HEIDENHAIN

Technical Manual
TNC 407
TNC 415
TNC 425

Valid for the NC software types
259 96 (TNC 415 A)
259 97 (TNC 415 E)
243 02 (TNC 407)
up to version 09

and
259 93 (TNC 415 B/TNC 425)
259 94 (TNC 415 F/TNC 425 E)
243 03 (TNC 407)
up to version 12

and
280 54 (TNC 415 B/TNC 425)
280 56 (TNC 415 F/TNC 425 E)
280 58 (TNC 407)
up to version 06

January 98

208 732 21 · 6.5 · 1/98 · S · Printed in Germany · Subject to change without notice
(208 732 E2)

Preface

This Technical Manual is intended for manufacturers and distributors of machine tools.
It contains all the necessary information for the assembly, electrical installation, start-up and PLC-programming for the HEIDENHAIN contouring controls.

When hardware or software is improved in these HEIDENHAIN contouring controls you will receive a free delivery of updated information. Please arrange and insert this updated information in your manual without delay. This will ensure that your manual always remains at the current revision level.

You can use extracts from this manual for your machine documentation. An enlargement of the manual format (17 cm x 24 cm) by a factor of 1.225 will produce pages in DIN A4 format.

No documentation can be perfect. Like all living things it must grow and change. Among other things, it lives from your impulses and suggestions for improvement. Please help us by letting us know your ideas.

DR. JOHANNES HEIDENHAIN GmbH
Department PE
PO Box 1260
D-83292 Traunreut
Germany

Contents Technical Manual TNC 407, TNC 415 B, TNC 425

Update Information No. 20

New NC software

In May 1996 the following new NC software versions were released:

TNC 407	280 58x **06** 280 590 **06**
TNC 415B / TNC 425	280 54x **06** 280 550 **06**
TNC 415F / TNC 425E	280 56x **06** 280 570 **06**

Hardware

New Design for TNC 407
A new hardware design was introduced with the new TNC 410, TNC 426 and TNC 430 controls. This changed design can also be used for the TNC 407. The visual display unit and keyboard unit with the new design are gray in color and require a larger front-panel cutout.

The new **BC 120** visual display unit is connected to the TNC 407 with a new cable (Id. Nr. 312 878 ..) and adapter (Id. Nr. 313 434 02). The matching **TE 400 B** keyboard unit has the Id Nr. 313 038 02. A matching gray **MB 420** machine operating panel (Id. Nr. 293 757 12) is also available.

Documentation

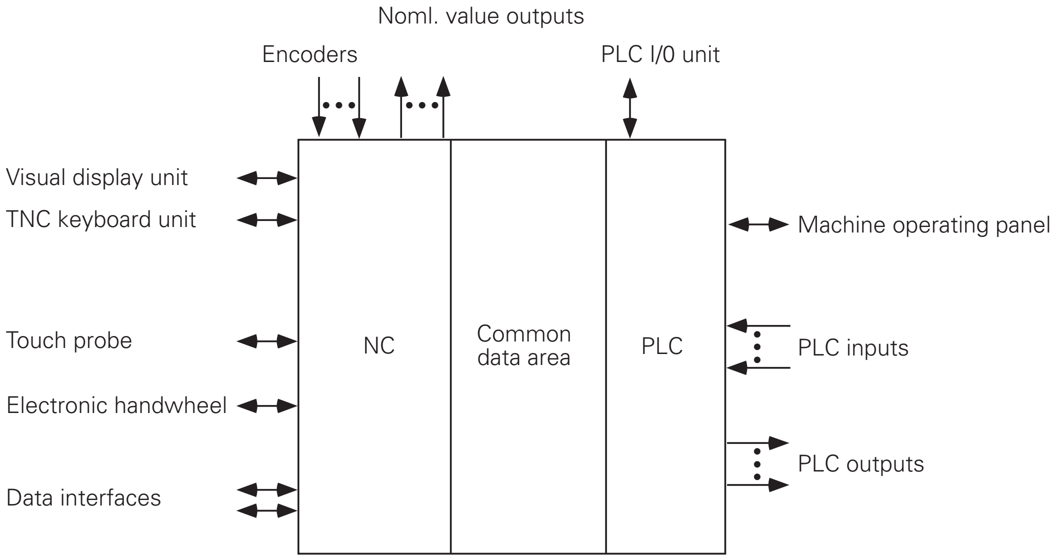
No replacement pages will be issued for this Technical Manual. You will find the updated documentation on the CD-ROM entitled "TNCguide OEM" (Id. Nr. 208 935 92), available from January 1998. Please contact HEIDENHAIN if you have not yet received this CD-ROM.

Introduction – Contents

1 Hardware concept

The HEIDENHAIN-contouring controls TNC 407 and TNC 415 are designed for use with drilling and milling machines.

The HEIDENHAIN-TNCs consist of several units. The principal subassembly is the logic unit. The logic unit is joined to the other units and the TNC accessories by connecting cables.



The logic unit contains the electronics for both the NC and the PLC sections of the control.
The common data area contains the machine parameters and the PLC markers and words. The machine parameters define the hardware configuration of the machine (ranges of travel, acceleration, number of axes etc.). The PLC markers and words are used for the exchange of information between the NC and the PLC.

2 Technical data TNC 407/TNC 415 B/TNC 425

	TNC 407	TNC 415B/TNC 425
Axes	3, 4 or 5 plus spindle S[1)] (NC axes and PLC axes can be defined as desired)	
Program input	In HEIDENHAIN Plain Language and to DIN/ISO	
Memory for part program	6000 blocks approx.	12 000 blocks approx.
Positions	Nominal positions in Cartesian or polar coordinates, dimensions absolute and incremental	
Input and display resolution	to 1 µm	to 0.1 µm
Interpolation		
Linear interpolation	3 of 5 axes	5 of 5 axes
Circular interpolation	2 of 5 axes	3 of 5 axes [1)]
Helix	Yes	
Rigid Tapping	Yes[1)]	
Block processing time[2)]	25 ms	4 ms
Look Ahead	• Defined rounding of discontinuous contour transitions (e.g. 3D surfaces) • Collision viewing with the SL cycle for "open" contours • Advance calculation of geometry for feed-rate adjustment	
Free contour programming FK	In HEIDENHAIN Plain Language with graphic support	
Coordinate transformations	Shift and/or rotate coordinate system, mirroring, reduce and enlarge - also axis-specific	
Tilting the working plane	–	Yes[1)]
Subprogram functions	Program section repeats, subprograms, program calls	
Fixed cycles	Pecking, tapping, slot cutting, rectangular and circular pockets, SL cycles (milling cycles whose contour descriptions are stored in subprograms); the machine tool manufacturer can also integrate customized macros[1)]	

[1)] These functions must be implemented by the OEM.
[2)] 3D-straight line without radius compensation

	TNC 407	**TNC 415 B/TNC 425**
Q-Parameters	Mathematical functions (=, +, -, x, ÷, sin α, cos α, angle α from sin α and cos α, $\sqrt{a}$, $\sqrt{a^2 + b^2}$, tan α, arcsin, arctan, arccos, a^n, e^n, ln, log, absolute value of a number, the constant π, negation, truncate places before or after the decimal point) logical comparisons (=,≠, >, <), Parenthetical calculations	
Program test	By graphic simulation of the part program	
Parallel operation	Yes, no graphics	Yes, with graphics
File management	up to 100 files: programs in HEIDENHAIN and DIN/ISO format, also tool [1], PLC datum shift, pallet tables[1] and text files	
Tool compensation	Tool length, tool radius in machining plane	
	–	Three-dimensional tool compensation with surface normal vectors
Central tool file	Various tool tables for 254 tools max. each, with flexible pocket coding, tool life monitoring and sister tool organization[1]	
Data interfaces	V.24/RS 232 C and V.11/RS 422 • "Blockwise transfer": programs that exceed the control capacity can be downloaded block by block and simultaneously executed. • Extended data interface with LSV/2 protocol for external TNC operation across the data interface	
Baud rate	38400; 19200; 9600; 4800; 2400; 1200; 600; 300; 150; 110	
Keyboard	TE 400 with integral QWERTY keyboard	
Screen	BC 110 14" colour monitor 640 x 490 pixels	
Logic unit	LE 407	LE 415B / LE 425
Axis control	Feed pre-control or operation with servo lag	
	–	**TNC 425**: additional digital speed control
Cycle times		
Contour interpolation	6 ms	3 ms
Fine interpolation	–	**TNC 415 B**: 0.6 ms (contour) **TNC 425:** 0.6 ms (speed)
Position control resolution	0.1 µm	

[1] These functions must be implemented by the OEM.

	TNC 407	TNC 415 B/TNC 425
Integral PLC		
PLC inputs	56 + 1 "Control is ready" input; (Option: + 64*) per PL)	
PLC outputs	31 + 1 "Control is ready" output "; (Option: + 31*) per PL) .Two PL 410 max. can be connected	
Option: analog inputs ± 10 V	4 per PL 410[1)] or PA	
Option: Inputs for thermistors	4 per PL 410[1)] or PA	
PLC program memory	Approx. 8000 logic commands	
PLC cycle time	24 ms	20 ms

Error compensation

- linear axis error compensation
- non-linear axis error compensation
- compensation of reversal spikes in circular movements
- compensation of thermal expansion
- backlash compensation
- stiction compensation
- offset compensation

Position encoders	HEIDENHAIN incremental linear and angle encoders (preferably with distance-coded reference marks) also HEIDENHAIN incremental rotary encoders	
Reference mark evaluation	Following a power interruption, automatic reference value input if reference marks are traversed	
Max. traverse	± 100 000 mm	
Max. traversing speed	300 m/min	
Feed-rate and spindle override	0 to 150% with two potentiometers at the control panel	

Accessories		
Electronic handwheel	1 x HR 330	Portable handwheel
or	1 x HR 130	Integral handwheel
or	up to 3 x HR 150	Integral handwheel with adapter HRA 110
Diskette unit	FE 401	
Touch trigger 3D probe	TS 120/TS 511	
Measuring 3D probe	—	TM 110
Touch probe for tool inspection	TT 110	

*) PL 410 B: Active analog inputs reduce the number of PLC inputs by 8 and the number of PLC outputs by 2.

	TNC 407	TNC 415 B/TNC 425
Digitizing with optional software module in TNC	• With TS 120 touch trigger 3D probe	• With TM 110 measuring touch probe
Export versions		TNC 415F / TNC 425E: Linear interpolation 4 of 5 axes, input/display resolution 1 µm
Power consumption		
NC	24 W approx.	36 W approx.
PLC	6 W approx.	
PL 410 B	25 W approx.	
BC 110 B	70 W max.	
Ambient temperature		
Operation	0 to 45°C	
Storage	–30 to 70°C	
Weights		
Logic unit	8.0 kg	10.0 kg
TE 400	2.4 kg	
BC 110 B	11.0 kg	
PL 410 B	3.1 kg	

3 Software

The logic unit contains separate software for the NC section and the PLC section. The software is identified by an 8-figure number.

After switching on the control, the NC, PLC and Software-Options software numbers are displayed on the screen. The software number can also be directly requested with the aid of the MOD function.

3.1 NC-Software

3.1.1 NC-Software number

The 8-figure NC software number identifies the type of control, the dialog language (language of the country) and the version of the software.

259 96 0 01

Software type

National language

- 0 = German
- 1 = Czech
- 2 = French
- 3 = Italian
- 4 = Spanish
- 6 = Swedish
- 7 = Danish
- 8 = Finnish
- 9 = Dutch

Software version

In addition to the above-listed languages, the TNC can always use **English,** which may be selected via the machine parameter MP7230.

3.1.2 Software types

Due to restrictions on the export of the TNC 415 B, HEIDENHAIN can also deliver a special export version. This export version is differentiated from the standard control through the installed software type. With this software type, the control offers different features in respect of linear interpolation and the entry/display resolutions.

	Linear interpolation	Entry/Display Resolution
TNC 415 B (Standard)	5 of 5 axes	up to 0.1 µm
TNC 415 F (Export)	4 of 5 axes	up to 1 µm

The TNC 407 does not fall under the export restrictions, therefore no export version is necessary.

New functions will be introduced only in the following software types:

TNC 415B / TNC 425:	280 54
TNC 415F / TNC 425 E:	280 56
TNC 407:	280 58

These software types are therefore supplied as standard.

3.1.3 Software option

HEIDENHAIN offers the "Digitising with TS 120" function as a software option (see chapter "Machine integration"). An additional software protection module is installed in controls supplied with this software option. The Id.-Nr. of the logic unit indicates another version. If the software module is installed, the option is indicated on the screen under the NC and PLC software numbers.

Logic units that have already been delivered can be retrofitted with the software protection module. Please contact HEIDENHAIN if you wish to buy this option for your existing control. The proper component model must be ordered for a specific hardware model.

Digitizing with TS 120:

<table>
<tr><th></th><th>Id.-Nr. of the installation kit</th><th>Id.-Nr. of software module</th><th>NC software</th></tr>
<tr><td>TNC 407</td><td>265 313 01</td><td rowspan="3">246 051 01</td><td>243 02, 243 03, 280 58</td></tr>
<tr><td>TNC 415A</td><td>265 314 01</td><td>259 96</td></tr>
<tr><td>TNC 415B
TNC 425</td><td>286 405 01</td><td>280 54</td></tr>
</table>

Digitizing with TM 110:

	Id.-Nr. of the installation kit	Id.-Nr. of software module	NC software
TNC 415B TNC 425	286 405 02	246 051 03	280 54

3.1.4 Software/Hardware

Various hardware versions of the logic units LE 407 and LE 415 have until now been delivered (please refer to the chapter 3 ”Mounting and electrical installation” section 1). The new software types are not compatible with all hardware versions. The valid combinations are shown in the following tables.

TNC 415A / TNC 415E:

		Software Type			
Id.-Nr.	LE	243 05	259 91	259 96	259 97
251 481 78	415 E	-	-	-	X
251 481 79	415 A	-	-	X	-
251 481 88	415 E	-	X	-	X
251 481 89	415 A	X	-	X	-
251 481 98	415 E	-	X	-	-
251 481 99	415 A	X	-	-	-
258 993 78	415 E	-	-	-	X
258 993 79	415 A	-	-	X	-
258 993 88	415 E	-	X	-	X
258 993 89	415 A	X	-	X	-
258 993 98	415 E	-	X	-	X
258 993 99	415 A	X	-	X	-
264 429 78	415 E	-	-	-	X
264 429 79	415 A	-	-	X	-
264 429 98	415 E	-	X	-	X
264 429 99	415 A	X	-	X	-

TNC 415B / TNC 415F:

Id.-Nr.: 267 223 xy

TNC 425 / TNC 425E:

Id.-Nr.: 267 214 xy

x = Identifier for hardware change
y = Version
- 3 = Export version with software module "Digitizing with TS 120"
- 4 = Standard version with software module "Digitizing with TS 120"
- 7 = Standard version with software module "Digitizing with TS 120"
- 8 = Export version without option
- 9 = Standard version without option

TNC 407:

Id.-Nr. of the LE	Software Type 243 07	243 02	243 03	280 58
256 113 99	X	-	-	-
255 444 79	-	X	X	X
255 444 89	X	X	X	X
255 444 99	X	X	X	X
261 092 79	-	X	X	X
261 092 89	X	X	X	X
261 092 99	X	X	X	X
264 430 24	X	X	X	X
264 430 29	X	X	X	X
264 430 79	-	X	X	X
264 430 99	X	X	X	X

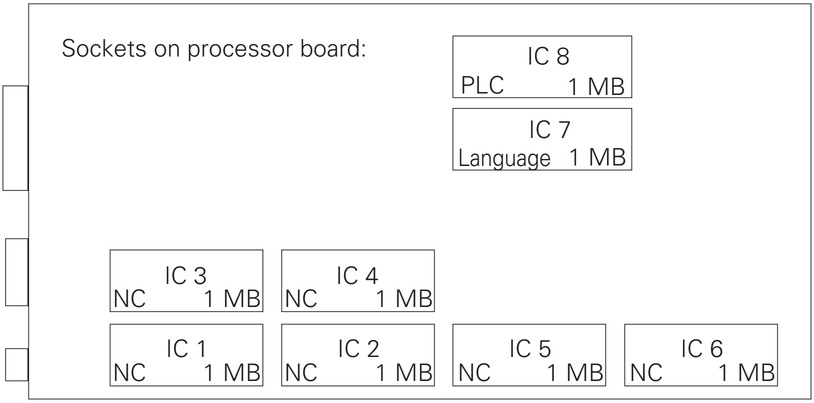
3.2 PLC-Software

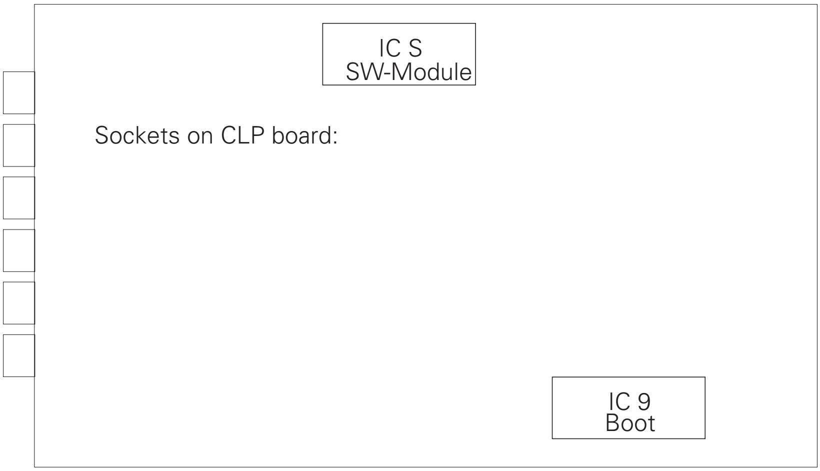
The PLC software is produced by the manufacturer of the machine. Either HEIDENHAIN or the manufacturer of the machine can store this software in EPROMs. HEIDENHAIN assigns PLC software numbers to the machine manufacturers on request. HEIDENHAIN can archive the specific PLC programs in a data bank, so that the installation of the correct PLC program is assured if a control has to be exchanged.

The PLC EPROM must be written in the format of the corresponding software type! Beginning with software types 280 54, 280 56 and 280 58, either a 1 MB or a 2 MB EPROM can be used for the PLC.

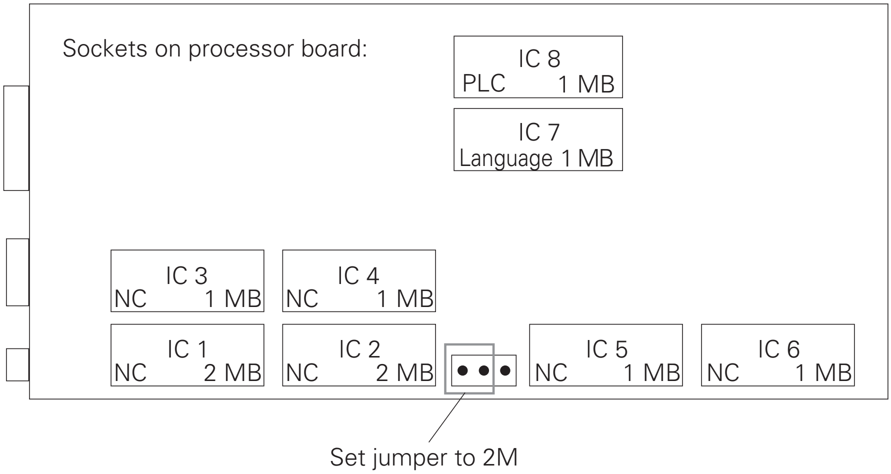
3.3 EPROM sockets

3.3.1 TNC 415 A/TNC 415 E

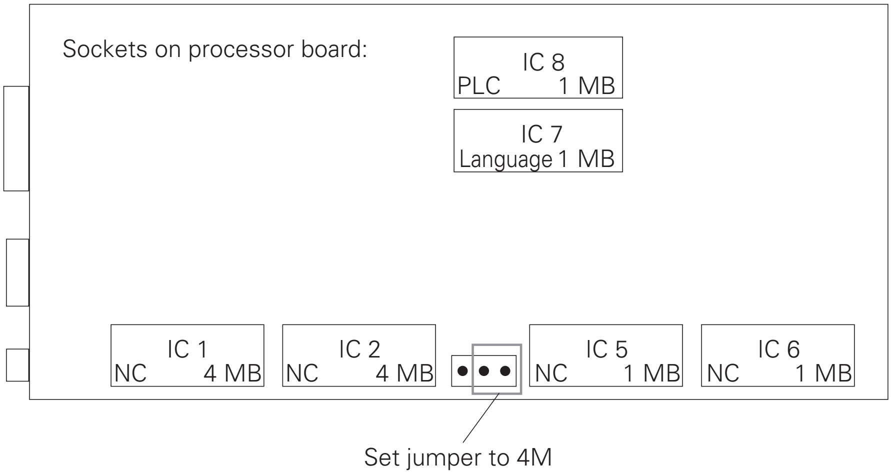


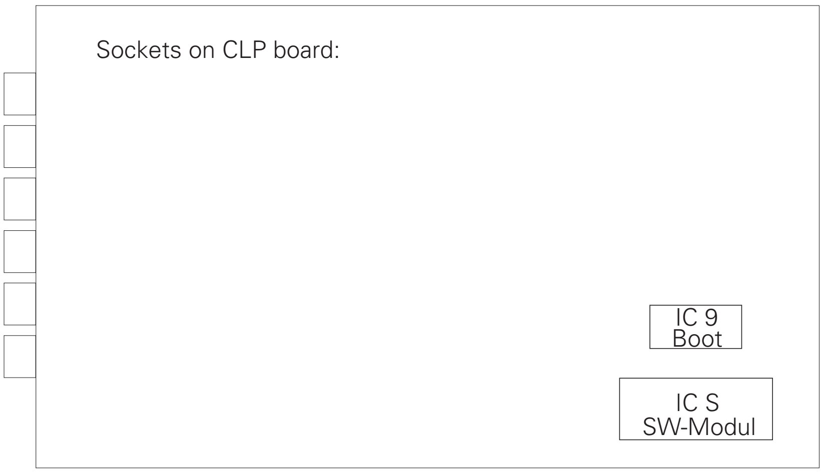


3.3.2 TNC 415 B/TNC 415 F and TNC 425 A/TNC 425 E

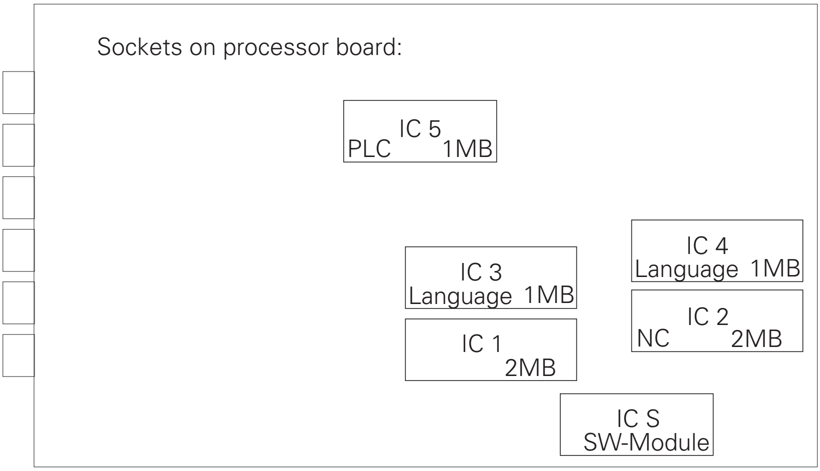


or





3.3.3 TNC 407



3.4 Software replacement

Before replacing the software on a TNC the data of all file types, the machine parameter list, the non-linear axis error compensation table and the PLC program (if in the RAM) must be backed up.

You can use the TNC BACKUP routine to do this. In the "Machine Parameter Editor" mode (code number 95 148) press the MOD key to display the menu for configuring the data interface. The "BACKUP DATA" and "RESTORE DATA" soft keys will be offered.

Use "BACKUP DATA" to transmit all operating parameters and the data of all file types across the data interface and store them in the $BACKUP.ANC file. When software replacement is complete, use "RESTORE DATA" to download the data to the TNC again.

3.5 Releases

New NC software versions are released by HEIDENHAIN at irregular intervals.

3.5.1 Software types 243 05, 259 91 and 243 07

Standard version TNC 415 A

NC-Software-Version	Release
243 05x 02	2/90
243 05x 03	4/90
243 05x 04	5/90
243 05x 05	8/90
243 05x 08	9/90
243 05x 10	2/91
243 05x 11	4/91
243 05x 12	5/91
243 05x 13	6/91
243 05x 14	11/91
243 05x 15	3/92
243 05x 16	7/93

Export version TNC 415 E

NC-Software-Version	Release
259 91x 08	11/90
259 91x 10	2/91
259 91x 11	4/91
259 91x 12	5/91
259 91x 13	6/91
259 91x 14	11/91
259 91x 15	3/92
259 91x 16	7/93

TNC 407

243 07x 03	7/90
243 07x 05	10/90
243 07x 07	2/91
243 07x 08	4/91
243 07x 09	5/91
243 07x 10	6/91
243 07x 11	11/91
243 07x 12	3/92
243 07x 13	7/93

New releases of these software types will be made only to correct errors.

