling the ASUB outside program operation, 1/K1/2-90

Cancel alarm cleared, 1/A2/5-77

Cancel alarms, 1/A2/2-16

Carrying out SERUPRO in a group of channels, 1/K1/2-44

Cascaded block search (SW 6.1 and higher), 1/K1/2-31

CFC, 1/V1/2-8

CFCIN, 1/V1/2-8

CFINE, 1/K2/1-9, 1/K2/2-43

CFTCP, 1/V1/2-8

Channel, 1/B1/2-6, 1/K1/1-5, 1/K1/2-13
Channel configuration, 1/K1/2-13
Channel machine data, 1/K1/2-129
Channel status, 1/K1/2-75
Gaps, 1/K1/2-9, 1/K1/2-11
VDI signal, 1/K1/2-129

Channel axes, 1/K2/2-15

Channel axis identifiers with FXS, 1/F1/2-7

Channel number (FC9: ChanNo), 1/A2/5-82

Channel number of machine control panel on MMC, 1/A2/5-73

Channel-specific NCK alarm is active, 1/A2/2-8

Channel-specific signals, 1/A2/5-83

Clear cancel alarms, 1/A2/5-72

Clear recall alarms, 1/A2/5-72

Closed-Loop Control, 1/G2/2-43

Collision detection, 1/W1/2-60

COM1, 1/A2/5-74, 1/A2/5-79

COM1 or COM2, 1/K4/2-6

COM2, 1/A2/5-74, 1/A2/5-79

Communication
MMC 100, 1/K4/2-6
MMC 100/101/102, 1/K4/1-3
MMC 101/102, 1/K4/1-3

Communication log, 1/D1/2-28

Communication with global data (GD), 1/P3/2-33

Commutation limit, 1/B2/2-15

Concurrent axes, 1/P3/2-54

Configurability of machine control panel, hand-held unit, 1/P3/2-66

Configurable gear adaption, 1/S1/2-28

Constant cutting rate (G96), 1/V1/2-11

Constant speed G9, 1/V1/2-11

Container axes, 1/K1/2-14

Contamination signal, 1/A3/2-23

Continuous-path mode, 1/B1/2-13

Continuous-path mode, exact stop and LookAhead (B1)
Alarms, 1/B1/7-66
Axis-specific machine data, 1/B1/4-56
Axis-specific signals, 1/B1/5-61
Channel-specific machine data, 1/B1/4-50
Channel-specific signals, 1/B1/5-61
General machine data, 1/B1/4-49
Interface signals, 1/B1/7-63
Machine data, 1/B1/7-64

Contour deviation, 1/D1/2-9

Contour error, 1/A3/2-5

Contour violation, 1/A3/2-28, 1/A3/2-31

Contrast, 1/A2/2-35

Control, 1/B2/2-15
Normal power-up, 1/K1/2-127

© Siemens AG, 2003. All rights reserved

Control deviation, 1/D1/2-9

Control direction, 1/G2/2-34

Control power-up, 1/K1/2-127
  With default values, 1/K1/2-127

Controller mode, 1/D1/2-11

Controller parameter set switchover, 1/A2/6-113

Controller parameter set switchover (checkback signal), 1/A2/5-99

Controller parameter set switchover (request), 1/A2/5-91

Conversion of basic system, 1/G2/2-13

Coordinate system, 1/A3/2-36

Coordinate systems, 1/K2/1-5, 1/K2/1-6, 1/K2/1-7, 1/K2/2-32

Counting
  Algorithm, 1/K1/2-129
  Pulse, 1/K1/2-129

Coupling two FM-NCs with one PLC, Coupling two FM-NCs with one PLC, 1/P3/8-268

Coupling two FM-NC Modules to one PLC, 1/P3/8-267
  Basic program error messages, 1/P3/8-281
  Coupling with the components MMC/MSTT/OP, 1/P3/8-270
  Function of the basic program for two FM-NCs, 1/P3/8-275
  Interface structure, 1/P3/8-271
  Memory space, 1/P3/8-281
  PLC/MCP/HHU interface, 1/P3/8-273
  PLC/MMC interface, 1/P3/8-273
  PLC/NCK interface, 1/P3/8-271
  Special features of blocks when using two FM-NCs, 1/P3/8-276
    FB1: RUN_UP, 1/P3/8-276
    FB2...5: Read/write NC variables, PI services, GUD variables, 1/P3/8-278
    FC 15, 16, 18: POS_AX, PART_AX, SpinCtrl, 1/P3/8-279
    FC 9: ASUB, 1/P3/8-280

Couplings, 1/K1/2-51

CRC error, 1/D1/2-18

CTRANS, 1/K2/1-9, 1/K2/2-43

Current controller active, 1/A2/2-31, 1/A2/5-99

Current gear stage, 1/D1/2-11

Cutter material memory, 1/D2/2-10

Cutting edge number, 1/W1/2-10

Cutting edge position, relevant, 1/W1/2-36

Cyclic operation, 1/P3/2-50

Cyclical communication, 1/K4/2-15

Cyclical signal exchange, 1/A2/2-5, 1/P3/1-5

D

D functions, 1/W1/2-5

D number structure, – flat (without tool management), 1/W1/2-19

D numbers, Allocation of free ..., 1/W1/2-10

Darken screen, 1/A2/2-15, 1/A2/5-71

Data backup, 1/D1/2-6

Data channel, high-speed, 1/A2/2-41

Data read-off, 1/D1/2-13

Data sets, 1/D2/2-19

DC link status, 1/D1/2-18

DC link voltage, 1/B2/2-15

Decoding single block, 1/K1/2-22

Decoding single block SBL2 with implicit preprocessing stop, 1/K1/2-97

Default passwords, 1/A2/2-11

Default value for path feed, 1/V1/4-63

DEFAULT_FEED, 1/V1/4-63

Define number of skippable program levels, 1/K1/4-132

Defining the reference for the path override, 1/V1/4-57

Definition of Terms, B-9

Delete distance-to-go, 1/A2/2-19

Delete distance-to-go (axis-specific)/Spindle reset, 1/A2/5-91

Delete distance-to-go (channel-specific), 1/A2/5-83

Diagnostic tools
  General machine data, 1/D1/4-35
  Machine data, 1/D1/7-42

Diagnostic tools (D1), 1/D1/1-3
  Alarms, 1/D1/2-5
  External, 1/D1/1-3
  Integrated, 1/D1/1-3
  Interface signals, 1/D1/7-41

Diagnostics, 1/D1/1-3, 1/D1/2-8
  611D start-up tool, 1/D1/2-31
  Archiving of data, 1/D1/2-31
  For alarms, 1/D1/2-14, 1/D1/2-24
  Of operational state errors, 1/D1/2-15

© Siemens AG, 2003. All rights reserved

Diagnostics buffer on PLC, 1/P3/2-33

DIAMOF, 1/P1/1-3, 1/P1/2-6

DIAMON, 1/P1/1-3, 1/P1/2-6

Direction of offset of tool for geometry axes with scratching, 1/K2/4-86

DISC, 1/W1/2-55

Display geo axes of channel first, 1/K1/4-131

Display mode of zero offset, 1/K2/4-87

Display of orientation axes as Euler angle, 1/K2/4-84

Display resolution, 1/A2/2-36, 1/G2/2-9

Displayed channel number of MMC, 1/A2/5-78

DL auxiliary function for sum offset, 1/H2/2-11

Drive configuration, 1/G2/2-30

Drive enable, 1/D1/2-17

Drive parameter set selection, 1/A2/2-32

Drive parameter set selection A, B, C, 1/A2/5-94

DRIVE ready, 1/A2/2-34, 1/A2/5-103

Drive service display, 1/D1/2-15

Drive status, 1/D1/2-18

DRIVE_AX_RATIO_DENOM, 1/S1/4-76

DRIVE_AX_RATIO2_NUMERA, 1/S1/4-77

Dry run feed, 1/K1/2-24, 1/V1/2-33

Dry run feedrate (DryRun), 1/S1/2-62

DRY_RUN_FEED_MODE, MD 42101, 1/V1/4-63

DRYRUN_MASK, 1/V1/2-34

Dynamic response adaptation, 1/G2/2-45

E

EMERGENCY STOP, 1/N2/1-3, 1/N2/2-5
- Acknowledgment, 1/N2/2-8
- PLC/NCK, 1/N2/2-6
- Sequence, 1/N2/2-7

Enable pulses, 1/D1/2-18

ENABLE_ALARM_MASK, 1/D1/4-37

ENABLE_CHAN_AX_GAP, 1/K2/4-92

ENC_IS_DIRECT2, 1/S1/4-76

Encoder coding, 1/G2/2-37

Encoder monitoring functions, 1/A3/2-21
- Encoder frequency, 1/A3/2-21
- Zero marks, 1/A3/2-22

Error, 1/A2/5-78, 1/A2/5-81

Error RS232C, 1/A2/5-80

Event-controlled program sequences, 1/K1/2-80

Event-driven signal exchange, 1/P3/1-5, 1/P3/1-6

Exact stop, 1/B1/2-9

Exact stop criteria, 1/B1/2-10
- Exact stop coarse, 1/A3/2-8, 1/B1/2-11
- Exact stop fine, 1/A3/2-9, 1/A3/2-10, 1/B1/2-11

Exact stop criterion, 1/B1/2-13

Exact stop limit fine, 1/A3/2-8

EXACT_POS_MODE, MD 20550, 1/B1/4-54

Example of auxiliary function output, 1/H2/2-20

Execute external subroutine, 1/K1/2-109, 1/K1/2-110

Execution from external source, extension (SW 4 and higher), 1/K1/2-109

EXPONENT_LIMIT, 1/A2/4-52

EXPONENT_SCIENCE, 1/A2/4-52

EXTCALL, 1/K1/2-110

External data communications connector, 1/K4/1-3

External protection zone, 1/A3/2-38

External subroutine, 1/K1/2-110

F

FB 10: Safety relay, 1/P3/4-152

FB 11: Brake test, 1/P3/4-155

FB 2: GET, 1/P3/4-102

FB 29: Signal recorder and data trigger diagnostics, 1/P3/4-159

FB 3: PUT, 1/P3/4-109

FB 4: PI_SERV General PI services, 1/P3/4-115

FB 5: GETGUD Read GUD variable, 1/P3/4-138

FB 7: PI_SERV2 General PI services, 1/P3/4-143

FB 9 M:N control unit switchover, 1/P3/4-147

FB 9: Operating unit switchover, 1/P3/4-89

FC 10: AL_MSG, 1/P3/4-182

FC 13: BHGDisp, 1/P3/4-185

FC 15: POS_AX, 1/P3/4-188

FC 16: PART_AX, 1/P3/4-193

© Siemens AG, 2003. All rights reserved

FC 17: YDelta, 1/P3/4-197
FC 18: SpinCtrl, 1/P3/4-200
FC 19: MCP_IFM, 1/P3/4-209
FC 2: GP_HP, 1/P3/4-163
FC 22: TM_DIR, 1/P3/4-224
FC 24: MCP_IFM2, 1/P3/4-226
FC 25: MCP_IFT, 1/P3/4-230
FC 3: GP_PRAL, 1/P3/4-164
FC 5: GP_DIAG, 1/P3/4-167
FC 7: TM_REV, 1/P3/4-168
FC 8: TM_TRANS, 1/P3/4-171
FC 9: ASUB, 1/P3/4-179
FC9: Start (measuring in Jog mode), 1/A2/5-82
Feed disable, 1/V1/2-26
Feed disable DB21, ... DBX6.0, 1/V1/2-26
Feed FB, Blockwise, 1/V1/2-42
Feed for chamfer/rounding, 1/V1/2-40
Feed override, 1/B1/2-7, 1/R1/2-15
Feed override switch, 1/V1/2-27
Feed stop, 1/K1/2-75
Feed/spindle stop, 1/V1/2-26
Feedrate, 1/B1/2-7, 1/R1/2-15
Feedrates
  Dry run feed, 1/V1/2-33
  Feed control, 1/V1/2-26
  Feed disable, 1/V1/2-26
  Feed override, 1/V1/2-27, 1/V1/2-32
  Feed/spindle stop, 1/V1/2-26
  Feeds in one block, 1/V1/2-34
  G93, G94, G95, 1/V1/2-9
  Path feed F, 1/V1/2-7
  Positioning axes FA, 1/V1/2-24
  Positioning feed G94, 1/V1/2-25
  Rigid tapping G331/G332, 1/V1/2-23
  Spindle override, 1/V1/2-30
  Tapping with compensation chuck G63, 1/V1/2-23
  Thread cutting G33, 1/V1/2-14
FGROUP, 1/K2/2-19, 1/K2/2-22
Field weakening, 1/B2/2-15
File types
  BIN, 1/K4/2-6
  BOT, 1/K4/2-6
  COM, 1/K4/2-6
  CPA, 1/K4/2-6
  DIR, 1/K4/2-6
  GIA, 1/K4/2-6
  GUD, 1/K4/2-6
  IKA, 1/K4/2-6
  INI, 1/K4/2-6
  LUD, 1/K4/2-6
  MPF, 1/K4/2-6
  OPT, 1/K4/2-6
  RPA, 1/K4/2-6
  SEA, 1/K4/2-6
  SPF, 1/K4/2-6
  SYF, 1/K4/2-6
  TEA, 1/K4/2-6
  UFR, 1/K4/2-6
  WPD, 1/K4/2-6
Fine interpolation, 1/G2/2-44
Fine offset, 1/K2/1-9, 1/K2/2-43
FIPO, 1/G2/2-44
Fixed feed values, 1/V1/2-39
Flat D number structure, 1/W1/2-19
  D number programming, 1/W1/2-21
  Read D number from the PLC, 1/W1/2-20
Floppy disk drive, 1/K4/2-10
FM, 1/P3/8-268
Following error, 1/D1/2-9
Following error compensation (feedforward control), speed feedforward control, 1/G2/2-51
Follow-up, 1/A2/2-21
Follow-up mode, 1/A2/2-20, 1/A2/5-87, 1/A3/2-6, 1/A3/2-11
Follow-up mode active, 1/A2/2-30, 1/A2/5-97
Foreground language, 1/A2/2-35
FRAME, 1/K2/1-5, 1/K2/2-15, 1/K2/2-42
  FRAME chaining, 1/K2/2-67
  MIRROR, 1/K2/2-54
  SCALE, 1/K2/2-54

© Siemens AG, 2003. All rights reserved

Function overview of inch/metric switchover (SW 5 and higher), 1/G2/2-21
Data backup, 1/G2/2-19
Rounding machine data, 1/G2/2-20
Synchronized actions, 1/G2/2-15

G

G groups, 1/K1/2-69
G0 logic in G96, 1/V1/4-60
G40, 1/W1/2-39
G41, 1/W1/2-39
G42, 1/W1/2-39
G450/G451, 1/W1/2-54
G451, 1/W1/2-56
G460, 1/W1/2-68
G461, 1/W1/1-4, 1/W1/2-69
G462, 1/W1/1-4, 1/W1/2-69
G58, 1/K2/1-9, 1/K2/2-43
G59, 1/K2/1-9, 1/K2/2-43
G91, Frames, 1/K2/3-82
G91 extension, offset external (WOE), 1/W1/2-97
Gap, 1/K1/2-11
Gear stage change from NC, 1/S1/2-28
Gear stages, 1/S1/2-28
General machine data, 1/A3/3-57
Geometry axes, 1/B1/2-5, 1/B1/2-14, 1/B1/2-18, 1/K2/2-15, 1/K2/2-19, 1/K2/2-35, 1/P1/1-3
Geometry model, 1/D2/2-6
GMMC_INFO_NO_UNIT, 1/K1/4-143
GMMC_INFO_NO_UNIT_STATUS, 1/K1/4-143
GroupSerupro, 1/K1/2-42, 1/K1/2-45

H

Handheld unit (HHU), 1/K4/2-11
Hardware faults, 1/A3/2-23
Hardware limit switches, 1/A3/2-24
Heatsink temperature prewarning, 1/A2/2-34, 1/A2/5-106
Heatsink temperature warning, 1/D1/2-21
Helical interpolation, 1/K2/2-22
Helix interpolation, 1/K2/2-22
Hide several program levels, 1/K1/4-131
High-speed data channel, 1/A2/2-41
Higraph first error display, 1/A2/5-73
HMI Embedded, 1/K4/2-6
Hold, 1/A2/2-21
HPU address, 1/K4/2-13

I

ID check, 1/K1/2-129
Implicit continuous-path mode, 1/B1/2-17
Implicit exact stop, 1/B1/2-14
INC, 1/P1/2-6
Inch measuring system, 1/G2/2-13
Influence path axes individually, 1/K1/2-67
Influence SERUPRO behavior, 1/K1/2-37
Initial setting, 1/B1/2-7
Initial settings, 1/K1/2-69
Input resolution, 1/G2/2-9
Integrator disable, 1/A2/2-33, 1/D1/2-19
Integrator ncontroller disabled, 1/A2/2-34
Interactive Programming
Adaptation of simulation data (SW 3.1 and higher), 1/D2/2-14
Adaptation of simulation data (SW 3.2 and higher), 1/D2/2-18
Adapting the technology memory, 1/D2/2-10
Archiving of configuration data, 1/D2/2-31
Area of application, 1/D2/2-5
Axis-specific machine data, 1/D2/7-40
Basic procedure, 1/D2/1-3
Channel-specific machine data, 1/D2/7-39
Compatibility, 1/D2/2-7
Configuring the user interface, 1/D2/2-8
Creation of a tool catalog, 1/D2/2-11
Customizing the axis names, 1/D2/2-13
Example, 1/D2/6-37
General conditions, 1/D2/2-7
General machine data, 1/D2/7-39
Matching the tool offset data, 1/D2/2-27
Milling, 1/D2/2-12
Restriction, 1/D2/2-7
Setting up, 1/D2/2-12

© Siemens AG, 2003. All rights reserved

Simulation data, special conditions, 1/D2/2-22
Supplementary conditions, 1/D2/3-33
Turning, 1/D2/2-12

Interactive programming (D2), 1/D2/1-3

Interface signal 840Di, 1/A2/2-9

Interface signal PC Shutdown, 1/A2/2-9

Channel-specific NCK alarm is active, 1/A2/5-84

Interface signals
Activate channel-specific protection zone, 1/A3/5-73
Activate dry run feed, 1/V1/5-66
Activate M01, 1/K1/5-175
Activate machine-related protection zone, 1/A3/5-73
Activate program test, 1/K1/5-176
Activate referencing, 1/R1/5-65
Activate single block, 1/K1/5-175
Active drive parameter set, 1/A2/2-34
Active G function of groups 1 to 60, 1/K1/5-188
Active machine function REF, 1/K1/5-175
Active machine function REPOS, 1/K1/5-174
Active machine function TEACH IN, 1/K1/5-174
Active mode AUTOMATIC, 1/K1/5-173
Active mode JOG, 1/K1/5-173
Active mode MDA, 1/K1/5-173
Active spindle control mode, 1/S1/5-107
Active spindle mode oscillation mode, 1/S1/5-107
Active spindle positioning mode, 1/S1/5-107
Actual direction of rotation clockwise, 1/S1/5-105
Actual gear stage A to C, 1/S1/5-99
All axes referenced, 1/R1/5-66
All channels in Reset state, 1/K1/5-174
AUTOMATIC mode, 1/K1/5-169
Block search active, 1/K1/5-184
Change gear, 1/S1/5-104
Channel status active, 1/K1/5-187
Channel status interrupted, 1/K1/5-187
Channel status reset, 1/K1/5-187
Channel-specific protection zone preactivated, 1/A3/5-74
Channel-specific protection zone violated, 1/A3/5-75
Clamping in progress, 1/A3/5-69
Delete S value, 1/S1/5-98
Dry run feedrate selected, 1/V1/5-71
Enable protection zones, 1/A3/5-72
Encoder limit frequency exceeded, 1/A3/5-71
F function for positioning axis, 1/V1/5-79
Feed disable, 1/V1/5-70
Feed hold (geometry axis 1 to 3), 1/V1/5-71
Feed override, 1/V1/5-66
Feed override (axis-specific), 1/V1/5-73
Feed override active, 1/V1/5-71
Feed override for rapid traverse selected, 1/V1/5-72
Feedrate hold/spindle stop (axis-specific), 1/V1/5-78
For operator panel, 1/A2/2-15
Gear changed, 1/S1/5-99
Hardware limit switches plus and minus, 1/A3/5-70
Invert M3/M4, 1/S1/5-100
JOG mode, 1/K1/5-169
M function for spindle, 1/S1/5-108
M00/M01 active, 1/K1/5-183
M01 selected, 1/K1/5-181
M02/M30 active, 1/K1/5-184
Machine function REF, 1/K1/5-172
Machine function REPOS, 1/K1/5-171
Machine function TEACH IN, 1/K1/5-171
Machine-related protection zone preactivated, 1/A3/5-73
Machine-related protection zone violated, 1/A3/5-74
MDA mode, 1/K1/5-169
Mode group ready, 1/K1/5-174
Mode group reset, 1/K1/5-171
Mode group stop, 1/K1/5-170
Mode group stop axes plus spindles, 1/K1/5-170
Mode switchover disable, 1/K1/5-170
n (act) < n(x), 1/A2/2-35
NC Start, 1/K1/5-178
NC Start disable, 1/K1/5-178
NC Stop, 1/K1/5-179
NC Stop at block limit, 1/K1/5-178
NC Stop axes plus spindles, 1/K1/5-180
Oscillation speed, 1/S1/5-101
Oscillation via PLC, 1/S1/5-101, 1/S1/5-102, 1/S1/5-103
Override active, 1/V1/5-77
Program level abort, 1/K1/5-178
Program status aborted, 1/K1/5-187
Program status interrupted, 1/K1/5-186, 1/K1/5-189
Program status running, 1/K1/5-185
Program status stopped, 1/K1/5-186
Program status wait, 1/K1/5-186
Program test active, 1/K1/5-185
Program test selected, 1/K1/5-181
Ramp-function generator rapid stop active, 1/A2/2-33
Rapid traverse override, 1/V1/5-68
Rapid traverse override active, 1/V1/5-70
Read-in disable, 1/K1/5-177
Reference point approach delay, 1/R1/5-67
Reference point value 1 to 4, 1/R1/5-66
Referenced/synchronized 1, 1/R1/5-67

© Siemens AG, 2003. All rights reserved

Referenced/synchronized 2, 1/R1/5-67
Referencing active, 1/R1/5-65
Reset, 1/K1/5-181
Resynchronize spindle 1 and 2, 1/S1/5-98
Resynchronize spindle during positioning 2 and 1, 1/S1/5-100
Revolutional feedrate active, 1/V1/5-79
Rigid tapping active, 1/S1/5-108
S function for spindle, 1/S1/5-108
Second software limit switch plus or minus, 1/A3/5-70
Selected machine function REF, 1/K1/5-173
Selected mode AUTOMATIC, 1/K1/5-172
Selected mode JOG, 1/K1/5-172
Selected mode MDA, 1/K1/5-172
Selected mode TEACH IN, 1/K1/5-172, 1/K1/5-173
Set counterclockwise/set clockwise, 1/S1/5-100
Set gear stage A to C, 1/S1/5-104
Setpoint speed increased, 1/S1/5-106
Setpoint speed limited, 1/S1/5-106
Skip block, 1/K1/5-176
Speed limit exceeded, 1/S1/5-106
Spindle in setpoint range, 1/S1/5-105
Spindle override, 1/V1/5-75
Spindle reset/delete distance-to-go, 1/S1/5-97
Spindle/no axis, 1/S1/5-103
Velocity/spindle speed limitation, 1/A3/5-70
Zero offset, 1/K2/5-102