3.5.2 Software types 259 96, 259 97 and 243 02

TNC 415 A: 259 96x 01
TNC 415 E: 259 97x 01
TNC 407: 243 02x 01 Release 8/91

New functions:

– Digitising with TS 120
– Rigid tapping
– Input resolution and display step 0.0001 mm for TNC 415 A
– Re-approaching the contour
– Compensation of reversal spikes in circular movements
– New format for PLC-EPROM
– Machine parameter editor
– FN15: PRINT
– New machine parameter for setting screen colours

– Thermal expansion compensation
– Machine datum
– "Free rotation"
– Changing the gear range through the PLC
– RPM upper limit per gear range
– Change in the organisation of PLC error messages
– Marker for special tool
– Interruption of PLC positioning
– Fast PLC input
– Arc end-point tolerance
– Locking of file types
– D596 rapid traverse from PLC
– D276 code number
– Axis-specific monitoring functions
– Compensation of tool length in the position display
– PLC: module 7031, module 9032, module 9083, module 9093, module 9094

TNC 415 A: 259 96x 02
TNC 415 E: 259 97x 02
TNC 407: 243 02x 02 Release 1/92

New functions:

– If MP7490 = 0, the datum is effective for all traverse ranges
– Non-linear characteristic curve for override potentiometer
– New process for traversing reference marks (MP1350.x = 3)

TNC 415 A: 259 96x 03
TNC 415 E: 259 97x 03
TNC 407: 243 02x 03 Release 3/92

New functions:

– ROT display in the status window
– PLC error messages if more than one of the markers M2485 to M2487 is set
– MP4070 has been added
– Handwheel symbol on screen for selected axis
– MP7640 (machine with handwheel) has new input values
– MP340 has been added (only for TNC 407)
– Module 9041 has been introduced (only for TNC 415 A)
– Dialog texts in Czech

TNC 415 A: 259 96x 04
TNC 415 E: 259 97x 04
TNC 407: 243 02x 04 Release 5/92

New functions:

- Multipoint axis error compensation: the maximum distance between compensation points was increased to 2^{23} .
- The maximum input range for position values was increased to –99 999.9999 mm to +99 999,9999 mm.
- The override adjustment in 2 % steps was dropped.
- The minimum input value for MP6130 was changed to 1 mm and for MP6140 to 0.001 mm.
- MP440 was dropped.
- Mid-program startup (block scan) now works for ISO programs during blockwise transfer with simultaneous execution.
- New Module 9035

TNC 415 A: 259 96x 05
TNC 415 E: 259 97x 05
TNC 407: 243 02x 05 Release 5/92

New functions:

- The input value 4 was added to MP7480.
- The distance D to an auxiliary point with the coordinates PDX and PDY is now entered without algebraic sign.

TNC 415 A: 259 96x 06
TNC 415 E: 259 97x 06
TNC 407: 243 02x 06 Release 9/92

New functions:

- Electronic Handwheel mode
  A new marker 2826 has been introduced. Setting this marker disables the evaluation of the counting pulses received by the handwheel. The handwheel cannot be used to position when the marker is set.

- Transfer values to PLC
  Numerical values can be entered in the new machine parameters MP4230.00 to MP4230.31 and MP4231.0 to MP4231.31 which the PLC can read with module 9032.

MP4230.0 to MP4230.31 MP4231.0 to MP4231.31	Transfer value to PLC with module 9032 Entry: –99999.9999 to +99999.9999

- DIN/ISO Programming

  The maximum permitted length of an NC block in DIN/ISO has been increased from 130 to 150 characters.

TNC 415 A: 259 96x 07
TNC 415 E: 259 97x 07
TNC 407: 243 02x 07 Release 11/92

New functions:

– Because it is run from standstill, the spindle orientation at the beginning of the "Rigid Tapping" cycle is always executed in the direction of rotation that reaches the target by the shortest route. Previously the direction of rotation with each spindle orientation was selected by marker M2656.

TNC 415 A: 259 96X 08
TNC 415 E: 259 97X 08
TNC 407: 243 02X 08 Release 1/93

New functions:

– If the PLC simultaneously issues two commands from memory areas PLC positioning (M2704 to M2708), confirm Q parameter (M2713) and PLC datum shift (M2716), then the NC flashes the message "Error in PLC program 1R" as these commands use the same memory areas for data transfer.

  PLC positioning in more than one axis counts as a single command. If marker M2719 (switch-over from word processing to TNC 355 mode) has the value 0, the same checks will be run with the corresponding TNC 355 strobes.

TNC 415 A: 259 96X 09
TNC 415 E: 259 97X 09
TNC 407: 243 02X 09 Release 10/93

New functions:

– Machine parameters MP951.x and MP7450 were introduced for calculating PLC positionings during block scan.

3.5.3 Software types 259 93, 259 94 and 243 03

TNC 415 B/TNC 425: 259 93x 04
TNC 415 F/TNC 425 E: 259 94x 04 Release 11/92

New functions:

- New functions for programmed contour approach and withdraw
- "Working Plane" cycle for machines with swivel head
- "Contour Train" cycle
- "Scaling factor axis-specific" cycle
- Three-dimensional tool compensation in DNC mode
- Extended tool management
- Extended additional status displays for file information, positions, tool data and coordinate transformation
- Stiction compensation
- PLC axes
- Extended PLC command set
- Length of filenames increased to 16 characters
- Automatic correction of centre offset for probe system
- Oscilloscope function
- New organization for PLC files
- Additional machine parameters for data transfer to PLC
- BACKUP and RESTORE functions
- Q parameter for tool length (Q114)
- Axis positions after programmed probe cycle in Q parameters Q115 to Q119
- New input format for machine parameters

TNC 415 B/TNC 415: 259 93x 05
TNC 415 F/TNC 425 E: 259 94x 05 Release 11/93

New functions:

- Status information can now be read with module 9035. See "TNC 407/TNC 415 B/TNC 425 Description of the Differences from TNC 415".
- The minimum input range of machine parameters MP6120, MP6350, MP6360 has been changed from 80 mm/min to 10 mm/min.
- MP7300 has been extended as follows:

MP7300 Delete status display, Q parameter and tool data
Entry: 0 to 7
0 = delete status display, Q parameter and tool data when selecting a program
1 = delete status display, Q parameter and tool data with M02, M30, END PGM and selecting a program
2 = delete status display and tool data when selecting a program
3 = delete status display and tool data with M02, M30, END PGM and selecting a program
4 = delete status display and Q parameter when selecting a program
5 = delete status display and Q parameter with M02, M30, END PGM and selecting a program
6 = delete status display when selecting a program
7 = delete status display with M02, M30, END PGM and selecting a program

TNC 407: 243 03x 05 Release 11/92

New functions:

All functions as for TNC 415 B except "Working Plane" cycle and three-dimensional tool compensation.

TNC 415 B/TNC 425: 259 93x 06
TNC 415 F/TNC 425 E: 259 94x 06
TNC 407: 243 03x 06 Release 12/92

New functions:

– MP7411 is used to select whether to use the tool data (length, radius, axis) from the last TOOL CALL block or from the calibrated data of the probe system in a touch probe block.

– **MP7411** **Tool data in touch probe block**
  Entry: 0 or 1
  0 = In the touch probe block the current tool data are overwritten with the calibrated data of the probe system.
  1 = Current tool data are retained even with a touch probe block.

– The displayable area for FK graphics has been restricted to -30,000 mm to + 30,000 mm. The maximum edge length is 30,000 mm.

TNC 415 B/TNC 425: 259 93X 07
TNC 415 F/TNC 425 E: 259 94X 07
TNC 407 243 03X 07 Release 3/93

New functions:

– PLC module 9033 has been introduced. This module allows the user to select a particular machine parameter file in a SUBMIT job and then execute a Reset so the control system boots up with this MP file.

– If the PLC simultaneously issues two commands from memory areas PLC positioning (M2704 to M2708), confirm Q parameter (M2713) and PLC datum shift (M2716), then the NC flashes the message "Error in PLC program 1R" as these commands use the same memory areas for data transfer.

– PLC positioning in more than one axis counts as a single command. If marker M2719 (switch-over from word processing to TNC 355 mode) has the value 0, the same checks will be run with the corresponding TNC 355 strobes.

– The offset for a variable-speed spindle is now adjusted with a new algorithm. A single offset adjustment is made two seconds after the nominal value has reached target position. Every second thereafter the offset voltage is increased by 0.152 mV or decreased by 0.152 mV when servo lag is greater or less than zero respectively, and if the voltage computed from the product of servo lag and Kv factor is greater than 0.152 mV. This means that the offset voltage changes by 1 mV in 7 seconds approx.

TNC 415 B/TNC 425:	**259 93x 08**	
TNC 415 F/TNC 425 E:	**259 94x 08**	**This versions was supplied from 6/93 only when**
TNC 407:	**243 03x 08**	**expressly requested by the customer.**

New functions:

– LSV2 protocol
– PLC axes
– Help files
– Synchronized axes
– New compensation value table (simultaneous compensation of sag and ballscrew pitch error)
– Cycle 3 "Slot Milling" modified
– Cycle 27 "Cylinder Surface" new
– "Working Plane": Displays are referenced to the tilted coordinate system; Touch probe functions in the tilted coordinate system: Datum setting in the tilted system
– M94 new (modulo 360°)
– M103 new (reduced feed rate during plunge cutting)
– M105/M106 new (second set of k_V factors)
– M112 new (Tolerance field for "Look ahead")
– M116 new (feed rate for rotary axes in mm/min)
– M118 new (Handwheel overlapping)
– Min. and max. memory for DNC operation
– Calculation with parentheses and expanded parameter functions
– MP7470 was eliminated
– Graphic depiction of the workpiece blank in the working space
– M114 new (automatic compensation of machine geometry during machining with tilted axes)
– Module 9150 new
– Input/Output of tool and pocket tables
– Module 9035 was expanded

TNC 415 B/TNC 425:	**259 93x 09**	
TNC 415 F/TNC 425 E:	**259 94x 09**	**This version was supplied from 8/93 only**
TNC 407:	**243 03x 09**	**when expressly requested by the customer.**

New functions:

– The export versions TNC 415F and TNC 425E were improved with linear interpolation in 4 of 5 axes (previously 3 of 5 axes).

TNC 415 B/TNC 425:	**259 93x 10**	
TNC 415 F/TNC 425 E:	**259 94x 10**	
TNC 407:	**243 03x 10**	**Release 9/93**

TNC 415 B/TNC 425:	**259 93x 11**	
TNC 415 F/TNC 425 E:	**259 94x 11**	**This version was supplied from 4/94 only**
TNC 407:	**243 03x 11**	**when expressly requested by the customer.**

New functions:

– The PLC module 9036 was expanded. The handwheel assignment can now be switched through the PLC to any desired axis; the assignment is indicated in the status window by the position of the handwheel symbol.
– After activation of Cycle 19 "Working plane" the offset is corrected only in the axis that is moved. Previously the offset was corrected simultaneously in all axes during execution of the first block.
– With M112 it is now possible to enter a limit angle A in addition to the tolerance T.

TNC 415 B/TNC 425:	**259 93x 12**	
TNC 415 F/TNC 425 E:	**259 94x 12**	
TNC 407:	**243 03x 12**	**Release 5/94**

3.5.4 Software types 280 54, 280 56 and 280 58

TNC 415 B/TNC 425: 280 54x 01
TNC 415 F/TNC 425 E: 280 56x 01
TNC 407: 280 58x 01 Release 6/94

New functions:

- Digitizing with TM 110
- Program structuring
- Tool measuring with TT 110
- Complete NC block with Actual Position Capture key
- M124 new

TNC 415 B/TNC 425: 280 54x 02
TNC 415 F/TNC 425 E: 280 56x 02
TNC 407: 280 58x 02 Release 11/94

New functions:

- A "PLC" column was added to the tool table. As in the pocket table, relevant PLC data can be entered here. This column is activated with machine parameter **MP7266.17**.
- A datum, tool, or pocket table can be searched for to certain values using **Module 9092**.
- The PLC can activate the geometry data of the tool from W264 with the new marker **M2717**.
- The elements "Number of cutting edges", "Tolerance for tool length", "Tolerance for tool radius", "Cutting direction of the tool" and "PLC status" were added to Modules 9093 and 9094.
- During feed pre-control, the positioning window is not evaluated until the current velocity is less than **MP1525**. In the previous software versions, the positioning window was not evaluated until the velocity fell below 0.5 mm/min. Increasing the input value in MP1525 increases the block processing time.
- Feed pre-control can be activated for all modes of operation with machine parameter **MP1391**. This becomes necessary when stiction compensation is to be active in the manual modes of operation.
- The maximum permissible positional difference between the two gantry axes is defined in **MP855**. When this value is exceeded, the blinking error message `GROSS POSITIONING ERROR <AXIS> # A` appears. The current positional deviation appears in the LAG display of the slave axis.
- **Module 9171** makes it possible to orient the spindle at a rotational speed defined by the PLC.
- **Modules 9040** and **9041** make it possible to read coordinates referenced to a shifted coordinate system.
- The jog increment can now be limited with Module 9036.
- With the new marker **M2827** an EMERGENCY STOP (control-is-ready PLC input signal) can be suppressed and, instead, all control loops opened and an NC stop executed.
- The new marker **M2830** opens all control loops and stops the NC.
- In a digitized data file that has been produced with the TM 110 touch probe, the BLK FORM is enlarged in the working plane by double the value of the deflection depth (MP6310).
- The radius compensation in the working plane is now also effective in NC blocks with 5-axis interpolation. The NC block can contain, besides the three non-parallel linear axes, also two rotary axes.
- Tool measurement with the TT 110 is possible only on machines with controlled spindle (Spindle orientation).

TNC 415 B/TNC 425:	**280 54x 03**	
TNC 415 F/TNC 425 E:	**280 56x 03**	
TNC 407:	**280 58x 03**	**This version was never released.**

TNC 415 B/TNC 425:	**280 54x 04**	
TNC 415 F/TNC 425 E:	**280 56x 04**	
TNC 407:	**280 58x 04**	**Release 3/95**

New functions:

– To make changes to the tool table it is no longer necessary to transfer the entire tool table. It now suffices to transfer only those tool data that have changed. These partial data can be transferred in the Program Run mode of operation

– For a rotary table display that is set to modulo 360°, the M function **M 126** positions the table by the shortest path. **M 127** positions the table, as before, by the path that does not cross over the 0° position.

The function for automatic **tool measurement** was expanded with the TT 110. The new fields L-OFFS, R-OFFS, LBREAK and RBREAK were introduced to the tool table (**MP7266.19** to **MP7266.21**). In addition, in Cycle 31 the radius is no longer entered and the fields L-OFFS, R-OFFS, LBREAK and RBREAK were added to the modules 9092, 9093 and 9094. The new marker **M2393** is set whenever the break tolerance is exceeded.

TNC 415 B/TNC 425: 280 54x 05
TNC 415 F/TNC 425 E: 280 56x 05
TNC 407: 280 58x 05 Release 3/96

Improvements:

- The Polish dialog language was added, Id. Nr. 280 590 xx, 280 550 xx and 280 570 xx.

- Contour Pocket cycle (cycle 6, 15, 16, 21, 22, 23, 24)
  MP7420
  Bit 4 = 0: After the contour pocket is machined, the TNC moves to the position that was last approached before the cycle was called.
  Bit 4 = 1: After the contour pocket is machined, the TNC moves the tool to clearance height.
- Module 9036 Limiting jog increment
  Transferred value:
  0 to 50 mm = Limiting jog increment
  –1; <–2 or >50 = Jog increment limit is canceled and the last increment entered is activated.
  –2 = Jog increment limit is canceled and the minimum from the last increment entered and the last limiting value is activated.

TNC 415 B/TNC 425: 280 54x 06
TNC 415 F/TNC 425 E: 280 56x 06
TNC 407: 280 58x 06 Release 5/96

Improvements:

- MP1925 Limit for integral factor of the speed controller new.

Mounting and electrical installation – Contents

1 Hardware components

The TNC 415B consists of the following hardware components:
- . LE 415B (Logic unit),
- . TE 400 (TNC keyboard),
- . BC 110 B (Visual display unit),
- . PL 410 B (max. 2 PLC-I/O boards, optional).

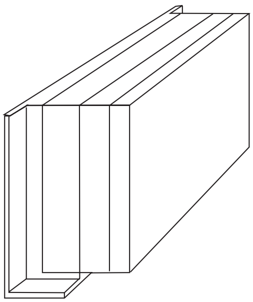
The export version which is offered is the TNC 415 F. In the TNC 415 F, an LE 415F is delivered in place of the LE 415B. The export software is built into the LE 415 F.

The TNC 407 consists of the following hardware components:
- . LE 407 A (Logic unit),
- . TE 400 (TNC keyboard),
- . BC 110 B (Visual display unit),
- . PL 410 B (max. 2 PLC I/O boards, optional).

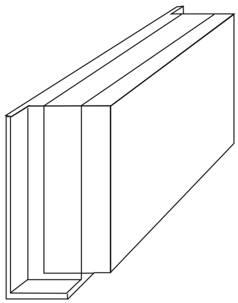
The TNC 407 is not subject to export restrictions. An export version is not required.

. Logic unit

LE 415 B (Id.-Nr. 267 223 ..)

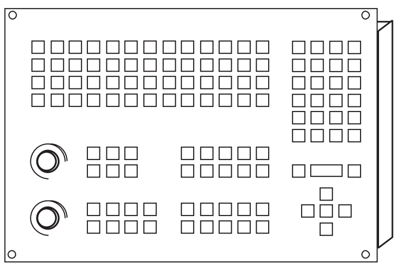


LE 407 A (Id.-Nr. 264 430 ..)



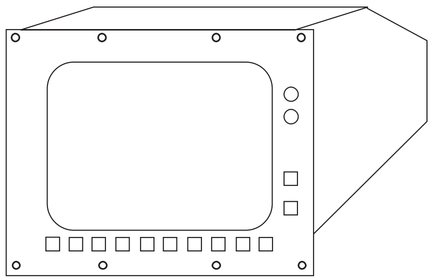
. TNC keyboard

TE 400 (Id.-Nr. 250 517 03)
Joined to the logic unit via connecting cable.



. VDU

BC 110 B (Id.-Nr. 260 520 01)
14 inch colour VDU with soft keys.
Joined to the logic unit and the TNC keyboard via connecting cable.



. PLC I/O unit (Option)

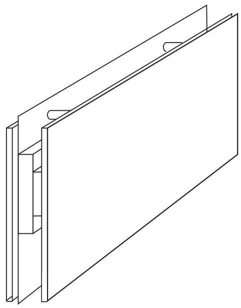
PL 410 B without analog inputs
(Id.-Nr. 263 371 12)

64 PLC inputs
31 PLC outputs
1 “Control is operational” output

PL 410 B with analog inputs
(Id.-Nr. 263 371 02)

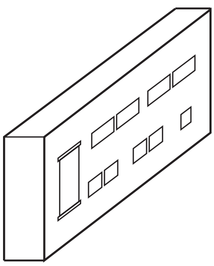
56 PLC inputs
29 PLC outputs
4 analog inputs ± 10V
4 inputs for Pt 100 thermistors
1 “Control is operational” output

Connected with the logic unit by cable. No more than two PL 410 B can be connected to the LE.



. PLC analog board (Option)

PA 110 (Id.-Nr. 262 651 01)
Connected with the logic unit via cable or with the first PLC I/O unit.
4 analog inputs for ± 10 V DC
4 analog inputs for Pt100 thermistors



1.1 Changes in the ID-number

If development or manufacturing requirements make it necessary to alter any of the hardware components, HEIDENHAIN will change the ID-numbers of the hardware components.

ID-Numbers assigned to date:

. VDU:

Id.-Nr. 254 740 01	BC 110	Separate power-supply for fan	until 5/91
Id.-Nr. 260 520 01	BC 110 B	Power supply for fan taken from VDU supply	since 11/90

. PLC-I/O board:

Id.-Nr. 252 855 01	PL 400	since 2/90
Id.-Nr. 263 371 01	PL 410	since 10/92
Id.-Nr. 263 371 02	PL 410 B with analog inputs (5 V interface)	since 10/94
Id.-Nr. 263 371 12	PL 410 B without analog inputs (5 V interface)	since 10/94

. TNC keyboard:

Id.-Nr. 250 517 01	TE 400	since 2/90
Id.-Nr. 250 517 03	TE 400 with APPR/DEP key	since 9/92

. Logic unit LE 415:

Id.-Nr.	Logic unit	Change	
251 481 99	LE 415		until 12/90
251 481 89	LE 415	Colour graphics chip; Expansion slot for software module	since 9/91 since 9/91
251 481 88	LE 415 E	Export version of the Id.-Nr. 251 481 89	
251 481 79	LE 415	Like Id.-Nr. 251 481 89, but with software module	since 9/91
251 481 78	LE 415 E	Export version of the Id.-Nr. 251 481 79	since 9/91
258 993 99	LE 415	Changed power supply and graphics chip	since 11/90
258 993 98	LE 415 E	Export version of Id.-Nr. 258 993 99	since 11/90
258 993 89	LE 415	Expansion slot for software module	since 9/91
258 993 88	LE 415 E	Export version of Id.-Nr. 258 993 89	since 9/91
258 993 79	LE 415	Like Id.-Nr. 251 481 89, but with software module	since 9/91
258 993 78	LE 415 E	Export version of Id.-Nr. 258 993 79	since 9/91
264 429 99	LE 415	Expansion slot for software module	since 6/91
264 429 98	LE 415 E	Export version of Id.-Nr. 264 429 99	since 6/91
264 429 79	LE 415	Like Id.-Nr. 264 429 99, but with software module	since 9/91
264 429 78	LE 415 E	Export version of Id.-Nr. 264 429 79	since 9/91
267 223 29	LE 415 B	New processor board	since 9/92
267 223 28	LE 415 F	Export version of Id.-Nr. 267 223 29	since 9/92
267 223 24	LE 415 B	Like Id.-Nr. 267 223 29 but with software module	since 9/92
267 223 23	LE 415 F	Export version of Id.-Nr. 267 223 24	
267 223 39	LE 415 B	Hardware change	since 2/93
267 223 38	LE 415 F	Export version of Id.-Nr. 267 223 39	since 2/93
267 223 34	LE 415 B	Like Id.-Nr. 267 223 39 but with software module	since 2/93
267 223 33	LE 415 F	Export version of Id.-Nr. 267 223 34	since 2/93
267 223 49	LE 415 B	Uninterruptible supply to control panel	since 5/93
267 223 48	LE 415 F	Export version of Id.-Nr. 267 223 49	since 5/93
267 223 44	LE 415 B	Like Id.-Nr. 267 223 49, but with software module	since 5/93
267 223 43	LE 415 F	Export version of Id.-Nr. 267 223 44	since 5/93

. Logic unit LE 407:

Id.-Nr.	Logic unit	Change	
256 113 99	LE 407 A	Pilot lot 32-bit bus	until 9/90
255 444 99	LE 407 A	16-bit bus; colour graphics chip	8/90 until 5/91
255 444 89	LE 407 A	Expansion slot for software module	since 9/91
255 444 79	LE 407 A	Like Id.-Nr. 255 444 89, but with software module	since 9/91
261 092 99	LE 407 A	32-bit bus	since 5/91
261 092 89	LE 407 A	Expansion slot for software module	since 9/91
261 092 79	LE 407 A	Like Id.-Nr. 261 092 89, but with software module	since 9/91
264 430 99	LE 407 A	Expansion slot for software module	since 6/91
264 430 79	LE 407 A	Like Id.-Nr. 264 430 99 but with software module	since 9/91
264 430 29	LE 407 A	Uninterruptible supply to control panel	since 5/93
264 430 24	LE 407 A	Like Id.-Nr. 264 430 29 but with software module	since 5/93

2 Assembly hints

2.1 Electrical noise immunity

Please note that the vulnerability of electronic equipment to noise increases with faster signal processing and higher sensitivity.
Please protect your equipment by observing the following rules and recommendations.

Noise voltages are mainly produced and transmitted by capacitive and inductive coupling. Electrical noise can be picked up by the inputs and outputs to the equipment, and the cabling.

Likely sources of interference are:

– Strong magnetic fields from transformers and electric motors,
– Relays, contactors and solenoid valves,
– High-frequency equipment, pulse equipment and stray magnetic fields from switch-mode power supplies,
– Mains leads and leads to the above equipment.

Electrical interference can be avoided by:

– A minimum distance between the logic unit (and its leads) and interfering equipment > 20 cm.
– A minimum distance between the logic unit (and its leads) and cables carrying interference signals > 10 cm.
  (Where signal cables and cables which carry interference signals are laid together in metallic ducting, adequate decoupling can be achieved by using a grounded separation screen)
– Screening according to DIN VDE 0160.
– Potential compensating lines-∅ ≥ 6 mm² (see earthing plan).
– Use of original HEIDENHAIN cables, connectors and couplings.

2.2 Heat generation and cooling

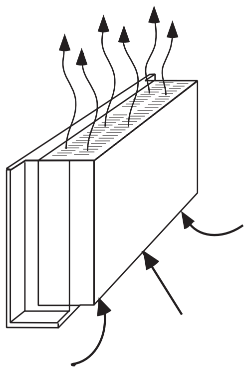
Please note that the reliability of electronic equipment is greatly reduced by continuous operation at elevated temperatures. Please make the necessary arrangements to keep within the permissible ambient temperature range.

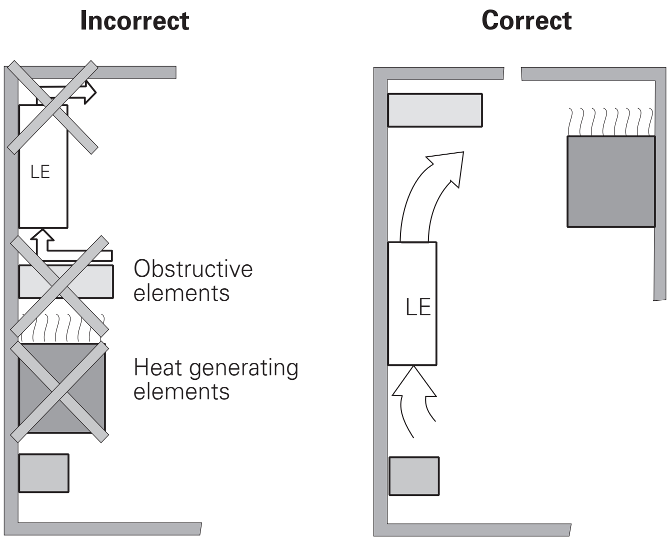
Permissible ambient temperature in operation: 0° C to 45° C

The following means may be employed to ensure adequate heat removal:

– Provide sufficient space for air circulation.
– Build in a ventilator fan to circulate the air inside the control cabinet. The fan must reinforce the natural convection. It must be mounted so that the warm air is extracted from the logic unit and no pre-warmed air is blown into the unit. The warmed-up air should flow over surfaces which have good thermal conductivity to the external surroundings (e.g. sheet metal).
– For a closed steel housing without assisted cooling, the figure for heat conduction is 3 Watt/m² of surface per °C air temperature difference between inside and outside.
– Use of a heat exchanger with separate internal and external circulation.

– Cooling by blowing external air through the control cabinet to replace the internal air. In this case the ventilator fan must be mounted so that the warm air is extracted from the control cabinet and only filtered air can be drawn in. HEIDENHAIN advises against this method of cooling, since the function and reliability of electronic assemblies are adversely affected by contaminated air (fine dust, vapours etc.). In addition to these disadvantages, a filter which is not adequately serviced leads to a loss in cooling efficiency. Regular servicing is therefore absolutely vital.





2.3 Humidity

Permissible humidity: < 75 % in continuous operation,
< 95 % for not more than 30 days p.a. (randomly distributed).

In tropical areas it is recommended that the TNC is not switched off, so that condensation is avoided on the circuit boards. The heat generation prevents condensation and has no further disadvantages.

2.4 Mechanical vibration

Permissible vibration: < 0.5 g

2.5 Mounting position

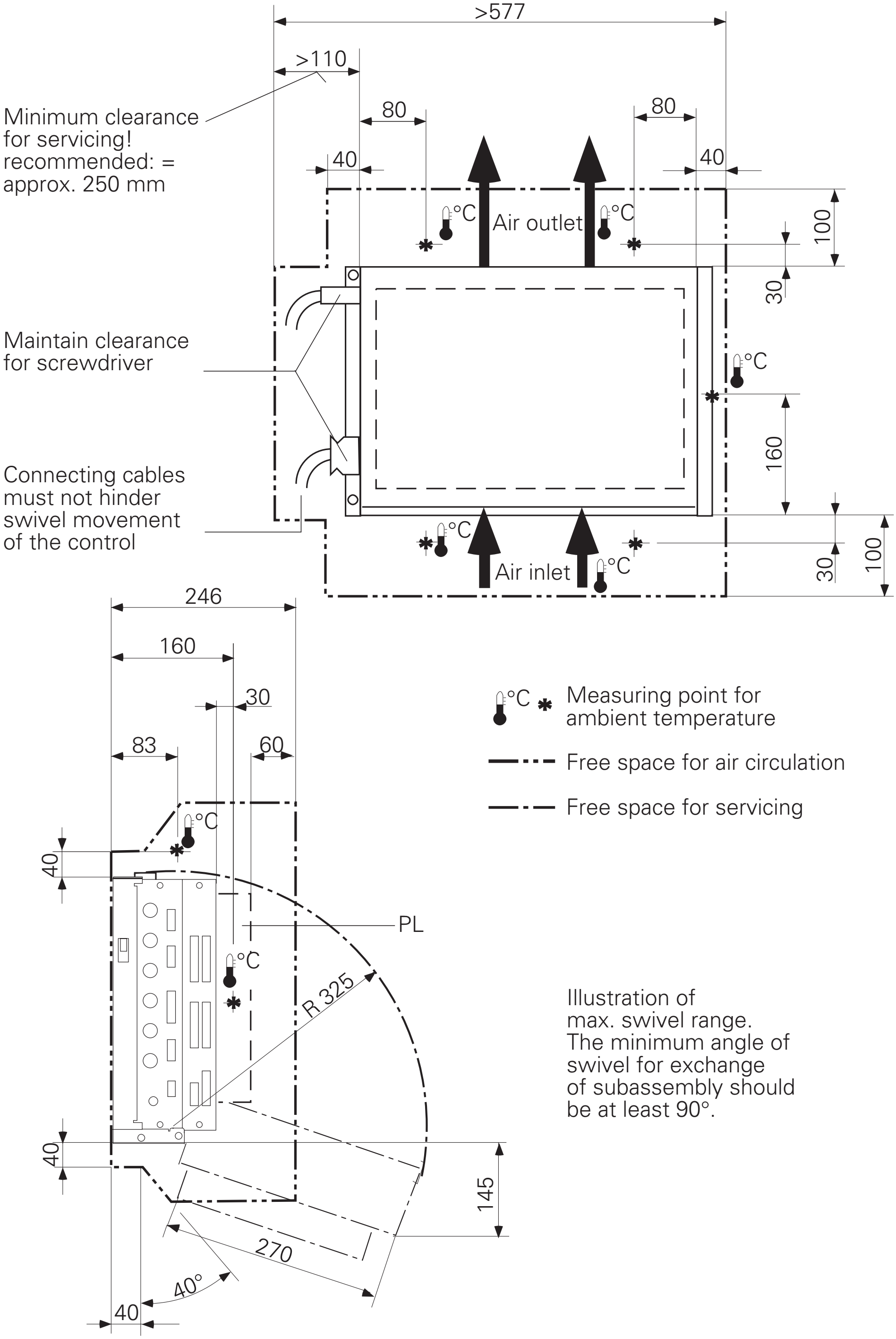
Note the following fundamental points on mounting:

- mechanical accessibility,
- permissible environmental conditions,
- electrical noise immunity,
- the electrical regulations which are in force in your country.

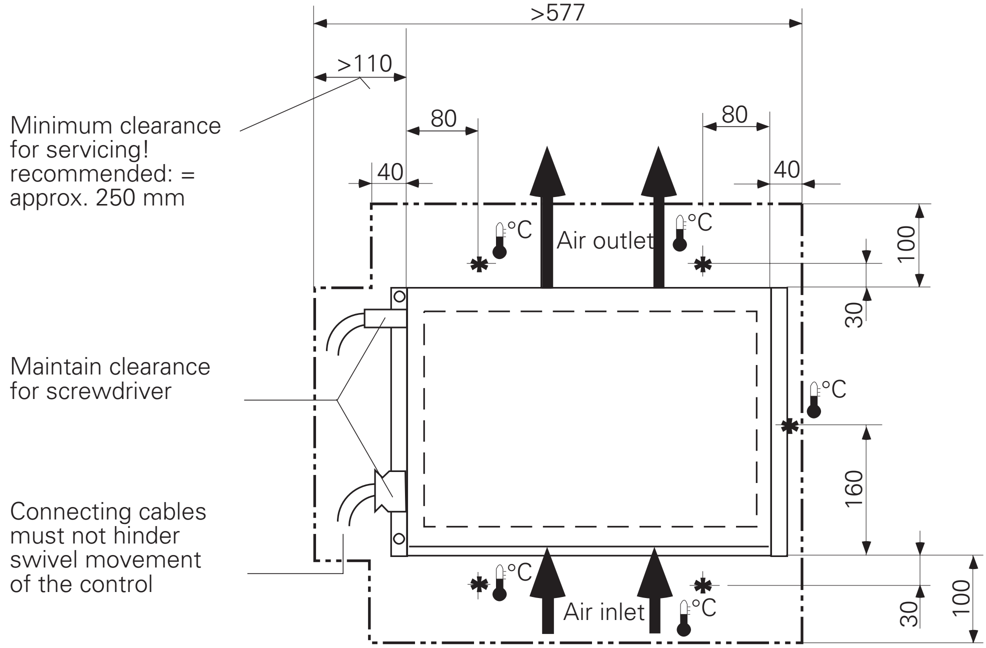
2.5.1 Logic unit

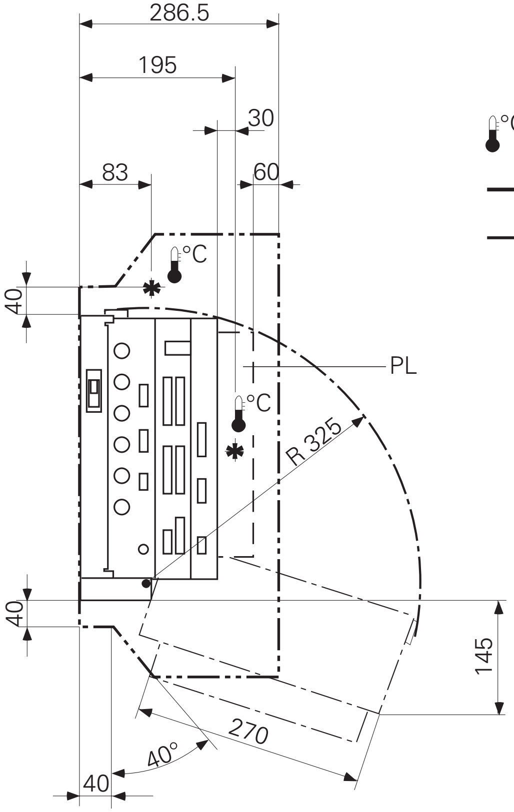
HEIDENHAIN recommends the following mounting position:

LE 407



LE 415





°C ✱ Measuring point for ambient temperature

—··— Free space for air circulation

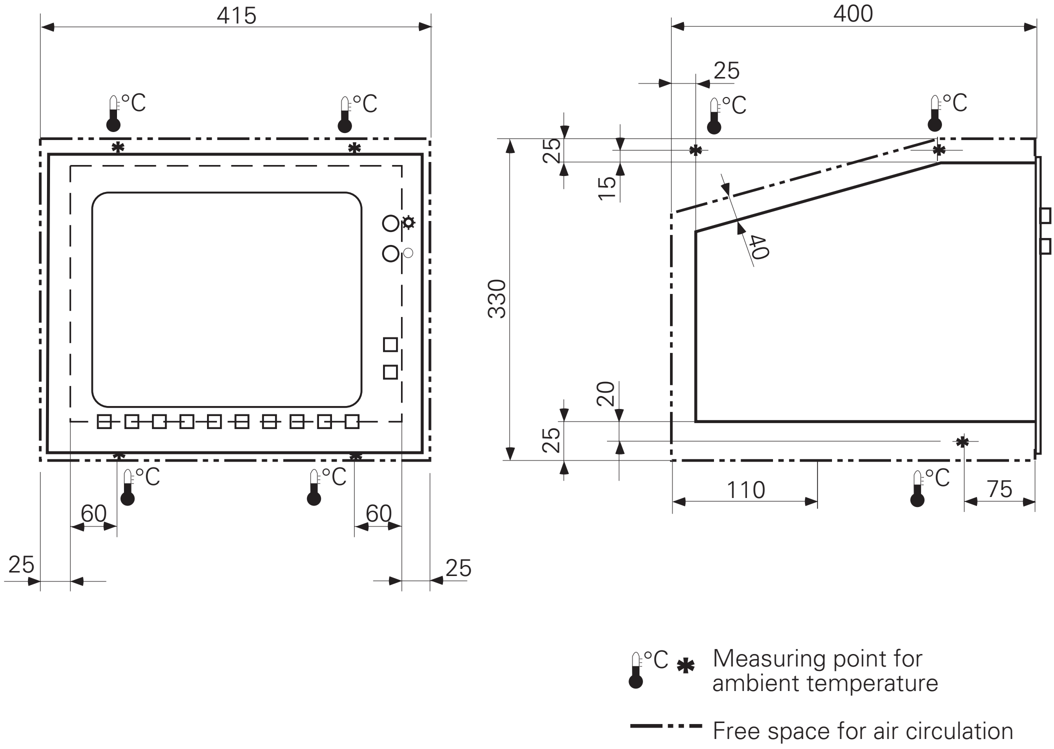
—·— Free space for servicing

Illustration of max. swivel range. The minimum angle of swivel for exchange of subassembly should be at least 90°.

2.5.2 Visual display unit (VDU)

When mounting the VDU it must be remembered that this unit is very sensitive to magnetic pick-up. The picture position and geometry can be disturbed by stray magnetic fields. Alternating fields cause periodic movement or distortion of the picture.

For this reason, keep a minimum distance of 0.5 m between the VDU casing and the source of any disturbance (e.g. permanent magnets, motors, transformers etc.)

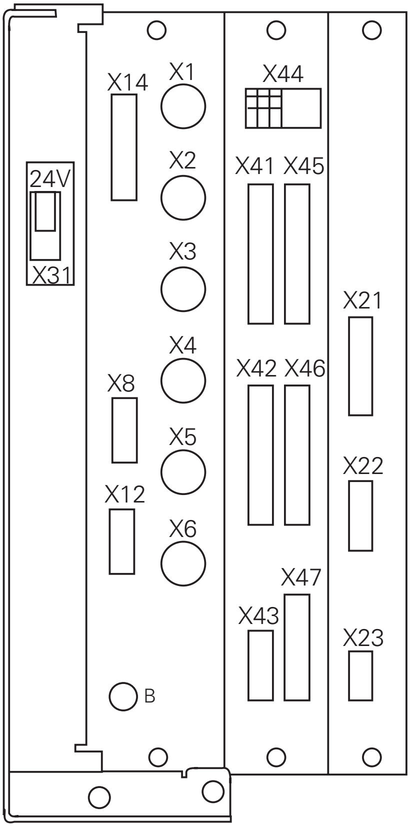


2.6 Degree of protection

When mounted, the visual display unit and the keyboard unit provide class IP54 protection against dust and splashwater.

3 Summary of connections

3.1 TNC 415



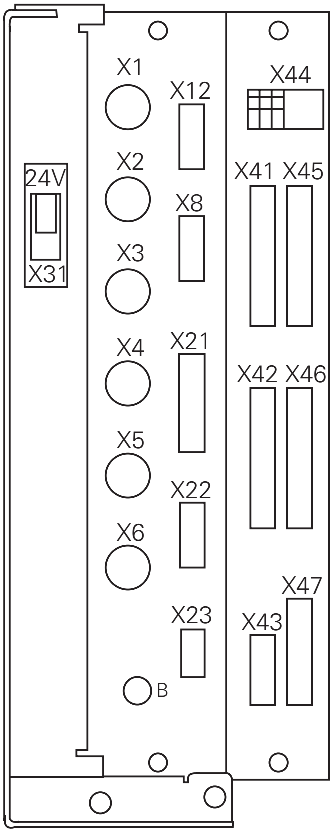
Control-loop board colour code
X1 = Measuring system 1 (~)
X2 = Measuring system 2 (~)
X3 = Measuring system 3 (~)
X4 = Measuring system 4 (~)
X5 = Measuring system 5 (~)
X6 = Measuring system S (⊓⊔)
X8 = Nominal value - output
1, 2, 3, 4, 5, S
X10 = Reference pulse inhibit
X12 = Touch trigger probe
X14 = Measuring touch probe
B = Signal ground

PLC and graphics board
X41 = PLC output
X42 = PLC input
X43 = VDU-screen (BC)
X44 = Power supply 24 V for PLC
X45 = TNC-keyboard (TE)
X46 = Machine control panel
X47 = PLC I/O board (PL)

Processor board
X21 = Data interface RS-232-C/V.24
X22 = Data interface RS-422/V.11
X23 = Electronic handwheel

X31 = Power supply 24 V DC for NC

3.2 TNC 407



Processor board
X1 = Measuring system 1 (~)
X2 = Measuring system 2 (~)
X3 = Measuring system 3 (~)
X4 = Measuring system 4 (~)
X5 = Measuring system 5 (⊓⊔)
X6 = Measuring system S (⊓⊔)
X12 = Triggering touch probe
X8 = Nominal value - output
1, 2, 3, 4, 5, S
X21 = Data interface RS-232-C/V.24
X22 = Data interface RS-422/V.11
X23 = Electronic handwheel
B = Signal ground

PLC and graphics board
X41 = PLC output
X42 = PLC input
X43 = VDU (BC)
X44 = Power supply 24 V for PLC
X45 = TNC keyboard (TE)
X46 = Machine control panel
X47 = PLC I/O board (PL)

X31 = Power supply 24 V DC for NC

4 Power supply

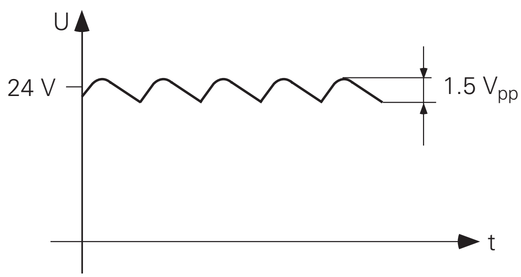
4.1 Logic unit and PLC I/O-board

		Supply voltage	Voltage range DC average	Max. current consumption	Power consumption
LE	NC section	24 Vdc [2]	Lower limit 20.4 V ⎓	LE 415: 1.5 A LE 407: 1.3 A	LE 415: ≈ 36 W LE 407: ≈ 24 W
	PLC section	24 Vdc [3]	Lower limit 31 V ⎓ [1]	2 A when half of the inputs/outputs are switched on simultaneously	≈ 48 W when half of the inputs/outputs are switched on simultaneously
PL 410 B	Digital inputs/out puts			20 A when half of the inputs/outputs are switched on simultaneously	≈ 480 W when half of the inputs/outputs are switched on simultaneously
	Analog inputs	24 Vdc [2]		≈ 100 mA	≈ 2.4 W
PA 110		24 Vdc [3]		≈ 100 mA	≈ 2.4 W

[1] Voltage surges up to V ⎓ for t < 100 ms are permissible.
[2] VDE 0160, 5.88 low-voltage electrical separation
[3] VDE 0160, Base insulation

4.1.1 NC power supply

The NC section of the LE must not be supplied from the machine control voltage supply! It requires an individual, external and separately generated supply voltage according to VDE 0160, 5.88 recommendations for low-voltage separation.



Use 24 V DC with a permissible AC component of 1.5 V_{PP} (recommended filter capacitor 10 000 µF/40 V DC).

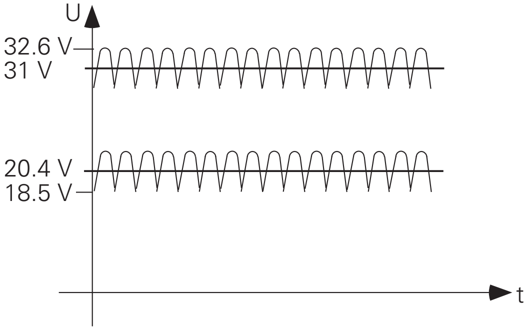
X31 power supply for NC

Connection terminals

Pin Number	Assignment
1	+ 24 V DC
2	0 V

4.1.2 PLC power supply

The PLC section (PLC inputs and outputs) of the LE and PL is run from the 24 V machine control voltage supply, generated according to VDE 0160 (base insulation). Superimposed AC components, such as those caused by a three-phase bridge rectifier without smoothing, are permissible up to a ripple factor of 5 % (see DIN 40110/10.75, Section 1.2).



The 0 V-line of the PLC-power supply must be grounded with an earth lead (∅ ≥ 6 mm²) to the main frame ground of the machine. The earth lead at the frame of the PL 410 must be directly connected to protective earth with an earth lead (∅ ≥ 6mm²). To prevent ground loops, the measured voltage at the analog inputs must not be grounded

X44 power supply for the PLC

Connection terminals

Pin Number	Assignment
1	+ 24 V DC switched off by EMERGENCY STOP
2	+ 24 V DC not switched off by EMERGENCY STOP
3	0 V

Power supply for the PL 400

Terminal	Assignment
X13	+24 V DC switched off by EMERGENCY STOP
X12	0 V
X3 Pin 12	+ 24 V DC not switched off by EMERGENCY STOP

Power supply for the PL 410

Terminal	Assignment	PL #1	or	PL #2
X9	0V			
X10	+24 V logic supply and for "Control ready"			
X11	+24 V supply for outputs	O32 - O39	or	O64 - O71
X12	+24 V supply for outputs	O40 - O47	or	O72 - O79
X13	+24 V supply for outputs	O48 - O55	or	O80 - O87
X14	+24 V supply for outputs	O56 - O62	or	O88 - O94

The routing and connection of the thermistors and analog inputs must be shockproof to VDE 0160 (Section 5.5.1). If this cannot be guaranteed, then both the PLC and the PL 410 must be supplied with voltage in accordance with VDE 0160, 5.88 recommendations for low-voltage electrical separation.

Power supply for the PL 410 B
Connections as on the PL 410, plus power supply at X23 in accordance with VDE 0160, 5.88 low-voltage electrical separation for analog inputs (the NC power supply can be used).

X23 Power supply for analog inputs
Connection terminals

Pin Number	Assignment
1	+ 24 V DC
2	0 V

Power supply for PA 110

Pin Number	Assignment
1	+24 V DC Not switched off by EMERGENCY STOP
2	0 V

The power for the PA 110 can be supplied according to VDE 0550, provided that the analog inputs and connections for the Pt 100 are shockproof according to VDE 0160 (Section 5.5.1). If this is not possible, the entire PLC power supply and the power for the PA 110 must be provided according to VDE 0551.

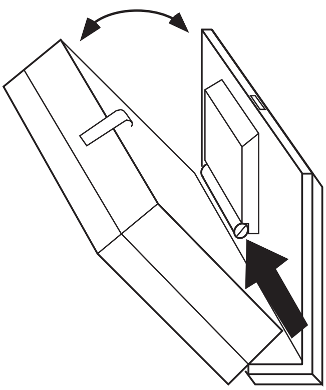
4.1.3 Buffer battery

The buffer battery is the potential source for the RAM memory for NC-programs, PLC-programs and machine parameters when the control is switched off.
If the

"EXCHANGE BUFFER BATTERY"

message appears, the batteries must be exchanged.

The 3 batteries may be found behind a screw cap in the power supply section of the logic unit.
As well as the batteries, the logic unit contains an additional energy store, mounted on the processor board, for buffering the memory contents. This means that the mains can be switched off when replacing the batteries. The energy store will ensure that the memory is retained while the batteries are exchanged.