Interface signals (A2), 1/A2/1-3

Interface signals for digital drives, 1/A2/2-32

Interface signals from and to channel, 1/A2/2-19

Interface signals from axis/spindle, 1/A2/2-30

Interface signals to axis/spindle, 1/A2/2-19

Interface signals/table, 1/A2/7-115

Internal protection zone, 1/A3/2-38

Interpolation buffer, reduced, 1/K1/2-103

Interpolation parameter sets, 1/G2/2-47

Interpolator, 1/K1/2-13

Interpolator end, 1/B1/2-11

Interpolatory axis grouping, 1/A2/2-27

Interrupt routines, 1/K1/1-6, 1/K1/2-86, 1/K1/4-131

Invert M3/M4, 1/S1/2-58

INVFRAME, 1/K2/2-68

J

Jerk limit, 1/B1/2-24

Jerk limitation, 1/B1/2-23
Smoothing method, 1/B2/2-13

Jerk limitation on the path, Example, 1/B1/7-63

JOG, 1/K1/2-16, 1/P1/2-6

K

Keep tool radius compensation constant, 1/W1/2-63

Key disable, 1/A2/2-15, 1/A2/5-71

Keyboard type, 1/A2/4-50

KEYBOARD_STATE, 1/A2/4-50

KEYBOARD_TYPE, 1/A2/4-50

Keyswitch, 1/A2/2-12

Keyswitch position, 1/A2/5-67

Kinematic transformation, 1/K2/2-35

Knee-shaped acceleration characteristic, Selection and deselection, 1/B2/2-16

KONT, 1/W1/2-40

Kv-Factor, 1/A3/2-5, 1/D1/2-14, 1/G2/2-44

L

Labels, 1/K1/2-79

Language commands
ACC, 1/B2/2-17
BRISK, 1/B2/2-8
SOFT, 1/B2/2-8

Last action block active, 1/K1/5-183

LEN_PROTOCOL_FILEX, 11420, 1/D1/4-37

LIFTFAST, 1/K1/2-87

LIFTFAST_WITH_MIRROR, 1/K1/4-148

Limit switch monitoring, 1/A3/2-24

Limit value for offsets fine, 1/K2/4-87

Limit value for wear fine, 1/K2/4-87

© Siemens AG, 2003. All rights reserved

Linear axis
With linear scale, 1/G2/2-38
With rotary encoder on motor, 1/G2/2-39
With rotary encoder on the machine, 1/G2/2-40

Linear drives, 1/K4/2-20

Linear feed (G94), 1/V1/2-9

Link axes, 1/K1/2-14

Load, 1/A2/5-81

Load parts program, 1/A2/5-75

Loader axes, 1/K2/2-19

Lockable data areas, 1/A2/2-13

Log book, 1/D1/2-29

Look Ahead, 1/B1/2-13, 1/B1/2-42
Block cycle problem, 1/B1/2-46
Continuous-path mode, 1/B1/2-42
Exact stop, 1/B1/2-42
Following block velocity, 1/B1/2-44
Override, 1/B1/2-45
Selection and deselection, 1/B1/2-46
Velocity profiles, 1/B1/2-44

LOOKAH_SMOOTH_WITH_FEED, MD 20462, 1/B1/4-51

Look Ahead, Number of blocks, 1/B1/2-43

Lubrication pulse, 1/A2/2-31, 1/A2/5-100

LVDS, 1/K4/1-3

M

M decoding acc. to list, 1/P3/2-58

M(d) less than M(dx), 1/A2/5-108

M_NO_FCT_CYCLE, 1/K1/4-137

M_NO_FCT_CYCLE_NAME, 1/K1/4-137

MA_AXES_SHOW_GEO_FIRST, 1/K1/4-131

MA_MAX_SKP_LEVEL, 1/K1/4-132

MA_ORIAXIS_EULER_ANGLE_NAME, 1/K2/4-84

MA_PRESET_MODE, 1/K1/4-132

MA_PREST_FRAMEIDX, 1/K2/4-84

MA_SCRATCH_DEFAULT_MODE, 1/K2/4-86

MA_STAT_DISPLAY_BASE, 1/K2/4-83

MA_TU_DISPLAY_BASE, 1/K2/4-83

Machine axes, 1/K2/2-14

Machine coordinate system, 1/A2/2-16

Machine coordinate system (MCS), 1/K2/1-7, 1/P1/2-6

Machine coordinate systems (MCS), 1/K2/2-34

Machine data, 1/A2/7-118

Machine data for operator panel, 1/A2/4-47

Machine data for processing programs from external, 1/K1/4-143

Machine data, channel-specific, 1/A2/4-60

Machine data, general, 1/A2/4-58

Machine tool axes, 1/K2/2-19

Machine zero, 1/R1/1-3

Main axes, 1/K2/2-21

Main technological application, 1/K1/2-15

Manual specification of a gear stage, 1/S1/2-33

Manual switchover of the basic system
General, 1/G2/2-17
Input resolution and calculation resolution, 1/G2/2-19
JOG and handwheel evaluation, 1/G2/2-19
Reference point, 1/G2/2-18
System data, 1/G2/2-18
Tool data, 1/G2/2-18

Manual switchover of the basic system (SW 5 and higher), 1/G2/2-17

Matching the machine data, 1/D2/2-16

Material memory, 1/D2/2-10

Maximum permissible path feed for compression, 1/V1/4-59

MAXNUM_USER_DATA_FLOAT, 1/P3/9-284

MAXNUM_USER_DATA_HEX, 1/P3/9-283

MAXNUM_USER_DATA_INT, 1/P3/9-283

MCS, 1/A2/2-16

MD_FILE_STYLE, 1/A2/4-60

MDA (MDI), 1/K1/2-16

Measuring systems, 1/G2/2-24

Memory requirements of basic PLC program, 1/P3/6-254
Maximum, 1/P3/6-256
Minimum, 1/P3/6-256

Message signals in DB2, 1/P3/5-247

Metric measuring system, 1/G2/2-13

Minimum speed, 1/D1/2-23

MIRROR_TOOL_LENGTH, 1/W1/4-149

© Siemens AG, 2003. All rights reserved

MIRROR_TOOL_WEAR, 1/W1/4-149

Mirroring, Retraction direction (lift fast), 1/K1/2-87

MM_ABSBLOCK, 1/K1/4-164

MM_ABSBLOCK_BUFFER_CONF, 1/K1/4-164

MM_NUM_VDIVAR_ELEMENTS, 1/P3/9-285

MM_SYSTEM_FRAME_MASK, 1/K2/4-99

MMC, 1/K1/2-128, 1/K1/2-129

MMC alarm, 1/A2/2-8

MMC alarm is active, 1/A2/5-68

MMC CPU1 Ready, 1/A2/2-7, 1/A2/5-68

Modal activation (FOCON/FOCOF), 1/F1/2-23

Modal execution from external source, 1/K1/2-109

Modally active synchronized actions, 1/K1/2-16

Mode change disable, 1/A2/5-77

Mode group, 1/K1/2-9, 1/P3/2-51
- Number, 1/K1/2-11, 1/A2/5-82
- Ready to operate, 1/K1/2-12
- Reset, 1/K1/2-12

Mode groups, 1/K1/2-16
- Group, 1/K1/1-5, 1/K1/2-10
- Interlocks, 1/K1/2-18
- Monitoring functions, 1/K1/2-18
- Operating mode switchover, 1/K1/2-19, 1/K1/2-130

Modify the torque FXS, 1/F1/2-22

Monitor type, 1/A2/2-35

Monitoring functions in the modes, 1/K1/2-18

Monitoring of static limits, 1/A3/2-24

Motion monitoring functions, 1/A3/2-5

Motion specification, 1/S1/2-58

Motor, 1/A2/2-32

Motor selection, 1/D1/2-20

Motor selection A, B, 1/A2/5-94

Motor selection in progress, 1/A2/5-95

Motor temperature, 1/D1/2-19

Motor temperature warning, 1/A2/2-34, 1/A2/5-105, 1/D1/2-21

MPI/DP interface (RS485), 1/K4/2-10, 1/K4/2-11

Multiple feed values in one block, 1/V1/2-34

Multiple starts with SERUPRO, 1/K1/2-42

N

n(act) equals n(set), 1/A2/5-110

n(act) less than n(min), 1/A2/5-110

n(act) less than n(x), 1/A2/5-110

NC
- NC instruction, 1/K1/2-70
- NC Start, 1/K1/2-70, 1/K1/2-128
- Program, 1/K1/2-128
- Reset, 1/K1/2-128, 1/K1/2-130
- Stop, 1/K1/2-75

NC failure, 1/P3/2-53

NC Ready, 1/A2/5-69

NC VAR selector, 1/P3/3-78
- Startup, installation, 1/P3/3-88

NC variables, 1/P3/3-83

NCK alarm handler, 1/D1/2-5

NCK alarm is active, 1/A2/2-8, 1/A2/5-69

NCK alarm with processing stop, 1/A2/2-8, 1/A2/5-84

NCK battery alarm, 1/A2/2-8, 1/A2/5-70

NCK CPU Ready, 1/A2/2-7, 1/A2/5-68

NCK machine data, axis/spindle-specific data, 1/A2/4-61

NC-specific signals, 1/A2/5-67

Non-cyclic communication, 1/K4/2-15

NORM, 1/W1/2-40

Normalization, 1/G2/2-7

Normalization of machine and setting data, 1/G2/2-10

Number of places to be displayed without exponent, 1/A2/4-52

Numerical basis for display of moving joint STAT, 1/K2/4-83

Numerical basis for display of rotary axis position TU, 1/K2/4-83

O

O.K., 1/A2/5-79, 1/A2/5-80

Offset number, 1/W1/2-11

Operator panel reset, 1/K1/2-81, 1/K1/2-83

© Siemens AG, 2003. All rights reserved

Orientation, 1/A3/2-36

Orientation axes, 1/B1/2-18

Orientation axis, 1/B1/2-5

Orientational toolholder, 1/W1/1-4, 1/W1/2-72, 1/W1/6-153

Overload factor, 1/B1/2-16

Override, 1/D1/2-10, 1/K1/2-127

Overstore, 1/H2/2-30

P

PA_ZOA_MODE, 1/K2/4-87

Parallel interface, 1/K4/5-32

Parallel interface (Centronics), 1/K4/2-8

Parameter Process Data Object (PPO), 1/K4/2-15

Parameter set (axis), 1/D1/2-11

Parameter sets of the position controller, 1/G2/2-46

Parameterization of basic block display, 1/K1/4-164

Parking axis, 1/D1/2-20

Parts program end, 1/K1/2-81

Parts program start, 1/K1/2-80, 1/K1/2-113

Parts program start and parts program end, 1/K1/2-82

Parts programs
  Parts program interruption, 1/K1/2-71
  Skipping of specific parts program blocks, 1/K1/2-25

Password, 1/A2/2-11
  resetting, 1/A2/2-11
  setting, 1/A2/2-11

Path axes, 1/B1/2-5, 1/K2/2-19, 1/K1/5-193

Path criterion, 1/B1/2-19

Path feed, 1/B1/2-7, 1/G2/2-6

PATH_TRANS_JERK_LIM, MD 32432, 1/B1/4-56

Path-related jerk limitation, 1/B2/2-7
  Acceleration with jerk limitation, 1/B2/2-7
  Profile with abrupt acceleration changes, 1/B2/2-9
  Selection and deselection, 1/B2/2-8

PDA area, 1/K4/2-15

Physical quantities, 1/G2/2-10

PI services, Overview, 1/P3/4-117

PKW area, 1/K4/2-15

PLC CPUs, properties, 1/P3/2-30

PLC index, 1/A2/5-75, 1/A2/5-76

PLC line offset, 1/A2/5-76

PLC messages, 1/P3/2-42

PLC program, 1/K1/2-127

PLC status, 1/D1/2-30
  Change in status, 1/D1/2-30
  Status display, 1/D1/2-30

PLC/NCK
  840D, 1/P3/2-30
  FM-NC, 1/P3/2-32
  PLC/MCP, 1/P3/2-44
  PLC/MMC, 1/P3/2-41

PLC/NCK interface, 1/A2/2-5, 1/P3/2-35

PLC_CYCLIC_TIMEOUT, 1/P3/9-283

PLC-CPU, Key data, 1/P3/2-7

Polynomial, Intersection procedure, 1/W1/2-67

POS, 1/B1/2-14, 1/K2/2-19, 1/K2/2-20

POSA, 1/K2/2-19, 1/K2/2-20

Position actual value measuring system 1/2, 1/D1/2-21

Position control, 1/G2/2-46

Position control direction, 1/A3/2-20

Position control loop, 1/G2/2-43

Position controller active, 1/A2/2-30, 1/A2/5-98

Position controller gain, 1/A3/2-9

Position measuring system, 1/A2/2-24

Position measuring system 1 and 2, 1/A2/5-88

Position setpoint, 1/D1/2-9

Positioning accuracy of the control system, 1/G2/2-7

Positioning axes, 1/G2/2-6, 1/K2/2-20, 1/P1/2-6

Positioning axis type POS, 1/B1/2-8

Positioning axis type POSA, 1/B1/2-8

Positioning window large, 1/A3/2-9

POWER ON, block search, 1/K1/2-27, 1/K1/7-198

Power-up, 1/K1/2-82

PRG_DEFAULT_DIR, 1/A2/4-49

Process alarm processing, 1/P3/2-52

© Siemens AG, 2003. All rights reserved

Processing programs from external, extension (SW 4 and higher), 1/K1/1-6

PROFIBUS DP Address space, 1/K4/2-16

PROFIBUS DP Master, Slave, 1/K4/2-15

PROG_EVENT_MASK, 1/K1/4-145

PROG_EVENT_MASK_PROPERTIES, 1/K1/4-145

Program
- Hide several Program levels, 1/K1/4-131
- Program action, 1/K1/2-76
- Program control, 1/K1/2-101
- Program operation mode, 1/K1/1-6, 1/K1/2-69
- Program status, 1/K1/2-73
- Program test, 1/K1/2-20
- Repeating program sections, 1/K1/2-79
- Runtime, 1/K1/1-7, 1/K1/2-127, 1/K1/7-202

Program code, 1/D2/2-6

Program control
- DRF, 1/K1/2-101
- DRY, 1/K1/2-101
- M01, 1/K1/2-101
- PRT, 1/K1/2-101
- ROV, 1/K1/2-101
- SBL1, 1/K1/2-101
- SKP, 1/K1/2-101

Program execution without setpoint outputs, 1/K1/2-20

Program interpreter, 1/D2/2-6

Program test, 1/K1/1-5

Program testing and SERUPRO, 1/S1/2-62

Programmable run-in and run-out paths, 1/V1/2-15, 1/V1/2-17

Programming and parameterizing tools, 1/P3/3-75

Programming area selection, 1/A2/5-73

Programming devices or PCs, 1/P3/3-75

PROTAREA_GEOAX_CHANGE_MODE, 1/A3/3-57

Protection level for access to tool carrier offsets..., 1/A2/4-52

Protection levels, 1/A2/2-13

Protection levels for MACHINE operating area for access to ..., 1/A2/4-57

Protection levels for PROGRAM operating area for access to ..., 1/A2/4-56

Protection zone
- Activation, 1/A3/2-41
- Deactivation, 1/A3/2-41
- Definition, 1/A3/2-36
- Enabling of, 1/A3/2-45
- Restrictions, 1/A3/2-52

Protection zone definition, Example, 1/A3/6-77

Protection zone definition with system variables, 1/A3/2-39

Protection zone for switchover of geo axes, 1/A3/3-57

Protection zone violation, 1/A3/2-45

Protection zones, 1/A3/1-3, 1/A3/2-32
- 2-dimensional, 1/A3/2-33
- 3-dimensional, 1/A3/2-33
- Channel-specific interface signals, 1/A3/7-82
- Channel-specific machine data, 1/A3/7-85
- Channel-specific signals, 1/A3/5-72
- Data storage, 1/A3/2-41
- General machine data, 1/A3/7-83
- Supplementary conditions, 1/A3/3-55

Protection zones for milling machine, 1/A3/2-35

Protection zones for turning machine, 1/A3/2-34

Protection zones, convex, 1/A3/2-39

Protection zones, concave, 1/A3/2-39

PS/2
- Keyboard interface, 1/K4/2-9
- Mouse interface, 1/K4/2-10

Pulse enable, 1/A2/2-33, 1/A2/5-95, 1/D1/2-17

Pulse multiplication factor, 1/G2/2-38

Pulses enabled, 1/A2/2-34, 1/A2/5-104

R

Ramp function generator rapid stop active, 1/A2/2-33

Ramp function generator rapid stop, 1/A2/2-32, 1/A2/5-92, 1/D1/2-17

Ramp function generator rapid stop active, 1/A2/5-100

Ramp-up function complete, 1/A2/2-34, 1/A2/5-107, 1/D1/2-21

Ramp-up phase, 1/D1/2-18, 1/D1/2-24

Ramp-up times, 1/A2/5-92

© Siemens AG, 2003. All rights reserved

Rapid stop (with open, 1/A3/2-7, 1/A3/2-11, 1/A3/2-18, 1/A3/2-19, 1/A3/2-21, 1/A3/2-22

Rapid traverse, 1/B1/2-7, 1/K1/2-128

Rapid traverse override, 1/B1/2-7

Rapid traverse override switch, 1/V1/2-28

Read/write NC variables, 1/P3/2-55

Read/write PLC variable, 1/A2/1-3, 1/A2/2-41

Reader's note, v

Read-in disable, 1/K1/2-75

Reapproach contour with controlled REPOS, 1/K1/2-67

Reapproach point, 1/K1/2-68

Recall alarm cleared, 1/A2/5-78

Recall alarms, 1/A2/2-16

Reciprocal feed (G93), 1/V1/2-9

Reduction of jerking, 1/B1/2-24

REF, REF active machine function, DB11, ... DBX5.2, 1/R1/2-6

References, C-21

Reference point, 1/R1/1-3

Referencing
- Absolute offset, 1/R1/2-23
- Axis-specific, 1/R1/2-6
- By means of actual value adjustment, 1/R1/2-33
- By parts program, 1/R1/2-6
- Channel-specific, 1/R1/2-6
- Reference cam, 1/R1/2-22
- Reference cam adjustment, 1/R1/2-14
- Reference mark, 1/R1/2-22, 1/R1/2-23
- With distance-coded marks, 1/R1/2-20, 1/R1/2-25
- With incremental measurement system, 1/R1/2-8

Replaceable geometry axes, 1/K2/2-15

REPOS Delay Ack, 1/K1/5-192

REPOS mode, 1/K1/2-68

REPOS offset, 1/K1/5-192

REPOS offset valid, 1/K1/5-192

REPOSDELAY, 1/K1/5-192, 1/K1/5-193

Repositioning neutral axes after SERUPRO, 1/K1/2-63

REPOSMODEEDGE, 1/K1/5-182

REPOSMODEEDGEACKN, 1/K1/5-190

Reset, 1/K1/2-75, 1/N2/2-9

Reset and starting response, 1/K1/4-154

RESET command, 1/K1/2-73

Reset response, 1/K1/7-198

Reset time, 1/K1/2-129

Resolutions, 1/G2/2-7

Revolutional feedrate (G95), 1/V1/2-9
- Revolutional feedrate active, 1/V1/2-10

Rigidity control (DSC), 1/K4/2-20

RMB, 1/K1/5-182

RME, 1/K1/5-182

RMI, 1/K1/5-182

RMN, 1/K1/5-182

RMNOTDEF, 1/K1/5-182

Rotary axes, 1/K2/2-19

Rotary axis
- With rotary encoder on motor, 1/G2/2-41
- With rotary encoder on the machine, 1/G2/2-42

Rough offset, 1/K2/1-9, 1/K2/2-43

Rounding, 1/B1/2-18

Rounding with contour tolerance, 1/B1/2-22

RS 232, 1/K4/1-3

RS 232 C external, 1/A2/5-74, 1/A2/5-79

RS 232 C OFF, 1/A2/5-74, 1/A2/5-80

RS 232 C stop, 1/A2/5-74, 1/A2/5-79

RS 232, 1/K4/5-31

RS 232 C ON, 1/A2/5-75, 1/A2/5-80

Run-in and run-out paths, programmable, 1/V1/2-15, 1/V1/2-17

Runtime
- Measurement; Example, 1/K1/2-128
- Of the NC program, 1/K1/2-128
- Of the path axes, 1/K1/2-128

S

Safe actual position, 1/D1/2-12

Safe input signals, 1/D1/2-12

Safe output signals, 1/D1/2-12

SAVE command, 1/K1/2-89

Screen bright, 1/A2/5-71

Screen is dark, 1/A2/5-77

© Siemens AG, 2003. All rights reserved

Selection, 1/A2/5-75, 1/A2/5-81

Selection of the cutting edge when changing tool, 1/W1/2-6

Semi-automatic procedure for matching the machine data, 1/D2/2-20

SER_CLASS_WRITE_TOA_SUPVIS, 1/A2/4-52

Serial interface
MMC 100: File types, 1/K4/2-6
MMC 101/102: File types, 1/K4/2-7

Serial interfaces, 1/K4/5-31

Serial interfaces (RS232), 1/K4/2-5

SERUPRO, 1/K1/2-36

SERUPRO sequence without defining a search target beforehand, 1/K1/2-46

SERUPRO_SPEED_FACTOR, 1/K1/4-151

Servo enable, 1/A2/2-25, 1/A2/5-90

Servo gain factor, 1/D1/2-9, 1/G2/2-45

Set REPOS response, 1/K1/2-61

SETINT, 1/K1/2-87

SETINT_ASSIGN_FASTIN, MD 21210, 1/K1/4-148

Setpoint output, 1/G2/2-23

Setpoint parameter set, 1/D1/2-20

Setpoint system, 1/G2/2-23

Setpoint/actual-value system, Configuration of drives for SINUMERIK 840Di, 1/G2/2-30

Setting data for processing programs from external, 1/K1/4-167

Setting the computer mode (turning, milling), 1/A2/4-51

Setting the constant cutting rate, 1/S1/2-59

Setting the speed, 1/S1/2-59

Setup mode, 1/D1/2-20

Setup mode active, 1/A2/2-33, 1/A2/5-100

Shift behavior of keyboard during booting, 1/A2/4-50

ShopMill control signal, 1/A2/5-74

Signal Descriptions, 1/A2/5-67

Signal distortion, 1/A3/2-5

Signaling states, 1/A2/1-3

Signals
Alarm signals, 1/A2/2-8
Axis/spindle-specific (DB31, ...), 1/A2/2-6
Channel-specific (DB21, ...), 1/A2/2-6
General (DB10), 1/A2/2-6
OP-specific (DB19), 1/A2/2-6
PLC/axes, spindles, 1/P3/2-40
PLC/mode group, 1/P3/2-37, 1/P3/2-38
PLC/NC, 1/P3/2-37
PLC/NCK channels, 1/P3/2-39
Ready signals, 1/A2/2-7

Signals from axis/spindle, 1/A2/2-33, 1/A2/5-97

Signals from channel, 1/A2/5-84

Signals from NC to PLC (DB10), 1/A2/5-68

Signals from operator panel, 1/A2/5-77

Signals from PLC to NC (DB10), 1/A2/5-67

Signals to axis/spindle, 1/A2/2-32, 1/A2/5-85

Signals to channel, 1/A2/5-83

Signals to operator panel, 1/A2/5-71

Simulation axes, 1/G2/2-24

Simulation selected, 1/A2/5-78

SINUMERIK 802D, 1/K1/2-129

SINUMERIK 810D powerline, vi

SINUMERIK 840D powerline, vi

SINUMERIK 840DiISRAM, 1/R1/3-45

SLASH_MASK, 1/A2/4-58

Slotting saw, 1/W1/2-28

Smooth approach and retraction
Approach movement, 1/W1/2-41
Meaning, 1/W1/2-41
Retraction movement, 1/W1/2-41

Smoothed actual current value, 1/D1/2-19

SOFT, 1/B2/2-8

Software limit switches, 1/A3/2-25

Special axes, 1/K2/2-19

Special spindle interface, 1/S1/2-55

Specify gear stage, 1/S1/2-28

© Siemens AG, 2003. All rights reserved

Speed
Actual point, 1/D1/2-10
Closed-Loop Control, 1/A3/2-21
Servo enable NC, 1/D1/2-17
Setpoint, 1/D1/2-10, 1/D1/2-19
Setpoint smoothing, 1/D1/2-19
Speed control loop, 1/G2/2-43
Speed controller active, 1/A2/2-31, 1/A2/5-99
Speed controller integrator disable, 1/A2/5-95
Speed controller integrator disabled, 1/A2/5-103
Speed setpoint adjustment, 1/G2/2-34
Speed setpoint output, 1/G2/2-34, 1/G2/2-47, 1/G2/2-50, 1/G2/2-56, 1/G2/2-60, 1/G2/2-61
Speed setpoint routing, 1/G2/2-25
Speed setpoint smoothing, 1/A2/2-32, 1/A2/5-93
Speed setpoint smoothing active, 1/A2/2-33, 1/A2/5-101
SPI(n, 1/S1/2-53
SPIND_POSIT_DELAY_TIME, 1/S1/4-87
Spindle commands, 1/S1/2-57
Spindle disable, 1/A2/2-19
Spindle job, 1/S1/2-56
Spindle override factor, 1/V1/2-30
Spindle reset, 1/A2/2-28
Spindle speed, 1/G2/2-6
Spindle speed limitation with G96, 1/V1/2-12
Spindle speed setpoint current, 1/D1/2-10
Spindle speed setpoint programmed, 1/D1/2-10
Spindle stop, 1/K1/2-75
Spindle-specific functions, 1/S1/2-61
SPOS, 1/B1/2-14
START command, 1/K1/2-70
Start-up and synchronization of NCK PLC, 1/P3/2-50
Stop bits, 1/K4/3-21
Stop block, 1/K1/2-41
STOP commands, 1/K1/2-71
Stop routine, 1/K1/2-41
STOP_LIMIT_FACTOR, 1/B1/4-58
Structure of MD files for data backup, 1/A2/4-60
SUMCORR_DEFAULT, 1/W1/4-140
SUPPRESS_ALARM_MASK, 1/D1/4-35
SUPPRESS_SCREEN_REFRESH, 10131, 1/A2/4-58
SW 5.2, 1/K1/1-7, 1/K1/2-127, 1/K1/7-202
Switch over MCS/WCS, 1/A2/5-78
Switch over measuring system, 1/A2/2-24
Symbolic programming of user program with interface DB, 1/P3/2-56
Synchronization, 1/B1/2-9
Synchronous axes, 1/B1/2-14, 1/B1/2-19, 1/K2/2-22
System variables, 1/A2/4-65, 1/A2/7-121
Channel-specific, 1/K1/2-127
NCK-specific, 1/K1/2-127
With read/write access, 1/K1/2-129

T

T function, 1/W1/2-5
T_NO_FCT_CYCLE_NAME, 1/K1/4-138
Tabulator length, 1/A2/4-50
TABULATOR_SIZE, 1/A2/4-50
Tachogenerator compensation, 1/A3/2-19
TEACH IN, 1/K1/2-16
Technical exponent representation in three steps, 1/A2/4-52
TECHNOLOGY, 1/A2/4-51
Technology memory, 1/D2/2-10
Test mode, 1/A3/2-17
Threshold speed, 1/D1/2-23
Threshold torque, 1/D1/2-21
Timer activation, 1/K1/2-129
Timers, Channel-specific, 1/K1/2-128
TOA, 1/W1/2-7
TOCARR_BASE_FINE_LIM_LIN, 1/W1/4-134
TOCARR_BASE_FINE_LIM_ROT, 1/W1/4-135
TOCARR_FINE_CORRECTION, 1/W1/4-152

© Siemens AG, 2003. All rights reserved

Tool, 1/W1/2-5
Base dimension/adapter dimension, 1/W1/2-34
Change, 1/W1/2-5
Change tool with “M06”, 1/W1/2-5
Cutting edge, 1/W1/2-26
Cutting edge position, 1/W1/2-29
DISC, 1/W1/2-55
Length compensation, 1/W1/2-30
Management, 1/K1/1-7, 1/K1/2-127, 1/K1/2-129
Operating time, 1/K1/2-128
Radius compensation, 1/W1/2-32
Select, 1/W1/2-5
T function, 1/W1/2-5
Tool clearance angle, 1/W1/2-35
Tool cutting edge, 1/W1/2-5
Tool radius compensation 2D (TRC), 1/W1/2-38
Tool type, 1/W1/2-27
Wear, 1/W1/2-33

Tool change
D function, 1/W1/2-6
Offset memory, 1/W1/2-7

Tool length compensation, 1/W1/2-30

Tool master data, 1/D2/2-11

Tool offset, Calculation, 1/W1/2-8

Tool offset selection, 1/A2/5-73

Tool operating data, 1/D2/2-11

Tool radius compensation, change alarm response, 1/W1/2-66

Tool radius compensation (TRC), 1/W1/2-38

Tool radius compensation: 2D, 1/W1/2-38
Approach and retraction behavior, 1/W1/2-40
Collision detection, 1/W1/2-60
Compensation at inside corners, 1/W1/2-58
Compensation at outside corners, 1/W1/2-54
Deselection, 1/W1/2-54
G451 intersection, 1/W1/2-56
Selection, 1/W1/2-39
Smooth approach and retraction, 1/W1/2-41
Transition circle, 1/W1/2-55
Variable compensation value, 1/W1/2-61, 1/W1/2-85

Tool revolver axes, 1/K2/2-19

Tool type, Slotting saw, 1/W1/2-28

TOOL_LENGTH_CONST, 1/K2/4-100, 1/W1/4-150

TOOL_REF_GEO_AXIS1, 1/K2/4-85

TOOL_REF_GEO_AXIS2, 1/K2/4-85

TOOL_REF_GEO_AXIS3, 1/K2/4-85

TOOL_TEMP_COMP, 1/W1/4-152

TOOL_TEMP_COMP_LIMIT, 1/W1/4-141

TOOL_TEMP_COMP_ON, 1/W1/4-141

Toolholder selection, 1/W1/2-72

Toolholder, orientational, 1/W1/2-72
Control system response on Reset, program start, REPOS, 1/W1/2-94
Programming, 1/W1/2-86, 1/W1/2-89, 1/W1/2-93, 1/W1/6-154, 1/W1/6-157
Supplementary conditions, 1/W1/2-85, 1/W1/2-89, 1/W1/2-94

Tools with relevant tool point direction, 1/W1/2-36

Torque limit, 1/D1/2-19

Torque limit 2, 1/A2/2-32, 1/A2/5-93

Torque limit 2 active, 1/A2/2-33, 1/A2/5-101

Torque limitation value, 1/D1/2-12

TRANS, 1/K2/1-9, 1/K2/2-43

Transfer time, 1/K1/2-63

Transformation, 1/K2/2-35

TRANSMIT, 1/K2/2-35

Transverse axes, 1/P1/1-3
Diameter programming, 1/P1/2-6
Geometry axes, 1/P1/2-5

Travel to fixed stop, 1/F1/1-3, 1/F1/2-5
Analog drives, 1/F1/2-28
Analog drives, diagrams, 1/F1/2-34
Analog drives, FXS deselection, 1/F1/2-36
Analog drives, FXS selection, 1/F1/2-34
Analog drives, SIMODRIVE 611A (FDD), 1/F1/2-28
Analog drives, SIMODRIVE 611A (MSD), 1/F1/2-30
Axis/spindle-specific signals, 1/F1/5-48
Axis-specific machine data, 1/F1/4-39
Axis-specific setting data, 1/F1/4-44
Deselection, 1/F1/2-11
Functional sequence, 1/F1/2-8
Interface signals, 1/F1/7-53
Monitoring window, 1/F1/2-9
Programming, 1/F1/2-6, 1/F1/2-8
RESET, 1/F1/2-13
Selection, 1/F1/2-8

© Siemens AG, 2003. All rights reserved

Travel to fixed stop (F1)
Alarms, 1/F1/2-10
Contour monitoring, 1/F1/2-19
Detailed Description, 1/F1/2-5
Function abort, 1/F1/2-13
Machine data, 1/F1/7-54
Positioning axes, 1/F1/2-19
POWER ON, block search, 1/F1/2-14
Setting data, 1/F1/2-10, 1/F1/2-18, 1/F1/7-54
Signal Descriptions, 1/F1/5-47
Status interrogation, 1/F1/2-19

Traversing ranges, 1/G2/2-6

Types of feed, 1/V1/1-3

Types of protection zone, 1/A3/2-32

U

U(ZK), 1/A2/2-35

UDC-link lower than warning threshold, 1/A2/5-112

Universal Serial Bus, 1/K4/1-3

Unload, 1/A2/5-75, 1/A2/5-81

Updating the REPOS offset within the valid scope, 1/K1/2-66

USB, interface, 1/K4/2-9

Useful Tips on Programming with STEP7, 1/P3/7-257
Copying data, 1/P3/7-258

Useful tips on programming with Step7
ANY and POINTER, 1/P3/7-258
Multi-instance DB, 1/P3/7-263
POINTER or ANY variable, 1/P3/7-261
Strings, 1/P3/7-264
Use of ANY and POINTER in FC, 1/P3/7-259, 1/P3/7-260

USER_CLASS_BASE_ZERO_OFF_MA, 1/K2/4-84

USER_CLASS_BASE_ZERO_OFF_PA, 1/K2/4-84

USER_CLASS_DIRECTORY1_M, 1/A2/4-57

USER_CLASS_DIRECTORY1_P, 1/A2/4-56

USER_CLASS_DIRECTORY2_M, 1/A2/4-57

USER_CLASS_DIRECTORY2_P, 1/A2/4-56

USER_CLASS_DIRECTORY3_M, 1/A2/4-57

USER_CLASS_DIRECTORY3_P, 1/A2/4-56

USER_CLASS_DIRECTORY4_M, 1/A2/4-57

USER_CLASS_DIRECTORY4_P, 1/A2/4-56

USER_CLASS_READ_TCARR, 1/A2/4-52

USER_CLASS_WRITE_MAG_WGROUP, 1/A2/4-52

USER_CLASS_WRITE_TCARR, 1/A2/4-52

USER_CLASS_WRITE_TOA_ASSDNO, 1/A2/4-52

USER_CLASS_WRITE_TOA_EC, 1/A2/4-52

USER_CLASS_WRITE_TOA_SC, 1/A2/4-52

USER_DATA_FLOAT[n], 1/P3/9-284

USER_DATA_HEX[n], 1/P3/9-284

USER_DATA_INT[n], 1/P3/9-284

User-defined system ASUBs, 1/K1/2-94

Using DMP modules, 1/K2/3-81

V

Value storage scratching and PRESET, 1/K2/4-84

Variable signal 1, 1/D1/2-23

Variable signaling function, 1/A2/2-35, 1/A2/5-111

Velocities, 1/B1/2-7, 1/G2/2-5

Velocity adjustment, 1/G2/2-34

Velocity for zero cycle blocks, 1/B1/2-7

Velocity reduction according to overload factor, 1/B1/2-16

W

WCS, 1/A2/2-16

WEAR_SIGN, 1/W1/4-150

WEAR_SIGN_CUTPOS, 1/W1/4-149

Working area limitation, 1/A3/2-28

Workpiece
Actual, 1/K1/2-129
Counter, 1/K1/1-7, 1/K1/2-129, 1/K1/7-202
Example, 1/K1/2-130
Counting, 1/K1/2-127
Setpoint, 1/K1/2-129
Total actual, 1/K1/2-129

© Siemens AG, 2003. All rights reserved

Workpiece coordinate system (WCS), 1/A2/2-16, 1/K2/1-7, 1/K2/2-42, 1/P1/1-3, 1/P1/2-6

WRITE_TOA_LIMIT_MASK, 1/K2/4-86

Z

Zeros and reference points, 1/K2/2-29

ZK1 messages, 1/D1/2-18, 1/D1/2-24

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

Complete Index (FB1, FB2, FB3)

Symbols

"Selfacting Serupro", 1/K1/2-46
$A_DNO[n], 1/W1/2-20
$A_PBB_IN[n], 2/A4/4-56
$A_PBB_OUT[n], 2/A4/4-57
$A_PBD_IN[n], 2/A4/4-56
$A_PBD_OUT[n], 2/A4/4-58
$A_PBR_IN[n], 2/A4/4-57
$A_PBR_OUT[n], 2/A4/4-58
$A_PBW_IN[n], 2/A4/4-56
$A_PBW_OUT[n], 2/A4/4-57

Numbers

10670, 3/F2/4-89
10672, 3/F2/4-89
13201, 2/M5/4-69
13210, 2/M5/4-69
2-axis SC special kinematic, 3/TE4/2-39
3-axis kinematics, Articulated-arm kinematics, 3/TE4/2-25
3-axis NR kinematic, 3/TE4/2-25, 3/TE4/2-27
3-axis RR kinematic, 3/TE4/2-26
3-axis CC kinematic, 3/TE4/2-22, 3/TE4/2-29, 3/TE4/2-36
3-axis CS kinematic, 3/TE4/2-24, 3/TE4/2-31
3-axis kinematics, 3/TE4/2-21, 3/TE4/2-28, 3/TE4/2-34, 3/TE4/2-38
SCARA kinematic, 3/TE4/2-22
3-axis SC kinematic, 3/TE4/2-23, 3/TE4/2-30
12104,12105, 3/F2/2-29
34104 – REFP_PERMITTED_IN_FOLLOWUP, 1/R1/4-57
4-axis kinematics
Articulated-arm kinematics, 3/TE4/2-32
SCARA kinematics, 3/TE4/2-29
4-axis NR kinematic, 3/TE4/2-32
5-axis kinematics, SCARA kinematics, 3/TE4/2-36
5-axis NR kinematic, 3/TE4/2-37
5-axis machining package, 3/F2/3-85
5-axis transformation
Channel-specific signals, 3/F2/5-113
Configuration of a machine, 3/F2/2-18
Data descriptions (MD, SD), 3/F2/4-87
Geometry of the machine, 3/F2/2-19
Interface signals, 3/F2/7-129
Machine data, 3/F2/7-130
Machine types, 3/F2/2-16
Singular positions, 3/F2/2-27
Tool orientation, 3/F2/2-23
611D Ready, 1/A2/2-7, 1/A2/5-69
7-layer model, 2/B3/1-9

A

A_COORDINATE_SYSTEM, 1/K2/4-86
Absolute compensation value meas. system 1 or 2, 1/D1/2-9
ABSBLOCK_FUNCTION_MASK, 1/K1/4-164
Absolute dimension tool length compensation geometry axis 1, 1/K2/4-85
Absolute dimension tool length compensation geometry axis 2, 1/K2/4-85
Absolute dimension tool length compensation geometry axis 3, 1/K2/4-85
Acceleration, 1/B2/1-3, 2/H1/2-6, 2/H1/2-17, 2/H1/2-34, 3/TE4/2-19
Kneeshaped acceleration characteristic, 1/B2/2-15
Programmable, 1/B2/2-17
Acceleration (B2), 1/B2/1-3
Alarms, 1/B2/7-34
Axisspecific machine data, 1/B2/4-23
Channelspecific machine data, 1/B2/4-28
Channelspecific setting data, 1/B2/4-29
Machine data, 1/B2/7-31
Setting data, 1/B2/7-33
System variables, 1/B2/7-34
Acceleration characteristic, 2/N4/2-16
Acceleration limitation, 1/B2/2-11
Acceleration profiles, 1/B2/2-5
Abrupt acceleration changes, 1/B2/2-5
Acceleration with jerk limitation, 1/B2/2-5
Acceleration reduction characteristic (FM-NC only), 2/S6/4-20
Acceleration time constant, 3/G3/2-7
Acceleration warning threshold, 3/M3/5-98
Access authorization, 1/A2/2-10, 3/V2/2-8
Access features, 1/A2/2-10
Access protection, 1/A2/2-10
ACN, 2/R2/2-12
ACP, 2/R2/2-12
Action single block, 1/K1/2-22
Activate coupling parameters, 2/S3/2-21
Activate following axis overlay, 3/M3/5-97
Activate temperature, Compensation, 2/K3/2-8
Activate/deactivate, 2/W4/2-22
Activating remote diagnostics later (MMC 100.2), 2/F3/2-89
Activating the rotation, 3/F2/2-67
Activation, 3/F2/2-42, 3/K6/2-7
After power ON, 2/S3/2-11
Methods, 2/S3/2-10
Of coupling, 2/S3/2-10
Activation and priorities, 1/K1/2-17
Activation of block skip, 1/A2/4-58
Activation of protection zones, Example, 1/A3/6-80

© Siemens AG, 2003. All rights reserved

Activation speed/-velocity for acceleration reduction, 2/S6/4-20
Activation/Deactivation, 3/V2/2-7
Active, 1/A2/5-81
Active data (frames) are immediately operative after editing, 1/K2/4-86
Active drive parameter set A, B, C, 1/A2/5-101
Active feedforward control, 3/K6/2-9
Active file system, 2/S7/2-5
Active meas. system, 1/D1/2-9
Active motor, 1/A2/2-34
Active motor A, B, 1/A2/5-102
Active or passive file system, 1/A2/5-76
Active state, 2/B3/1-14
Active/passive operating mode, 2/B3/5-142
Active/passive operating mode of MMC, 2/B3/5-142
ACTIVE_SEL_USER_DATA, 1/K2/4-86
Actual motor, 1/D1/2-21
Actual parameter set, 1/D1/2-20
Actual position value measuring system, 1/D1/2-9
Actual speed = set speed, 1/D1/2-23
Actual value synchronization, 1/A2/2-27
Actual value correction, 1/G2/2-24
Actual value in WCS, 1/A2/5-72
Actual value in workpiece coordinate system, 1/A2/2-16
Actual value processing, 1/G2/2-34, 1/G2/2-47, 1/G2/2-50, 1/G2/2-56, 1/G2/2-60, 1/G2/2-61
Actual value resolution, 1/G2/2-36
Actual value routing, 1/G2/2-25
Actual value sensing, 1/G2/2-23
Actual value system, 1/G2/2-23
  Workpiecerelated, 1/K2/2-77
Actual values, 1/A3/2-20, 2/M5/2-17
Actual-value coupling, 2/S3/2-34
Adaptation, Dynamic performance, 1/B1/1-3
Adapting active frames, 1/K2/2-56
Adapting the motor/load ratios, 1/G2/2-31
Adaptive control, 2/S5/2-29
  Additive control, 2/S5/2-29
  Example, 2/S5/2-32
  Multiplicative control, 2/S5/2-29
ADDFRAME, 1/K2/2-70
Additive frame in frame chain, 1/K2/2-70
Address, 2/B3/1-12
Air temperature alarm, 1/A2/2-8, 1/A2/5-70
Alarm
  Delete criterion, 1/D1/2-6
  Server, 2/B3/1-14, 2/B3/1-22
  Text file, 2/B3/1-18
  Text management, 2/B3/1-18
Alarm area selection, 1/A2/5-73
Alarm for preprocessing stop, 1/W1/2-66
ALARM_PAR_DISPLAY_TEXT, 1/D1/4-37
ALARM_REACTION_CHAN_NOREADY, 1/D1/4-37
Alarms, 1/A2/7-121, 2/B3/1-20, 2/B3/1-21, 2/B3/1-23
  Messages, 2/B3/1-18
Alternate interface, 2/N4/2-15
AMIRROR(C), 2/R2/2-21
Analog axis: alarms, 3/TE2/7-23, 3/TE3/7-62
Analog axis: brief description, 3/TE2/1-3, 3/TE3/1-3
Analog axis: detailed description, 3/TE2/2-5, 3/TE3/2-5
Analog axis: Hardware configuration, 3/TE4/2-6, 3/TE2/2-7
Analog axis: machine data, 3/TE2/7-24, 3/TE3/7-61
Analog axis: supplementary conditions, 3/TE2/3-13, 3/TE3/3-33
Analog inputs of the NCK, 2/A4/2-19
Analog inputs-/outputs on local P bus of FM-NC, 2/L1/2-13
Analog outputs of the NCK, 2/A4/2-22
Analog spindle 1, utilization in percent, 1/A2/5-73
Analog spindle 2, utilization in percent, 1/A2/5-73
Analog value representation, 2/A4/2-32
Analysis output, 3/K6/2-8
Angle, inclined axis, 2/M1/2-46
Angular offset
  LS/FS, 2/S3/2-34
  POSFS, 2/S3/2-11
Angularity error compensation, 2/K3/2-23
Applicability of MD 9203 to edge data and location-dependent offsets, 1/K2/4-86
Application tool, 2/B3/1-19
ASCALE, 2/R2/2-18
ASCII editor, 1/D2/2-6
Assigned timers, 1/P3/6-253
Assignment
  Bus nodes – d bus system, 2/B3/1-14
  By axis groups, 2/B3/1-27
  MMCs – NCUs, 2/B3/1-14
  Oscillation/infeed axis, 2/P5/2-23, 2/P5/2-24
Assignment between probe type and application, 2/M5/2-5
Assignment of DBs, 1/P3/6-251
Assignment of FBs and FCs, 1/P3/6-251
Assignment tool/toolholder, 1/W1/2-72
ASUBs, 1/P3/2-55
Asynchronous events, 1/A2/1-3
Asynchronous subroutines (ASUBs), 1/K1/2-86
ATRANS, 1/K2/1-9, 1/K2/2-43, 2/R2/2-18
AUTOMATIC, 1/K1/2-16
Automatic mode, 1/K1/2-128
Automatic start of an ASUB following block search (SW 6.1 and higher, 1/K1/2-31
Automatically activated pre-initiation time, 2/N4/2-15
Autonomous machine, 2/B3/1-23

© Siemens AG, 2003. All rights reserved

Autonomous single axis operations, 2/P2/6-42
Auxiliary function output, 1/B1/2-15
Detailed Description, 1/H2/2-5
Overstore, 1/H2/2-30
Power ON, block search, 1/H2/2-29
Auxiliary functions, 1/H2/1-3
Axis transformations, 3/F2/3-85
Axis transformation, Call and application, 3/F2/2-41
Electronic gear, 3/M3/3-82, 3/M3/3-83
Helix interpolation, 1/K2/3-81
Screw interpolation, 1/K2/3-81
Workpiecerelated actualvalue system, 1/K2/3-82
Zero offset, 1/K2/3-82
Availability of basic offset in Machine operating area, 1/K2/4-84
Availability of basic offset in Parameters operating area, 1/K2/4-84
AX_JERK_MODE, 1/B2/4-24
AXCHANGE_MASK, 2/K5/4-27
Axial measurement, 2/M5/2-58
Axial sources, 3/M3/2-68
Axis, 1/A2/2-19
Container, 2/B3/1-27, 2/B3/1-28
Container rotation active, 2/B3/5-146
Data, 2/B3/1-7
Holding the workpiece, 2/B3/1-27
Local, 2/B3/1-27, 2/B3/1-28
Physical, 2/B3/1-7, 2/B3/1-27
Ready, 2/B3/5-146
Rotary, 2/B3/1-25
Usable, 2/B3/1-7
Work-holding, 2/B3/1-28
Axis accelerated, 3/M3/5-97
Axis configuration, 1/K2/2-23
Axis direction, Changing, 3/TE4/2-18
Axis identifiers, 3/V2/2-12
Axis monitoring functions, 1/A3/1-3
Actual velocity, 1/A3/2-19
Axis/spindlespecific interface signals, 1/A3/7-81
Axis/Spindlespecific machine data, 1/A3/3-58, 1/A3/7-83
Axis/spindlespecific setting data, 1/A3/7-85
Axis/spindlespecific signals, 1/A3/5-69
Channelspecific machine data, 1/A3/7-85
Clamping, 1/A3/2-7, 1/A3/2-11
Contour, 1/A3/2-5
Following error, 1/A3/2-6
Positioning monitoring., 1/A3/2-7
Setting data, 1/A3/3-66
Speed setpoint, 1/A3/2-17
Supplementary conditions, 1/A3/3-55
Zero speed, 1/A3/2-7, 1/A3/2-10
Axis monitoring, protection zones (A3), Channel-specific machine data, 1/A3/3-57
Axis parameter sets, 1/G2/2-46
Axis sequence, Changing, 3/TE4/2-17
Axis types, 1/K2/2-13, 3/TE4/2-19
Axis zero points, Matching, 3/TE4/2-19
Axis/spindle disable, 1/A2/5-85
Axis/spindle lock DB31, ... DBX1.3, 1/V1/2-27
Axis/spindle replacement, 2/K5/1-3, 2/K5/2-13
Axis/spindle service display, 1/D1/2-8
Axis/spindle stops, 1/A2/2-30, 1/A2/5-98
Axis/spindlespecific signals, 1/A2/5-85
AXIS_VAR_SERVER_SENSITIVE, 2/B3/4-125
Axisrelated jerk limitation, 1/B2/2-12
Jerk limitation, 1/B2/2-12
Axisspecific machine data, 1/K2/4-99
Axisspecific signals, 1/K1/5-192
Axis transformation, 3/F2/1-7
AXRESET DONE, 2/P2/5-38