Type of batteries:
Three AA-size batteries, leak-proof,
IEC-Designation "LR6"

4.2 Visual display unit (VDU)

X3 = Mains supply connection

Mains supply voltage	110 V	220 V
Supply voltage range	85 to 132 V	170 to 264 V
Fuse rating	F 3.15 A	F 3.15 A
Frequency range	50 to 60 Hz	
Power consumption	70 W	

Connection	Assignment
L1	Live (BK)
N	Neutral (BL)
⏚	Protective earth (GN/YL)

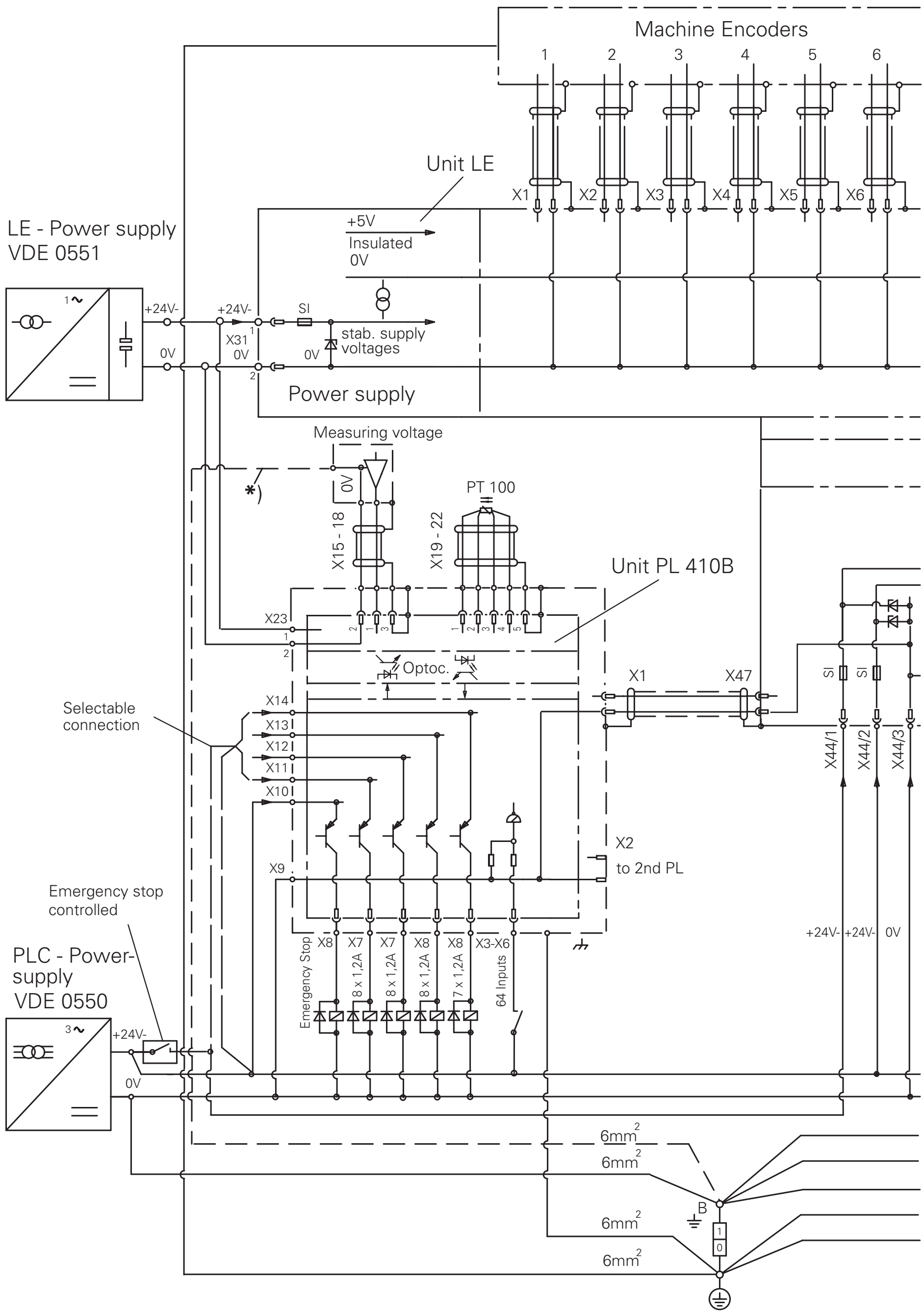
X4 = DC connections (only for BC 110, Id.-Nr. 254 740 01)

Pin Number	Assignment
1	+ 24 V
2	0 V

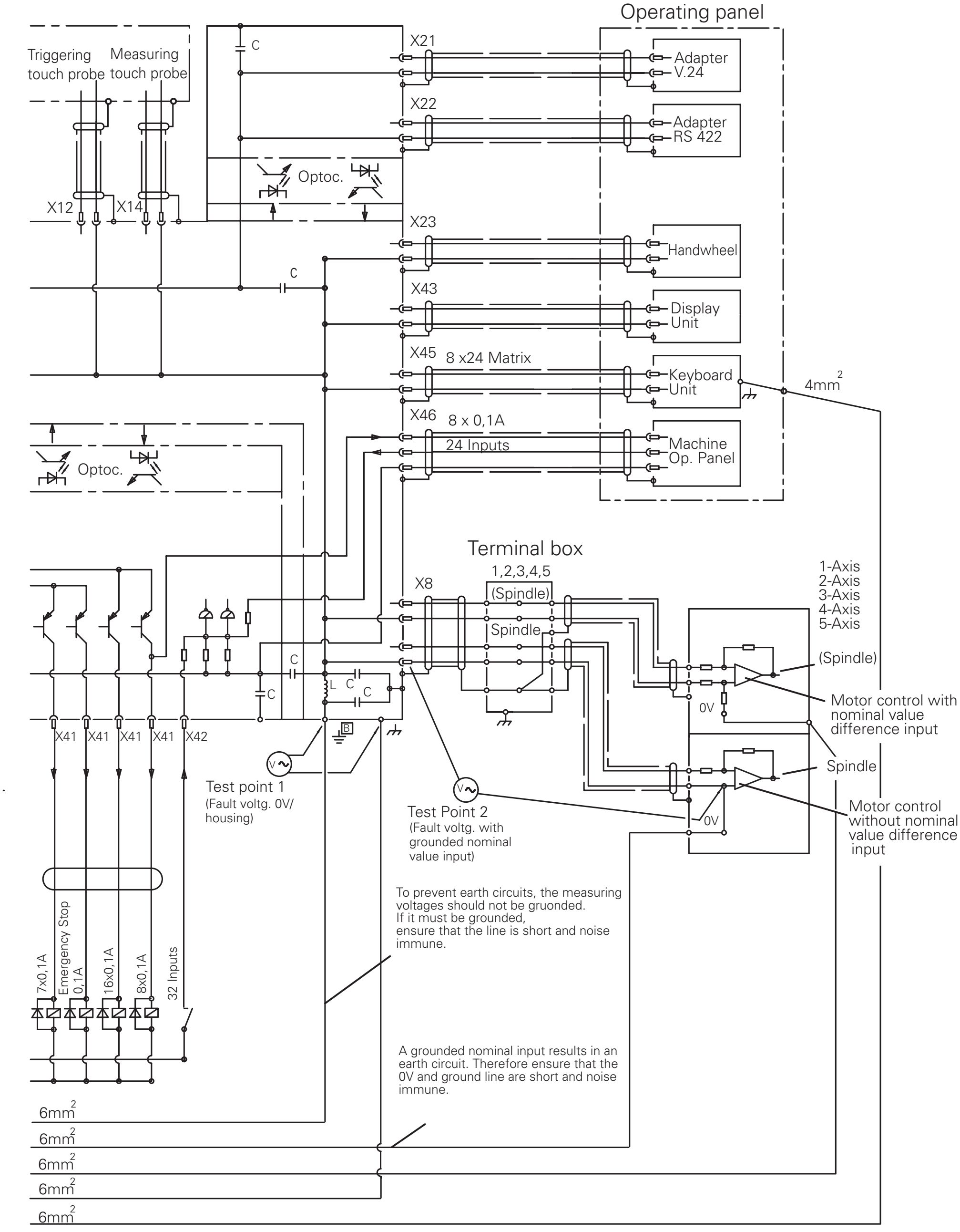
Power supply for integral fan:
The power supply for the fan must be connected separately to the BC 110 (Id.-Nr. 254 740 01).
The connection to the +24 V machine control voltage must be according to VDE 0550.
Permissible voltage range +18 to +28 V; power consumption 5 W at +24 V DC.

The power supply for the fan is taken internally from X3 in the BC 110 B (Id.-Nr. 260 520 01). There X4 is a DC output for test purposes (please do not connect!).

4.3 Earthing plan



*) In order to avoid an earth circuit, the measuring voltage should not be grounded. If it must be grounded, ensure that the ground line is short and noise immune.


Operating panel
Triggering touch probe
Measuring touch probe
C
X21
Adapter V.24
X22
Adapter RS 422
Optoc.
X12
X14
X23
Handwheel
C
X43
Display Unit
X45
8 x24 Matrix
Keyboard Unit
4mm²
X46
8 x 0,1A
24 Inputs
Machine Op. Panel
Optoc.
Terminal box
1,2,3,4,5
(Spindle)
Spindle
X8
1-Axis
2-Axis
3-Axis
4-Axis
5-Axis
(Spindle)
C
L
C
C
C
0V
Motor control with nominal value difference input
X41
X41
X41
X41
X42
Spindle
Test point 1
(Fault voltg. 0V/ housing)
Test Point 2
(Fault voltg. with grounded nominal value input)
0V
Motor control without nominal value difference input
To prevent earth circuits, the measuring voltages should not be gruonded.
If it must be grounded,
ensure that the line is short and noise immune.
7x0,1A
Emergency Stop 0,1A
16x0,1A
8x0,1A
32 Inputs
A grounded nominal input results in an earth circuit. Therefore ensure that the 0V and ground line are short and noise immune.
6mm²
6mm²
6mm²
6mm²
6mm²

5 Measuring systems

The HEIDENHAIN-contouring controls are designed for the installation of incremental linear and angular measuring systems.

The control controls the actual position with a measuring step of 0.0001 mm or 0.0001°. Measuring systems and encoders with a graduation period of 0.001 mm or 0,001° to 0.1 mm or 0.1° may be used.

It does **not** matter whether the measuring system or encoder has one or several reference marks. However, HEIDENHAIN recommends the use of measuring systems with distance-coded reference marks, since the traversing distance when referencing is thereby reduced to a minimum. See Chapter "Machine Integration".

The current requirement per measuring system input must not exceed 300 mA. The maximum current requirement for all measuring system inputs together is limited to 1.2 A.

5.1 Linear measuring systems

Measurement of length is best performed by a linear measuring system or encoder. Insofar as it is compatible with the accuracy requirements, linear measurement can also be made using a rotary encoder on the ballscrew.

HEIDENHAIN recommends use of the following linear measuring systems:

LS 103 C, LS 106 C, LS 405 C, LS 406 C, LS 706 C, LB 326, ULS 300 C.

For linear measurement with the aid of a rotary encoder and a ballscrew you could use, for example, an ROD 450.

5.2 Angular measuring systems

For direct angular measurement in the A, B or C axes the following incremental angular measuring systems are available: ROD 250 C, ROD 700 C, RON 255 C, and RON 705 C.

In order to meet accuracy requirements, HEIDENHAIN recommends line counts of at least 18 000.

5.3 Measuring system inputs for sinusoidal signals (7 to 16 µApp)

The LE 415 can have five measuring systems and the LE 407 four measuring systems connected with sinusoidal inputs (7 to 16 μA_{pp}). Maximum input frequency is 50 kHz.

5.3.1 Connector assignments

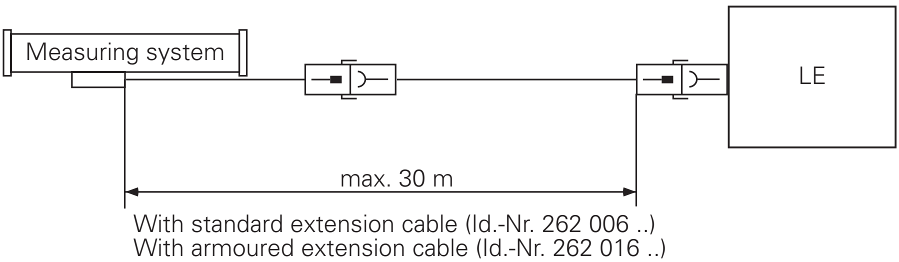
LE 407: X1, X2, X3, X4 measuring system 1, 2, 3, 4
LE 415: X1, X2, X3, X4, X5 measuring system 1, 2, 3, 4, 5

Flange socket with female connector insert (9-pin)

Pin Number	Assignment
1	I_1+
2	I_1-
5	I_2+
6	I_2-
7	I_0+
8	I_0-
3	+ 5 V
4	0 V
9	Inner screen
Housing	Outer screen = housing

5.3.2 Connecting cable

Please use only HEIDENHAIN measuring system cables, connectors and couplings. Standard HEIDENHAIN extension cables enable a maximum distance of 60 m (200 ft) to be covered.



5.4 Measuring system inputs for square-wave signals

One measuring system with square-wave signals can be connected to the LE 415. Two such systems can be connected to the LE 407. Maximum input frequency is 300 kHz.

5.4.1 Connector assignments

LE 407: X5, X6 measuring system 5, S
LE 415: X6 measuring system S

Flange socket with female connector insert (12-pin)

Pin Number	Assignment
5	U_{a1}
6	$\overline{U_{a1}}$
8	U_{a2}
1	$\overline{U_{a2}}$
3	U_{a0}
4	$\overline{U_{a0}}$
7	$\overline{U_{aS}}$
2	+ 5 V (U_P)
12	+ 5 V (U_P)
11	0 V (U_N)
10	0 V (U_N)
9 (contact spring)	screen = housing

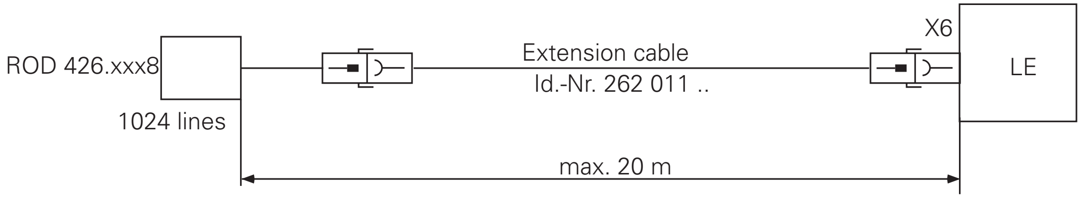
5.4.2 Connecting cable

Please use only HEIDENHAIN-measuring system cables, connectors and couplings.

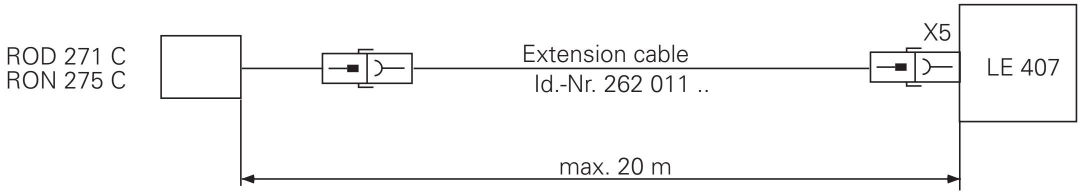
In order to be able to connect a measuring system to the square-wave signal input of the logic unit, the sinusoidal signal from the measuring system must be converted to a square-wave signal. This conversion is performed by the interpolation and digitizing electronics (EXE). The interpolation and digitizing electronics is either integrated into the measuring system or made as an independent unit.

If the interpolation and digitizing electronics does not have its own power supply, it can be supplied from the logic unit. In order to ensure a correct supply voltage, the total length of the connecting cable between the interpolation and digitizing electronics and the logic unit must be limited (see the following diagram).

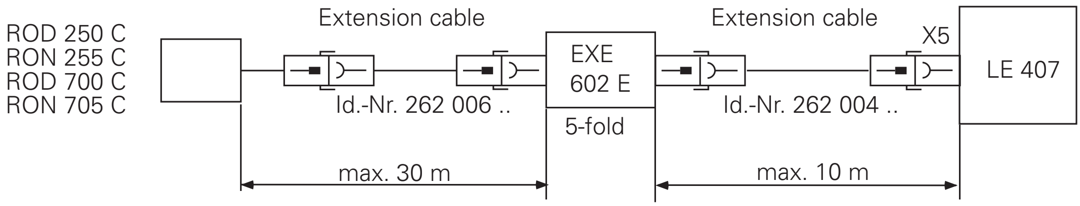
Spindle orientation:



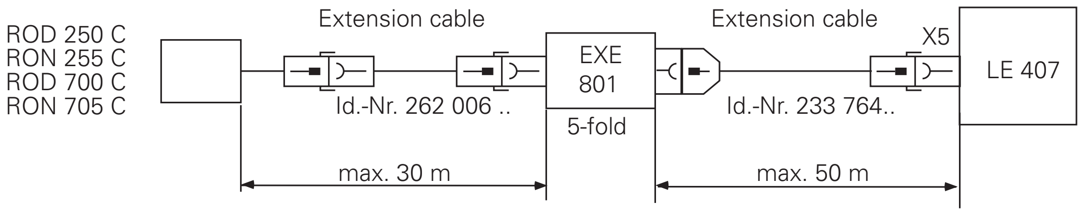
Angle encoders:



or:



or:

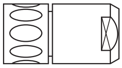
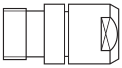
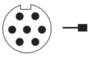
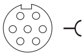
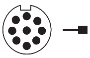
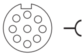
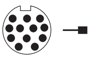
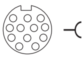


If necessary, linear measuring systems can also be connected to the X5 connector on the LE 407 via interpolation and digitizing electronics.

5.5 Measuring system connections

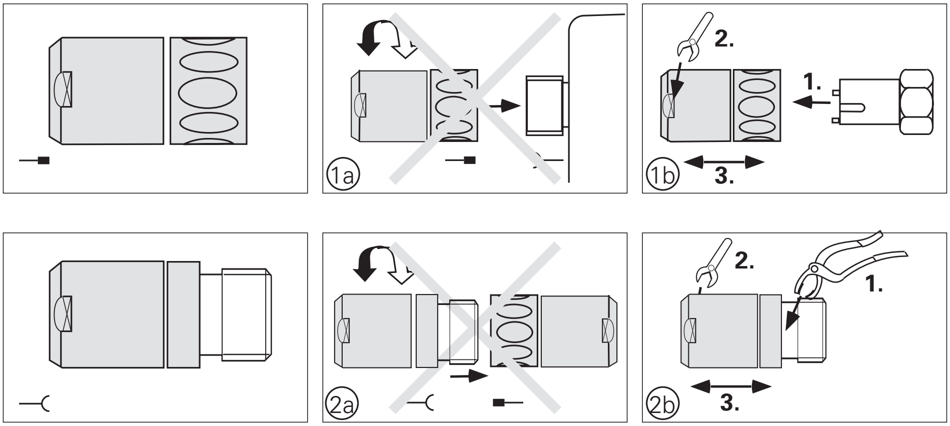
Please observe the directions in the assembly instructions for the particular measuring system which is being employed.

Measuring system cables must be laid without any intermediate clamping. Please use only the HEIDENHAIN-connectors and couplings for making connections.

Type		**Connector**	**Coupling**
Pin number	Cable Ø	 for PUR cable	 for PUR cable
7-pole 	6 mm 8 mm	237 524 20 237 524 24	— —
	6 mm 8 mm	— 237 524 21	237 525 11 —
9-pole 	6 mm 8 mm	237 524 03 237 524 02	— —
	6 mm 8 mm	237 524 15 237 524 10	237 525 07 237 525 04
12-pole 	6 mm 8 mm	237 524 06 237 524 07	237 525 01 237 525 03
	6 mm 8 mm	237 524 14 237 524 12	237 525 09 237 525 06

Assembly of the connector 237 524 ..

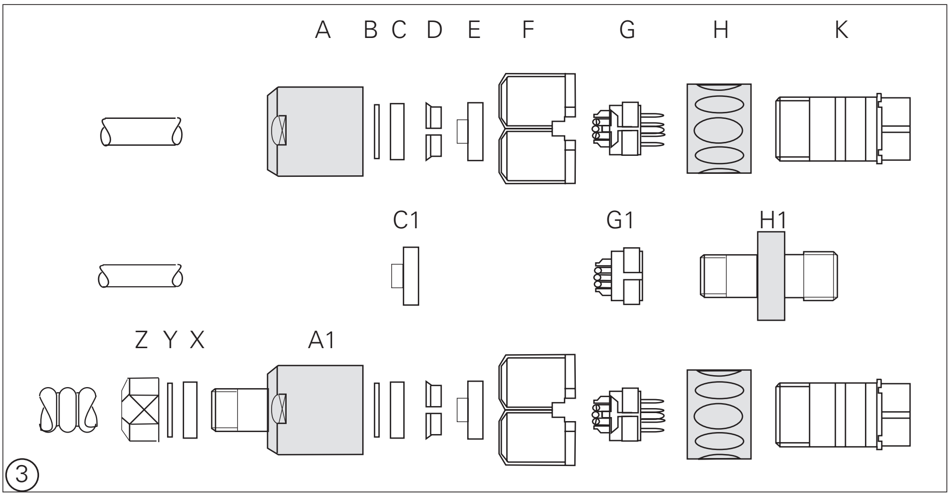
Assembly of the coupling 237 525 ..



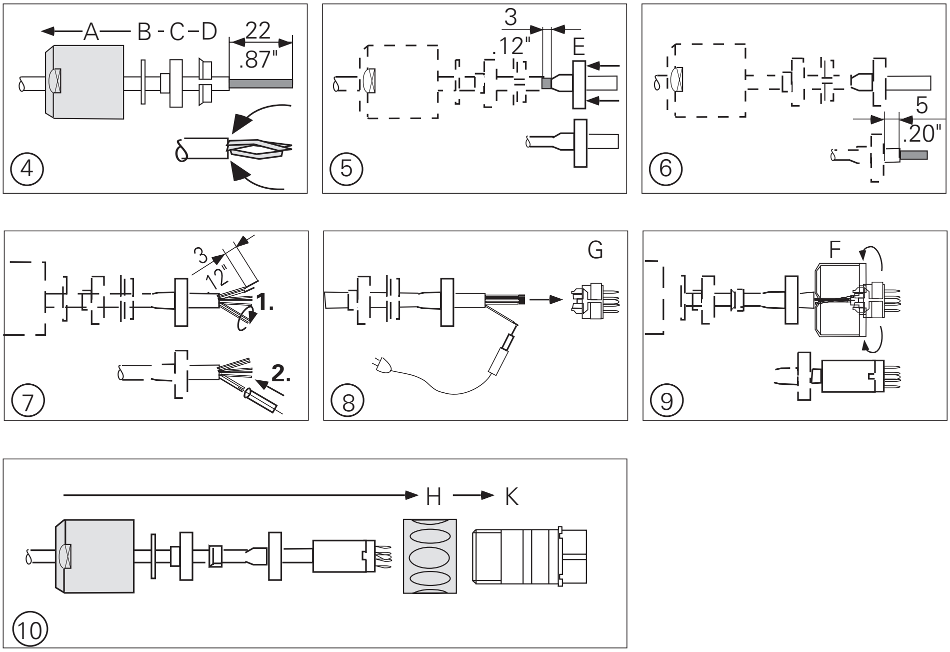
1a + 2a Do not open connector or coupling with a mating connector!

1b The special assembly tool Id.-Nr. 236 148 01 and a 22 mm spanner are absolutely necessary to assemble the connector.

2b An adjustable pipe-wrench with plastic jaws is required to assemble the coupling.



3 The diagram shows the various component parts of the connector and the coupling, and the two different versions of the screw connections for the armoured version PG7 and PG9. The screw connection PG9 with the Id.-Nr. 209 629 01, consisting of the parts X_1, Y_1, Z_1 , must be ordered separately.



4 Push parts A – D on to the cable, alternatively assemble the screw connection for the armouring according to diagram 3. Strip back 22 mm of the outside sleeving. Unpick the outer screen and fold back.

5 Cut off the outer screen to a length of 3 mm and slide the screen contact sleeve E between the internal sleeving and the braided screen.

6 Cut back the internal sleeving to a length of 5 mm.

7 1. twist the inner screen together.
2. insulate the twisted inner screen with heat-shrinkable sleeving.

8 Strip off the insulation on all leads for 3 mm, tin and solder in accordance with the connection diagram to G or G1.

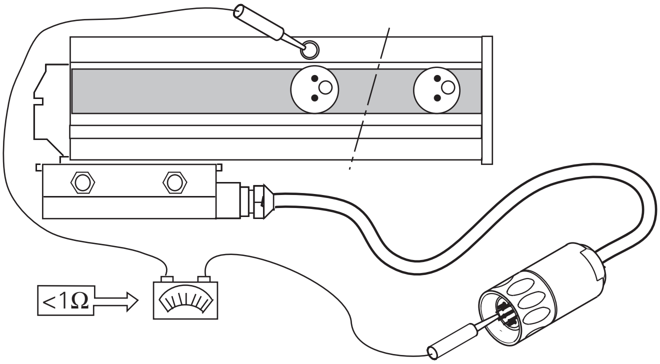
9 Assemble part F.