B

Backlash compensation
2nd measuring system, 2/K3/2-13
Display, 2/K3/2-12
Mechanical backlash, 2/K3/2-12
Negative backlash, 2/K3/2-13
Positive backlash, 2/K3/2-12
Backup battery, 2/B3/1-19
Basic axis configuration, 3/TE4/2-12
Basic block display, 1/K1/4-164
Basic coordinate system (BCS), 1/K2/1-7, 1/K2/2-35
Basic setting for Program directory, 1/A2/4-49
Basic setting of acceleration pattern (FM-NC), 2/S6/4-20
Basic system, 1/G2/2-13
Basic tool orientation, 1/W1/6-156
Baud rate, 1/K4/3-21, 2/B3/1-8
Beam sag compensation, 2/K3/2-23
Behavior at inner corners, 3/W5/2-25
Behavior at outer corners, 3/W5/2-24
Behavior at pole, 3/F2/2-27
Beveled hand with elbow, 3/TE4/2-14
Bi-directional probe, 2/M5/2-6
Block, 1/K1/2-75
Program operation mode, 1/K1/2-22
Search, 1/K1/2-26
Single block suppression, 1/K1/2-98, 1/K1/2-100
Block change, 1/B1/2-8, 1/B1/2-10
Block change point, 1/B1/2-13
Block change response, 2/S3/2-10
Block cycle time, 3/G3/2-7
Block search, 1/K1/1-5, 1/K1/1-6, 2/M5/2-58, 2/P2/2-8
Block search via program test, 1/K1/2-36
Block SPARPI, 1/K1/2-44
Block-related limit (FOC), 1/F1/2-23
Blockwise feed FB, 1/V1/2-42
BRISK, 1/B2/2-8

© Siemens AG, 2003. All rights reserved

Bus
Address, 2/B3/1-6
Capacity, 2/B3/1-8
Design, 2/B3/1-23
Node, 2/B3/1-8, 2/B3/1-12
Nodes, 2/B3/1-11, 2/B3/1-14, 2/B3/1-17
Properties, 2/B3/1-8
System, 2/B3/1-10, 2/B3/1-11, 2/B3/1-12, 2/B3/1-14
Termination, 2/B3/1-23
Type, 2/B3/1-6
Bus addresses, 1/K4/2-13
Bus addresses on SINUMERIK 840D, 1/P3/2-46
Bus addresses on SINUMERIK FMNC, 1/P3/2-47

C

Cabling, 2/B3/1-13
Calculation method, 2/M5/2-21
Calculation resolution, 1/G2/2-9
Calipers, 1/V1/2-38, 2/S5/2-28
Call, 3/V2/2-10
Call condition, 3/V2/2-10
Calling the ASUB outside program operation, 1/K1/2-90
Cam positions
Axis/cam assignment, 2/N3/2-15
Setting cam positions, 2/N3/2-14
Cam range/cam pair, 2/N3/2-5
Cam signal output
Independent, timer-controlled, 2/N3/2-20
Timer-controlled, 2/N3/2-19
Cam signals
Activation of signal output, 2/N3/2-17
For linear axes, 2/N3/2-6, 2/N3/2-10
For modulo rotary axes, 2/N3/2-7
Hardware assignment, 2/N3/2-18
Lead/delay times, 2/N3/2-15
Minus, 2/N3/2-17
Output to NCK I/Os, 2/N3/2-18
Output to PLC, 2/N3/2-17
Plus, 2/N3/2-17
Cancel alarm cleared, 1/A2/5-77
Cancel alarms, 1/A2/2-16
Carrying out SERUPRO in a group of channels, 1/K1/2-44
Cartesian PTP travel, 3/F2/2-79
STAT address, 3/F2/2-80
TU address, 3/F2/2-81
Cascaded block search (SW 6.1 and higher), 1/K1/2-31
Categories of kinematics, 3/TE4/2-5
Centerless Grinding
Activation/deactivation, 3/S8/2-5
Calculating the speed of the regulating wheel, 3/S8/2-5
What does the function do?, 3/S8/1-4
Centerless grinding, 3/S8/1-3
Central hand, 3/TE4/2-14
Central MMC, 2/B3/1-23
CFC, 1/V1/2-8
CFCIN, 1/V1/2-8
CFINE, 1/K2/1-9, 1/K2/2-43
CFTCP, 1/V1/2-8
Chained transformations, 2/M1/2-53
Activation, 2/M1/2-55
Disabling, 2/M1/2-55
Example, 2/M1/2-55, 2/M1/6-85
Number, 2/M1/2-54
Special characteristics, 2/M1/2-55
Chaining direction, 2/M1/2-54
Chaining rule, 2/W4/2-10
Change defined angular offset, 2/S3/2-10
Change in insertion depth, 3/W5/2-16
Change in orientation, 3/W5/2-10, 3/W5/2-25
Change protection for coupling characteristics, 2/S3/2-8
Channel, 1/B1/2-6, 1/K1/1-5, 1/K1/2-13, 2/K5/1-3, 2/P2/2-6
Channel configuration, 1/K1/2-13
Channel machine data, 1/K1/2-129
Channel status, 1/K1/2-75
Gaps, 1/K1/2-9, 1/K1/2-11
Menu, 2/B3/1-19
Name, 2/B3/1-19
VDI signal, 1/K1/2-129
Channel axes, 1/K2/2-15
Channel axis identifiers with FXS, 1/F1/2-7
Channel number (FC9: ChanNo), 1/A2/5-82
Channel number of machine control panel on MMC, 1/A2/5-73
Channel synchronization, 2/K5/2-6
Channel-specific machine data, 3/V2/4-15
Channels, number of, 2/B3/1-5, 2/B3/1-10
Channelspecific NCK alarm is active, 1/A2/2-8
Channelspecific signals, 1/A2/5-83
Chat, 2/F3/2-106
Circularity test, 2/K3/2-49, 2/K3/2-50, 2/K3/2-80
Display, 2/K3/2-82
Measurement, 2/K3/2-81
Parameterization, 2/K3/2-80
Representation, 2/K3/2-81
Clamping axis/spindle, 2/B3/1-28
Clamping protection zone, 2/N4/2-32
Clear cancel alarms, 1/A2/5-72
Clear recall alarms, 1/A2/5-72
Clearance control: alarms, 3/TE1/7-49, 3/TE5/7-29, 3/TE6/7-25
Clearance control: brief description, 3/TE1/1-3, 3/TE5/1-3
Clearance control: detailed description, 3/TE1/2-5
Clearance control: detailed description, 3/TE0/2-5, 3/TE6/2-5
Clearance control: machine data, 3/TE1/7-53, 3/TE5/7-30
Clearance control: programming, 3/TE1/2-13

© Siemens AG, 2003. All rights reserved

Clearance control: signals, 3/TE1/7-54, 3/TE5/7-30
Clearance control: suplementary conditions, 3/TE1/3-21
Clearance control: supplementary conditions, 3/TE5/3-13
Clearance control:signal descriptions, 3/TE1/5-41, 3/TE5/5-17, 3/TE6/5-21
Closed kinematic loop, 3/TE4/2-10
ClosedLoop Control, 1/G2/2-43
Collision detection, 1/W1/2-60
COM1, 1/A2/5-74, 1/A2/5-79
COM1 or COM2, 1/K4/2-6
COM2, 1/A2/5-74, 1/A2/5-79
Combination of different bus systems, 2/B3/1-14
Communication
MMC 100, 1/K4/2-6
MMC 100/101/102, 1/K4/1-3
MMC 101/102, 1/K4/1-3
Communication log, 1/D1/2-28
Communication with global data (GD), 1/P3/2-33
Commutation limit, 1/B2/2-15
Comparator inputs, 2/A4/2-7, 2/A4/2-34
Compensation table, 2/K3/2-19
Compensations, Additional torque for electronic weight compensation, 2/K3/4-97
Compilation, 3/V2/2-7
Complete machining, 2/M1/2-9
Computer
Link, 2/B3/1-6, 2/B3/2-34
Module, 2/B3/1-11
Computing capacity, 2/B3/1-25
Concurrent axes, 1/P3/2-54
Configurability of machine control panel, hand-held unit, 1/P3/2-66
Configurable gear adaption, 1/S1/2-28
Configuration, 2/B3/1-19
Configuration file, 2/B3/1-11, 2/B3/2-40, 2/B3/2-57, 2/B3/2-63
NETNAMES.INI, 2/B3/6-169
Number, 2/B3/2-57
Configuring the software
for HT6, 2/F3/2-83
for MMC 100.2, 2/F3/2-77
Conn_1, 2/B3/1-17
Connection
of probe, 2/M5/2-7
Operating area, 2/B3/1-21
to 810D, 2/M5/2-9
to 840D, 810D, 2/M5/2-7
to 840Di, 2/M5/2-8
to FM-NC NCU 570.2, 2/M5/2-11
Connection setup, 2/F3/2-94
for Windows 3.1X, 2/F3/2-101
for Windows 95, 2/F3/2-94
for Windows NT, 2/F3/2-99
in MS-DOS (MMC 100.2), 2/F3/2-103
Connector
X101, 2/B3/1-17
X122, 2/B3/1-21
Constant cutting rate (G96), 1/V1/2-11
Constant grinding wheel peripheral speed (GWPS), 2/W4/2-30
Constant speed G9, 1/V1/2-11
Constant workpiece speed for centerless grinding, 3/S8/2-5
Container axes, 1/K1/2-14
Contamination signal, 1/A3/2-23
Continuous
Dressing, 2/W4/2-18
Jogging, 2/H1/2-10
Measurements, 2/M5/2-63
Mode, 2/T1/2-6
Traversal
Continuous mode, 2/H1/2-10
Jog mode, 2/H1/2-10
Continuouspath mode, 1/B1/2-13
Continuouspath mode, exact stop and LookAhead (B1)
Alarms, 1/B1/7-66
Axisspecific machine data, 1/B1/4-56
Axisspecific signals, 1/B1/5-61
Channelspecific machine data, 1/B1/4-50
Channelspecific signals, 1/B1/5-61
General machine data, 1/B1/4-49
Interface signals, 1/B1/7-63
Machine data, 1/B1/7-64
Contour deviation, 1/D1/2-9
Contour error, 1/A3/2-5
Contour handwheel, 2/H1/2-28
Pulses per detent position, 2/H1/4-44
Contour tunnel monitoring, 3/K6/2-7
Contour violation, 1/A3/2-28, 1/A3/2-31
Contrast, 1/A2/2-35
Control, 1/B2/2-15
Normal powerup, 1/K1/2-127
Unit, 2/B3/1-6, 2/B3/1-7, 2/B3/1-10, 2/B3/1-12, 2/B3/1-13, 2/B3/1-15, 2/B3/1-20
Unit switchover, 2/B3/1-14
Control deviation, 1/D1/2-9
Control direction, 1/G2/2-34
Control of JOG traverse functions, 2/H1/2-8
Control panel, Main, 2/B3/1-6
Control powerup, 1/K1/2-127
With default values, 1/K1/2-127
Control velocity continuously, 2/S5/2-20
Controller mode, 1/D1/2-11
Controller parameter set switchover, 1/A2/6-113
Controller parameter set switchover (checkback signal), 1/A2/5-99
Controller parameter set switchover (request), 1/A2/5-91
Controlling synchronous spindle coupling via PLC, 2/S3/2-15

© Siemens AG, 2003. All rights reserved

Conversion of basic system, 1/G2/2-13
Coordinate system, 1/A3/2-36
Coordinate systems, 1/K2/1-5, 1/K2/1-6, 1/K2/1-7, 1/K2/2-32, 2/H1/2-7
Corner, 3/T3/1-3
Corner in area, 3/T3/6-20
COROS OP, 2/B3/1-13
Counting
  Algorithm, 1/K1/2-129
  Pulse, 1/K1/2-129
Coupled Axes
  Behavior in operating modes, 3/M3/2-12
  Programming, 3/M3/2-11
Coupled axes, Axis types, 3/M3/2-9
Coupled motion, Interface signals, 3/M3/2-13
Coupling
  Characteristics, 2/S3/2-8
  Define new, 2/S3/2-19
  Options, 2/S3/2-6
  Parameters, 2/S3/2-25
  Permanent coupling configuration, 2/S3/2-19
  Type, 2/S3/2-8
  User-defined, 2/S3/2-19
Coupling two FM-NCs with one PLC, Coupling two FM-NCs with one PLC, 1/P3/8-268
Coupling two FMNC Modules to one PLC, 1/P3/8-267
  Basic program error messages, 1/P3/8-281
  Coupling with the components MMC/MSTT/OP, 1/P3/8-270
  Function of the basic program for two FMNCs, 1/P3/8-275
  Interface structure, 1/P3/8-271
  Memory space, 1/P3/8-281
  PLC/MCP/HHU interface, 1/P3/8-273
  PLC/MMC interface, 1/P3/8-273
  PLC/NCK interface, 1/P3/8-271
  Special features of blocks when using two FMNCs, 1/P3/8-276
    FB1: RUN_UP, 1/P3/8-276
    FB2...5: Read/write NC variables, PI services, GUD variables, 1/P3/8-278
    FC 15, 16, 18: POS_AX, PART_AX, SpinCtrl, 1/P3/8-279
    FC 9: ASUB, 1/P3/8-280
Couplings, 1/K1/2-51
CRC error, 1/D1/2-18
Cross-communication, 2/B3/1-29
CTRANS, 1/K2/1-9, 1/K2/2-43
Current controller active, 1/A2/2-31, 1/A2/5-99
Current gear stage, 1/D1/2-11
Curve tables
  Axis types, 3/M3/2-15
  Behavior in operating modes, 3/M3/2-25
  Interface signals, 3/M3/2-26
  Programming, 3/M3/2-17
CUT3DC, 3/W5/2-28
CUT3DF, 3/W5/2-28
CUT3DFF, 3/W5/2-28
CUT3DFS, 3/W5/2-28
Cutter material memory, 1/D2/2-10
Cutter shapes, 3/W5/2-18
Cutting edge number, 1/W1/2-10
Cutting edge position, relevant, 1/W1/2-36
Cycle times – Defaults, 3/G3/2-6
Cycle Times – Example, 3/G3/2-6
Cycle-independent path-synchronous switch signal output, Brief description, 3/TE8/1-3
Cycle/independent path/synchronous switch signal output, 3/TE8/2-5
Cycles with parameter, 3/V2/2-9
Cycles without parameter, 3/V2/2-9
Cyclic operation, 1/P3/2-50
Cyclical communication, 1/K4/2-15
Cyclical signal exchange, 1/A2/2-5, 1/P3/1-5
Cylinder coordinate system, 2/M1/2-29
Cylinder generated surface, 2/M1/1-6

D

D functions, 1/W1/2-5
D number structure, – flat (without tool management), 1/W1/2-19
D numbers, Allocation of free ..., 1/W1/2-10
Darken screen, 1/A2/2-15, 1/A2/5-71
Data
  Backup via RS-232, 2/B3/1-17, 2/B3/1-21
  Exchange, 2/B3/1-16
  Management server, 2/B3/1-14
Data backup, 1/D1/2-6
Data channel, high-speed, 1/A2/2-41
Data Exchange Time, 3/G3/2-10
Data readoff, 1/D1/2-13
Data sets, 1/D2/2-19
DC link
  Backup, 3/M3/2-76
  Energy balance, 3/M3/2-77
DC link backup, 3/M3/2-72
DC link status, 1/D1/2-18
DC link voltage, 1/B2/2-15
Deactivation
  of coupling, 2/S3/2-11
  while spindles are moving, 2/S3/2-11
Deceleration methods, 3/K6/1-3, 3/K6/2-7
Decoding single block, 1/K1/2-22
Decoding single block SBL2 with implicit preprocessing stop, 1/K1/2-97
Default, 2/B3/1-13
Default passwords, 1/A2/2-11
Default settings, 2/S3/2-21
Default value for path feed, 1/V1/4-63
DEFAULT_FEED, 1/V1/4-63
Define number of skippable program levels, 1/K1/4-132
Defined short-circuits, 2/A4/4-46
Defining the reference for the path override, 1/V1/4-57

© Siemens AG, 2003. All rights reserved

Definition, EG axis grouping, 3/M3/2-47
Definition of a joint, 3/TE4/2-8
Definition of synchronous spindles, 2/S3/2-7
Delay time for ESR single axis, 2/P2/4-35
Delete couplings, 2/S3/2-21
Delete distance to go, 1/A2/2-19
Delete distancetogo (axisspecific)/Spindle reset, 1/A2/5-91
Delete distancetogo (channelspecific), 1/A2/5-83
Descriptions of kinematics, 3/TE4/2-21
Deselecting synchronous mode, 2/S3/2-11
Deselection methods, 2/S3/2-12
Deviation in synchronism, 3/M3/2-41
Diagnostic tools
  General machine data, 1/D1/4-35
  Machine data, 1/D1/7-42
Diagnostic tools (D1), 1/D1/1-3
  Alarms, 1/D1/2-5
  External, 1/D1/1-3
  Integrated, 1/D1/1-3
  Interface signals, 1/D1/7-41
Diagnostics, 1/D1/1-3, 1/D1/2-8
  611D startup tool, 1/D1/2-31
  Archiving of data, 1/D1/2-31
  for alarms, 1/D1/2-14, 1/D1/2-24
  of operational state errors, 1/D1/2-15
Diagnostics buffer on PLC, 1/P3/2-33
DIAMOF, 1/P1/1-3, 1/P1/2-6
DIAMON, 1/P1/1-3, 1/P1/2-6
Digital inputs of the NCK, 2/A4/2-12
Digital inputs-/outputs on local P bus of FM-NC, 2/L1/2-11
Digital outputs of the NCK, 2/A4/2-14
Direct connection, 2/B3/1-18
Direction of offset of tool for geometry axes with scratching, 1/K2/4-86
Disable synchronization, 2/S3/5-41
  (DB31, ... DBX31.5), 2/S3/2-15
DISC, 1/W1/2-55
Display geo axes of channel first, 1/K1/4-131
Display mode of zero offset, 1/K2/4-87
Display of orientation axes as Euler angle, 1/K2/4-84
Display resolution, 1/A2/2-36, 1/G2/2-9
Displayed channel number of MMC, 1/A2/5-78
Distance control, 1D/3D, 3/TE1/2-10
DL auxiliary function for sum offset, 1/H2/2-11
Double addressing, 2/B3/1-16
DP cycle, 3/G3/2-9
DP cycle time, 3/G3/2-10
DRAM, 2/S7/2-6
  Memory requirements, 2/S7/2-25
Dressing during machining process, 2/W4/2-18
DRF, 2/H1/1-3, 2/H1/2-38, 2/T1/2-21
  Offset, 2/H1/2-38
Drift, 2/K3/2-87
Drift compensation, 2/K3/2-87
DRIFT_LIMIT, 2/K3/2-87
Drive configuration, 1/G2/2-30
Drive control, 2/B3/1-27
Drive enable, 1/D1/2-17
Drive parameter set selection, 1/A2/2-32
Drive parameter set selection A, B, C, 1/A2/5-94
DRIVE ready, 1/A2/2-34
Drive Ready, 1/A2/5-103
Drive service display, 1/D1/2-15
Drive status, 1/D1/2-18
DRIVE_AX_RATIO_DENOM, 1/S1/4-76
DRIVE_AX_RATIO2_NUMERA, 1/S1/4-77
Drum, 2/B3/1-27
Drum/rotary switching axis, 2/B3/1-25
Dry run feed, 1/K1/2-24, 1/V1/2-33
Dry run feedrate, 2/P2/2-32
Dry run feedrate (DryRun), 1/S1/2-62
DRY_RUN_FEED_MODE, MD 42101, 1/V1/4-63
DRYRUN_MASK, 1/V1/2-34
Dx, 3/G3/2-10
Dyn. stiffness control: delay, 2/K3/4-105
Dynamic
  MMC property, 2/B3/1-13
  Response adaptation, 2/S3/2-33
  Switchover, 2/B3/1-6
Dynamic response adaptation, 1/G2/2-45
Dynamic Stiffness Control Config., MD 32642, 2/K3/4-105

E

EG, Electronic gear, 3/M3/2-38
EG axis grouping
  Activating, 3/M3/2-48
  Defining, 3/M3/2-47
  Deactivating, 3/M3/2-52
  Deleting, 3/M3/2-53
EG axis groupings, 3/M3/2-39
Electronic gear, 3/M3/1-6, 3/M3/2-38
  System variables, 3/M3/2-54
Electronic weight compensation, 2/K3/2-89
EMERGENCY STOP, 1/N2/1-3, 1/N2/2-5
  Acknowledgment, 1/N2/2-8
  PLC/NCK, 1/N2/2-6
  Sequence, 1/N2/2-7
Enable pulses, 1/D1/2-18
ENABLE_ALARM_MASK, 1/D1/4-37
ENABLE_CHAN_AX_GAP, 1/K2/4-92
ENC_IS_DIRECT2, 1/S1/4-76
Encoder coding, 1/G2/2-37
Encoder monitoring functions, 1/A3/2-21
  Encoder frequency, 1/A3/2-21
  Zero marks, 1/A3/2-22
Encoder switchover, 3/K6/2-7
End of motion criterion with block search, 2/P2/2-27
Equidistant index intervals, 2/T1/2-12
Error, 1/A2/5-78, 1/A2/5-81
Error RS232C, 1/A2/5-80

© Siemens AG, 2003. All rights reserved

ESR, 3/M3/2-60
Activation, 3/M3/2-69
Logic operation, 3/M3/2-69
Trigger sources, 3/M3/2-68
ESR_DELAY_TIME1, MD 21380, 3/M3/4-87
ESR_DELAY_TIME2, MD 21381, 3/M3/4-88
ESR_REACTION, MD 37500, 3/M3/4-89
Euler, 3/TE4/2-43
Euler angle, 3/F2/2-23
Eventcontrolled program sequences, 1/K1/2-80
Eventdriven signal exchange, 1/P3/1-5, 1/P3/1-6
Exact stop, 1/B1/2-9
Exact stop criteria, 1/B1/2-10
Exact stop coarse, 1/A3/2-8, 1/B1/2-11
Exact stop fine, 1/A3/2-9, 1/A3/2-10, 1/B1/2-11
Exact stop criterion, 1/B1/2-13
Exact stop limit fine, 1/A3/2-8
EXACT_POS_MODE, MD 20550, 1/B1/4-54
Example
Cont. mesurement on completion of prog. traversing movement, 2/M5/6-74
Continuous measurements modally over several blocks, 2/M5/6-75
Continuous measurements with deletion of distance-to-go, 2/M5/6-75
Measuring mode 1, 2/M5/6-73
Measuring mode 2, 2/M5/6-74
One operator panel, three NCUs, 2/B3/6-170
TRAANG, 2/M1/6-83
TRACYL, 2/M1/6-80
TRANSMIT, 2/M1/6-79
Two operator panels, one NCU, 2/B3/6-169
Example of auxiliary function output, 1/H2/2-20
Example of functional test, 2/M5/6-76
Example of functional test on probe, 2/M5/2-65
Example: Clearance control, 3/TE1/6-45, 3/TE5/6-19, 3/TE6/6-23
Example: Tangential control, 3/T3/6-19, 3/TE2/6-17, 3/TE3/6-53
Execute external subroutine, 1/K1/2-109, 1/K1/2-110
Execution from external source, extension (SW 4 and higher), 1/K1/2-109
Expansions to punching and nibbling functions, 2/N4/2-15
EXPONENT_LIMIT, 1/A2/4-52
EXPONENT_SCIENCE, 1/A2/4-52
EXTCALL, 1/K1/2-110
Extended stop and retract, 3/M3/2-60
External data communications connector, 1/K4/1-3
External protection zone, 1/A3/2-38
External subroutine, 1/K1/2-110

F

Face milling, 3/W5/2-18
Factor of acceleration reduction, 2/S6/4-20
Fast analog NCK inputs, 2/A4/2-20
Faults, 2/B3/1-18
FB 10: Safety relay, 1/P3/4-152
FB 11: Brake test, 1/P3/4-155
FB 2: GET, 1/P3/4-102
FB 29: Signal recorder and data trigger diagnostics, 1/P3/4-159
FB 3: PUT, 1/P3/4-109
FB 4: PI_SERV General PI services, 1/P3/4-115
FB 5: GETGUD Read GUD variable, 1/P3/4-138
FB 7: PI_SERV2 General PI services, 1/P3/4-143
FB 9 M:N control unit switchover, 1/P3/4-147
FB 9: Operating unit switchover, 1/P3/4-89
FC 10: AL_MSG, 1/P3/4-182
FC 13: BHGDisp, 1/P3/4-185
FC 15: POS_AX, 1/P3/4-188
FC 16: PART_AX, 1/P3/4-193
FC 17: YDelta, 1/P3/4-197
FC 18: SpinCtrl, 1/P3/4-200
FC 19: MCP_IFM, 1/P3/4-209
FC 2: GP_HP, 1/P3/4-163
FC 22: TM_DIR, 1/P3/4-224
FC 24: MCP_IFM2, 1/P3/4-226
FC 25: MCP_IFT, 1/P3/4-230
FC 3: GP_PRAL, 1/P3/4-164
FC 5: GP_DIAG, 1/P3/4-167
FC 7: TM_REV, 1/P3/4-168
FC 8: TM_TRANS, 1/P3/4-171
FC 9: ASUB, 1/P3/4-179
FC9: Start (measuring in Jog mode), 1/A2/5-82
Feed disable, 1/V1/2-26
Feed disable DB21, ... DBX6.0, 1/V1/2-26
Feed FB, Blockwise, 1/V1/2-42
Feed for chamfer/rounding, 1/V1/2-40
Feed override, 1/B1/2-7, 1/R1/2-15
Feed override switch, 1/V1/2-27
Feed stop, 1/K1/2-75
Feed/spindle stop, 1/V1/2-26
Feedforward control, 2/K3/1-3, 2/K3/2-39
Feedrate, 1/B1/2-7, 1/R1/2-15, 2/P2/2-28
Feedrate override, 2/H1/2-6, 2/P2/2-11, 2/P2/2-23, 2/P2/2-27, 2/P2/2-28
Feedrate/rapid traverse override, 2/H1/2-33

© Siemens AG, 2003. All rights reserved

Feedrates
Dry run feed, 1/V1/2-33
Feed control, 1/V1/2-26
Feed disable, 1/V1/2-26
Feed override, 1/V1/2-27, 1/V1/2-32
Feed/spindle stop, 1/V1/2-26
Feeds in one block, 1/V1/2-34
G93, G94, G95, 1/V1/2-9
Path feed F, 1/V1/2-7
Positioning axes FA, 1/V1/2-24
Positioning feed – G94, 1/V1/2-25
Rigid tapping G331/G332, 1/V1/2-23
Several in one block, 2/S5/2-25
Spindle override, 1/V1/2-30
Tapping with compensation chuck G63, 1/V1/2-23
Thread cutting G33, 1/V1/2-14
FGROUP, 1/K2/2-19, 1/K2/2-22
Field weakening, 1/B2/2-15
File, Transfer, 2/F3/2-105
File types
BIN, 1/K4/2-6
BOT, 1/K4/2-6
COM, 1/K4/2-6
CPA, 1/K4/2-6
DIR, 1/K4/2-6
GIA, 1/K4/2-6
GUD, 1/K4/2-6
IKA, 1/K4/2-6
INI, 1/K4/2-6
LUD, 1/K4/2-6
MPF, 1/K4/2-6
OPT, 1/K4/2-6
RPA, 1/K4/2-6
SEA, 1/K4/2-6
SPF, 1/K4/2-6
SYF, 1/K4/2-6
TEA, 1/K4/2-6
UFR, 1/K4/2-6
WPD, 1/K4/2-6
Fine interpolation, 1/G2/2-44
Fine offset, 1/K2/1-9, 1/K2/2-43
Fine/coarse synchronism, 2/S3/2-17
FIPO, 1/G2/2-44
Fixed coupling configuration, 2/S3/2-6
Fixed feed values, 1/V1/2-39
Flat D number structure, 1/W1/2-19
D number programming, 1/W1/2-21
Read D number from the PLC, 1/W1/2-20
Floppy disk drive, 1/K4/2-10
FM, 1/P3/8-268
FM 354 as 5th axis, 2/L1/2-8
Interfaces and front-panel elements, 2/L1/2-9
Machine data, 2/L1/2-10
FM-NC, 2/B3/1-22
Following error, 1/D1/2-9
Following error compensation, 2/K3/2-39
(feedforward control)
Axial following errors, 2/K3/2-39
Feedforward control methods, 2/K3/2-39
Speed feedforward control, 2/K3/2-41
Torque feedforward control, 2/K3/2-45
Dynamic response adaptation, 2/K3/2-47
Parameter, 2/K3/2-40
Parameters, 2/K3/2-46
Following error compensation (feedforward control), speed feedforward control, 1/G2/2-51
Followup, 1/A2/2-21
Followup mode, 1/A2/2-20, 1/A2/5-87, 1/A3/2-6, 1/A3/2-11
Followup mode active, 1/A2/2-30, 1/A2/5-97
Foreground language, 1/A2/2-35
FRAME, 1/K2/1-5, 1/K2/2-15, 1/K2/2-42
FRAME chaining, 1/K2/2-67
MIRROR, 1/K2/2-54
SCALE, 1/K2/2-54
Frame, 3/TE4/2-6
Friction Compensation, (quadrant error compensation), Start-up, 2/K3/2-50
Friction compensation, 2/K3/2-48
(quadrant error compensation)
Amplitude adaptation, 2/K3/2-49
Characteristic parameters, 2/K3/2-50
Quadrant errors, 2/K3/2-48
Conventional, 2/K3/2-49
Full-/half-step, 2/S6/3-17
Function overview of inch/metric switchover (SW 5 and higher), 1/G2/2-21
Data backup, 1/G2/2-19
Rounding machine data, 1/G2/2-20
Synchronized actions, 1/G2/2-15
Function Retrace Support, 3/TE7/2-5
Function viewer PC MS-DOS, 2/F3/2-112
Functional testing of probe, 2/M5/2-65
Functionality, 3/V2/2-5

G

G groups, 1/K1/2-69
G0 logic in G96, 1/V1/4-60
G40, 1/W1/2-39
G41, 1/W1/2-39
G42, 1/W1/2-39
G450, 3/W5/2-10
G450/G451, 1/W1/2-54
G451, 1/W1/2-56, 3/W5/2-13
G460, 1/W1/2-68
G461, 1/W1/1-4, 1/W1/2-69
G462, 1/W1/1-4, 1/W1/2-69
G58, 1/K2/1-9, 1/K2/2-43
G59, 1/K2/1-9, 1/K2/2-43

© Siemens AG, 2003. All rights reserved

G91, Frames, 1/K2/3-82
G91 Extension, Machine zero, 3/F2/2-61
G91 extension, offset external (WOE), 1/W1/2-97
Gantry Axes
Start-up, 3/G1/2-18
Terms, 3/G1/2-5
Gantry axes, 3/G1/1-3
Differences between “gantry axes” and “coupled axes” functions, 3/G1/2-25
Referencing and synchronization, 3/G1/2-10
Gap, 1/K1/2-11
GC, 3/G3/2-10
Gear stage change from NC, 1/S1/2-28
Gear stages, 1/S1/2-28
General, 2/M5/2-5
General machine data, 1/A3/3-57, 2/K5/4-27, 3/V2/4-13
General notes, 3/V2/2-5
General reset, 2/B3/1-19
General sources, 3/M3/2-68
Generator operation, 3/M3/2-72, 3/M3/2-78
Generic 5-axis transformation, 3/F2/2-42
Geometry axes, 2/H1/2-7
in JOG mode, 2/H1/2-33
Geometry axes, 1/B1/2-5, 1/B1/2-14, 1/B1/2-18, 1/K2/2-15, 1/K2/2-19, 1/K2/2-35, 1/P1/1-3
Geometry model, 1/D2/2-6
Geometry monitoring, 2/W4/2-27
Global control telegramm, 3/G3/2-10
Global data, 2/B3/1-16
GMMC_INFO_NO_UNIT, 1/K1/4-143
GMMC_INFO_NO_UNIT_STATUS, 1/K1/4-143
Grinding operations, 2/M1/2-40
Grinding tools, 2/W4/2-5, 2/W4/2-14
Grinding-specific tool monitoring, 2/W4/2-27
Groove machining, 2/M1/2-28
Groove side compensation, 2/M1/1-6
Groove traversing-section, 2/M1/2-30
GroupSerupro, 1/K1/2-42, 1/K1/2-45
GWPS, 2/W4/2-30
in all operating modes, 2/W4/2-32

H

Hand axes
Configuration, 3/TE4/2-13
Parameterization, 3/TE4/2-13
Handheld programming unit (HPU), 2/B3/1-13, 2/B3/1-16
Handheld unit (HHU), 1/K4/2-11
Handling Transformation Package
Alarms, 3/TE4/7-76
Brief Description, 3/TE4/1-3
Channel-specific machine data of standard system, 3/TE4/4-57
Channel-specific signals, 3/TE4/5-69
Configuring data, 3/TE4/2-17
Creating alarm texts, 3/TE4/3-53
Data Description, 3/TE4/4-57
Data Fields, Lists, 3/TE4/7-75
Detailed Description, 3/TE4/2-5
Example, 3/TE4/6-71
General machine data, 3/TE4/2-9
Interface signals, 3/TE4/7-75
Limitations of function, 3/TE4/3-54
NC machine data, 3/TE4/7-75
Options, 3/TE4/3-53
Signal Descriptions, 3/TE4/5-69
Start-up, 3/TE4/6-71
Supplementary conditions, 3/TE4/3-53
Handwheel
840Di, 2/H1/4-43
Assignment, 2/H1/2-15
Connection, 2/H1/2-15
Path definition, 2/H1/2-28
Selection from MMC, 2/H1/2-16
Traversal in JOG, 2/H1/2-15
Velocity specification, 2/H1/2-28
Via actual-value input, 2/H1/2-25
Handwheel override in AUTOMATIC mode
Path definition, 2/H1/2-19
Programming and activation, 2/H1/2-22
Velocity override, 2/H1/2-20
Hardware configuration
Internet, 2/F3/1-18
ISDN, 2/F3/1-15
Modem, 2/F3/1-15
Network, 2/F3/1-17
Windows with Gateway, 2/F3/1-19
Hardware faults, 1/A3/2-23
Hardware limit switches, 1/A3/2-24, 2/H1/2-35
Heatsink temperature prewarning, 1/A2/2-34, 1/A2/5-106
Heatsink temperature warning, 1/D1/2-21
Helical interpolation, 1/K2/2-22
Helix interpolation, 1/K2/2-22
HHU, 2/B3/1-16
Hide several program levels, 1/K1/4-131
High-speed data channel, 1/A2/2-41
Higraph first error display, 1/A2/5-73
Hirth tooth system, 2/T1/2-13

© Siemens AG, 2003. All rights reserved

HMI Embedded, 1/K4/2-6
Hold, 1/A2/2-21
Homogeneous network, 2/B3/1-20
Host computer, 2/B3/1-6, 2/B3/2-34
HPU, 2/B3/1-13, 2/B3/1-16
HPU address, 1/K4/2-13

I

I/O device interface, 2/M5/2-10, 2/M5/2-12
ID check, 1/K1/2-129
Identification, operator panels, 2/B3/2-58
Identification of axis sequence, 3/F2/2-18
Implicit continuouspath mode, 1/B1/2-17
Implicit exact stop, 1/B1/2-14
INC, 1/P1/2-6, 2/T1/2-5, 2/T1/2-6
Inch measuring system, 1/G2/2-13
Inch or metric units of measurement, 2/M5/2-24
Inclined axis, 2/M1/3-61
  TRAANG, 2/M1/2-40
Incremental jogging, 2/H1/2-12
Incremental jogging (INC)
  Continuous mode, 2/H1/2-12
  Jog mode, 2/H1/2-12
Independent drive, 3/M3/2-73
Independent drive generator operation, 3/M3/2-62
Independent drive reactions, 3/M3/2-62
Independent drive retract, 3/M3/2-62, 3/M3/2-74
Independent drive stop, 3/M3/2-63
Indexing axes, 2/T1/2-5
  Coded position, 2/T1/2-10
  Coded positions, 2/T1/2-7
  Continuous traversal, 2/T1/2-6
  Handwheel, 2/T1/2-6
  Incremental traversal (INC), 2/T1/2-6
  Parameterization, 2/T1/2-9
  Programming, 2/T1/2-10
  Reference point approach, 2/T1/2-5, 2/T1/2-21
  Start-up, 2/T1/2-18
  Traversal by PLC, 2/T1/2-8
Infeed, 2/P5/1-3
  at 2 reversal points, 2/P5/2-21
  at reversal point, 2/P5/2-19
  in reversal point range, 2/P5/2-19
Influence path axes individually, 1/K1/2-67
Influence SERUPRO behavior, 1/K1/2-37
Initial learning, 2/K3/2-69
Initial setting, 1/B1/2-7
Initial settings, 1/K1/2-69
Input resolution, 1/G2/2-9
Input Time, 3/G3/2-10
Input values, 2/M5/2-16
Insertion depth ISD, 3/W5/2-8
Installing
  Remote diagnosis on the control system, 2/F3/1-51, 2/F3/1-53, 2/F3/2-65
  the PC card MMC 100.2, 2/F3/2-76, 2/F3/2-77, 2/F3/2-83
  the TCP/IP protocol, 2/F3/1-34
Integrator disable, 1/A2/2-33, 1/D1/2-19
Integrator ncontroller disabled, 1/A2/2-34
Interactive Programming
  Adaptation of simulation data (SW 3.1 and higher), 1/D2/2-14
  Adaptation of simulation data (SW 3.2 and higher), 1/D2/2-18
  Adapting the technology memory, 1/D2/2-10
  Archiving of configuration data, 1/D2/2-31
  Area of application, 1/D2/2-5
  Axisspecific machine data, 1/D2/7-40
  Basic procedure, 1/D2/1-3
  Channelspecific machine data, 1/D2/7-39
  Compatibility, 1/D2/2-7
  Configuring the user interface, 1/D2/2-8
  Creation of a tool catalog, 1/D2/2-11
  Customizing the axis names, 1/D2/2-13
  Example, 1/D2/6-37
  General conditions, 1/D2/2-7
  General machine data, 1/D2/7-39
  Matching the tool offset data, 1/D2/2-27
  Milling, 1/D2/2-12
  Restriction, 1/D2/2-7
  Setting up, 1/D2/2-12
  Simulation data, special conditions, 1/D2/2-22
  Supplementary conditions, 1/D2/3-33
  Turning, 1/D2/2-12
Interactive programming (D2), 1/D2/1-3
Interface, 2/B3/1-11, 2/H1/2-8, 2/M5/2-10
Interface signal, 840Di, 1/A2/2-9
Interface signal, PC Shutdown, 1/A2/2-9
  Channelspecific NCK alarm is active, 1/A2/5-84
Interface signals, 2/K5/5-29
  Activate channelspecific protection zone, 1/A3/5-73
  Activate dry run feed, 1/V1/5-66
  Activate M01, 1/K1/5-175
  Activate machinerelated protection zone, 1/A3/5-73
  Activate program test, 1/K1/5-176
  Activate referencing, 1/R1/5-65
  Activate single block, 1/K1/5-175
  Active drive parameter set, 1/A2/2-34
  Active G function of groups 1 to 60, 1/K1/5-188
  Active machine function REF, 1/K1/5-175
  Active machine function REPOS, 1/K1/5-174
  Active machine function TEACH IN, 1/K1/5-174

© Siemens AG, 2003. All rights reserved

Active mode AUTOMATIC, 1/K1/5-173
Active mode JOG, 1/K1/5-173
Active mode MDA, 1/K1/5-173
Active spindle control mode, 1/S1/5-107
Active spindle mode oscillation mode, 1/S1/5-107
Active spindle positioning mode, 1/S1/5-107
Actual direction of rotation clockwise, 1/S1/5-105
Actual gear stage A to C, 1/S1/5-99
All axes referenced, 1/R1/5-66
All channels in Reset state, 1/K1/5-174
AUTOMATIC mode, 1/K1/5-169
Block search active, 1/K1/5-184
Change gear, 1/S1/5-104
Channel status active, 1/K1/5-187
Channel status interrupted, 1/K1/5-187
Channel status reset, 1/K1/5-187
Channelspecific protection zone preactivated, 1/A3/5-74
Channelspecific protection zone violated, 1/A3/5-75
Clamping in progress, 1/A3/5-69
Delete S value, 1/S1/5-98
Dry run feedrate selected, 1/V1/5-71
Enable protection zones, 1/A3/5-72
Encoder limit frequency exceeded, 1/A3/5-71
F function for positioning axis, 1/V1/5-79
Feed disable, 1/V1/5-70
Feed hold (Geometry axis 1 to 3), 1/V1/5-71
Feed override, 1/V1/5-66
Feed override (axisspecific), 1/V1/5-73
Feed override active, 1/V1/5-71
Feed override for rapid traverse selected, 1/V1/5-72
Feedrate hold/spindle stop (axisspecific), 1/V1/5-78
For operator panel, 1/A2/2-15
Gear changed, 1/S1/5-99
Hardware limit switches plus and minus, 1/A3/5-70
Invert M3/M4, 1/S1/5-100
JOG mode, 1/K1/5-169
M function for spindle, 1/S1/5-108
M00/M01 active, 1/K1/5-183
M01 selected, 1/K1/5-181
M02/M30 active, 1/K1/5-184
Machine function REF, 1/K1/5-172
Machine function REPOS, 1/K1/5-171
Machine function TEACH IN, 1/K1/5-171
Machinerelated protection zone preactivated, 1/A3/5-73
Machinerelated protection zone violated, 1/A3/5-74
MDA mode, 1/K1/5-169
Measuring status, 2/M5/5-71
Mode group ready, 1/K1/5-174
Mode group reset, 1/K1/5-171
Mode group stop, 1/K1/5-170
Mode group stop axes plus spindles, 1/K1/5-170
Mode switchover disable, 1/K1/5-170
n (act) < n(x), 1/A2/2-35
NC Start, 1/K1/5-178
NC start disable, 1/K1/5-178
NC Stop, 1/K1/5-179
NC Stop at block limit, 1/K1/5-178
NC Stop axes plus spindles, 1/K1/5-180
Oscillation speed, 1/S1/5-101
Oscillation via PLC, 1/S1/5-101, 1/S1/5-102, 1/S1/5-103
Override active, 1/V1/5-77
Probe actuated, 2/M5/5-71
Program level abort, 1/K1/5-178, 2/K5/5-29
Program status aborted, 1/K1/5-187
Program status interrupted, 1/K1/5-186, 1/K1/5-189
Program status running, 1/K1/5-185
Program status stopped, 1/K1/5-186
Program status wait, 1/K1/5-186
Program test active, 1/K1/5-185
Program test selected, 1/K1/5-181
Rampfunction generator rapid stop active, 1/A2/2-33
Rapid traverse override, 1/V1/5-68
Rapid traverse override active, 1/V1/5-70
Readin disable, 1/K1/5-177
Reference point approach delay, 1/R1/5-67
Reference point value 1 to 4, 1/R1/5-66
Referenced/synchronized 1, 1/R1/5-67
Referenced/synchronized 2, 1/R1/5-67
Referencing active, 1/R1/5-65
Reset, 1/K1/5-181
Resynchronize spindle 1 and 2, 1/S1/5-98
Resynchronize spindle during positioning 2 and 1, 1/S1/5-100
Revolutional feedrate active, 1/V1/5-79
Rigid tapping active, 1/S1/5-108
S function for spindle, 1/S1/5-108
Second software limit switch plus or minus, 1/A3/5-70
Selected machine function REF, 1/K1/5-173
Selected mode AUTOMATIC, 1/K1/5-172
Selected mode JOG, 1/K1/5-172
Selected mode MDA, 1/K1/5-172
Selected mode TEACH IN, 1/K1/5-172, 1/K1/5-173
Set counterclockwise/Set clockwise, 1/S1/5-100
Set gear stage A to C, 1/S1/5-104
Setpoint speed increased, 1/S1/5-106
Setpoint speed limited, 1/S1/5-106
Skip block, 1/K1/5-176
Speed limit exceeded, 1/S1/5-106
Spindle in setpoint range, 1/S1/5-105
Spindle override, 1/V1/5-75

© Siemens AG, 2003. All rights reserved

Spindle reset/Delete distancetogo, 1/S1/5-97
Spindle/no axis, 1/S1/5-103
Velocity/spindle speed limitation, 1/A3/5-70
Zero offset, 1/K2/5-102
Interface signals (A2), 1/A2/1-3
Interface signals for digital drives, 1/A2/2-32
Interface signals from and to channel, 1/A2/2-19
Interface signals from axis/spindle, 1/A2/2-30
Interface signals to axis/spindle, 1/A2/2-19
Interface signals – Table, 1/A2/7-115
Intermediate block, 3/T3/1-3
Intermediate blocks, 3/W5/2-28
Internal protection zone, 1/A3/2-38
Interpolation, 2/B3/1-27
Interpolation buffer, reduced, 1/K1/2-103
Interpolation cycle, 2/B3/1-8, 2/B3/1-29
Interpolation functions, 2/P2/2-10, 2/P2/6-41
Interpolation of the angle of rotation, 3/F2/2-66
Interpolation of the rotation vector, 3/F2/2-66
Interpolation parameter sets, 1/G2/2-47
Interpolator, 1/K1/2-13
Interpolator cycle, 3/G3/2-7
Interpolator cycle, 840Di, 3/G3/2-10
Interpolator end, 1/B1/2-11
Interpolatory axis grouping, 1/A2/2-27
Interpolatory compensation
Linear interpolation, 2/K3/2-17
Methods, 2/K3/2-15
Interrupt routines, 1/K1/1-6, 1/K1/2-86, 1/K1/4-131
Intersection behavior for 3D compensation, 3/W5/2-13
Invert M3/M4, 1/S1/2-58
INVFRAME, 1/K2/2-68
IPO cycle, 3/G3/2-8
IS “Feedrate stop/spindle stop” (DB31, ... DBX4.3), 2/S3/2-16
ISD, 3/W5/2-8

J

Jerk limit, 1/B1/2-24
Jerk limitation, 1/B1/2-23
Smoothing method, 1/B2/2-13
Jerk limitation on the path, Example, 1/B1/7-63
JOG, 1/K1/2-16, 1/P1/2-6, 2/H1/2-36, 2/T1/2-5
JOG mode, 2/H1/2-5, 2/H1/2-33, 2/T1/2-6