10 Push the connector together.

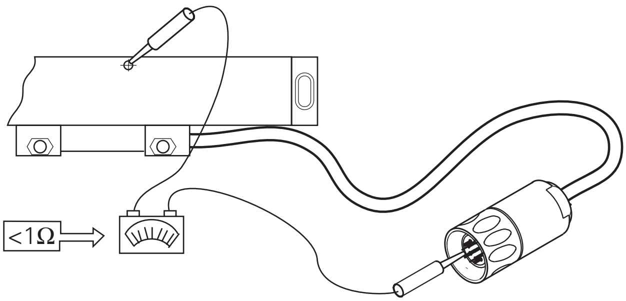
The following points must be observed when assembling the measuring system:

.The inner screen (pin 9) must not make any electrical connection with the outer screen (connector housing).
.The outer screen of the measuring system cable must have an electrical connection with the connector housing.
.The measuring system is grounded through its mechanical fixings, the mounting block in the case of encapsulated systems, and the housing of the scale.
.When using external pulse-forming electronics (EXE) the ground must be electrically connected with the frame of the machine. Necessary cable cross-section ≥ ∅ 6 mm².
.Encapsulated linear measurement systems should be connected to compressed air.

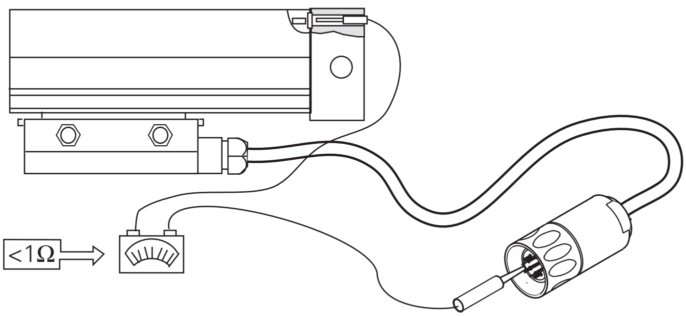
Please check that the mounting block, the sensor unit and the scale housing all have a good electrical connection with the chassis/frame of the machine. Since the connector for the measuring system and the mounting block are connected by the outer screen of the connector cable, this test can be carried out between the measuring system connector and the measuring system housing. During this test the connector must not be plugged in to the logic unit, as this would cause a grounding via the logic unit.



LS 107 C



LS 403 C, LS406 C



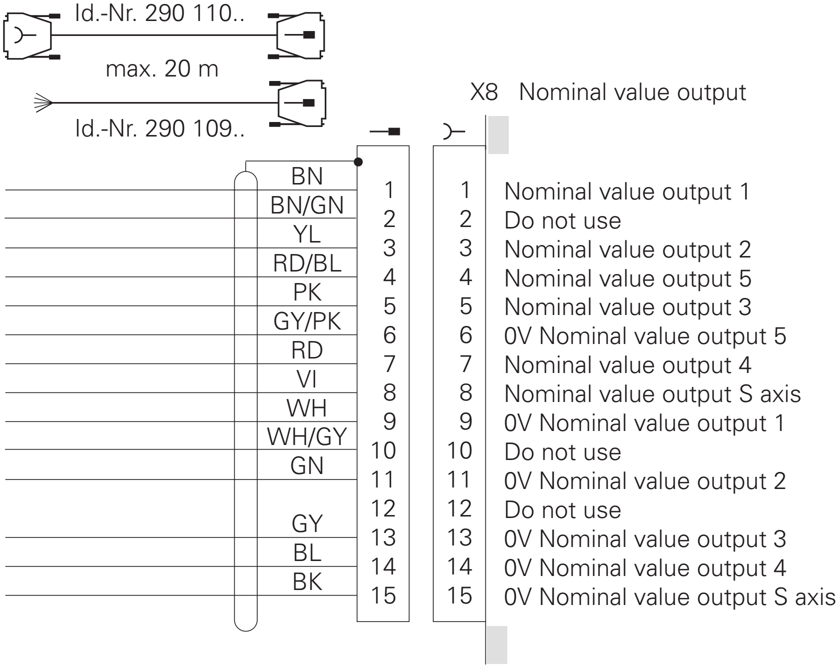
LS 704 C

6 Nominal value output

HEIDENHAIN contouring controls control the position loop servo with a nominal value potential of ±10 volts.

Maximum loading of the nominal value outputs: 2 mA
Maximum load capacitance: 2 nF

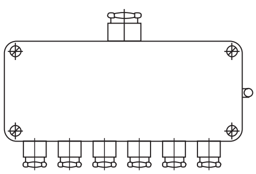
6.1 Connector assignment



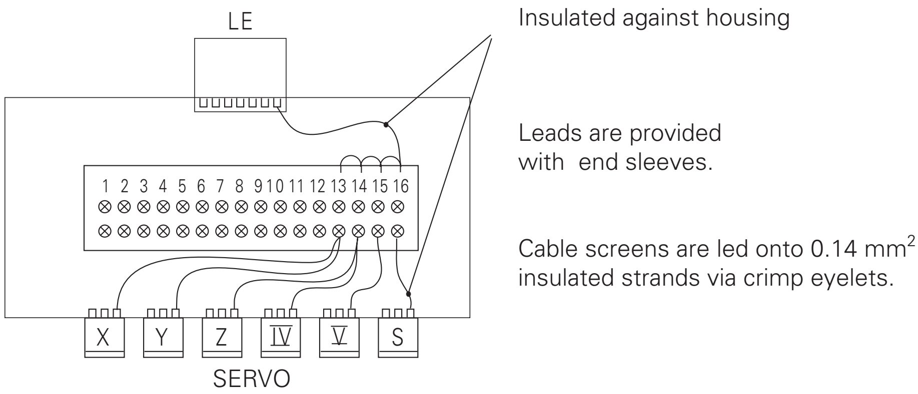
6.2 Connecting cable

HEIDENHAIN offers a connecting cable with a connector at one end (Id.-Nr. 290 109 ..).

The connecting cable to the nominal value outputs may not have more than one intermediate terminal. The terminal must be made in an earthed connection box. This is necessary when the cable must branch to physically separate servo inputs. It is only possible to earth the screening of the servo leads in this way. If required, suitable connection boxes are available from HEIDENHAIN with the Id.-Nr. 251 249 01.



The casing of the connection box must be electrically connected with the frame of the machine.
The 0 V of the nominal value differential input must be joined to signal ground, (cable cross-section ≥ Ø 6 mm², see also under "Earthing plan").
Suggested solution for connecting and wiring the screening in the connection box:



Pin number	Assignment	
1	Nominal value output	X axis
2	Nominal value output 0 V	X axis
3	Nominal value output	Y axis
4	Nominal value output 0 V	Y axis
5	Nominal value output	Z axis
6	Nominal value output 0 V	Z axis
7	Nominal value output	IV axis
8	Nominal value output 0 V	IV axis
9	Nominal value output	V axis
10	Nominal value output 0 V	V axis
11	Nominal value output	S axis
12	Nominal value output 0 V	S axis
13	Screen connection	
14	Screen connection	
15	Screen connection	
16	Screen connection	

HEIDENHAIN recommends that the logic unit and the connection box be connected by HEIDENHAIN-cable Id.-Nr. 290 109 ..

If the manufacturers want to use their own cable, HEIDENHAIN offers a 15 pin Sub-D connector with solderable leads (Id.-Nr. 243 971 ZY).

7 Reference pulse inhibit input

The reference pulse inhibit input can be used to suppress the evaluation of the reference pulse input of each of the six measuring systems. This is achieved by simply applying a positive potential (13 V to +30.3 V) to the appropriate reference pulse inhibit input. Pin 9 of the female connector X10 (reference pulse inhibit) must be connected to 0 V of the PLC-power supply.

The reference pulse inhibit inputs are only available in the LE 415 A. These inputs are normally unused. See also Chapter "Machine Integration".

7.1 Connector assignment

X10 Reference pulse inhibit (LE 415 A only)

Flange socket with female connector insert (9-pin)

Pin number	Assignment
1	Screen
2	Reference pulse inhibit input X1
3	Reference pulse inhibit input X2
4	Reference pulse inhibit input X3
5	Reference pulse inhibit input X4
6	Reference pulse inhibit input X5
7	Reference pulse inhibit input X6
8	Do not use
9	0 V (PLC)

7.2 Connecting cable

Standard commercially available screened cables can be used for the connections to the reference pulse inhibit inputs.

HEIDENHAIN can deliver a 9-pin D-subminiature connector (Id.-Nr. 244 503 ZY) for this purpose.

8 Touch probe system input

The following 3D touch probe systems are available from HEIDENHAIN:

- TS 120 with cable transmission and integrated APE (interface electronics)
- TS 511 with infrared transmission of the trigger signal and connectable via APE (interface electronics)
- TT 110 for workpiece measurement

- The TNC 415 and TNC 425 can also support the measuring touch probe system TM 110

For start-up and adjustment of the 3D-touch probe systems see Chapter "Machine Integration".

8.1 Connector assignment

X12 Touch probe system

D-subminiature female connector (15-pin)

Pin number	Signal designation
1	Inner screen (0 V)
3	Ready/standby
4	Start
5	+15 V ± 10 % (U_P)
6	+ 5 V ± 5 % (U_P)
7	$\overline{\text{Battery warning}}$
8	0 V (U_N)
9	Trigger signal
10	$\overline{\text{Trigger signal}}$[2]
2, 11 to 15	Do not use

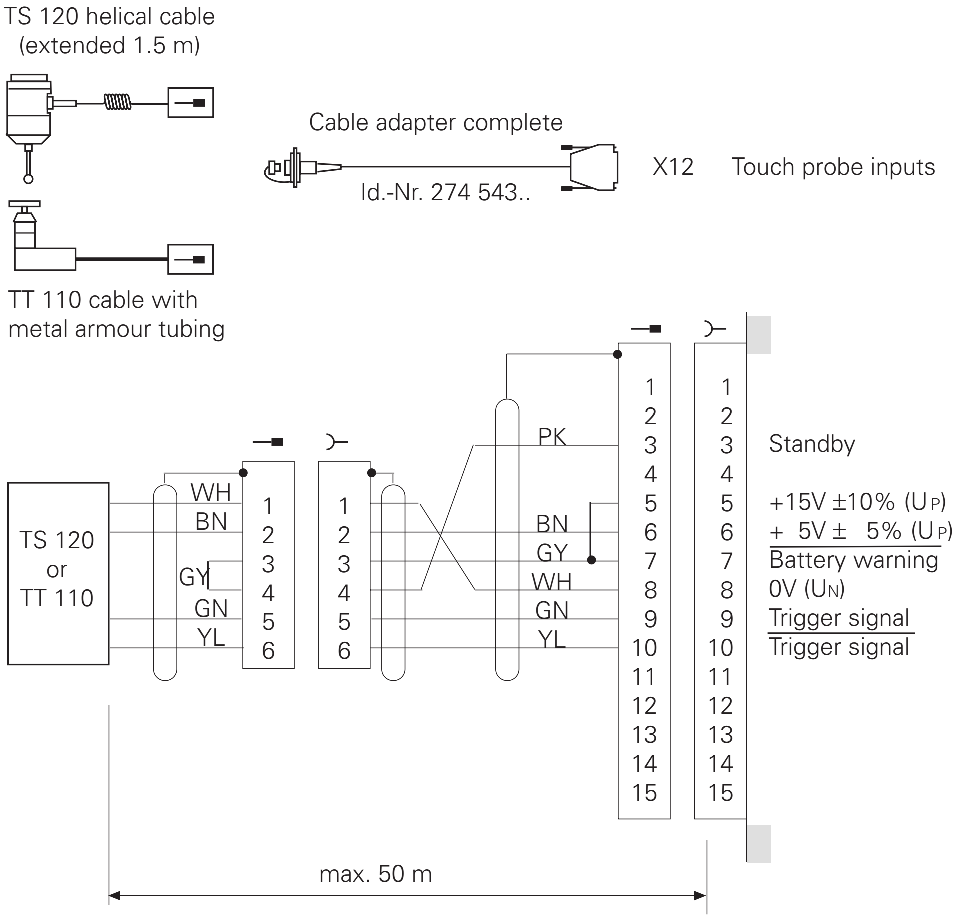
2 Stylus in rest position = signal high

8.2 Connection of the touch probe system

Please use only HEIDENHAIN connecting cables and adapters for the connection to the touch probe system.

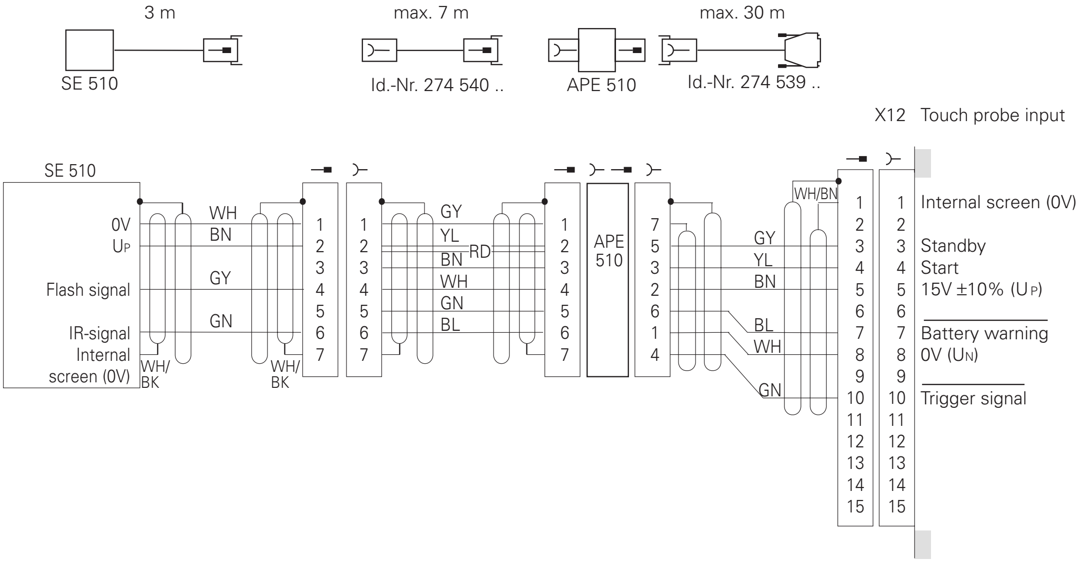
8.2.1 TS 120 or TT 110

The touch probe system TS 120 is connected directly to the logic unit via a cable adapter.
See also under the heading "Mounting dimensions".



8.2.2 TS 511

The TS 511 touch probe system can only function together with a transmitter/receiver unit (SE 510) and interface electronics (APE 510).

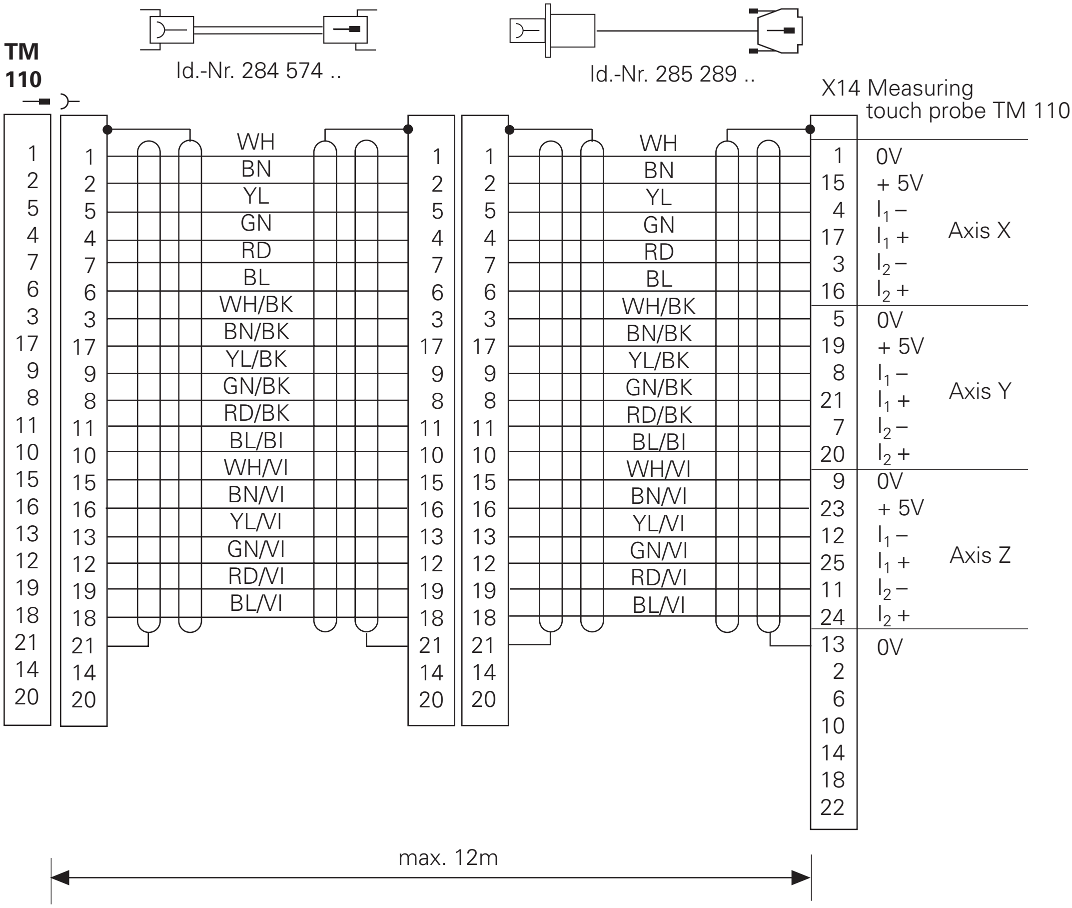


The signals may be inverted by changing the switch positions S1 to S4 in the APE 510. See operating instructions TS 511.

Please install the transmitter/receiver unit SE 510 either insulated from, or electrically connected to the machine, as it must take up a definite potential, also under vibration. The earthing screw of the APE 510 must be joined to the machine signal ground by a potential compensating lead (≥ ∅ 6 mm²). See also under the heading "Earthing plan".

8.2.3 TM 110

The TM 110 measuring touch probe can be mounted to the TNC 415 B and TNC 425. A special software module (optional) is required for digitizing with TM 110.



9 Data interface

The HEIDENHAIN contouring controls TNC 407 and TNC 415 have two data interfaces:

one RS-232-C/V.24 data interface and
one RS-422/V.11 data interface.

Both interfaces may be utilised. The operator decides which of the two interfaces he wishes to use. See also chapter "Data interface".

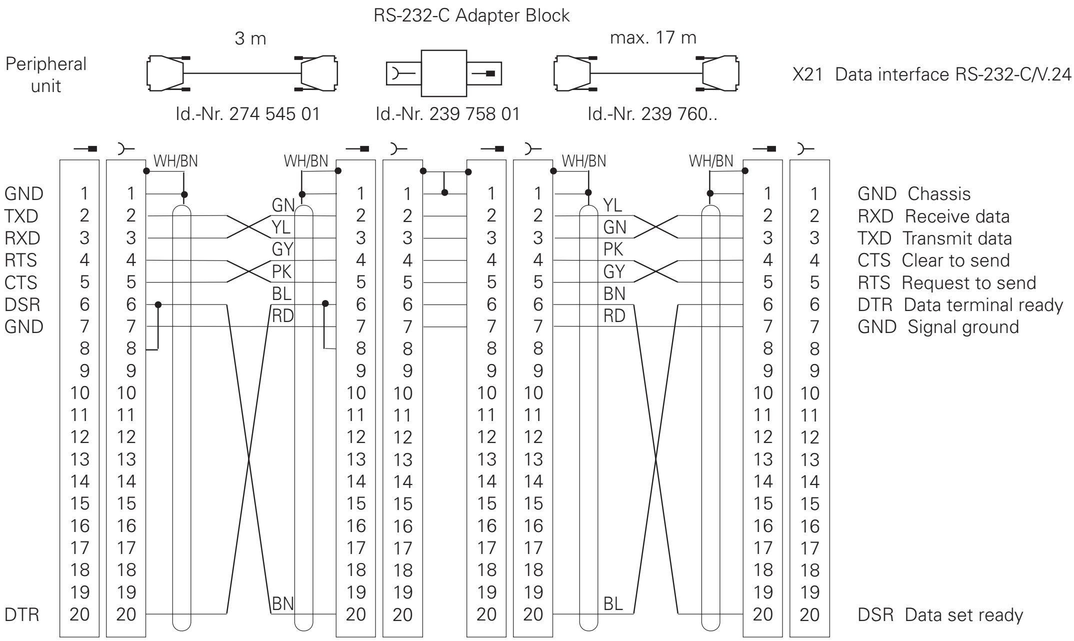
9.1 RS-232-C/V.24 data interface

HEIDENHAIN guarantees that, if properly connected, the serial data interface RS-232-C/V.24 will transmit data correctly up to a distance of 20 m between the logic unit and the peripheral unit.

The connection to the peripheral unit is made via a cable adapter which is attached to either the operating console or the control cabinet. See also under the heading "Mounting dimensions".

This cable adapter (Id.-Nr. 239 758 01) is connected to the logic unit with the HEIDENHAIN cable Id.-Nr. 239 760 ..

For connection to the peripheral unit HEIDENHAIN offers a standard connecting cable (Id.-Nr. 274 545 01) , length 3 m.



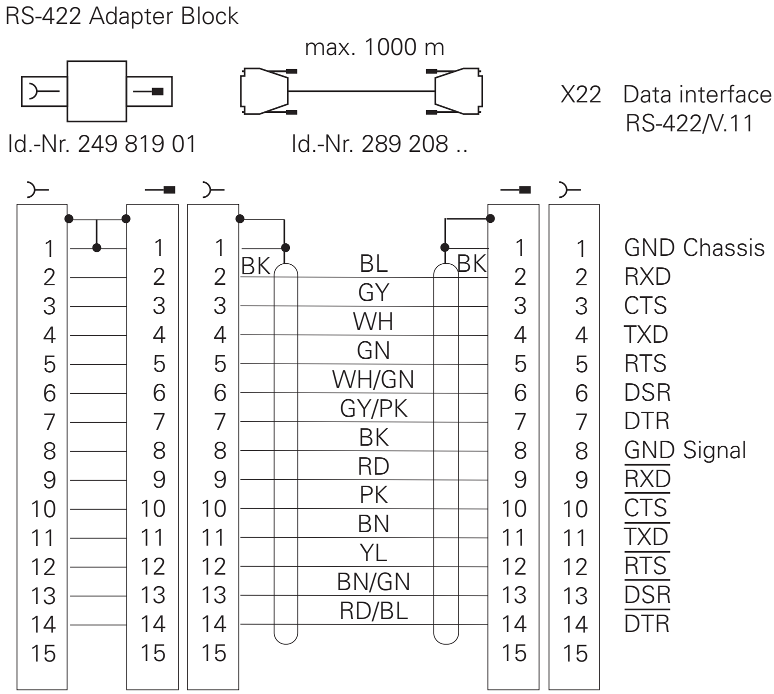
If your peripheral unit has a connector layout that differs from the above, the HEIDENHAIN connecting cable cannot be used.

9.2 RS-422/V.11 data interface

If used correctly, the RS-422/V.11 serial data interface will ensure error-free data transmission up to a distance of 1000 m between logic unit and peripheral unit.

The connection to the peripheral unit is made via a cable adapter which is attached to either the operating console or the control cabinet. See also under the heading "Mounting dimensions".

The cable adapter is connected to the logic unit with a HEIDENHAIN connecting cable.



The pin assignment is the same at both ends of the connecting cable.
In the RS-422 adapter the pins of the female connectors are connected one-to-one.
The pin assignment in the cable adapter is therefore the same as in the X22 connector in the logic unit.

The following cable type must be used for the connection to the peripheral unit:
LIYCY 7 x 2 x 0.14 Cu

For the cable connection HEIDENHAIN offers a 15-pin D-subminiature connector (Id.-Nr. 243 971 ZY).

10 Handwheel input

The following handwheels can be connected to the HEIDENHAIN contouring controls.

1 integral handwheel HR 130, or
3 integral handwheels HR 150 using handwheel adapter HRA 110, or
1 portable handwheel HR 330, or
1 portable handwheel HR 332

10.1 Pin assignment

X23 Handwheel input
D-subminiature female connector (9-pin)

Pin number	Assignment
2	0 V
4	+ 12 V ± 0.6 V (Uv)
6	DTR
7	TxD
8	RxD
9	DSR
1, 3, 5	Do not use

10.2 Portable handwheel HR 330

The HR 330 is a portable handwheel with keys for the 5 axes, rapid traverse, direction of traverse and EMERGENCY STOP.