K

Keep tool radius compensation constant, 1/W1/2-63
Key disable, 1/A2/2-15, 1/A2/5-71
Keyboard, 2/B3/1-11
Keyboard type, 1/A2/4-50
KEYBOARD_STATE, 1/A2/4-50
KEYBOARD_TYPE, 1/A2/4-50
Keyswitch, 1/A2/2-12
Keyswitch position, 1/A2/5-67
Kinematic transformation, 1/K2/2-35, 3/F2/2-15, 3/TE4/2-5
Configuration, 3/TE4/2-9
Start-up, 3/TE4/6-74
Kinematics of machines, 3/F2/2-16
Knee-shaped acceleration characteristic, 2/S6/2-8
Activation, 2/S6/2-12
Equivalent path characteristic, 2/S6/2-11
G64-Block transition, 2/S6/2-13
G64-Braking ramp, 2/S6/2-14
Parameterization of axis characteristic, 2/S6/2-8
Path characteristic, 2/S6/2-10
Kneeshaped acceleration characteristic, Selection and deselection, 1/B2/2-16
KONT, 1/W1/2-40
Kv factor, 1/A3/2-5, 1/D1/2-14, 1/G2/2-44

L

Labels, 1/K1/2-79
Language command
SPN, 2/N4/2-22
SPP, 2/N4/2-20
Language commands
ACC, 1/B2/2-17
BRISK, 1/B2/2-8
SOFT, 1/B2/2-8
Large batch production, 2/B3/1-25
Laser cutting: clearance control, 3/TE1/2-5
Last action block active, 1/K1/5-183
Learning
ON / OFF, 2/K3/2-66
the neural network, 2/K3/2-65
LEN_PROTOCOL_FILEX, 11420, 1/D1/4-37
LIFTFAST, 1/K1/2-87
LIFTFAST_WITH_MIRROR, 1/K1/4-148
Limit angle for the fifth axis, 3/F2/2-27
Limit switch monitoring, 1/A3/2-24
Limit value for offsets fine, 1/K2/4-87
Limit value for wear fine, 1/K2/4-87
Linear axis, 2/T1/2-9
with linear scale, 1/G2/2-38
with rotary encoder on motor, 1/G2/2-39
with rotary encoder on the machine, 1/G2/2-40
Linear drives, 1/K4/2-20
Linear feed (G94), 1/V1/2-9
Link axes, 1/K1/2-14
Link axis, 2/B3/1-27, 2/B3/1-28
Link changeover, instant, 2/B3/1-17
Link frames, 3/TE4/2-15
Link module, 2/B3/1-28, 2/B3/1-29
Link variables, 2/B3/1-7
Global, 2/B3/1-29

© Siemens AG, 2003. All rights reserved

Link-changeover, attempt, 2/B3/1-21
Little/big endian representation for PLC I/O, 2/A4/4-54
Load, 1/A2/5-81
Load parts program, 1/A2/5-75
Loader axes, 1/K2/2-19
Local NCU, 2/B3/1-22
Local P-bus, 2/L1/2-5
Configuration, 2/L1/2-5, 2/L1/2-6, 2/L1/2-7
FM 354 as 5th axis, 2/L1/2-7
Inputs/-outputs, 2/L1/2-7
Lockable data areas, 1/A2/2-13
Log book, 1/D1/2-29
Longitudinal grooves, 2/M1/1-6
Look Ahead, 1/B1/2-13, 1/B1/2-42
Block cycle problem, 1/B1/2-46
Continuouspath mode, 1/B1/2-42
Exact stop, 1/B1/2-42
Following block velocity, 1/B1/2-44
Override, 1/B1/2-45
Selection and deselection, 1/B1/2-46
Velocity profiles, 1/B1/2-44
LOOKAH_SMOOTH_WITH_FEED, MD 20462, 1/B1/4-51
LookAhead, Number of blocks, 1/B1/2-43
Lubrication pulse, 1/A2/2-31, 1/A2/5-100
LVDS, 1/K4/1-3

M

M command, 2/W3/1-3
M decoding acc. to list, 1/P3/2-58
M:N concept, 2/B3/1-10
M:N switchover, 2/B3/1-13
M(d) less than M(dx), 1/A2/5-108
M_NO_FCT_CYCLE, 1/K1/4-137
M_NO_FCT_CYCLE_NAME, 1/K1/4-137
MA_AXES_SHOW_GEO_FIRST, 1/K1/4-131
MA_MAX_SKP_LEVEL, 1/K1/4-132
MA_ORIAXIS_EULER_ANGLE_NAME, 1/K2/4-84
MA_PRESET_MODE, 1/K1/4-132
MA_PREST_FRAMEIDX, 1/K2/4-84
MA_SCRATCH_DEFAULT_MODE, 1/K2/4-86
MA_STAT_DISPLAY_BASE, 1/K2/4-83
MA_TU_DISPLAY_BASE, 1/K2/4-83
Machine
Control panel, 2/B3/1-11, 2/H1/2-8
Rear panel, 2/B3/1-20
State, global, 2/B3/1-29
Machine axes, 1/K2/2-14
Machine coordinate system, 1/A2/2-16
Machine coordinate system (MCS), 1/K2/1-7, 1/P1/2-6
Machine coordinate systems (MCS), 1/K2/2-34
Machine data, 1/A2/7-118, 3/V2/7-20
10134 (MM_NUM_MMC_UNITS), 2/B3/1-18
20000 (CHAN_NAME), 2/B3/1-19
for analog inputs-/outputs, 2/L1/4-18
for digital inputs-/outputs, 2/L1/4-18
for the 5th axis, 2/L1/4-17
TRAANG, 2/M1/4-71
TRACYL, 2/M1/4-69
Transformation-specific, 2/M1/4-64
TRANSMIT, 2/M1/4-67
Machine data for operator panel, 1/A2/4-47
Machine data for processing programs from external, 1/K1/4-143
Machine data in the transformation standard set, channel-specific, 3/TE4/4-58
Machine data, channelspecific, 1/A2/4-60
Machine data, general, 1/A2/4-58
Machine kinematics, 3/F2/2-42
Swiveling linear axis, 3/F2/1-9
Machine tool axes, 1/K2/2-19
Machine types, 3/F2/2-16, 3/F2/2-18, 3/F2/2-43
Machine zero, 1/R1/1-3
Machining, face-end, 2/M1/1-5
Main axes, 1/K2/2-21
Main control panel, 2/B3/1-14
Main technological application, 1/K1/2-15
Manual specification of a gear stage, 1/S1/2-33
Manual stroke initiation, 2/N4/5-43
Manual switchover of the basic system
General, 1/G2/2-17
Input resolution and calculation resolution, 1/G2/2-19
JOG and handwheel evaluation, 1/G2/2-19
Reference point, 1/G2/2-18
System data, 1/G2/2-18
Tool data, 1/G2/2-18
Manual switchover of the basic system (SW 5 and higher), 1/G2/2-17
Manual traverse in JOG, 2/H1/2-5
Master Application Cycle, 3/G3/2-10
Master axis, 3/G1/2-5
Master computer, 2/B3/1-7
Master Time, 3/G3/2-10
Master value coupling
Axis types, 3/M3/2-30
Interface signals, 3/M3/2-37
Programming, 3/M3/2-33
Response in operating modes, 3/M3/2-36
Master, slave communication, 2/B3/1-6
Matching the machine data, 1/D2/2-16
Material memory, 1/D2/2-10
Maximum permissible path feed for compression, 1/V1/4-59
Maximum stepper motor frequency, 2/S6/3-17, 2/S6/4-19
MAXNUM_USER_DATA_FLOAT, 1/P3/9-284

© Siemens AG, 2003. All rights reserved

MAXNUM_USER_DATA_HEX, 1/P3/9-283
MAXNUM_USER_DATA_INT, 1/P3/9-283
MCP, 2/B3/1-11, 2/B3/1-17
  Control unit, 2/B3/1-6
MCP changeover, 2/B3/1-14, 2/B3/2-50
MCP switchover disable, 2/B3/5-141
MCS, 1/A2/2-16
MCS Coupling: Brief Description, 3/TE6/1-3
MCS coupling: Supplementary Conditions, 3/TE6/3-13
MD 37500, ESR_REACTION, 3/M3/4-89
MD_FILE_STYLE, 1/A2/4-60
MDA (MDI), 1/K1/2-16
MEAC, 2/M5/2-63
MEAS, MEAW commands, 2/M5/2-13
MEAS_PROBE_LOW_ACTIVE, 2/M5/4-69
MEAS_PROBE_SOURCE, 2/M5/4-69
MEAS_TYPE, 2/M5/4-69
Measurement
  Accuracy, 2/M5/2-65
  Results, 2/M5/2-61
Measuring
  Cycles, 2/M5/2-22
  Mode 1, 2/M5/2-59
  Mode 2, 2/M5/2-59
  Points, 2/M5/2-17
Measuring systems, 1/G2/2-24
Memory
  Capacity, Hardware configuration 840Di, 2/S7/2-7
  Configuration, 2/S7/2-8
  Organization, 2/S7/2-8
Memory required, 3/V2/2-8
Memory requirements of basic PLC program, 1/P3/6-254
  Maximum, 1/P3/6-256
  Minimum, 1/P3/6-256
Menu
  Connections/Service, 2/B3/1-18
  Start-up/MMC/Operator panel, 2/B3/1-19
Message signals in DB2, 1/P3/5-247
Metric measuring system, 1/G2/2-13
Milling/drilling unit, 2/B3/1-25
Minimum speed, 1/D1/2-23
Minimum time interval between two consecutive strokes, 2/N4/2-16, 2/N4/4-39
MIRROR(C), 2/R2/2-21
MIRROR_TOOL_LENGTH, 1/W1/4-149
MIRROR_TOOL_WEAR, 1/W1/4-149
Mirroring, Retraction direction (lift fast), 1/K1/2-87
Mixing axes of setpoint output with actual value input, 2/L1/3-15
MM_ABSBLOCK, 1/K1/4-164
MM_ABSBLOCK_BUFFER_CONF, 1/K1/4-164
MM_NUM_MMC_UNITS, 2/B3/1-18
MM_NUM_VDIVAR_ELEMENTS, 1/P3/9-285
MM_SERVO_FIFO_SIZE, MD 18720, 2/B3/4-125, 2/B3/4-128
MM_SYSTEM_FRAME_MASK, 1/K2/4-99
MMC, 1/K1/2-128, 1/K1/2-129, 2/B3/1-18, 2/B3/1-19
  Control unit, 2/B3/1-6
  Operator action, 2/B3/1-13
  PLC interfaces, 2/B3/1-11
  Properties static/dynamic, 2/B3/1-13
  State, 2/B3/1-11
  Switchover, 2/B3/2-43
MMC 1 requests active operating mode, 2/B3/5-141
MMC 1 switchover disable, 2/B3/5-141
MMC 100 / MMC 102 / 103, 2/B3/1-16
MMC 100.2/102/103, 2/B3/1-11
MMC alarm, 1/A2/2-8
MMC alarm is active, 1/A2/5-68
MMC CPU1 Ready, 1/A2/2-7, 1/A2/5-68
Modal activation (FOCON/FOCOF), 1/F1/2-23
Modal execution from external source, 1/K1/2-109
Modally active synchronized actions, 1/K1/2-16
Mode change disable, 1/A2/5-77
Mode group, 1/K1/2-9, 1/P3/2-51, 2/K5/1-3, 2/K5/2-5
  Number, 1/K1/2-11
  Ready to operate, 1/K1/2-12
  Reset, 1/K1/2-12
Mode group number, 1/A2/5-82
Mode groups, 1/K1/2-16
  Group, 1/K1/1-5, 1/K1/2-10
  Interlocks, 1/K1/2-18
  Monitoring functions, 1/K1/2-18
  Operating mode switchover, 1/K1/2-19, 1/K1/2-130
Modes, 2/H1/2-36
Modified activation of machine data, 2/T1/2-15
Modify the torque FXS, 1/F1/2-22
Modular machine concept, 2/B3/1-5
Modulo, 2/T1/2-9
Modulo 360, 2/R2/2-9
Monitor type, 1/A2/2-35
Monitoring functions, 2/H1/2-35
Monitoring functions in the modes, 1/K1/2-18
Monitoring of
  Stepper motor rotation with BERO, 2/S6/2-7
  The input signal, 2/N4/2-15
Monitoring of static limits, 1/A3/2-24
Mono-directional probe, 2/M5/2-6
Motion control, 2/B3/1-27
Motion monitoring functions, 1/A3/2-5
Motion specification, 1/S1/2-58
Motion-synchronous actions, 2/S5/2-6
  Conditions, 2/S5/2-8
  Principle, 2/S5/2-6
  Real-time variables, 2/S5/2-10
  Structure, 2/S5/2-7
Motion-synchronous conditions, 2/P5/2-17
Motor, 1/A2/2-32
Motor selection, 1/D1/2-20

© Siemens AG, 2003. All rights reserved

Motor selection A, B, 1/A2/5-94
Motor selection in progress, 1/A2/5-95
Motor temperature, 1/D1/2-19
Motor temperature warning, 1/A2/2-34, 1/A2/5-105, 1/D1/2-21
MPI, 2/B3/1-12, 2/B3/1-16, 2/B3/1-21
MPI, Network rules, 2/B3/1-24
MPI/DP, interface, 1/K4/2-11
MPI/DP interface (RS485), 1/K4/2-10
Multi-directional probe (3D), 2/M5/2-6
Multi-face machining, 2/B3/1-25
Multi-point interface (MPI), 2/B3/1-12
Multi-spindle machine, 2/B3/1-25
Multi-spindle turning machine, 2/B3/1-5
Multiple feed values in one block, 1/V1/2-34
Multiple feedrate values in one block, 2/S5/2-25
Multiple starts with SERUPRO, 1/K1/2-42

N

n(act) equals n(set), 1/A2/5-110
n(act) less than n(min), 1/A2/5-110
n(act) less than n(x), 1/A2/5-110
NC
  Address, 2/B3/1-12, 2/B3/1-16
  NC instruction, 1/K1/2-70
  NC Start, 1/K1/2-70, 1/K1/2-128
  Production center, 2/B3/1-5
  Program, 1/K1/2-128
  Stop, 1/K1/2-75
  Reset, 1/K1/2-128, 1/K1/2-130
NC address, 2/B3/1-17
NC failure, 1/P3/2-53
NC Ready, 1/A2/5-69
NC VAR selector, 1/P3/3-78
  Startup, installation, 1/P3/3-88
NC variables, 1/P3/3-83
NC-controlled extended stop, 3/M3/2-64
NC-controlled retraction, 3/M3/2-66
NC-PLC data exchange, 2/B3/1-19
NCK alarm handler, 1/D1/2-5
NCK alarm is active, 1/A2/2-8, 1/A2/5-69
NCK alarm with processing stop, 1/A2/2-8
NCK alarm with processing stop present, 1/A2/5-84
NCK battery alarm, 1/A2/2-8, 1/A2/5-70
NCK CPU Ready, 1/A2/2-7, 1/A2/5-68
NCK I/Os, 2/A4/2-5
NCK machine data, axis/spindlespecific data, 1/A2/4-61
NCspecific signals, 1/A2/5-67
NCU, 2/B3/1-21
  Grouping, 2/B3/1-7
  Link, 2/B3/1-6, 2/B3/1-10, 2/B3/1-25, 2/B3/2-34
  Operation, 2/B3/1-22
  Replacement, 2/B3/1-19
NCU link active, 2/B3/5-146
NCU link axis active, 2/B3/5-146
NCU-NCU communication, 2/B3/1-25
NCU_LINKNO, 2/B3/4-125
NETNAMES.INI, 2/B3/1-11, 2/B3/1-14, 2/B3/1-15, 2/B3/1-17, 2/B3/1-19, 2/B3/2-57
  Syntax, 2/B3/2-57
Network rules, 2/B3/1-13, 2/B3/1-24
Networked NCUs, 2/B3/1-25
Neural quadrant error compensation, 2/K3/2-57
  Optimization, 2/K3/2-72
  Parameterization, 2/K3/2-59
  Start-up, 2/K3/2-69
Noncyclic communication, 1/K4/2-15
NORM, 1/W1/2-40
Normalization, 1/G2/2-7
Normalization of machine and setting data, 1/G2/2-10
Number of
  Analog inputs-/outputs, 2/L1/3-15
  Axes, 2/B3/1-5, 2/B3/1-10
  Bus nodes, 2/B3/1-17
  Chained transformations, 2/M1/2-54
  Direct read inputs bytes of PLC I/Os, 2/A4/4-52
  Direct write output bytes of PLC I/Os, 2/A4/4-52
  Synchronous spindles, 2/S3/2-5
Number of places to be displayed without exponent, 1/A2/4-52
Numerical basis for display of moving joint STAT, 1/K2/4-83
Numerical basis for display of rotary axis position TU, 1/K2/4-83

O

O.K., 1/A2/5-79, 1/A2/5-80
OEM solution, 2/B3/1-22
Offline
  Request, 2/B3/2-44
  State, 2/B3/1-14
Offset number, 1/W1/2-11
Online
  Changeover, 2/B3/1-17
  State, 2/B3/1-14
Online tool offset, 2/W4/2-18
Online tool radius compensation, 2/W4/2-26
OP030, 2/B3/1-16, 2/B3/1-20
OP030/OP031/OP032, 2/B3/1-11
Operating
  Area, 2/B3/1-17, 2/B3/1-18, 2/B3/1-21, 2/B3/1-22
  Unit, 2/B3/1-5

© Siemens AG, 2003. All rights reserved

Operating mode changeover, 2/M5/2-58
Operating mode changeover rejected, 2/B3/5-142
Operator, 2/B3/1-14
Component, 2/B3/1-18
Components Manual, 2/B3/1-12, 2/B3/1-23
Display, 2/B3/1-21
Interface, 2/B3/1-12
Panel, 2/B3/1-10, 2/B3/1-11, 2/B3/1-12, 2/B3/1-14, 2/B3/1-16, 2/B3/1-17, 2/B3/1-18, 2/B3/1-19, 2/B3/1-20, 2/B3/1-21, 2/B3/1-23
Panel interface (OPI), 2/B3/1-12
Screen, 2/B3/1-22
Station, 2/B3/1-10
Operator panel, Main secondary, 2/B3/1-11
Operator panel reset, 1/K1/2-81, 1/K1/2-83
Operator's Guide, 2/B3/1-12
OPI, 2/B3/1-12, 2/B3/1-16, 2/B3/1-23
Network rules, 2/B3/1-24
Options in synchronous mode, 2/S3/2-6
ORIC, 3/W5/2-10, 3/W5/2-25
ORID, 3/W5/2-12
Orientation, 1/A3/2-36, 3/F2/2-48
Orientation axes, 1/B1/2-18, 3/F2/1-11, 3/F2/1-12, 3/F2/2-56
Definition, 3/F2/1-11
Introduction, 3/F2/1-11, 3/F2/1-12
Programming, 3/F2/2-58
Orientation axis, 1/B1/2-5
Orientation direction, 3/F2/2-65
Orientation in TCS and MCS, 3/F2/2-23
Orientation path in pole vicinity, 3/F2/2-28
Orientation programming, 3/F2/2-23
Orientation transformation, 3/F2/2-15
Orientation vectors, 3/F2/2-62
Orientational toolholder, 1/W1/1-4, 1/W1/2-72, 1/W1/6-153
ORIMKS, 3/F2/2-24, 3/TE4/2-44
ORIWKS, 3/F2/2-24, 3/TE4/2-44
OS, 2/P5/2-10
OSCILL, 2/P5/2-17, 2/P5/2-23, 2/P5/2-24
Oscillation, 2/P5/1-3
Asynchronous, 2/P5/1-3, 2/P5/2-6
Continuous infeed, 2/P5/1-3
Infeed, 2/P5/2-25
PLC control, 2/P5/2-12
With synchronized actions, 2/P5/2-16, 2/P5/6-37, 2/P5/6-39
Oscillation axis, 2/P5/1-3
Oscillation movement
Restarting, 2/P5/2-23
Stopping, 2/P5/2-22
OSCTRL, 2/P5/2-11
OSE, 2/P5/2-12
OSNSC, 2/P5/2-12
OSP, 2/P5/2-10
OST, 2/P5/2-10
Outer corners/inner corners, 3/W5/2-9
Output Time, 3/G3/2-10
Overlaid movement, 2/S3/2-9
Overload factor, 1/B1/2-16
Override, 1/D1/2-10, 1/K1/2-127
Overstore, 1/H2/2-30

P

P-bus, 2/B3/1-8
PA_ZOA_MODE, 1/K2/4-87
Parallel interface, 1/K4/5-32
Parallel interface (Centronics), 1/K4/2-8
Parameter Process Data Object (PPO), 1/K4/2-15
Parameter set (axis), 1/D1/2-11
Parameter sets of the position controller, 1/G2/2-46
Parameterization of basic block display, 1/K1/4-164
Parameterization of machine geometry, 3/TE4/2-10
Parameterization of stepper motor frequency, 2/S6/2-15
Parking axis, 1/D1/2-20
Parts program, 2/B3/1-28, 2/B3/1-29
Parts program end, 1/K1/2-81
Parts program start, 1/K1/2-80, 1/K1/2-113
Parts program start and parts program end, 1/K1/2-82
Parts programs
Parts program interruption, 1/K1/2-71
Skipping of specific parts program blocks, 1/K1/2-25
Passive
File system, 2/S7/2-5
State, 2/B3/1-14
Password, 1/A2/2-11
Resetting, 1/A2/2-11
Setting, 1/A2/2-11
Path axes, 1/B1/2-5, 1/K2/2-19, 2/P2/2-7
Path axis, 1/K1/5-193
Path criterion, 1/B1/2-19
Path definition by handwheel, 2/H1/2-28
Path feed, 1/B1/2-7, 1/G2/2-6
Path-synchronous switch signal output, Brief description, 3/TE8/1-3
Path/synchronous switch signal output, 3/TE8/2-5
PATH_TRANS_JERK_LIM, MD 32432, 1/B1/4-56
Pathrelated jerk limitation, 1/B2/2-7
Acceleration with jerk limitation, 1/B2/2-7
Profile with abrupt acceleration changes, 1/B2/2-9
Selection and deselection, 1/B2/2-8
PDA area, 1/K4/2-15
Peripheral milling, 3/W5/2-8
Peripheral surface transformation, 2/M1/3-61