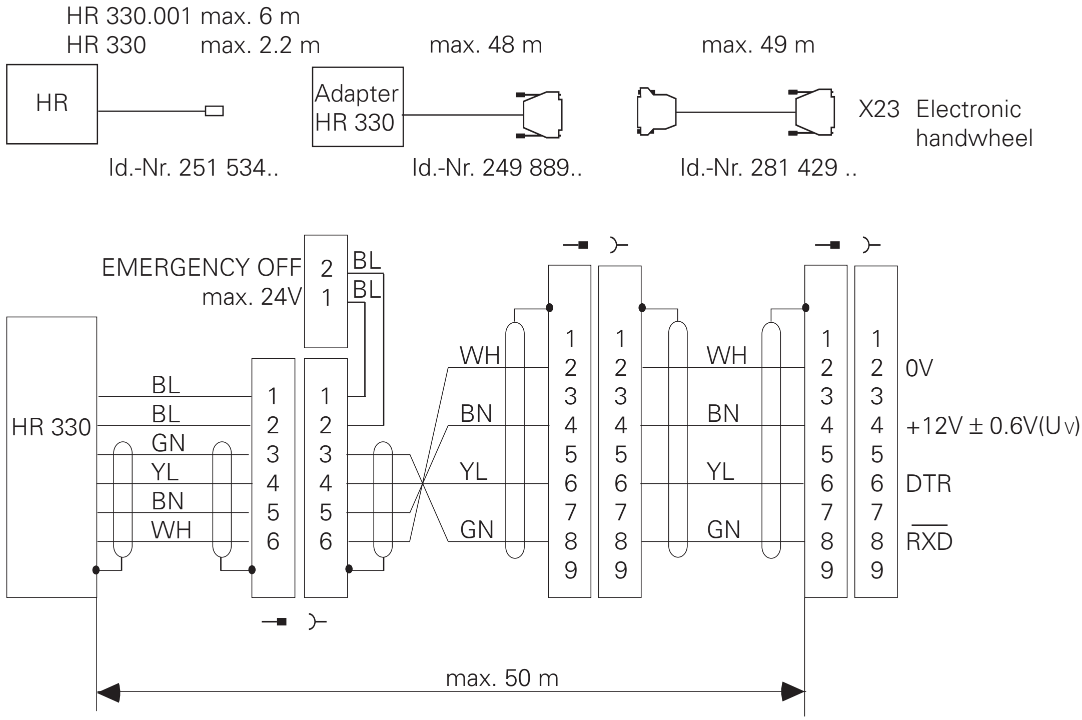
The HR 330 is connected to the logic unit by means of the cable adapter Id.-Nr. 249 889 ..
See also under the heading "Mounting dimensions".

The HEIDENHAIN extension cable Id.-Nr. 281 429 .. may be used to increase the connection distance.

The HR 330 is available in 2 versions:

. HR 330	helical cable (stretched length 2.2 m)	Id.-Nr. 251 534 11
. HR 330.001	normal cable (max. 6 m)	Id.-Nr. 251 534 12

Dummy plug for EMERGENCY STOP circuit Id.-Nr. 271 958 02



The adapter includes a cable with a 9 pin connector from the connection to the logic unit and two terminals for connecting the 24 V of the EMERGENCY STOP control circuit (max. load 1.2 A).
See also the Section "Mounting dimensions".

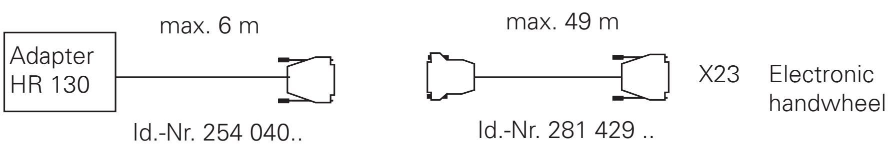
10.3 Integral handwheel HR 130

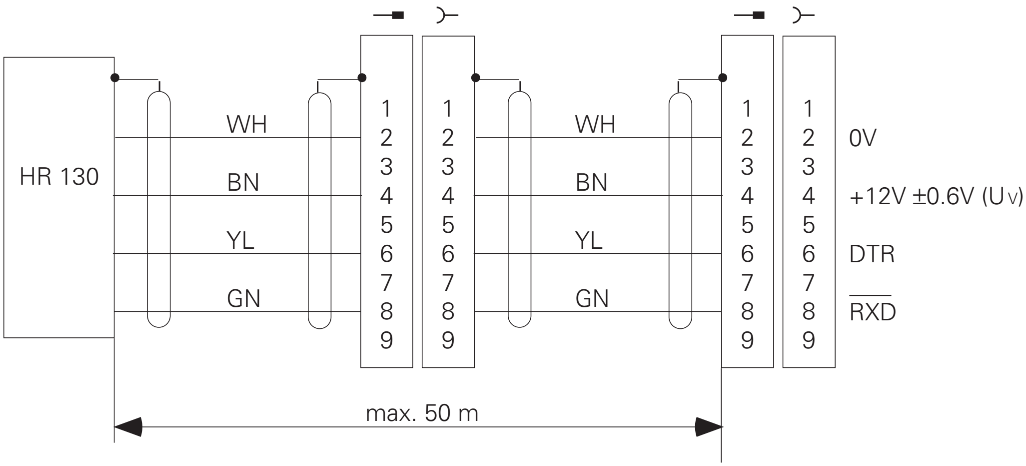
The HR 130 is the integral version of the HR 330, without the keys for the axes, rapid traverse etc.

It may be attached directly to the logic unit, or via an extension cable (Id.-Nr. 249 814 ..).

The HR 130 (Id.-Nr. 254 040 ..) is available in several versions (standard cable length 1 m):
- small knob; axial cable exit: version 01
- small knob; radial cable exit: version 02
- large knob; axial cable exit: version 03
- large knob; radial cable exit: version 04
- ergonomic knob; radial cable exit: version 05

You will find dimension drawings for the knobs at the end of this chapter.





10.4 Portable handwheels with permissive buttons

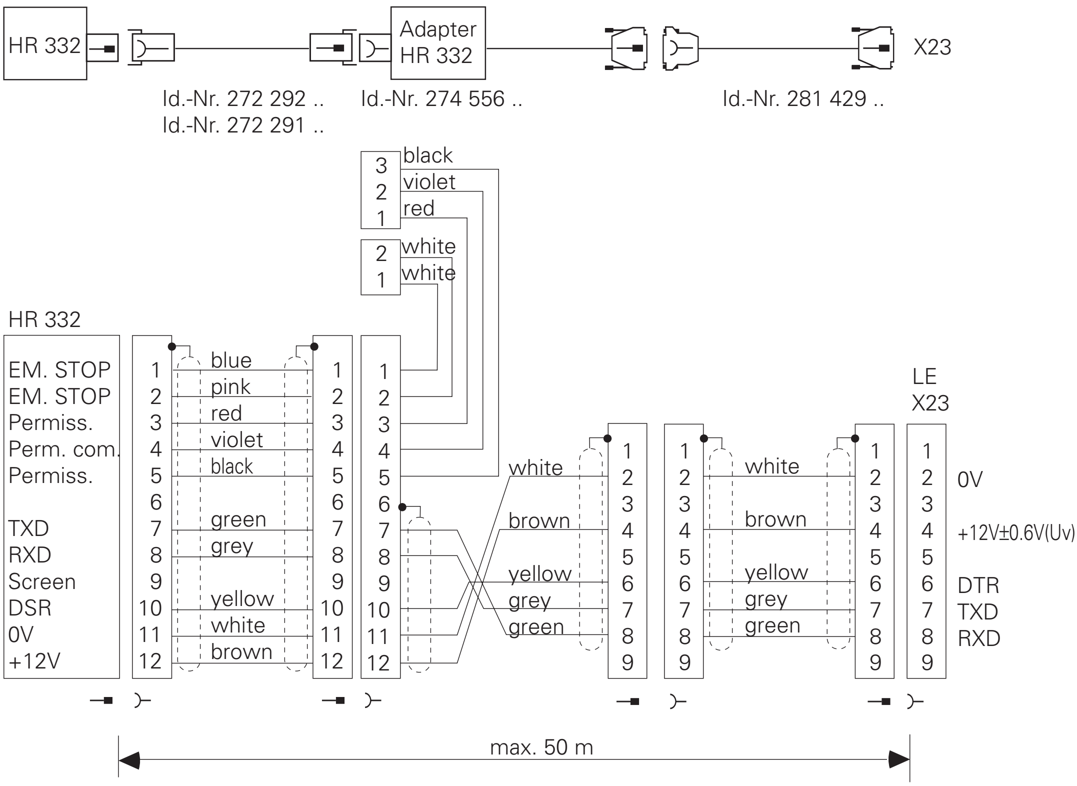
HR 332

Handwheel HR 332 has two enable buttons and a keypad that can be evaluated by the PLC (see "Handwheel" section in the chapter entitled "Machine Integration"). There are different models of the handwheel with different key labelling - please contact HEIDENHAIN for further details.

HR 332 is connected to a cable adapter on the panel by means of a 5 m plug-in connecting cable. The connecting cable is available with or without metal armouring.

Maximum cable length between HR and LE is 50 m.

Connecting cable to HR 332	Id.-Nr. 272 291 ..
Connecting cable to HR 332, with metal armouring	Id.-Nr. 272 292 ..
Cable adapter HR 332/LE	Id.-Nr. 274 556 ..
Extension cable for cable adapter	Id.-Nr. 281 429 ..
Dummy plug for EMERGENCY STOP circuit	Id.-Nr. 271 958 01
Handwheel holder	Id.-Nr. 268 268 03
HR 332	Id.-Nr. 266 064 21



The adapter has a cable with a 9-pin plug for connection to the logic unit, two terminals for the 24 V of the EMERGENCY STOP circuit of the control (max. load 1.2 A) and 3 terminals for the enable circuit. The enable buttons are N/O contacts (24 V/1.2 A).

HR 410

The HR 410 is a portable electronic handwheel with:

- Five axis-selection keys
- Traverse direction keys
- Three keys with predefined traverse speeds (slow, medium, fast)
- Actual-position-capture key
- Three keys for machine functions to be determined by the machine tool builder
- Two permissive buttons
- EMERGENCY STOP button
- Holding magnets

Dummy plug for EMERGENCY STOP circuit Id. Nr. 271 958 03

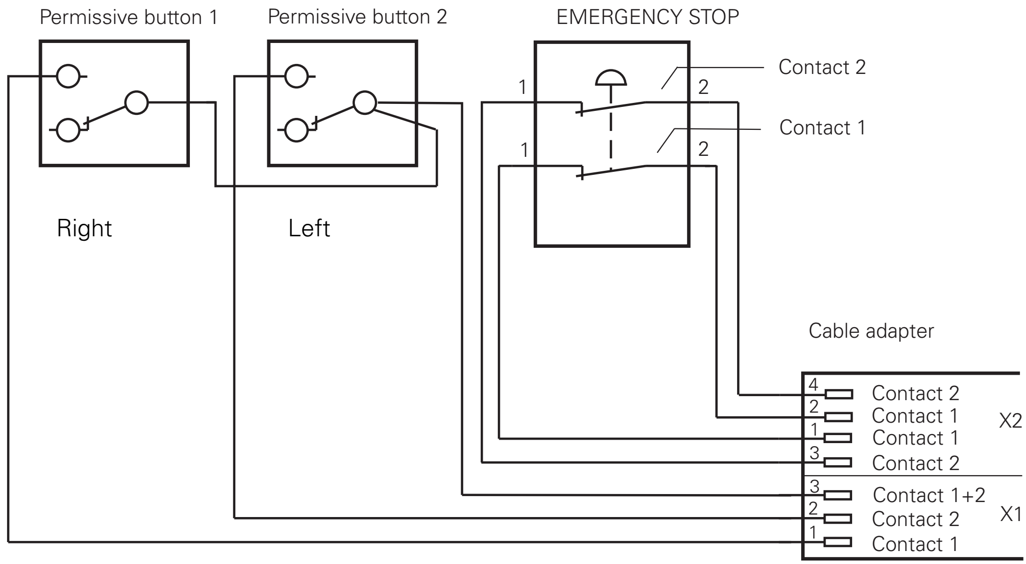
In order to be able to mount the HR 410, you will need a TNC with at least software version 280 540 03, 280 560 03 0r 280 580 03.

The adapter includes plug-in terminal strips for the contacts of the EMERGENCY STOP button and permissive button (maximum load 1.2 A).

Extension Cable Id. Nr. 281 429..			Adapter Cable Id. Nr. 296 466..			Connecting Cable Id. Nr. 296 467 05			HR 410 Id. Nr. 296 469 01	
D-sub connector (male) 9-pin		**D-sub connector (female) 9-pin**	**D-sub connector (male) 9-pin**		**Coupling on mounting base (female) 18-pin**	**Connector (male) 18-pin**		**Connector (female) 18-pin**	**Connector (male) 18-pin**	
Housing	Shield	Housing	Housing	Shield	Housing	Housing	Shield	Housing	Housing	Shield
2	White	2	2	White	E	E	White	E	E	
4	Brown	4	4	Brown	D	D	Brown	D	D	
6	Yellow	6	6	Yellow	B	B	Yellow	B	B	
7	Gray	7	7	Gray	A	A	Gray	A	A	
8	Green	8	8	Green	C	C	Green	C	C	
					6	6	WH/BK	6	6	
					7	7	YL/BK	7	7	
					5	5	WH/RD	5	5	
					4	4	WH/BL	4	4	
					2	2	WH/GN	2	2	
					3	3	WH/YL	3	3	
					1	1	WH/BN	1	1	

WH/BN	3	Contact 1 + 2	
WH/YL	2	Contact 2 (left)	Permissive button
WH/GN	1	Contact 1 (right)	
WH/BL	1	Contact 1	
WH/RD	2	Contact 1	EMERGENCY STOP
YL/BK	3	Contact 2	
WH/BK	4	Contact 2	

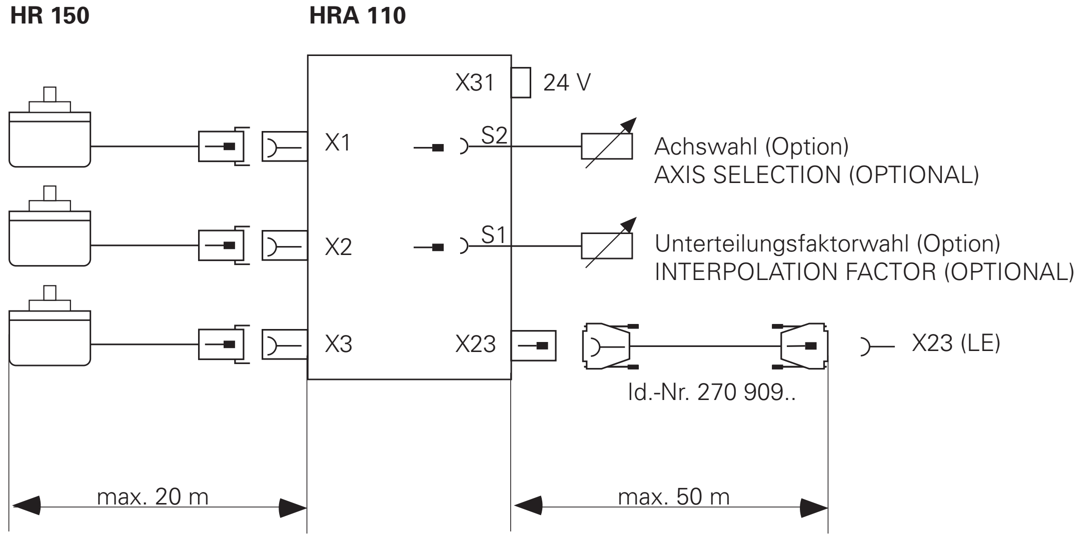
Internal wiring of the permissive button and EMERGENCY STOP contacts of the HR 410:



10.5 Handwheel adapter HRA 110

The HRA 110 can be used to connect 2 or 3 handwheels HR 150 to one LE.

The first two handwheels are dedicated to axes X and Y. The third handwheel can be assigned to axes X, Y, Z, IV or V using a step switch (optional) or machine parameters. (See "Handwheel" section in chapter "Machine integration").



A second step switch (optional) can be used to select the interpolation factor for the handwheels. The interpolation factor of the step switch must be evaluated in the PLC - it is displayed on screen but cannot be altered with the keyboard.
However the interpolation factor can be set for specific axes without a step switch by using the keyboard as before.

Pin assignments

Handwheel inputs X1, X2, X3

Pin number	Assignment
1	I_1+
2	I_1–
3	I_2+
4	I_2–
5	I_0+
6	I_0–
7	+ 5 V
8	0 V
9	Inner screen
Housing	Outer screen

Handwheel adapter output X23
Pin assignment

Pin number	Assignment
1	RTS
2	0 V
3	CTS
4	+12V + 0,6 V (Uv)
5	Do not use
6	DSR
7	RxD
8	TxD
9	DTR
Housing	Outer screen

NC supply connection X31
Pin assignment

Pin number	Assignment
1	+ 24 V
2	0 V

The 24 V power supply of the PLC must not be connected in the handwheel adapter, since that would bridge the electrical separation of PLC inputs and outputs. The handwheel adapter must be powered with the 24 V supply from the NC block of the LE (VDE 0551). See chapter "Power supply".

11 PLC inputs/outputs

The following configurations of PLC inputs/outputs are possible with the HEIDENHAIN contouring controls.

PLC Inputs	PLC Outputs	Analogue Inputs	Thermistors	Components
56	31	–	–	LE
119	62	–	–	LE + 1 PL 400
182	93	–	–	LE + 2 PL 400
119	62	4	4	LE + 1 PL 400 + PA 110
120	62	–	–	LE + 1 PL 410 / PL 410 B
112	60	4	4	LE + 1 PL 410 / PL 410 B
184	93	–	–	LE + 2 PL 410 / PL 410 B
176	91	4	4	LE + 2 PL 410 / PL 410 B
168	89	8	8	LE + 2 PL 410 / PL 410 B
120	62	4	4	LE + 1 PL 410 + PA 110
112	60	8	8	LE + 1 PL 410 + PA 110
56	31	4	4	LE + PA 110

	PL 410 PL 410 B (Id.-Nr. 263 371 12)	PL 410 PL 410 B (Id.-Nr. 263 371 02)	PA 110	PL 400
PLC inputs	64	56	–	63
PLC outputs	31	29	–	31
Analogue inputs	–	4	4	–
Inputs for thermistors	–	4	4	–
"Control is operational" output	1	1	–	1

The analogue inputs of the PL 410 / PL 410 B they must be activated by a DIL switch on the PL and a machine parameter in the TNC. When the analogue inputs are active, two outputs (O61/O62 or O93/O94 on PL #2) and eight inputs (I120 to I127 or I248 to I255 on PL #2) of the PLC cannot be used.

One PLC extension can be mounted on the logic unit. The second PLC extension must be installed next to the logic unit in the switch cabinet. It is not possible to combine the PL 410 B with the PL 410 or PA 110.

11.1 Technical data

11.1.1 PLC Inputs

	Logic Unit / PL 410	PL 410 B	PL 400
Voltage ranges: "1" signal: Ui "0" signal: Ui	 13 V to 30.2 V –20 V to 3.2 V		 16.5 V to 30 V –20 V to 4 V
Current ranges: "1" signal: Ii "0" signal: Ii	 3.8 mA to 8.9 mA 1.0 mA at Ui=3.2 V	 2.5 mA to 6 mA 0,65 mA at Ui=3.2 V	 6.2 mA to 12.6 mA 1.6 mA at Ui=4 V

11.1.2 PLC Outputs

Transistor outputs with current limiter

	Logic Unit	PL 400 / PL 410 / PL 410 B
Min. output voltage for "1" signal	3 V under supply voltage	
Nominal operating current per output	0.1 A	1.2 A

Permitted load: resistive load; inductive load only with quenching diode parallel to inductance.

Not more than one output may be shorted on the logic unit at any one time.
One shorted output causes no overload.

No more than half of the PLC outputs may be driven at once (usage factor 0.5).

11.1.3 Analogue inputs

The PA 110, the PL 410 and the PL 410 B (Id.-Nr 263 371 02) have 4 analogue inputs (± 10 V) and 4 inputs for Pt 100 thermistors. These inputs must be activated by a DIL switch on the PL 410 / PL 410 B and a machine parameter in the TNC. When the analogue inputs are active, two outputs (O61/O62 or O93/O94 on PL #2) and eight inputs (I120 to I127 or I248 to I255 on PL #2) of the PLC cannot be used. (See chapter "Machine integration", section "Analogue inputs").

Voltage range	-10 V to +10 V
Input resistance	> 250 k Ω
Resolution	100 mV
Internal value range	-100 to +100

11.1.4 Inputs for Pt 100 Thermistors

See chapter "Machine integration", section "Analogue inputs".

Constant current	5 mA
Temperature range	0°C to 100°C
Resolution	0.5°C
Internal value range	0 to 200

11.2 Connector assignment

Power supply see section "PLC power supply"

11.2.1 PLC inputs

The PLC inputs I0 to I31 are on connector X42 (PLC input).
The PLC inputs I128 to I152 are on the connector for the machine control panel (X46).
See also the heading "Machine control panel".

X42 PLC input D-sub 37-pin female connection

Pin number	Assignment
1	I0
2	I1
3	I2
4	I3 Acknowledgment for "Control is operational" test
5	I4
6	I5
7	I6
8	I7
9	I8
10	I9
11	I10
12	I11
13	I12
14	I13
15	I14
16	I15
17	I16
18	I17

Pin number	Assignment
19	I18
20	I19
21	I20
22	I21
23	I22
24	I23
25	I24
26	I25
27	I26
28	I27
29	I28
30	I29
31	I30
32	I31
33, 34	Do not use
35, 36, 37	0 V (PLC) test output; Do not use
Housing	Outer screen

11.2.2 PLC output

The PLC outputs O0 to O30 and the "Control operational" output are on connector X41 (PLC output). The PLC outputs O0 to O7 are also to be found on the connector for the machine control panel (X46). See also under heading "Machine control panel".

X41 PLC output
D-sub 37-pin female connection

Pin number	Assignment	Pin number	Assignment
1	O0	19	O18
2	O1	20	O19
3	O2	21	O20
4	O3	22	O21
5	O4	23	O22
6	O5	24	O23
7	O6	25	O24[1)]
8	O7	26	O25[1)]
9	O8	27	O26[1)]
10	O9	28	O27[1)]
11	O10	29	O28[1)]
12	O11	30	O29[1)]
13	O12	31	O30[1)]
14	O13	32	Do not use
15	O14	33	0 V (PLC) test output; do not use
16	O15	34	Control is operational[1)]
17	O16	35, 36, 37	24 V (PLC) test output; do not use
18	O17	Housing	Outer screen

[1)] Not switched off by EMERGENCY STOP

11.2.3 PL connection

X47 PLC I/O Board
D-sub 37-pin male connection

Pin number	Assignment	Pin number	Assignment
1, 2, 3	0 V	13	Screen
4	Serial IN 2	14, 15, 16	+ 12 V (from PL)
5, 6, 17, 18	Do not use	19	Serial IN 1
7	$\overline{\text{RESET}}$	20	EMERGENCY STOP
8	$\overline{\text{WRITE EXTERN}}$	21	$\overline{\text{Serial OUT}}$
9	WRITE EXTERN	22	Serial OUT
10	$\overline{\text{A5}}$	23	$\overline{\text{A4}}$
11	$\overline{\text{A3}}$	24	$\overline{\text{A2}}$
12	$\overline{\text{A1}}$	25	$\overline{\text{A0}}$

11.3 Connector assignment on the PL

Power supply see section "PLC power supply"

11.3.1 Connection to LE or PL #1

X1 to PL 410 / PL 410 B
X10 to PL 400
D-sub 25-pin female connector

Pin number	Assignment
1, 2, 3	0 V
4	Serial IN 2
5, 6, 17, 18	Do not use
7	$\overline{\text{RESET}}$
8	$\overline{\text{WRITE EXTERN}}$
9	WRITE EXTERN
10	$\overline{\text{A5}}$
11	$\overline{\text{A3}}$
12	$\overline{\text{A1}}$

Pin number	Assignment
13	Screen
14, 15	+ 12 V (from PL)
16	PL identification
19	Serial IN 1
20	EMERGENCY STOP
21	$\overline{\text{Serial OUT}}$
22	Serial OUT
23	$\overline{\text{A4}}$
24	$\overline{\text{A2}}$
25	$\overline{\text{A0}}$
Housing	External shield

11.3.2 Connection of PL #2

X2 at the PL 410 / PL 410 B D-sub 25-pin male connector
X11 at the PL 400 D-sub 25-pin female connector

Pin number	Assignment
1, 2, 3	0 V
4-6, 14-18	Do not use
7	$\overline{\text{RESET}}$
8	$\overline{\text{WRITE EXTERN}}$
9	WRITE EXTERN
10	$\overline{\text{A5}}$
11	$\overline{\text{A3}}$
12	$\overline{\text{A1}}$

Pin number	Assignment
13	Screen
19	Serial IN 1
20	EMERGENCY STOP
21	$\overline{\text{Serial OUT}}$
22	Serial OUT
23	$\overline{\text{A4}}$
24	$\overline{\text{A2}}$
25	$\overline{\text{A0}}$

11.3.3 PLC inputs/outputs on the PL 400

The PLC inputs/outputs on the PL 400 are spread over nine connectors assigned as follows:

Cables with a core cross-section ≥ dia. 0.14 mm² Cu must be used to connect the PLC inputs and outputs. Maximum cable length 20 m.