© Siemens AG, 2003. All rights reserved

PG diagnostics, 2/B3/1-13
Physical axis, 2/B3/1-7
Physical quantities, 1/G2/2-10
PI services, Overview, 1/P3/4-117
PKW area, 1/K4/2-15
PLC
Address, 2/B3/1-12, 2/B3/1-16
Basic routine, 2/B3/1-16, 2/B3/1-17, 2/B3/1-23
Controls axis, 2/P2/5-39
CPU 314, 2/B3/1-12, 2/B3/1-19, 2/B3/1-22
CPU 315, 2/B3/1-12, 2/B3/1-16
Local I/Os, 2/B3/1-6
Master, 2/B3/1-29
PLC communication, 2/B3/1-6, 2/B3/1-7
Service display, 2/M5/2-14, 2/M5/2-62
Slave, 2/B3/1-29
PLC CPUs, properties, 1/P3/2-30
PLC index, 1/A2/5-75, 1/A2/5-76
PLC line offset, 1/A2/5-76
PLC messages, 1/P3/2-42
PLC program, 1/K1/2-127
PLC status, 1/D1/2-30
Change in status, 1/D1/2-30
Status display, 1/D1/2-30
PLC/NCK
840D, 1/P3/2-30
FMNC, 1/P3/2-32
PLC/MCP, 1/P3/2-44
PLC/MMC, 1/P3/2-41
PLC/NCK interface, 1/A2/2-5, 1/P3/2-35
PLC-CPU 315, 2/B3/1-16, 2/B3/1-21
PLC_CYCLIC_TIMEOUT, 1/P3/9-283
PLCCPU, Key data, 1/P3/2-7
PLCIO_IN_UPDATE_TIME, MD 10398, 2/A4/4-53
PLCIO_LOGIC_ADDRESS_IN, MD 10395, 2/A4/4-52
PLCIO_LOGIC_ADDRESS_OUT, MD 10397, 2/A4/4-53
PLCIO_NUM_BYTES_IN, MD 10394, 2/A4/4-52
PLCIO_NUM_BYTES_OUT, MD 10396, 2/A4/4-52
PLCIO_TYPE_REPRESENTATIION, MD 10399, 2/A4/4-54
PO_SYSCLOCK_TIME_RATIO, 3/G3/2-16
Pole, 3/F2/2-27
Polynomial, Intersection procedure, 1/W1/2-67
Polynomial definition, 2/S5/2-19
Polynomial interpolation, 3/F2/2-62
Polynomial, 5th degree, 3/F2/2-66
POS, 1/B1/2-14, 1/K2/2-19, 1/K2/2-20
POSA, 1/K2/2-19, 1/K2/2-20
POSCTRL_DESVAL_DELAY, MD 10065, 2/B3/4-124
POSCTRL_DESVAL_DELAY_INFO, MD 32990, 2/B3/4-130
POSCTRL_SYSCLOCK_TIME_RATIO, 3/G3/2-15
Position, Control, 2/B3/1-27
Position actual value measuring system 1/2, 1/D1/2-21
Position control, 1/G2/2-46
Position control cycle, 3/G3/2-7
840Di, 3/G3/2-10
Position control cycle offset, 840Di, 3/G3/2-11
Position control direction, 1/A3/2-20
Position control loop, 1/G2/2-43
Position controller active, 1/A2/2-30, 1/A2/5-98
Position controller gain, 1/A3/2-9
Position measuring system, 1/A2/2-24
Position measuring system 1 and 2, 1/A2/5-88
Position setpoint, 1/D1/2-9
Position switching signals, 2/N3/2-5
Position-time cams, Properties, 2/N3/2-22
Positioning accuracy of the control system, 1/G2/2-7
Positioning axes, 1/G2/2-6, 1/K2/2-20, 1/P1/2-6, 2/P2/1-3, 2/P2/2-5
Axis interpolator, 2/P2/2-10
Axis-specific signals, 2/P2/2-29
Block change, 2/P2/2-19
Channel-specific signals, 2/P2/2-29
Concurrent positioning axis, 2/P2/2-9
Dependence of positioning axes, 2/P2/2-8
FC15, 2/P2/2-29
Independence of path and positioning axes, 2/P2/2-7
Path interpolator, 2/P2/2-10, 2/P2/2-23
Positioning axis type 1, 2/P2/2-32
Programming, 2/P2/2-30
Type 1 positioning axis, 2/P2/2-7, 2/P2/2-19
Type 2 positioning axis, 2/P2/2-7, 2/P2/2-21
Velocity, 2/P2/2-28
Positioning axis type POS, 1/B1/2-8
Positioning axis type POSA, 1/B1/2-8
Positioning window large, 1/A3/2-9
POSP, 2/P5/2-17, 2/P5/2-25
Power, Supply, 2/B3/1-11
Power failure detection, 3/M3/2-70
Power ON, 2/H1/2-40
Power ON, block search, 1/K1/2-27, 1/K1/7-198
Power supply connection, 2/M5/2-12
Power-up, 2/B3/1-14, 2/B3/1-19
Powerup, 1/K1/2-82
Preconditions, 2/K5/2-14
Preprocessing, Machine data, 3/V2/4-13
Prerequisites for synchronous mode, 2/S3/2-13
Preset actual value memory, 2/M5/2-15
PRG_DEFAULT_DIR, 1/A2/4-49
Probe types, 2/M5/2-5
Process alarm processing, 1/P3/2-52
Processing programs from external, extension (SW 4 and higher), 1/K1/1-6
Profibus, 2/B3/1-16, 2/B3/1-28
DP I/Os, 2/A4/2-6
Interface, 2/B3/1-25
PROFIBUS DP – Address space, 1/K4/2-16

© Siemens AG, 2003. All rights reserved

PROFIBUS DP – Master, Slave, 1/K4/2-15
PROFIBUS shutdown handling, 3/G3/2-18
PROFIBUS_SHUTDOWN_TYPE, 3/G3/2-18
PROG_EVENT_MASK, 1/K1/4-145
PROG_EVENT_MASK_PROPERTIES, 1/K1/4-145
Program
Hide several Program levels, 1/K1/4-131
Program action, 1/K1/2-76
Program control, 1/K1/2-101
Program operation mode, 1/K1/1-6, 1/K1/2-69
Program status, 1/K1/2-73
Program test, 1/K1/2-20
Repeating program sections, 1/K1/2-79
Runtime, 1/K1/1-7, 1/K1/2-127, 1/K1/7-202
Server, 2/B3/1-22
Program code, 1/D2/2-6
Program control, 2/P2/2-8
DRF, 1/K1/2-101
DRY, 1/K1/2-101
M01, 1/K1/2-101
PRT, 1/K1/2-101
ROV, 1/K1/2-101
SBL1, 1/K1/2-101
SKP, 1/K1/2-101
Program coordination, 2/K5/2-6
Program coordination, example, 2/K5/2-8
Program execution without setpoint outputs, 1/K1/2-20
Program interpreter, 1/D2/2-6
Program test, 1/K1/1-5
Program testing and SERUPRO, 1/S1/2-62
Programmable block change, 2/S3/2-22
Programmable contour accuracy, 3/K6/1-5, 3/K6/2-9
Application, 3/K6/2-9
Programmable run-in and run-out paths, 3/TE1/2-17, 3/TE1/2-18
Programmable runin and runout paths, 1/V1/2-15, 1/V1/2-17
Programming, 2/M5/2-60
Programming and parameterizing tools, 1/P3/3-75
Programming area selection, 1/A2/5-73
Programming devices or PCs, 1/P3/3-75
PROTAREA_GEOAX_CHANGE_MODE, 1/A3/3-57
Protection level, 2/B3/1-20
Service, 2/B3/1-18
Protection level for access to tool carrier offsets..., 1/A2/4-52
Protection levels, 1/A2/2-13
Protection levels for MACHINE operating area for access to ..., 1/A2/4-57
Protection levels for PROGRAM operating area for access to ..., 1/A2/4-56
Protection zone
Activation, 1/A3/2-41
Deactivation, 1/A3/2-41
Definition, 1/A3/2-36
Enabling of, 1/A3/2-45
Restrictions, 1/A3/2-52
Protection zone definition, Example, 1/A3/6-77
Protection zone definition with system variables, 1/A3/2-39
Protection zone for switchover of geo axes, 1/A3/3-57
Protection zone violation, 1/A3/2-45
Protection zones, 1/A3/1-3, 1/A3/2-32
2-dimensional, 1/A3/2-33
3-dimensional, 1/A3/2-33
Channelspecific interface signals, 1/A3/7-82
Channelspecific machine data, 1/A3/7-85
Channelspecific signals, 1/A3/5-72
Data storage, 1/A3/2-41
General machine data, 1/A3/7-83
Supplementary conditions, 1/A3/3-55
Protection zones for milling machine, 1/A3/2-35
Protection zones for turning machine, 1/A3/2-34
Protection zones, convex, 1/A3/2-39
Protection zones, concave, 1/A3/2-39
Protocol layers, 2/B3/1-8
PS/2
Keyboard interface, 1/K4/2-9
Mouse interface, 1/K4/2-10
PS/2 keyboard/mouse interface, 1/K4/2-9
Pulse enable, 1/A2/2-33, 1/A2/5-95, 1/D1/2-17
Pulse multiplication factor, 1/G2/2-38
Pulses between two BERO-edges for monitoring stepper motor rotation, 2/S6/4-19
Pulses enabled, 1/A2/2-34, 1/A2/5-104
Punching and nibbling
Language commands, 2/N4/2-11
Path segmentation, 2/N4/2-19

Q

Quadrant error compensation, 2/K3/2-48, 2/K3/2-57
Quantization of characteristic, 2/K3/2-62
Quick start-up, 2/K3/2-77

R

Ramp-function generator rapid stop active, 1/A2/2-33
Rampfunction generator rapid stop, 1/A2/2-32, 1/A2/5-92, 1/D1/2-17

© Siemens AG, 2003. All rights reserved

Rampfunction generator rapid stop active, 1/A2/5-100
Rampup function complete, 1/A2/2-34, 1/A2/5-107, 1/D1/2-21
Rampup phase, 1/D1/2-18, 1/D1/2-24
Rampup times, 1/A2/5-92
Rapid stop (with open, 1/A3/2-7, 1/A3/2-11, 1/A3/2-18, 1/A3/2-19, 1/A3/2-21, 1/A3/2-22
Rapid traverse, 1/B1/2-7, 1/K1/2-128
Rapid traverse override, 1/B1/2-7, 2/H1/2-6, 2/P2/2-28
Rapid traverse override switch, 1/V1/2-28
Read
  Current angular offset, 2/S3/2-11, 2/S3/2-23
  Measurement results in PP, 2/M5/2-14
  Offset, 2/S3/2-16
Read/write NC variables, 1/P3/2-55
Read/write PLC variable, 1/A2/1-3, 1/A2/2-41
Readin disable, 1/K1/2-75
Reading the current coupling status, 2/S3/2-23
Reapproach contour with controlled REPOS, 1/K1/2-67
Reapproach point, 1/K1/2-68
Recall alarm cleared, 1/A2/5-78
Recall alarms, 1/A2/2-16
Reciprocal feed (G93), 1/V1/2-9
Reduction of jerking, 1/B1/2-24
REF, REF active machine function, DB11, ... DBX5.2, 1/R1/2-6
Reference point, 1/R1/1-3
Reference point approach, 2/H1/2-40, 2/S6/1-3, 2/T1/2-5
Reference point approach for stepper motors, 2/S6/2-5
  BERO_edge, 2/S6/2-6
  Synchronization with reference point-BERO, 2/S6/2-5
  With incremental encoder, 2/S6/2-5
  Without incremental encoder, 2/S6/2-5
Referencing
  Absolute offset, 1/R1/2-23
  Axisspecific, 1/R1/2-6
  By means of actual value adjustment, 1/R1/2-33
  By parts program, 1/R1/2-6
  Channelspecific, 1/R1/2-6
  Reference cam, 1/R1/2-22
  Reference cam adjustment, 1/R1/2-14
  Reference mark, 1/R1/2-22, 1/R1/2-23
  With distancecoded marks, 1/R1/2-20, 1/R1/2-25
  With incremental measurement system, 1/R1/2-8
Relearning, 2/K3/2-70
Remote control, 2/F3/2-104
Remote diagnosis
  Availability, 2/F3/3-115
  Communication, 2/F3/1-13
  Exit, 2/F3/2-107, 2/F3/2-111
  General, 2/F3/1-13
  Hardware configuration, 2/F3/1-15
  Help, 2/F3/2-107
  Keyboard, 2/F3/1-13
Remote diagnosis (F3), 2/F3/1-5
  Software installation, 2/F3/1-50, 2/F3/2-65
  Start-up, 2/F3/1-50, 2/F3/2-65
Remote diagnosis MMC 100.2, 2/F3/1-8
Remote diagnostics
  for HT6, 2/F3/1-10
  MMC 100.2, 2/F3/1-10
Replaceable geometry axes, 1/K2/2-15
REPOS, 2/M5/2-58
REPOS Delay Ack, 1/K1/5-192
REPOS mode, 1/K1/2-68
REPOS offset, 1/K1/5-192
REPOS offset valid, 1/K1/5-192
REPOSDELAY, 1/K1/5-192, 1/K1/5-193
Repositioning neutral axes after SERUPRO, 1/K1/2-63
REPOSMODEEDGE, 1/K1/5-182
REPOSMODEEDGEACKN, 1/K1/5-190
Reset, 1/K1/2-75, 1/N2/2-9
Reset and starting response, 1/K1/4-154
RESET command, 1/K1/2-73
Reset response, 1/K1/7-198
Reset time, 1/K1/2-129
Resolutions, 1/G2/2-7
Response to setpoint changes, 2/S3/2-32
Restart, 2/B3/1-17
Retrace Support, 3/TE7/2-5
Retrace/Continue Machining Support: Signal Descriptions, 3/TE7/5-31
Retract, 3/M3/1-7
Retract and stop, 3/M3/2-60
Reversal points, 2/P5/1-3
Revolutional feedrate (G95), 1/V1/2-9
  Revolutional feedrate active, 1/V1/2-10
Rigidity control (DSC), 1/K4/2-20
RMB, 1/K1/5-182
RME, 1/K1/5-182
RMI, 1/K1/5-182
RMN, 1/K1/5-182
RMNOTDEF, 1/K1/5-182
Rotary
  Axis, 2/R2/2-5, 2/T1/2-9
  Button pad, 2/B3/1-11
  Indexing machine, 2/B3/1-5, 2/B3/1-25

© Siemens AG, 2003. All rights reserved

Rotary axes, 1/K2/2-19, 2/R2/1-3
Absolute dimension programming, 2/R2/2-12, 2/R2/2-16
Axis addresses, 2/R2/2-6
Feedrate, 2/R2/2-8
Incremental dimension programming, 2/R2/2-16, 2/R2/2-17
Mirroring, 2/R2/2-21
Modulo 360, 2/R2/2-9
Modulo conversion, 2/R2/2-12, 2/R2/2-16
Operating range, 2/R2/2-7
Positioning display, 2/R2/2-8
Software limit switches, 2/R2/2-21
Start-up, 2/R2/2-19
Units of measurement, 2/R2/2-7
Rotary axis
with rotary encoder on motor, 1/G2/2-41
with rotary encoder on the machine, 1/G2/2-42
Rotation, 3/TE4/2-7
Rotation monitoring, 2/S6/1-3
Error, 2/S6/5-21
ON/OFF, 2/S6/5-21
Rotation of the orientation vector, 3/F2/2-64, 3/F2/2-65
Rough offset, 1/K2/1-9, 1/K2/2-43
Rounding, 1/B1/2-18
Rounding with contour tolerance, 1/B1/2-22
RPY, 3/F2/2-23, 3/TE4/2-43
RS 232, 1/K4/1-3
RS-232-C external, 1/A2/5-74, 1/A2/5-79
RS-232-C OFF, 1/A2/5-74, 1/A2/5-80
RS-232-C stop, 1/A2/5-74, 1/A2/5-79
RS232, 1/K4/5-31
RS232C ON, 1/A2/5-75, 1/A2/5-80
Run-in and run-out paths, programmable, 3/TE1/2-17, 3/TE1/2-18
Runin and runout paths, programmable, 1/V1/2-15, 1/V1/2-17
Runtime, 2/B3/1-14
Measurement; Example, 1/K1/2-128
of the NC program, 1/K1/2-128
of the path axes, 1/K1/2-128
Runtime optimization, 3/V2/2-5

S

Safe actual position, 1/D1/2-12
Safe input signals, 1/D1/2-12
Safe output signals, 1/D1/2-12
Sag compensation, Compensation values in grid structure, 2/K3/2-32
SAVE command, 1/K1/2-89
SCALE, 2/R2/2-18
Scratching, 2/M5/2-15
Screen bright, 1/A2/5-71
Screen is dark, 1/A2/5-77
Second operator panel, 2/B3/1-20
Secondary
Control panel, 2/B3/1-6, 2/B3/1-14
Unit, 2/B3/1-20
Selecting synchronous mode, 2/S3/2-10
Selection, 1/A2/5-75, 1/A2/5-81
Selection of the cutting edge when changing tool, 1/W1/2-6
Selection of type of interpolation, 3/F2/2-62
Selection/deselection, 3/W5/2-28
Semiautomatic procedure for matching the machine data, 1/D2/2-20
SER_CLASS_WRITE_TOA_SUPVIS, 1/A2/4-52
Serial interface
MMC 100: File types, 1/K4/2-6
MMC 101/102: File types, 1/K4/2-7
Serial interfaces, 1/K4/5-31
Serial interfaces (RS232), 1/K4/2-5
SERUPRO, 1/K1/2-36
SERUPRO sequence without defining a search target beforehand, 1/K1/2-46
SERUPRO_SPEED_FACTOR, 1/K1/4-151
Service case, 2/B3/1-19
Service display for FS, 2/S3/2-34
Service/installation and start-up, 2/B3/1-13
Servo enable, 1/A2/2-25, 1/A2/5-90
Servo enabling on SM-axis, 2/S6/3-17
Servo gain factor, 1/D1/2-9, 1/G2/2-45, 2/K3/2-47
Servo gain for SM-control without measuring system, 2/S6/3-17
Servo gain for SM-control with measuring system, 2/S6/3-17
SERVO_FIFO_SIZE, MD 10087, 2/B3/4-125
Set REPOS response, 1/K1/2-61
SETINT, 1/K1/2-87
SETINT_ASSIGN_FASTIN, MD 21210, 1/K1/4-148
Setpoint exchange: detailed description, 3/TE5/2-5
Setpoint output, 1/G2/2-23
Setpoint parameter set, 1/D1/2-20
Setpoint switchover, 3/TE5/1-3
Setpoint system, 1/G2/2-23
Setpoint/actualvalue system, Configuration of drives for SINUMERIK 840Di, 1/G2/2-30
Setting data for processing programs from external, 1/K1/4-167
Setting the computer mode (turning, milling), 1/A2/4-51
Setting the constant cutting rate, 1/S1/2-59
Setting the speed, 1/S1/2-59
Setup mode, 1/D1/2-20
Setup mode active, 1/A2/2-33, 1/A2/5-100
Several NCUs, 2/B3/1-21, 2/B4/1-4
SW 3.5 and higher, 2/B3/1-23

© Siemens AG, 2003. All rights reserved

Several operator panels
Alarm text storage, 2/B3/2-62
Alarms/messages, 2/B3/2-61, 2/B3/2-66
Applications, 2/B4/1-3
Availability, 2/B3/3-120, 2/B4/4-17
Buses, 2/B3/2-58
Compatibility, 2/B3/2-41, 2/B3/2-60
Configuration files, Number, 2/B3/2-57
Configurations, 2/B3/2-56
Connections, 2/B3/2-58
Defaults, 2/B3/2-41, 2/B3/2-60
Features in operation, 2/B4/2-15
Implementation in SW 3.1, 2/B3/1-20
Implementation SW 3.1, 2/B4/1-3
Link check, 2/B3/2-62
NCU components, 2/B3/2-59
No. of communications partners, 2/B3/4-123
Operational characteristics, 2/B3/1-20
Operator interface, 2/B3/2-66
Operator panel components, 2/B3/2-59
Switchover of connection, 2/B3/2-61, 2/B3/2-62
User interfaces, 2/B4/1-4
Shift behavior of keyboard during booting, 1/A2/4-50
ShopMill control signal, 1/A2/5-74
Signal, Transformation active, 2/M1/5-77, 2/M5/5-71
Signal Descriptions, 1/A2/5-67
Signal distortion, 1/A3/2-5
Signaling states, 1/A2/1-3
Signals, 2/B3/1-7
Alarm signals, 1/A2/2-8
Axis/spindlespecific (DB31, ...), 1/A2/2-6
Channelspecific (DB21, ...), 1/A2/2-6
For digital and analog inputs-/outputs, 2/L1/5-19
For the 5th axis, 2/L1/5-19
General (DB10), 1/A2/2-6
OPspecific (DB19), 1/A2/2-6
PLC/axes, spindles, 1/P3/2-40
PLC/mode group, 1/P3/2-37, 1/P3/2-38
PLC/NC, 1/P3/2-37
PLC/NCK channels, 1/P3/2-39
Ready signals, 1/A2/2-7
Signals from axis/spindle, 1/A2/2-33, 1/A2/5-97
Signals from channel, 1/A2/5-84
Signals from NC to PLC (DB10), 1/A2/5-68
Signals from operator panel, 1/A2/5-77
Signals from PLC to NC (DB10), 1/A2/5-67
Signals to axis/spindle, 1/A2/2-32, 1/A2/5-85
Signals to channel, 1/A2/5-83
Signals to operator panel, 1/A2/5-71
SIMATIC, 2/B3/2-39, 2/B3/2-60
Simulation axes, 1/G2/2-24
Simulation selected, 1/A2/5-78
SINCOM, 2/B3/1-6, 2/B3/2-34
Single block, 2/P2/2-8, 2/P2/2-32
Single-axis rotary table, 3/F2/2-17
Single-axis swivel head, 3/F2/2-17
Singular positions, 3/F2/2-27, 3/TE4/2-48
Singularities, 3/F2/2-51
SINUMERIK
810D, 2/B3/1-22
840D Installation Guide, 2/B3/1-13
SINUMERIK 802D, 1/K1/2-129
SINUMERIK 840DiSRAM, 1/R1/3-45
SLASH_MASK, 1/A2/4-58
Slide, 2/B3/1-25
Slimline screen, 2/B3/1-11
Slotting saw, 1/W1/2-28
Smooth approach and retraction
Approach movement, 1/W1/2-41
Meaning, 1/W1/2-41
Retraction movement, 1/W1/2-41
Smoothed actual current value, 1/D1/2-19
SOFT, 1/B2/2-8
Soft key, 2/B3/1-17
Conn..., 2/B3/1-17
Softkey, 2/B3/1-11, 2/B3/1-21, 2/B3/1-22
Software cams, 2/N3/2-5
Software limit switches, 1/A3/2-25, 2/H1/2-35, 2/R2/2-21, 2/T1/2-21
Sparking-out strokes, 2/P5/1-3
Special 2-axis NR kinematic, 3/TE4/2-42
Special 3-axis SC kinematic, 3/TE4/2-40
Special 4-axis SC kinematic, 3/TE4/2-41
Special axes, 1/K2/2-19
Special kinematics, 3/TE4/2-38
Special spindle interface, 1/S1/2-55
Specify gear stage, 1/S1/2-28
Speed, 2/H1/2-34
Actual point, 1/D1/2-10
ClosedLoop Control, 1/A3/2-21
Servo enable NC, 1/D1/2-17
Setpoint, 1/D1/2-10, 1/D1/2-19
Setpoint smoothing, 1/D1/2-19
Speed control, 2/S6/1-3, 2/S6/2-8
Activation, 2/S6/2-14
Constant speed time, 2/S6/2-14
Knee-shaped acceleration characteristic, 2/S6/2-8
Response to acceleration rate changes, 2/S6/2-14
Speed control loop, 1/G2/2-43
Speed controller active, 1/A2/2-31, 1/A2/5-99
Speed controller integrator disable, 1/A2/5-95
Speed controller integrator disabled, 1/A2/5-103
Speed feedforward control, 2/K3/2-39
Speed monitoring, 2/W4/2-28
Speed ratio, 2/S3/2-7
Speed setpoint adjustment, 1/G2/2-34
Speed setpoint output, 1/G2/2-34, 1/G2/2-47, 1/G2/2-50, 1/G2/2-56, 1/G2/2-60, 1/G2/2-61
Speed setpoint routing, 1/G2/2-25
Speed setpoint smoothing, 1/A2/2-32, 1/A2/5-93
Speed setpoint smoothing active, 1/A2/2-33, 1/A2/5-101

© Siemens AG, 2003. All rights reserved

Speed/acceleration limits, 2/S3/2-18
SPI(n, 1/S1/2-53
SPIND_POSIT_DELAY_TIME, 1/S1/4-87
Spindle
Replacement, 2/K5/2-13
Start-up, 2/S3/2-32
Spindle commands, 1/S1/2-57
Spindle disable, 1/A2/2-19
Spindle job, 1/S1/2-56
Spindle number, 2/W4/2-10
Spindle override factor, 1/V1/2-30
Spindle position with mono probe, 2/M5/2-6
Spindle reset, 1/A2/2-28
Spindle speed, 1/G2/2-6
Spindle speed limitation with G96, 1/V1/2-12
Spindle speed setpoint current, 1/D1/2-10
Spindle speed setpoint programmed, 1/D1/2-10
Spindle stop, 1/K1/2-75
Spindle traversal in JOG mode, 2/H1/2-34
Spindlespecific functions, 1/S1/2-61
SPOS, 1/B1/2-14
SRAM, 2/S7/2-6
Memory requirements, 2/S7/2-27
Standard alarm texts, 2/B3/1-18
Start address of
Direct read input bytes of PLC I/Os, 2/A4/4-52
Direct write output bytes of PLC I/Os, 2/A4/4-53
START command, 1/K1/2-70
Start operating area, 2/B3/1-17, 2/B3/1-21
Start-up, and installation MMC 100.2, 2/F3/2-76
Start-up, 2/B3/1-18, 2/B3/1-19
Of neural QEC, 2/K3/2-69
Tool, 2/B3/1-22
Startup and synchronization of NCK PLC, 1/P3/2-50
Static MMC property, 2/B3/1-13
Station/position change, 2/B3/1-28
Status query, HW outputs, 2/N3/2-18
Stepper motor control
Axis-specific machine data, 2/S6/4-19
Axis-specific signals, 2/S6/5-21
Interface signals, 2/S6/7-24
Machine data, 2/S6/7-23
Stepper motor drive without measuring system, 2/S6/3-18
Stop, 3/K6/2-7, 3/M3/1-7
Stop and retract, 3/M3/2-60
Stop bits, 1/K4/3-21
Stop block, 1/K1/2-41
STOP commands, 1/K1/2-71
Stop routine, 1/K1/2-41
STOP_LIMIT_FACTOR, 1/B1/4-58
Structure of MD files for data backup, 1/A2/4-60
Suitable probes, 2/M5/2-5
SUMCORR_DEFAULT, 1/W1/4-140
SUPPRESS_ALARM_MASK, 1/D1/4-35
SUPPRESS_SCREEN_REFRESH, 10131, 1/A2/4-58
Suppression
Algorithm, 2/B3/1-15, 2/B3/2-43
Mechanism, 2/B3/1-11
Priority, 2/B3/1-14
Rules, 2/B3/2-45
Strategy, 2/B3/1-15, 2/B3/2-44, 2/B3/2-49
SW 5.2, 1/K1/1-7, 1/K1/2-127, 1/K1/7-202
SW_CAM_MODE, 2/N3/4-33
Switch over MCS/WCS, 1/A2/5-78
Switch over measuring system, 1/A2/2-24
Switchover
Behavior OP030, MMC 100, MMC 102/103, 2/B3/1-17
Conditions, 2/B3/2-45
Swiveled linear axis
Application, 3/F2/2-31
Channel-specific MD, 3/F2/4-99
Kinematics variants, 3/F2/2-31
Parameter assignments, 3/F2/2-31
Pole, 3/F2/2-31
Swiveled linear axis
Type of machine, 3/F2/2-36
Zero position, 3/F2/2-34
Swiveling linear axis, 3/F2/1-9
Symbol name, 2/B3/1-14, 2/B3/1-19
Symbolic programming of user program with interface DB, 1/P3/2-56
Synchronism deviation, Scanning, 3/M3/2-43
Synchronization, 1/B1/2-9
Synchronization cycle, 2/B3/1-8
Synchronized
Actions, 2/B3/1-29
State reached, 2/S3/2-15
Synchronized axis, 3/G1/2-5
Synchronous
Procedure, 2/S5/2-15
Spindle pair, 2/S3/2-5
Synchronous axes, 1/B1/2-14, 1/B1/2-19, 1/K2/2-22
Synchronous mode, 2/S3/2-5
Activate, 2/S3/2-22
Control dynamics, 2/S3/2-27
Deactivate, 2/S3/2-22
Feedforward control, 2/S3/2-27
Knee-shaped acceleration characteristic, 2/S3/2-28
Multiple couplings, 2/S3/2-27
Response to alarms, 2/S3/2-28
Speed/acceleration limits, 2/S3/2-27
Syntax check, 3/V2/2-11
SYSCLOCK_SAMPL_TIME_RATIO, 3/G3/2-17
System clock cycle, 3/G3/2-5
840Di, 3/G3/2-10
System clock rate for FM NC with five axes-NC, 2/L1/3-15
System of units, SW cams, 2/N3/2-14
System requirements, 2/M5/2-13
System variable, 2/M5/2-14, 2/M5/2-61

© Siemens AG, 2003. All rights reserved

System variables, 1/A2/4-65, 1/A2/7-121, 2/B3/1-29
Channelspecific, 1/K1/2-127
Global, 2/B3/1-29
NCKspecific, 1/K1/2-127
with read/write access, 1/K1/2-129

T

T function, 1/W1/2-5, 2/W3/1-3
T_FL_WP, 3/TE4/2-16
T_IRO_RO, 3/TE4/2-15
T_NO_FCT_CYCLE_NAME, 1/K1/4-138
T_X3_P3, 3/TE4/2-16
Tabulator length, 1/A2/4-50
TABULATOR_SIZE, 1/A2/4-50
Tachogenerator compensation, 1/A3/2-19
Tangential Control, Applications, 3/T3/1-4
Tangential control, 3/T3/1-3
Activation of follow-up control, 3/T3/2-8
Following axis, 3/T3/2-8
Leading axis, 3/T3/2-8
Limit angle, 3/T3/2-13
Termination of follow-up control, 3/T3/2-10
TANGON, 3/T3/2-9
TDP, 3/G3/2-10
TDX, 3/G3/2-10
TEACH IN, 1/K1/2-16
Technical exponent representation in three steps, 1/A2/4-52
TECHNOLOGY, 1/A2/4-51
Technology memory, 1/D2/2-10
Temperature compensation
Coefficient tanß(T), 2/K3/2-11
Compensation equation, 2/K3/2-6
Deformation, 2/K3/2-5
Error curves, 2/K3/2-5
Monitoring functions, 2/K3/2-9
Parameter, 2/K3/2-7
Parameters, 2/K3/2-8
Position display, 2/K3/2-9
Sensor equipment, 2/K3/2-5
Temporary assignment, 2/B3/1-7
Terminal emulation, 2/F3/2-107, 2/F3/2-111
Test mode, 1/A3/2-17
Test program for testing repeat accuracy, 2/M5/6-76
Threshold for change in handwheel direction, 2/H1/4-44
Threshold speed, 1/D1/2-23
Threshold torque, 1/D1/2-21
Threshold values, 2/S3/2-18
Threshold values for coarse/fine synchronism, 2/S3/2-34
TI, 3/G3/2-10
Timer activation, 1/K1/2-129
Timers, Channelspecific, 1/K1/2-128
TM, 3/G3/2-10
TMAPC, 3/G3/2-10
TO, 3/G3/2-10
TOA, 1/W1/2-7
TOCARR_BASE_FINE_LIM_LIN, 1/W1/4-134
TOCARR_BASE_FINE_LIM_ROT, 1/W1/4-135
TOCARR_FINE_CORRECTION, 1/W1/4-152
Tolerance of BERO -edges in pulses for monitoring stepper motor rotation, 2/S6/4-19
Tool, 1/W1/2-5
Base dimension/adapter dimension, 1/W1/2-34
Change, 1/W1/2-5
Change tool with “M06”, 1/W1/2-5
Cutting edge, 1/W1/2-26
Cutting edge position, 1/W1/2-29
DISC, 1/W1/2-55
Length compensation, 1/W1/2-30
Management, 1/K1/1-7, 1/K1/2-127, 1/K1/2-129
Operating time, 1/K1/2-128
Radius compensation, 1/W1/2-32
Select, 1/W1/2-5
T function, 1/W1/2-5
Tool clearance angle, 1/W1/2-35
Tool cutting edge, 1/W1/2-5
Tool radius compensation 2D (TRC), 1/W1/2-38
Tool type, 1/W1/2-27
Wear, 1/W1/2-33
Tool change
Change point, 2/W3/1-3, 2/W3/2-7
D function, 1/W1/2-6
Fixed points, 2/W3/2-7
Offset memory, 1/W1/2-7
Tool diameter ($AC MEAS TYPE = 11), 2/M5/2-57
Tool length ($AC MEAS TYPE = 10), 2/M5/2-55
Tool length compensation, 1/W1/2-30
Tool master data, 1/D2/2-11
Tool offset, 2/P2/2-31
Calculation, 1/W1/2-8
Tool offset
for grinding operations, 2/W4/2-5
for grinding tools, 2/W4/2-5
Tool offset selection, 1/A2/5-73
Tool operating data, 1/D2/2-11
Tool orientation, 3/F2/1-11, 3/TE4/2-43, 3/W5/2-7
4-axis kinematics, 3/TE4/2-46
5-axis kinematic, 3/TE4/2-47
Tool orientation using orientation vectors, 3/F2/2-26
Tool programming, 3/TE4/2-51, 3/TE4/2-52
Tool Radius Compensation, 3/W5/1-3

© Siemens AG, 2003. All rights reserved

Tool radius compensation, change alarm response, 1/W1/2-66
Tool radius compensation (TRC), 1/W1/2-38
Tool radius compensation: 2D, 1/W1/2-38
Approach and retraction behavior, 1/W1/2-40
Collision detection, 1/W1/2-60
Compensation at inside corners, 1/W1/2-58
Compensation at outside corners, 1/W1/2-54
Deselection, 1/W1/2-54
G451 intersection, 1/W1/2-56
Selection, 1/W1/2-39
Smooth approach and retraction, 1/W1/2-41
Transition circle, 1/W1/2-55
Variable compensation value, 1/W1/2-61, 1/W1/2-85
Tool radius offset, 3/M3/2-15
Tool revolver axes, 1/K2/2-19
Tool tip at a fixed point in space, 3/F2/2-24
Tool type, Slotting saw, 1/W1/2-28
Tool types for grinding tools, 2/W4/2-9
TOOL_LENGTH_CONST, 1/K2/4-100, 1/W1/4-150
TOOL_REF_GEO_AXIS1, 1/K2/4-85
TOOL_REF_GEO_AXIS2, 1/K2/4-85
TOOL_REF_GEO_AXIS3, 1/K2/4-85
TOOL_TEMP_COMP, 1/W1/4-152
TOOL_TEMP_COMP_LIMIT, 1/W1/4-141
TOOL_TEMP_COMP_ON, 1/W1/4-141
Toolholder, Orientational, 1/W1/2-72
Toolholder selection, 1/W1/2-72
Toolholder, orientational
Control system response on Reset, program start, REPOS, 1/W1/2-94
Programming, 1/W1/2-86, 1/W1/2-89, 1/W1/2-93, 1/W1/6-154, 1/W1/6-157
Supplementary conditions, 1/W1/2-85, 1/W1/2-89, 1/W1/2-94
Toolholders, orientable, Programming, 3/F2/2-60
Tools with relevant tool point direction, 1/W1/2-36
Torque feedforward control, 2/K3/2-39
Torque limit, 1/D1/2-19
Torque limit 2, 1/A2/2-32, 1/A2/5-93
Torque limit 2 active, 1/A2/2-33, 1/A2/5-101
Torque limitation value, 1/D1/2-12
TRAANG
Activation, 2/M1/2-48
Availability, 2/M1/3-61
Brief description, 2/M1/1-7
Deactivation, 2/M1/2-48
Number of, 2/M1/2-42
Preconditions, 2/M1/2-41
Restrictions, 2/M1/2-49
Specific settings, 2/M1/2-44
TRAANG, Inclined axis, 2/M1/2-40
TRACYL, 2/M1/1-6, 2/M1/2-28
Activation, 2/M1/2-36
Availability, 2/M1/3-61
Axis image, 2/M1/2-33
Deactivation, 2/M1/2-37
Number of, 2/M1/2-30
Preconditions, 2/M1/2-30
Restrictions, 2/M1/2-37
Specific settings, 2/M1/2-33
TRANS, 1/K2/1-9, 1/K2/2-43, 2/R2/2-18
Transfer time, 1/K1/2-63
Transformation
Activation, 3/TE4/2-49
Actual-value display, 3/TE4/2-50
Chaining sequence, 2/M1/2-54
Deactivation, 3/TE4/2-49
End of program, 3/TE4/2-49
Transformation, 1/K2/2-35
Transformation active, 3/F2/2-41
Transformation group, 3/F2/2-41
Transformation types, 3/F2/2-22
Transformed axes, Number of, 3/TE4/2-17
Translation, 3/TE4/2-6
TRANSMIT, 1/K2/2-35, 2/M1/1-5, 2/M1/2-9
Activation, 2/M1/2-16
Availability, 2/M1/3-61
Axis image, 2/M1/2-13
Deactivation, 2/M1/2-16
Restrictions, 2/M1/2-17
Specific settings, 2/M1/2-13
Structures, Number of, 2/M1/2-10
Transverse axes, 1/P1/1-3, 2/H1/2-37
Diameter programming, 1/P1/2-6
Geometry axes, 1/P1/2-5
Transverse grooves, 2/M1/1-6
Travel to fixed stop, 1/F1/1-3, 1/F1/2-5
Analog drives, 1/F1/2-28
Analog drives, diagrams, 1/F1/2-34
Analog drives, FXS deselection, 1/F1/2-36
Analog drives, FXS selection, 1/F1/2-34
Analog drives, SIMODRIVE 611A (FDD), 1/F1/2-28
Analog drives, SIMODRIVE 611A (MSD), 1/F1/2-30
Axis/spindlespecific signals, 1/F1/5-48
Axisspecific machine data, 1/F1/4-39
Axisspecific setting data, 1/F1/4-44
Deselection, 1/F1/2-11
Functional sequence, 1/F1/2-8
Interface signals, 1/F1/7-53
Monitoring window, 1/F1/2-9
Programming, 1/F1/2-6, 1/F1/2-8
RESET, 1/F1/2-13
Selection, 1/F1/2-8

© Siemens AG, 2003. All rights reserved

Travel to fixed stop (F1)
Alarms, 1/F1/2-10
Contour monitoring, 1/F1/2-19
Detailed Description, 1/F1/2-5
Function abort, 1/F1/2-13
Machine data, 1/F1/7-54
Positioning axes, 1/F1/2-19
Power ON, block search, 1/F1/2-14
Setting data, 1/F1/2-10, 1/F1/2-18, 1/F1/7-54
Signal Descriptions, 1/F1/5-47
Status interrogation, 1/F1/2-19
Traverse against fixed stop, 2/S6/3-17
Traversing ranges, 1/G2/2-6
Tunnel size, 3/K6/2-7
Two-axis rotary table, 3/F2/2-17
Two-axis swivel head, 3/F2/2-16
Types of feed, 1/V1/1-3
Types of protection zone, 1/A3/2-32

U

U(ZK), 1/A2/2-35
UDClink lower than warning threshold, 1/A2/5-112
Universal milling head, 3/F2/1-11, 3/F2/2-37
Applications, 3/F2/2-37
Features, 3/F2/1-11
JOG, 3/F2/2-40
Parameterization, 3/F2/2-39
Universal Serial Bus, 1/K4/1-3
Unload, 1/A2/5-75, 1/A2/5-81
Update time for PLC I/O input cycle, 2/A4/4-53
Updating the REPOS offset within the valid scope, 1/K1/2-66
USB, interface, 1/K4/2-9
Useful Tips on Programming with STEP7, 1/P3/7-257
Copying data, 1/P3/7-258
Useful tips on programming with Step7
ANY and POINTER, 1/P3/7-258
Multiinstance DB, 1/P3/7-263
POINTER or ANY variable, 1/P3/7-261
Strings, 1/P3/7-264
Use of ANY and POINTER in FC, 1/P3/7-259, 1/P3/7-260
User
Alarm, 2/B3/1-18
Communication, 2/B3/1-7
User-defined coupling, 2/S3/2-6
USER_CLASS_BASE_ZERO_OFF_MA, 1/K2/4-84
USER_CLASS_BASE_ZERO_OFF_PA, 1/K2/4-84
USER_CLASS_DIRECTORY1_M, 1/A2/4-57
USER_CLASS_DIRECTORY1_P, 1/A2/4-56
USER_CLASS_DIRECTORY2_M, 1/A2/4-57
USER_CLASS_DIRECTORY2_P, 1/A2/4-56
USER_CLASS_DIRECTORY3_M, 1/A2/4-57
USER_CLASS_DIRECTORY3_P, 1/A2/4-56
USER_CLASS_DIRECTORY4_M, 1/A2/4-57
USER_CLASS_DIRECTORY4_P, 1/A2/4-56
USER_CLASS_READ_TCARR, 1/A2/4-52
USER_CLASS_WRITE_MAG_WGROUP, 1/A2/4-52
USER_CLASS_WRITE_TCARR, 1/A2/4-52
USER_CLASS_WRITE_TOA_ASSDNO, 1/A2/4-52
USER_CLASS_WRITE_TOA_EC, 1/A2/4-52
USER_CLASS_WRITE_TOA_SC, 1/A2/4-52
USER_DATA_FLOAT[n], 1/P3/9-284
USER_DATA_HEX[n], 1/P3/9-284
USER_DATA_INT[n], 1/P3/9-284
Userdefined system ASUBs, 1/K1/2-94
Using DMP modules, 1/K2/3-81
Utilization property, 2/B3/1-14

V

Value storage scratching and PRESET, 1/K2/4-84
Variable signal 1, 1/D1/2-23
Variable signaling function, 1/A2/2-35, 1/A2/5-111
VDI-Signals when stepper motor is used, 2/S6/3-18
Velocities, 1/B1/2-7, 1/G2/2-5
Velocity, 2/H1/2-6, 2/H1/2-16, 3/TE4/2-19
Velocity adjustment, 1/G2/2-34
Velocity and acceleration, 2/H1/2-36
Velocity coupling, 2/S3/2-34
Velocity for zero cycle blocks, 1/B1/2-7
Velocity increase, 3/TE4/2-48
Velocity reduction according to overload factor, 1/B1/2-16
Velocity warning threshold, 3/M3/5-97
Vertical axis, 2/K3/2-89
Viewer MS-DOS (MMC 100.2), 2/F3/2-92
Viewer PC functions, 2/F3/2-108
Vocabulary, 3/V2/2-12

W

WAITMC, 2/P2/2-31
WAITP, 2/P2/2-7, 2/P2/2-31
Oscillation axis, 2/P5/2-17
WAITP coordination, 2/P2/2-31
WCS, 1/A2/2-16
WEAR_SIGN, 1/W1/4-150
WEAR_SIGN_CUTPOS, 1/W1/4-149
Working area limitation, 1/A3/2-28, 2/H1/2-35, 2/R2/2-21

© Siemens AG, 2003. All rights reserved

Workpiece
Actual, 1/K1/2-129
Clamping, 2/B3/1-25, 2/B3/1-29
Counter, 1/K1/1-7, 1/K1/2-129, 1/K1/7-202
Example, 1/K1/2-130
Counting, 1/K1/2-127
Measuring, 2/M5/2-15
Setpoint, 1/K1/2-129
Total actual, 1/K1/2-129
Workpiece coordinate system, 1/A2/2-16
Workpiece coordinate system (WCS), 1/K2/1-7, 1/K2/2-42, 1/P1/1-3, 1/P1/2-6
Writ, Online tool offset discretely, 2/W4/2-24
Write
Continuously, 2/W4/2-20
Real-time variable, 2/S5/2-13
WRITE_TOA_LIMIT_MASK, 1/K2/4-86

Z

Zeros and reference points, 1/K2/2-29
ZK1 messages, 1/D1/2-18, 1/D1/2-24

© Siemens AG, 2003. All rights reserved

Notes

© Siemens AG, 2003. All rights reserved

To
SIEMENS AG
A&D MC BMS
P.O. Box 3180

D-91050 Erlangen, Germany

(Phone: ++49-(0)180-5050-222 [Hotline]
Fax: ++49-(0)9131-98-2176 [Documentation]
Email: motioncontrol.docu@erlf.siemens.de)

Suggestions

Corrections

For Publication/Manual:

SINUMERIK 840D/840Di/810D
Description of Functions
Basic Machine

Manufacturer/Service Documentation

Description of Functions

Order No.: 6FC5 297-6AC20-0BP3
Edition: 11.03

From

Name

Company/Dept.

Address

Phone: /

Fax: /

Should you come across any printing errors when reading this publication, please notify us on this sheet.
Suggestions for improvement are also welcome.

Suggestions and/or corrections

Overview of SINUMERIK 840D/840Di/810D Documentation

General Documentation

- *SINUMERIK* 840D/810D — Brochure
- *SINUMERIK* 840D/840Di/ 810D/ — Catalog Ordering Info. NC 60 ***)**
- *SINUMERIK SIROTEC SIMODRIVE* Accessories — Catalog Accessories NC-Z

User Documentation

- *SINUMERIK* 840D/810D/ FM-NC — **AutoTurn**
  - Short Guide
  - Programming/ Setup
- *SINUMERIK* 840D/840Di/ 810D — Operator's Guide **– HT 6**
- *SINUMERIK* 840D/840Di/ 810D — **Diagnostics Guide *)**
- *SINUMERIK* 840D/840Di/ 810D — **Operator's Guide *)**
  - Short Guide
  - HMI Embedded
  - HMI Advanced

User Documentation

- *SINUMERIK* 840D/840Di/ 810D — **Program. Guide**
  - Short Guide
  - Fundamentals ***)**
  - Advanced ***)**
  - Cycles
  - Measuring Cycles
  - ISO Turning/Milling
- *SINUMERIK* 840D/810D — Operator's Guide
  - **ManualTurn**
  - Short Guide ManualTurn
  - **ShopMill**
  - Short Guide ShopMill
  - **ShopTurn**
  - Short Guide ShopTurn
- *SINUMERIK* 840Di — **System Overview**

Manufacturer/Service Documentation

- *SINUMERIK* — **Configuring** (HW) ***)**
  - 810D
  - 840D
- *SINUMERIK* 840D/840Di/ 810D — **Operator Components** (HW) ***)**
- *SINUMERIK* 840D/810D — Description of Functions **– ManualTurn** **– ShopMill** **– ShopTurn**
- *SINUMERIK* 840D/840Di/ 810D — Description of Functions **Synchronized Actions**

Manufacturer/Service Documentation

- *SINUMERIK SIMODRIVE* 611D 840D/810D — Description of Functions **Drive Functions *)**
- *SINUMERIK* 840D/840Di/ 810D — **Description of Functions**
  - Basic Machine ***)**
  - Extended Functions
  - Special Functions
- *SINUMERIK* 840D/840Di/ 810D — Description of Functions **Tool Management**
- *SINUMERIK* 840D/810D — **Configuring Kit** HMI Embedded
- *SINUMERIK* 840D/810D — Description of Functions Operator Interface **OP 030**
- *SINUMERIK* 840D/840Di/ 810D — **Motion Control Information System**
  - Computer Link SinCom
  - Tool Information System SinTDI
  - DNC NC Program Management
  - Tool Data Inform./TDI Ident Connec.
  - Tool Data Information Communication

Manufacturer/Service Documentation

- *SINUMERIK SIMODRIVE* — Description of Functions SINUMERIK **Safety Integrated**
- *SINUMERIK* 840D — Description of Functions **Digitizing**
- *SINUMERIK SIMODRIVE* — **Installation & Start-Up Guide *)**
  - 810D
  - 840D/611D
  - HMI
- *SINUMERIK SIMODRIVE* 840D/840Di 810D 611D — **Lists *)**
- *SINUMERIK SIMODRIVE* 840D 611D — Description of Functions **Linear Motor**
- *SINUMERIK SIMODRIVE* 840D 611D — Description of Functions **– Hydraulics Module** **– Analog Module**
- *SINUMERIK SIMODRIVE SIROTEC* — **EMC** Guidelines

Electronic Documentation

- *SINUMERIK SIMODRIVE* 840D/840Di/ 810D 611, Motors — **DOC ON CD *)** The SINUMERIK System

Manufacturer/Service Documentation

- *SINUMERIK* 840D/840Di/ 810D — Description of Functions **ISO Dialects** for SINUMERIK
- *SINUMERIK* 840Di — **Manual** (HW + Installation and Start-Up)
- *SINUMERIK* 840D/810D — Description of Functions **Remote Diagnosis**
- *SINUMERIK* 840D/840Di/ 810D — **Manual @ Event**

***)** These documents are a minimum requirement

Siemens AG
Automation & Drives
Motion Control Systems
P.O. Box 3180, D-91050 Erlangen
Germany

www.ad.siemens.de

© Siemens AG, 2003
Subject to change without prior notice
Order No.: 6FC5297-6AC20-0BP3

Printed in Germany