X1

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	O32	O64
2	O33	O65
3	O34	O66
4	O35	O67
5	O36	O68
6	O37	O69
7	O38	O70
8	O39	O71
9	O40	O72
10	O41	O73
11	O42	O74
12	Do not use	

X2

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	O43	O75
2	O44	O76
3	O45	O77
4	O46	O78
5	O47	O79
6	O48	O80
7	O49	O81
8	O50	O82
9	O51	O83
10	O52	O84
11	O53	O85
12	Do not use	

X3

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	O54	O86
2	O55	O87
3	O56 2)	O88 2)
4	O57 2)	O89 2)
5	O58 2)	O90 2)
6	O59 2)	O91 2)
7	O60 2)	O92 2)
8	O61 2)	O93 2)
9	O62 2)	O94 2)
10	Control ready	
11	Do not use	
12	+24 V not interruptible by ext. EMERGENCY STOP 1)	

X4

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	I126	I254
2	I74	I202
3	I73	I201
4	I72	I200
5	I71	I199
6	I70	I198
7	I69	I197
8	I68	I196
9	I67	I195
10	I66	I194
11	I65	I193
12	I64	I192

1) +24V must always be connected even if the non-interruptible outputs are not used.
2) Outputs not interruptible by ext. EMERGENCY STOP

X5

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	I86	I214
2	I85	I213
3	I84	I212
4	I83	I211
5	I82	I210
6	I81	I209
7	I80	I208
8	I79	I207
9	I78	I206
10	I77	I205
11	I76	I204
12	I75	I203

X6

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	I98	I226
2	I97	I225
3	I96	I224
4	I95	I223
5	I94	I222
6	I93	I221
7	I92	I220
8	I91	I219
9	I90	I218
10	I89	I217
11	I88	I216
12	I87	I215

X7

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	I110	I238
2	I109	I237
3	I108	I236
4	I107	I235
5	I106	I234
6	I105	I233
7	I104	I232
8	I103	I231
9	I102	I230
10	I101	I229
11	I100	I228
12	I99	I227

X8

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	I122	I250
2	I121	I249
3	I120	I248
4	I119	I247
5	I118	I246
6	I117	I245
7	I116	I244
8	I115	I243
9	I114	I242
10	I113	I241
11	I112	I240
12	I111	I239

X9

Pin number	Assignment PL 400 #1	Assignment PL 400 #2
1	Do not use	
2	Do not use	
3	Do not use	
4	I125	I253
5	I124	I252
6	I123	I251

11.3.4 PLC inputs/outputs on the PL 410 / PL 410 B

X3

Pin number	Assignment PL 410 B #1 PL 410 #1	PL 410 B #2 PL 410 #2
1	I64	I192
2	I65	I193
3	I66	I194
4	I67	I195
5	I68	I196
6	I69	I197
7	I70	I198
8	I71	I199
9	I72	I200
10	I73	I201
11	I74	I202
12	I75	I203
13	I76	I204
14	I77	I205
15	I78	I206
16	I79	I207

X4

Pin number	Assignment PL 410 B #1 PL 410 #1	PL 410 B #2 PL 410 #2
1	I80	I208
2	I81	I209
3	I82	I210
4	I83	I211
5	I84	I212
6	I85	I213
7	I86	I214
8	I87	I215
9	I88	I216
10	I89	I217
11	I90	I218
12	I91	I219
13	I92	I220
14	I93	I221
15	I94	I222
16	I95	I223

X5

Pin number	Assignment PL 410 B #1 PL 410 #1	PL 410 B #2 PL 410 #2
1	I96	I224
2	I97	I225
3	I98	I226
4	I99	I227
5	I100	I228
6	I101	I229
7	I102	I230
8	I103	I231
9	I104	I232
10	I105	I233
11	I106	I234
12	I107	I235
13	I108	I236
14	I109	I237
15	I110	I238
16	I111	I239

X6

Pin number	Assignment PL 410 B #1 PL 410 #1	PL 410 B #2 PL 410 #2
1	I112	I240
2	I113	I241
3	I114	I242
4	I115	I243
5	I116	I244
6	I117	I245
7	I118	I246
8	I119	I247
9	I120[1)]	I248[1)]
10	I121[1)]	I249[1)]
11	I122[1)]	I250[1)]
12	I123[1)]	I251[1)]
13	I124[1)]	I252[1)]
14	I125[1)]	I253[1)]
15	I126[1)]	I254[1)]
16	I127[1)]	I255[1)]

[1)] These PLC inputs are not available when analogue inputs are active.

X7 Pin number	Assignment PL 410 B #1 PL 410 #1	PL 410 B #2 PL 410 #2
1	O32	O64
2	O33	O65
3	O34	O66
4	O35	O67
5	O36	O68
6	O37	O69
7	O38	O70
8	O39	O71
9	O40	O72
10	O41	O73
11	O42	O74
12	O43	O75
13	O44	O76
14	O45	O77
15	O46	O78
16	O47	O79

X8 Pin number	Assignment PL 410 B #1 PL 410 #1	PL 410 B #2 PL 410 #2
1	O48	O80
2	O49	O81
3	O50	O82
4	O51	O83
5	O52	O84
6	O53	O85
7	O54	O86
8	O55	O87
9	O56	O88
10	O57	O89
11	O58	O90
12	O59	O91
13	O60	O92
14	O61[1)]	O93[1)]
15	O62[1)]	O94[1)]
16	Control ready	

[1)] These PLC outputs are not available when analogue inputs are active.

11.3.5 Analogue inputs on the PL 410 / PL 410 B

X15,X16,X17,X18 analogue input

Pin number	Assignment
1	–10 V to +10 V
2	0 V (reference potential)
3	Screen

11.3.6 Inputs for thermistors on the PL 410 / PL 410 B

X19,X20,X21,X22 connection for Pt 100

Pin number	Assignment
1	I + constant current for Pt 100 (5 mA)
2	U + measuring input for Pt 100
3	U - measuring input for Pt 100
4	I - constant current for Pt 100
5	Screen

11.4 Connector assignment on the PA 110

Power supply see section "PLC power supply"

X2,X3,X4,X5 analogue inputs
Assignment as X15,X16,X17,X18 on PL 410

X7,X8,X9,X10 connection for Pt 100
Assignment as X19,X20,X21,X22 on PL 410

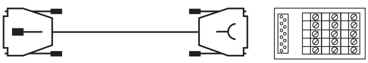
11.5 Connecting cable

Please use only HEIDENHAIN connecting cables.

11.5.1 Connection of PLC inputs/outputs on the LE

HEIDENHAIN recommends the installation of a transfer unit with a terminal strip in the switch cabinet.

max. 40 m



Id.-Nr. 263 954 ..

If a transfer unit is not required, use HEIDENHAIN connecting cable Id.-Nr. 244 005 ..

Assignment:

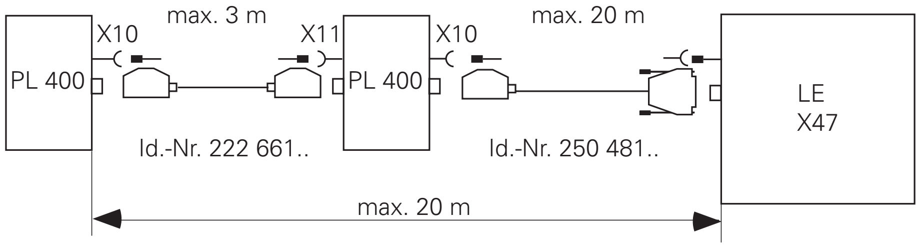
1	= green/red	14	= green/blue	27 =	yellow/black
2	= brown/black	15	= yellow	28 =	white/yellow
3	= white/black	16	= red	29 =	grey/blue
4	= green/black	17	= grey	30 =	pink/blue
5	= brown/red	18	= blue	31 =	pink/red
6	= white/red	19	= pink	32 =	brown/blue
7	= white/green	20	= white/grey	33 =	pink/green
8	= red/blue	21	= yellow/grey	34 =	brown
9	= yellow/red	22	= green/red	35 =	yellow/pink
10	= grey/pink	23	= white/pink	36 =	violet
11	= black	24	= grey/green	37 =	white
12	= pink/brown	25	= yellow/brown		
13	= yellow/blue	26	= grey/brown		

max. 25 m

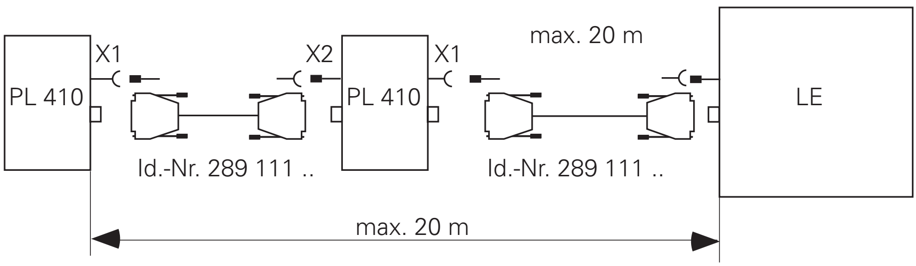
Id.-Nr. 244 005..

If the connector is to be mounted at the customer's facility, HEIDENHAIN can provide a 37-pin solderable connector (Id.-Nr. 243 937 ZY).

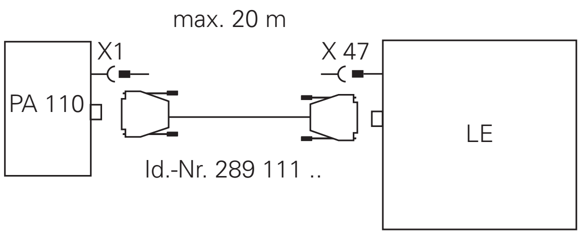
11.5.2 PL 400 connection

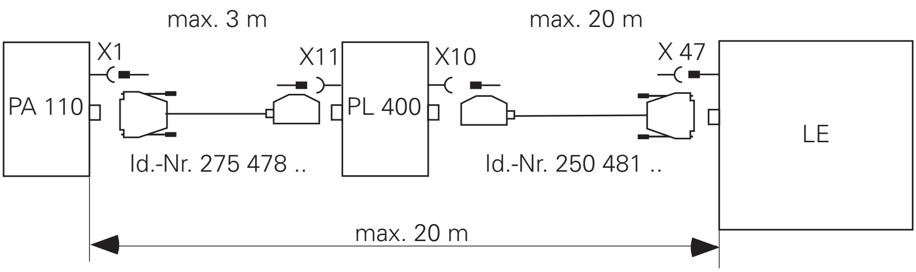


11.5.3 PL 410 / PL 410 B connection



11.5.4 PA 110 connection



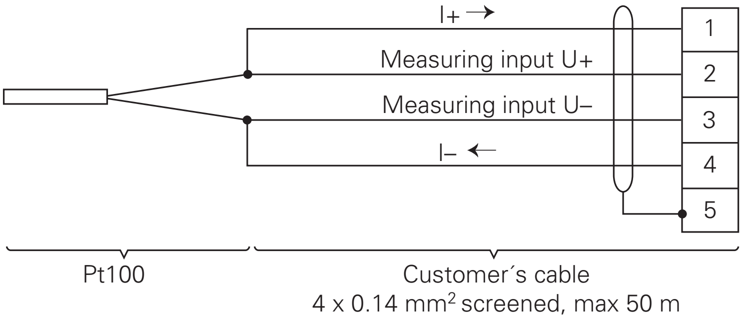


11.5.5 Connection to analogue inputs

Connecting cable 2 x 0.14 mm² screened, 50 m max.

11.5.6 Connection to inputs for thermistors

The Pt 100 thermistors must be connected in four-wire mode.



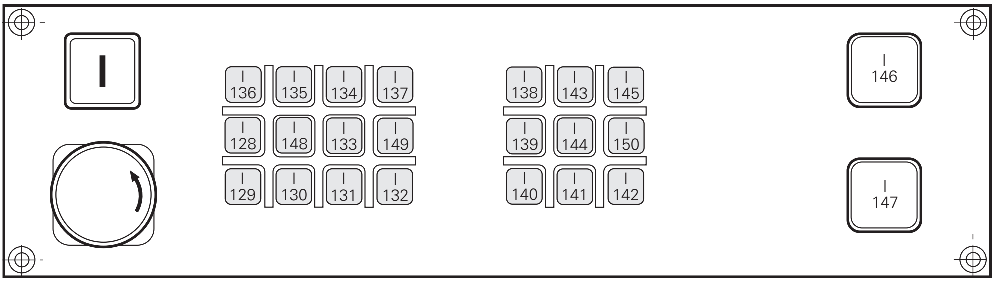
12 Machine operating panel

A separate 37-pin female connector (X46) is mounted on the logic unit for the connection to the machine operating panel. This connector includes the PLC inputs I128 to I152, the PLC outputs O0 to O7, as well as the 0 V and +24 V of the PLC power supply. The PLC inputs I128 to I152 may be connected only with the power supply from pins 36 and 37, since this power supply is internally secured as required.

HEIDENHAIN now offers a machine operating panel. It is installed beneath the TNC keyboard. The dimension drawings show the standard set of keys. Four additional black keys are supplied with the panel. The machine tool builder to use them replace unneeded axis keys, for example. Keys with other symbols are also available upon request.

MB 410	Id.-Nr. 293 757 02 (suited to BC 110 and TE 400 black)
MB 420	Id.-Nr. 293 757 12 (suited to BC 120 and TE 400 B gray)

Assignment of PLC inputs to the keys of the MB:



12.1 Pin connections

X46 Machine operating panel
Flange socket with female connector insert (37-pin)

Pin number	Assignment	Key on MB 410
1	I128	X–
2	I129	Y–
3	I130	Z–
4	I131	IV–
5	I132	V–
6	I133	X+
7	I134	Y+
8	I135	Z+
9	I136	IV+
10	I137	V+
11	I138	FN1
12	I139	FN2
13	I140	FN3
14	I141	FN4
15	I142	FN5
16	I143	Spindle on
17	I144	Spindle off
18	I145	Coolant on/off
19	I146	NC start
20	I147	NC stop
21	I148	Rapid traverse
22	I149	Black
23	I150	Black
24	I151	
25	I152	
26	O0	
27	O1	
28	O2	
29	O3	
30	O4	
31	O5	
32	O6	
33	O7	
34, 35	0 V (PLC)[1)]	
36, 37	+ 24 V (PLC)[2)]	
Housing	External shield	

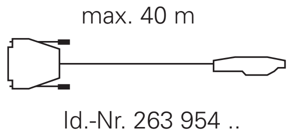
The PLC inputs I128 to I152 must be supplied with power only from pins 36 and 37, since this power supply is properly safeguarded.

1) Externally available PLC reference potential for the outputs O0-O7
2) Externally available (via fuse) PLC supply voltage for the inputs.

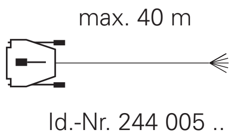
12.2 Connecting cable

Please use only HEIDENHAIN connecting cables.

HEIDENHAIN recommends that a 37-pin D-subminiature connector should be mounted on the machine operating panel. The machine operating panel can be connected to the logic unit with the standard HEIDENHAIN connecting cable Id.-Nr. 263 954 ..



If the machine operating panel does not have a 37-pin D-subminiature connector, the HEIDENHAIN connecting cable Id.-Nr. 244 005 .. may be used.



For the assignments of the multi-core conductors see "PLC inputs/outputs".

If for any reason the manufacturers of the machine have to produce their own cable, a 37-pin connector is available from HEIDENHAIN (Id.-Nr. 243 937 ZY).

13 TNC keyboard

The TNC keyboard TE 400 is connected to the logic unit by a connecting cable. Additionally, the Soft keys for the VDU (BC 110) are connected to the TNC keyboard by a flat cable. The flat cable is one of the items supplied with the VDU.

13.1 Pin connections

On the logic unit

X45 TNC keyboard (TE 400)
D-sub female connector (37-pin)

Pin Number	Assignment
1	RL0
2	RL1
3	RL2
4	RL3
5	RL4
6	RL5
7	RL6
8	RL7
9	RL8
10	RL9
11	RL10
12	RL11
13	RL12
14	RL13
15	RL14
16	RL15
17	RL16
18	RL17
19	RL18
20	SL0
21	SL1
22	SL2
23	SL3
24	SL4
25	SL5
26	SL6
27	SL7
28	RL19
29	RL20
30	Do not use
31	RL21
32	RL22
33	RL23
34	Spindle override (wiper)
35	Feed override (wiper)
36	+ 5 V override-potentiometer
37	0 V override-potentiometer
Housing	External screen

On the TNC keyboard

X2 for the connection to the logic unit

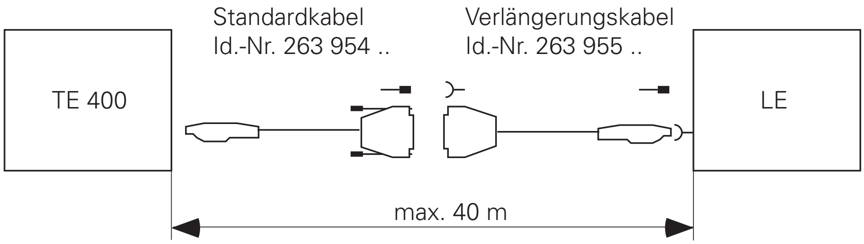
Pin connections as for X45 on the logic unit

X1 for the connection to the soft keys on the Visual Display Unit (BC)

Pin Number	Assignment
1	SL0
2	SL1
3	SL2
4	SL3
5	Do not use
6	RL15
7	RL14
8	RL13
9	RL12

13.2 Connecting cable

Please use only HEIDENHAIN connecting cables.



The flat cable between the TNC keyboard and the VDU is included in the package delivered with the VDU.

14 VDU

The power supply for the VDU-BC 110 B is described under the heading "Power supply". The video signals are transmitted via a connecting cable from the logic unit to the display unit. The soft keys for the VDU are connected to the TNC keyboard by a flat cable. This flat cable is one of the items supplied with the VDU.

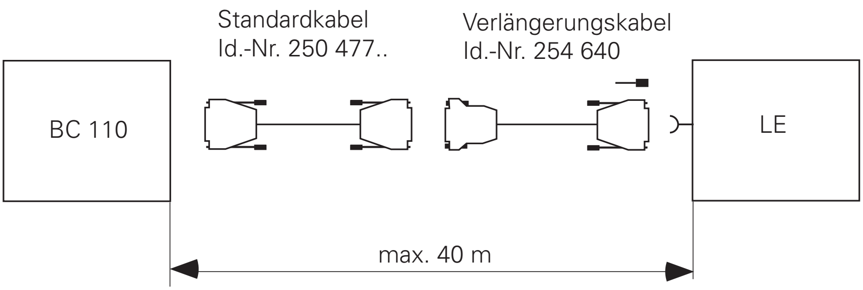
14.1 Pin connections

X43 VDU (BC 110)
D-sub female connector (15-pin)

Pin Number	Assignment
1, 8, 11	GND
2 to 6, 12, 13	Do not use
7	R
9	V SYNC
10	H SYNC
14	G
15	B
Housing	External shield

14.2 Connecting cable

Please use only HEIDENHAIN connecting cables.



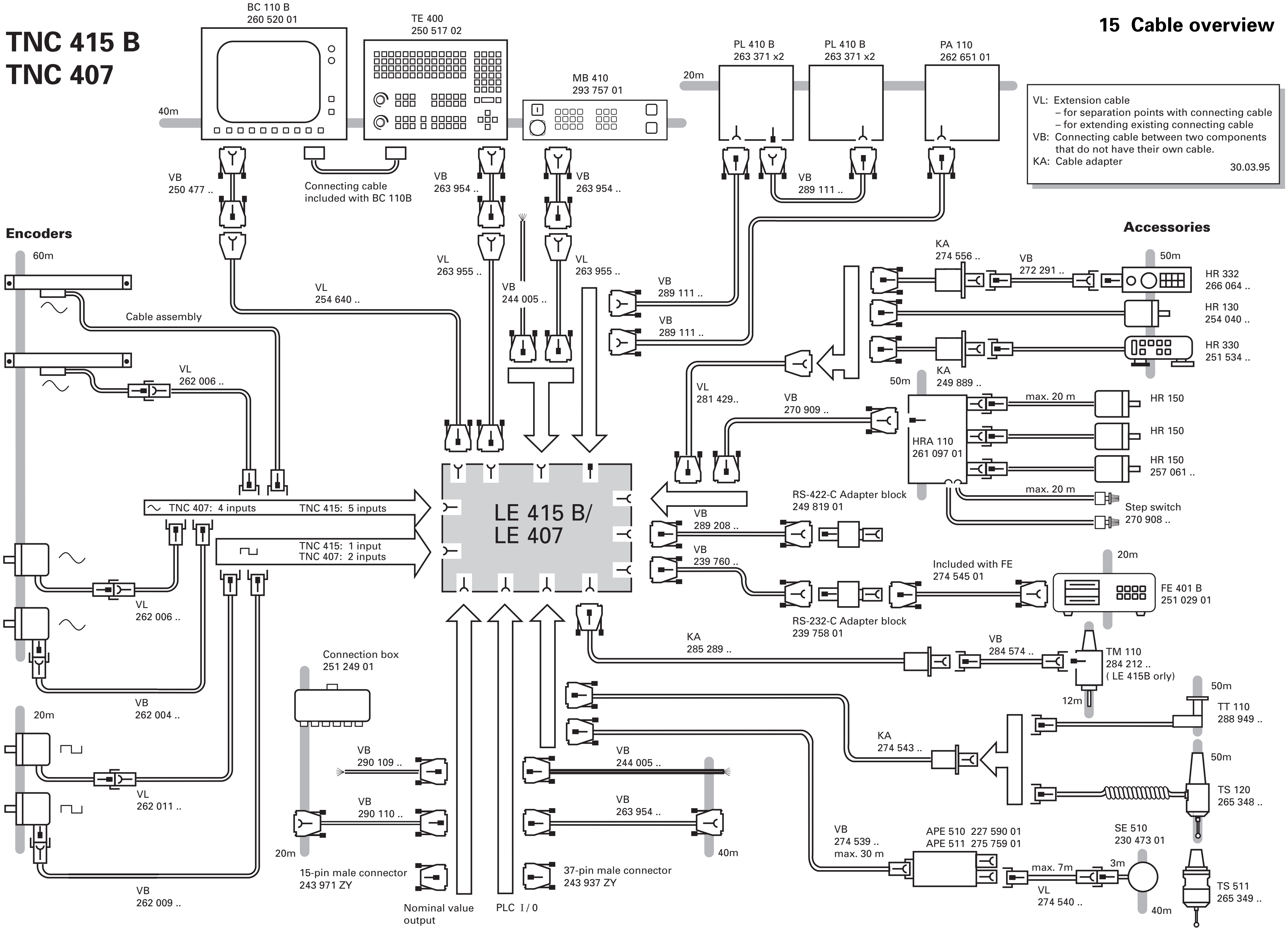
The flat connecting cable between the VDU and the TNC keyboard is one of the items supplied with the VDU.

14.3 Connecting the BC 120

X43 Visual Display Unit (BC 120)

Logic unit I		Adapter 313 434 02	Extension cable Id. Nr. 312 878 ..			BC 120
D-sub connector (female) 15-pin 2-row	**Assignment**	**2-row / 3-row**	**D-sub connector (male) 15-pin 3-row**		**D-sub connector (female) 15-pin 3-row**	**D-sub connector (male) 15-pin 3-row**
1	GND		1	Coax I Red	1	1
2	*Do not assign*		2	Coax I Green	2	2
3	*Do not assign*		3	Coax I Blue	3	3
4	*Do not assign*		4		4	4
5	*Do not assign*		5		5	5
6	*Do not assign*		6	Coax S Red	6	6
7	R		7	Coax S Green	7	7
8	GND		8	Coax S Blue	8	8
9	VSYNC		9		9	9
10	HSYNC		10	Gray	10	10
11	GND		11	Green	11	11
12	*Do not assign*		12		12	12
13	*Do not assign*		13	Pink	13	13
14	G		14	Yellow	14	14
15	B		15		15	15
Housing	External shield	Housing	Housing	External shield	Housing	Housing

TNC 415 B
TNC 407

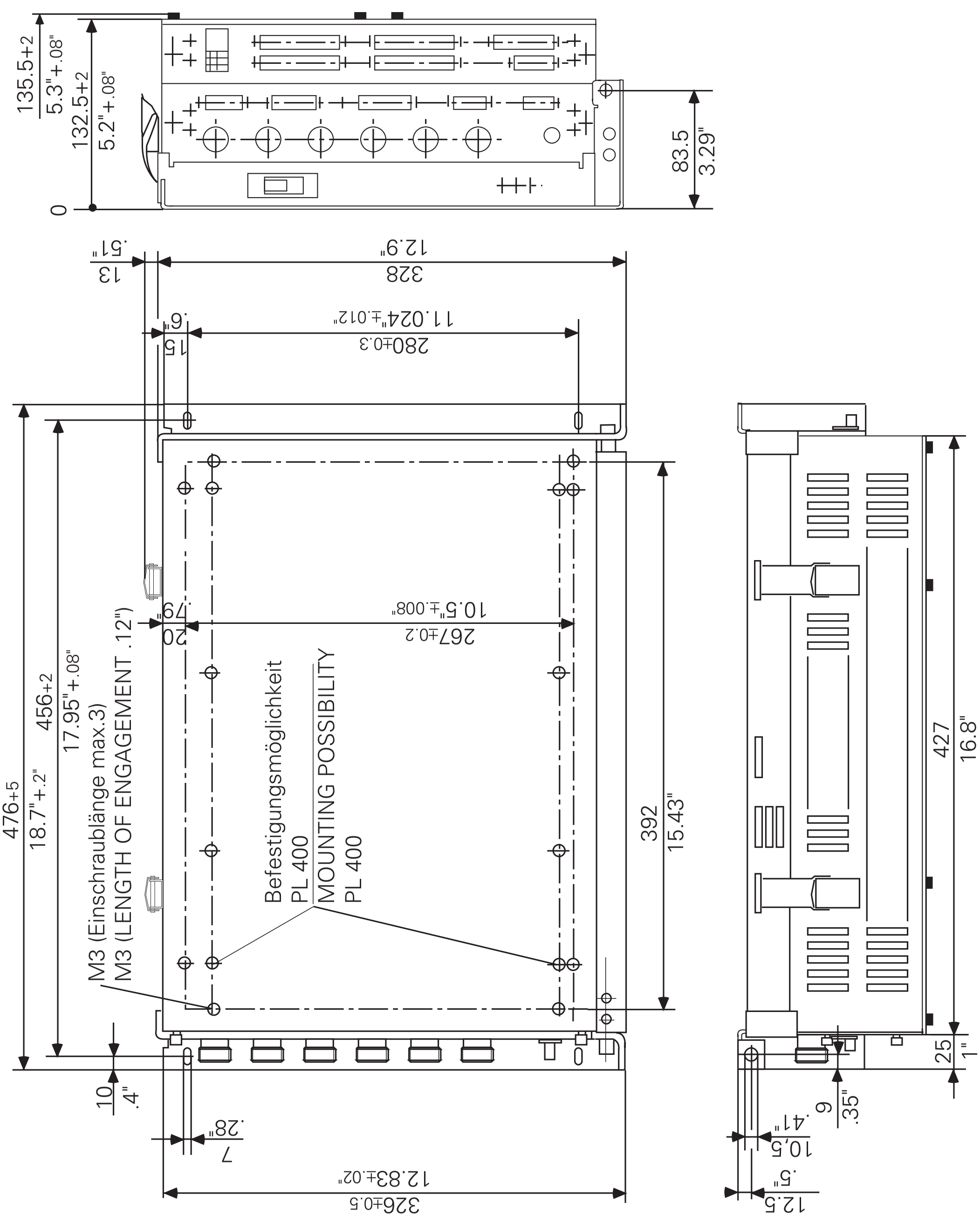


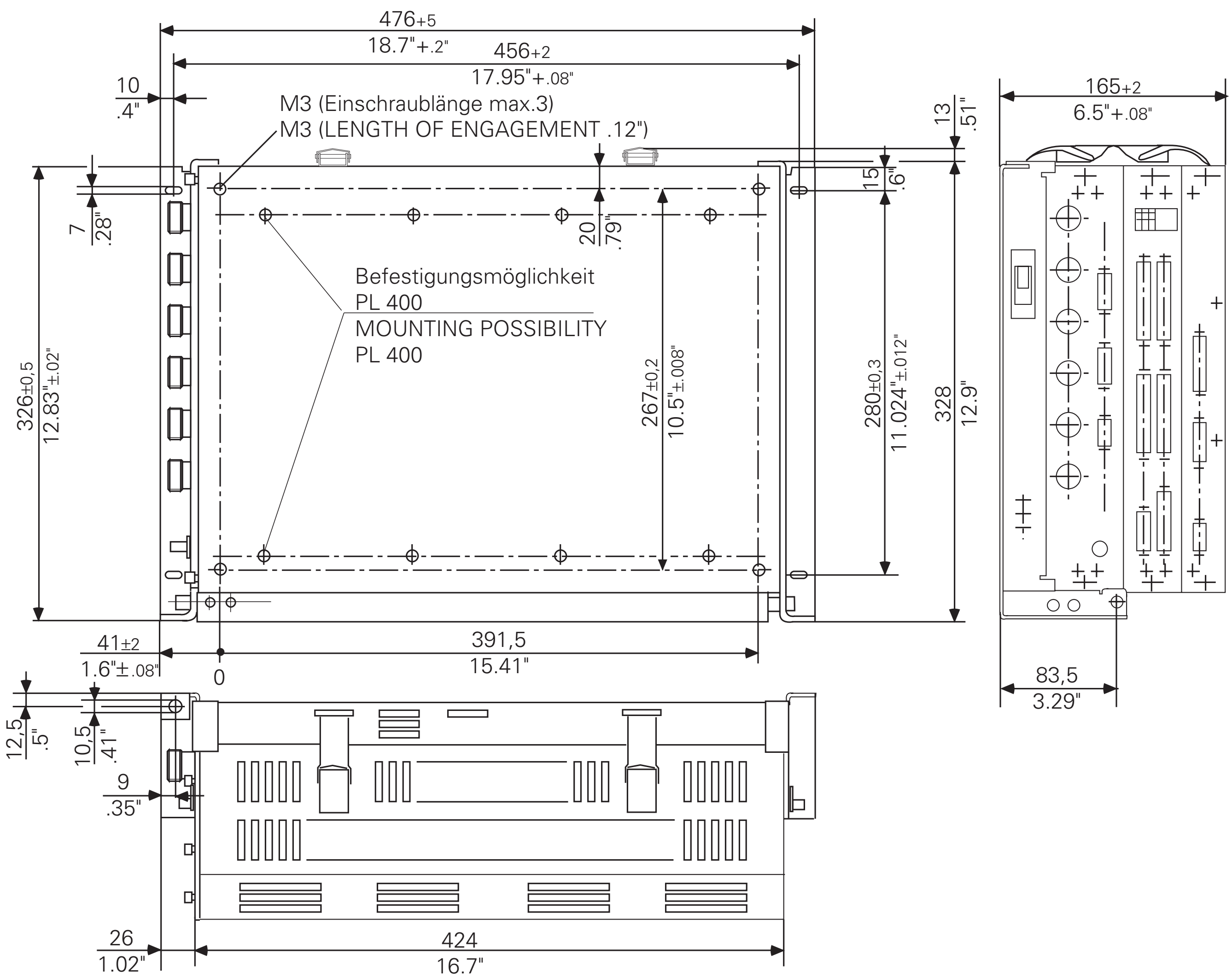
New ID numbers for connecting cables

Connecting cable	Old Id. Nr.	**New Id. Nr.**
LE - BC 110B	250 477 ..	**311 531 ..**
LE - BC 110 B (extension)	254 640 ..	**311 532 ..**
LE - encoder	262 006 ..	**309 774 ..**
LE - encoder	262 004 ..	**309 773 ..**
LE - encoder	262 009 ..	**298 399 ..**
LE - encoder	262 011 ..	**298 400 ..**
Connector male 15-pin	243 971 ZY	**315 650 03**
Connector male 37-pin	243 937 ZY	**315 650 07**

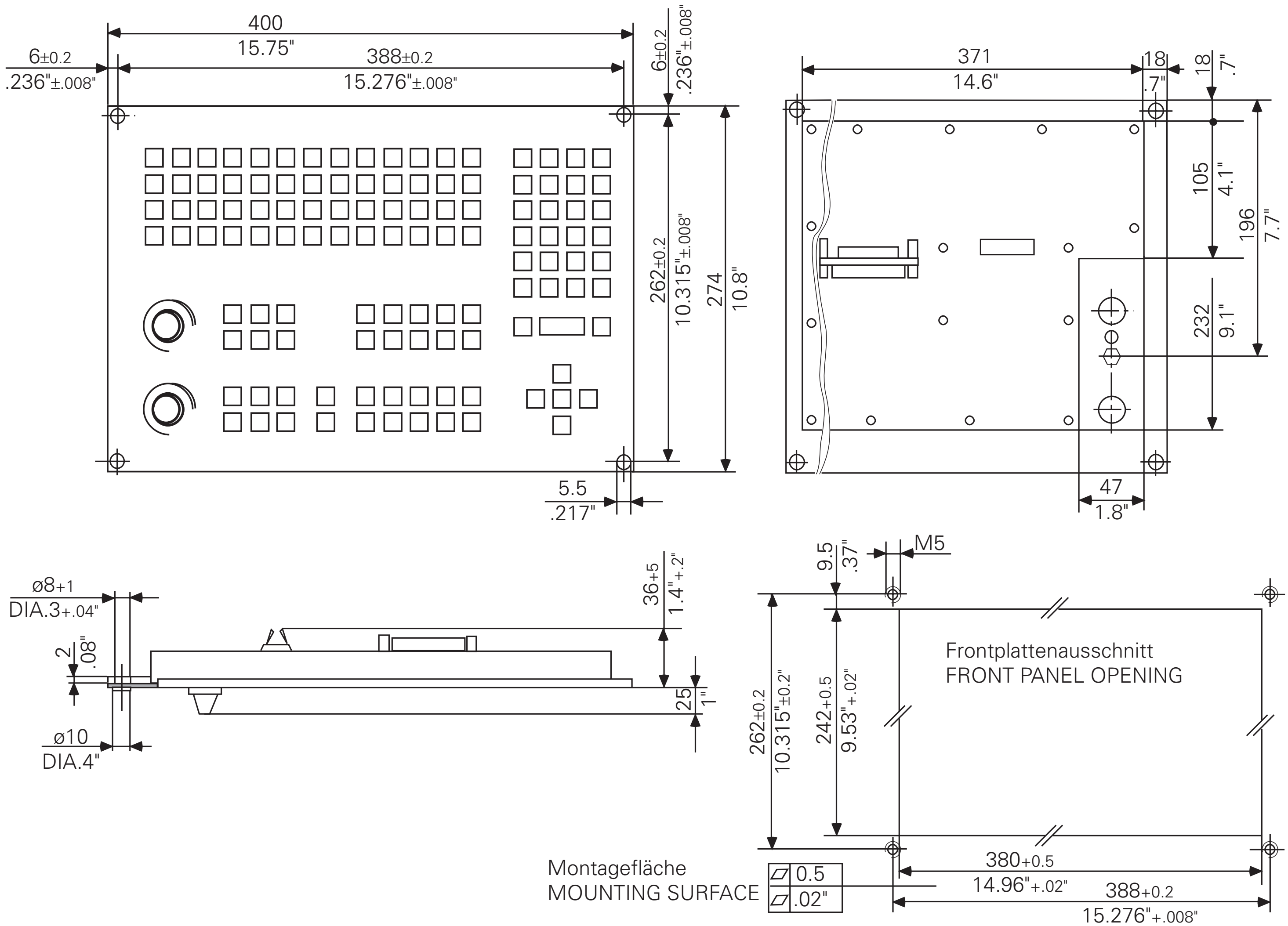
16 Dimensions

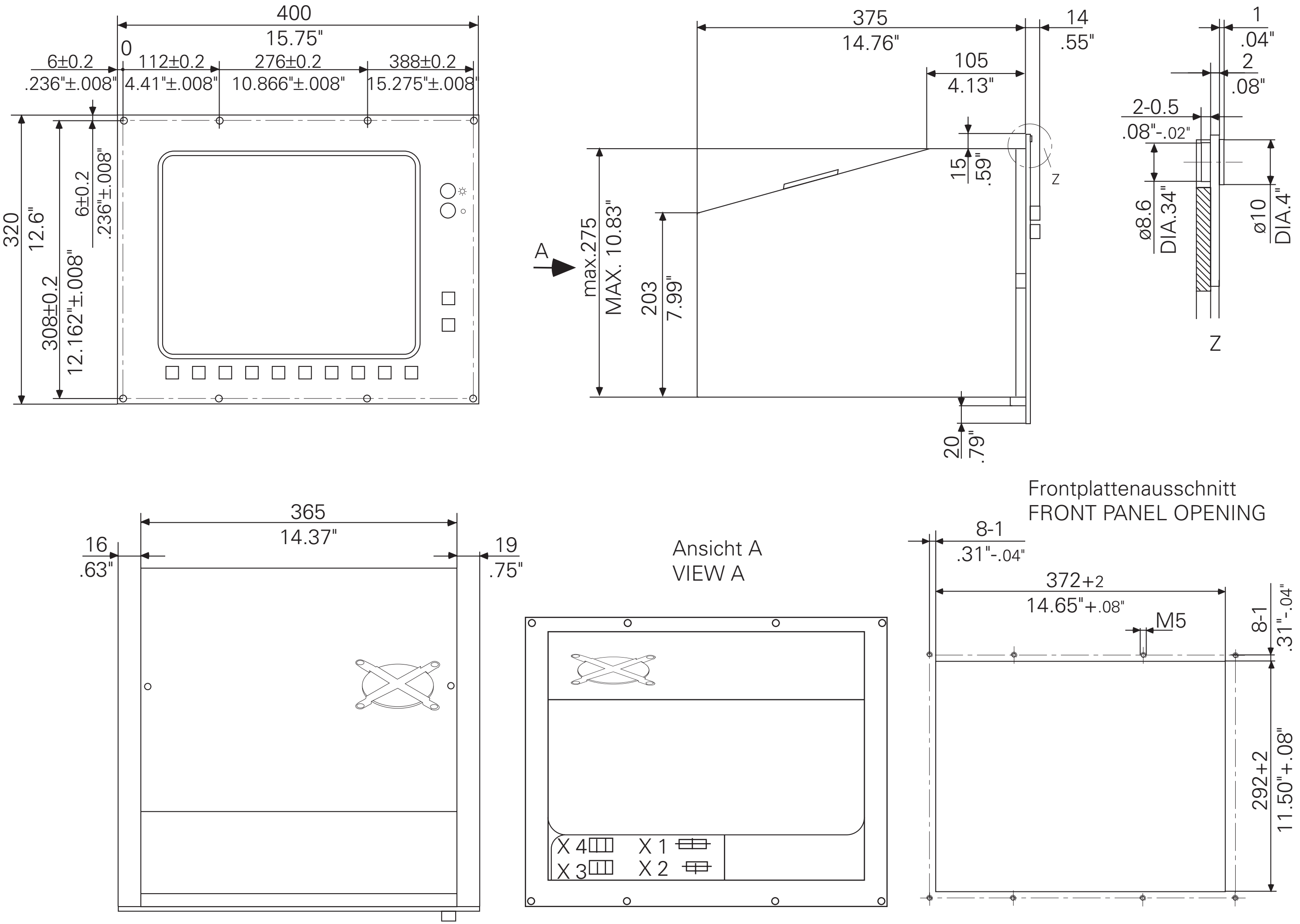
16.1 LE 407




476+5
18.7"+.2"
456+2
17.95"+.08"
10
.4"
M3 (Einschraublänge max.3)
M3 (LENGTH OF ENGAGEMENT .12")
13
.51"
15
.6"
7
.28"
20
.79"
Befestigungsmöglichkeit
PL 400
MOUNTING POSSIBILITY
PL 400
326±0,5
12.83"±.02"
267±0,2
10.5"±.008"
280±0,3
11.024"±.012"
328
12.9"
165+2
6.5"+.08"
41±2
1.6"±.08"
391,5
15.41"
0
83,5
3.29"
12,5
.5"
10,5
.41"
9
.35"
26
1.02"
424
16.7"

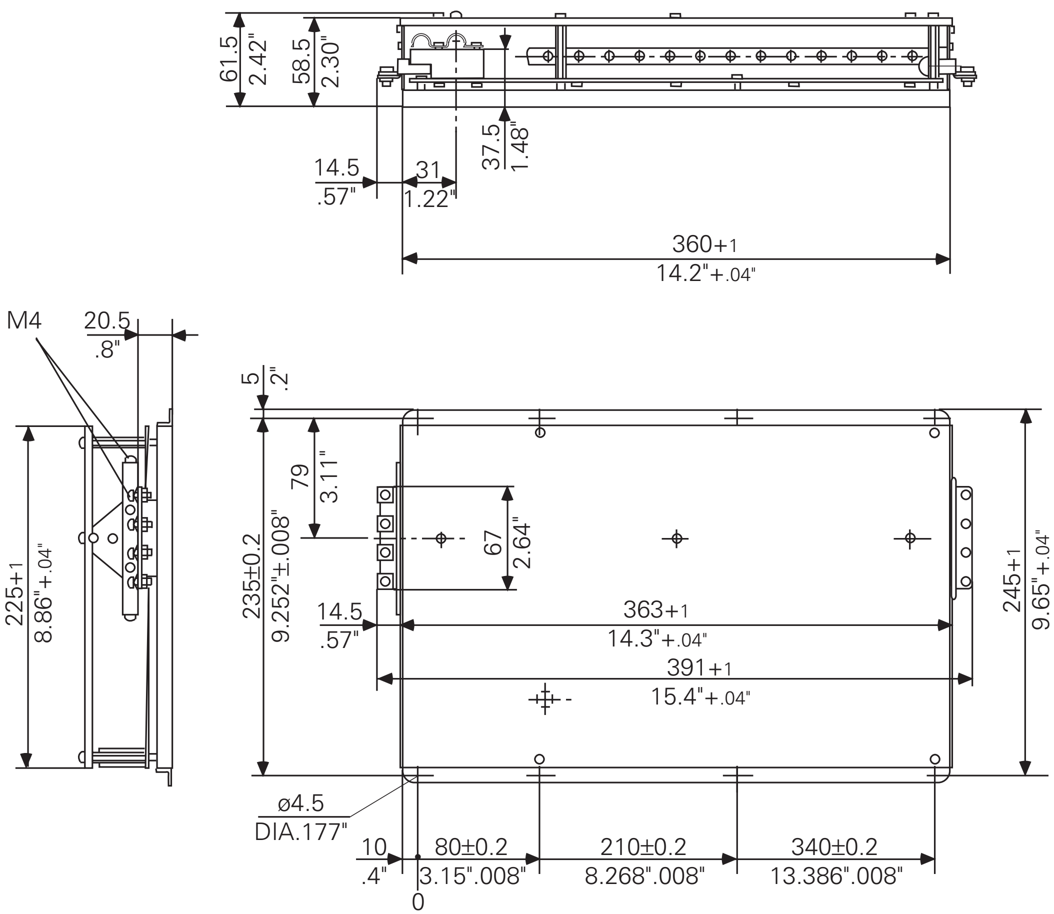
16.3 TE 400


400
15.75"
6±0.2
.236"±.008"
388±0.2
15.276"±.008"
6±0.2
.236"±.008"
262±0.2
10.315"±.008"
274
10.8"
5.5
.217"
371
14.6"
18
.7"
18
.7"
105
4.1"
196
7.7"
232
9.1"
47
1.8"
ø8+1
DIA.3+.04"
2
.08"
ø10
DIA.4"
36+5
1.4"+.2"
25
1"
9.5
.37"
M5
262±0.2
10.315"±0.2"
242+0.5
9.53"+.02"
Frontplattenausschnitt
FRONT PANEL OPENING
Montagefläche
MOUNTING SURFACE
0.5
.02"
380+0.5
14.96"+.02"
388+0.2
15.276"+.008"

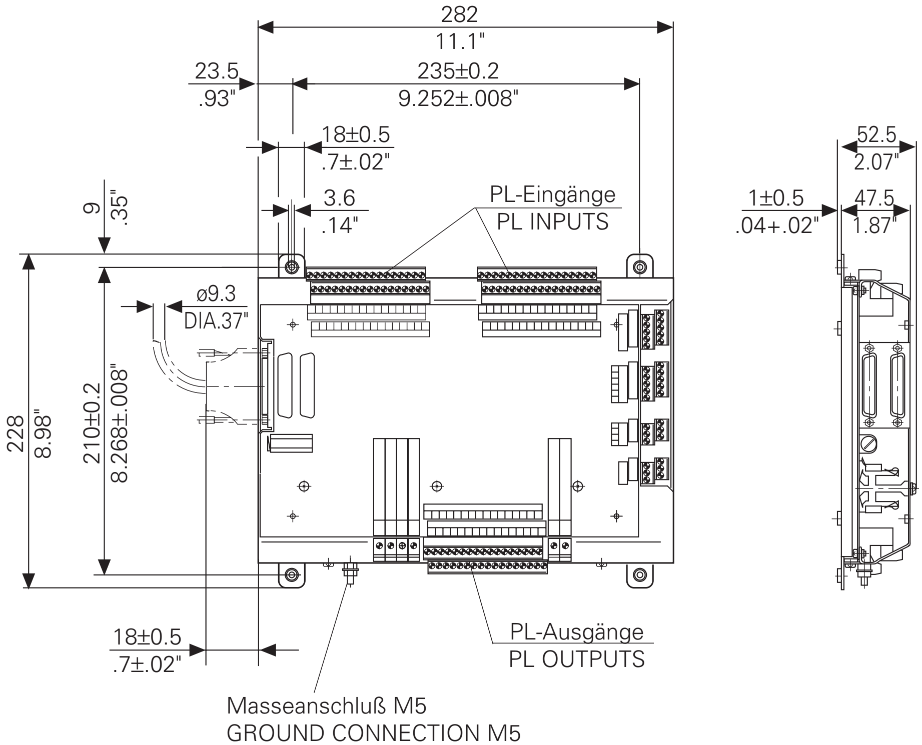

400
15.75"
0
6±0.2
.236"±.008"
112±0.2
4.41"±.008"
276±0.2
10.866"±.008"
388±0.2
15.275"±.008"
320
12.6"
308±0.2
12.162"±.008"
6±0.2
.236"±.008"
375
14.76"
14
.55"
105
4.13"
15
.59"
Z
A
max.275
MAX. 10.83"
203
7.99"
20
.79"
1
.04"
2
.08"
2-0.5
.08"-.02"
ø8.6
DIA.34"
ø10
DIA.4"
Z
365
14.37"
16
.63"
19
.75"
Ansicht A
VIEW A
X 4
X 1
X 3
X 2
Frontplattenausschnitt
FRONT PANEL OPENING
8-1
.31"-.04"
372+2
14.65"+.08"
M5
8-1
.31"-.04"
292+2
11.50"+.08"

16.5 PLC expansion boards

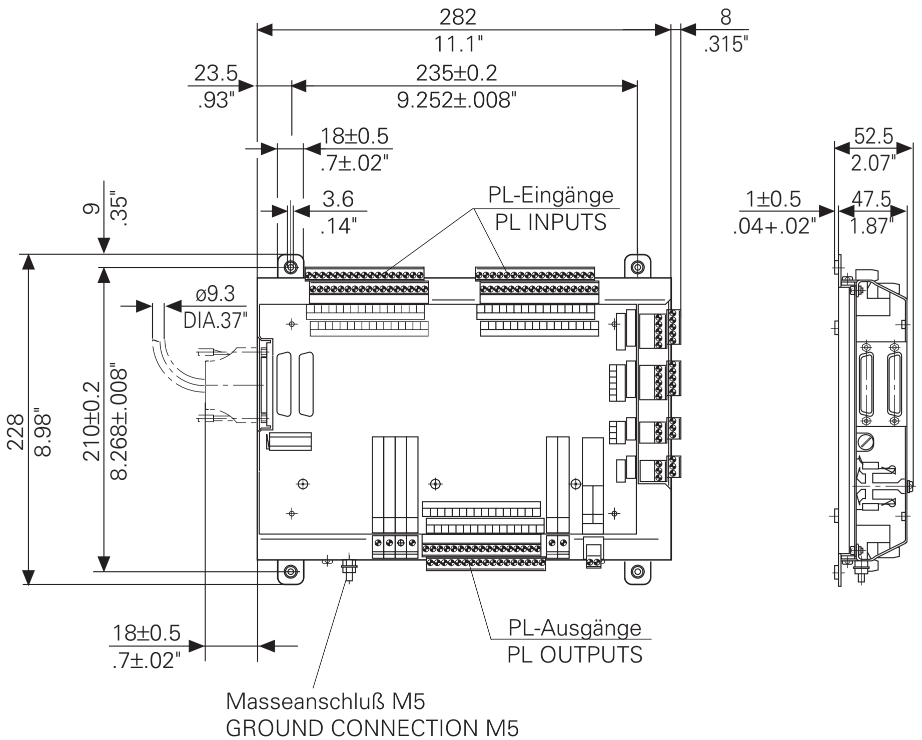
PL 400

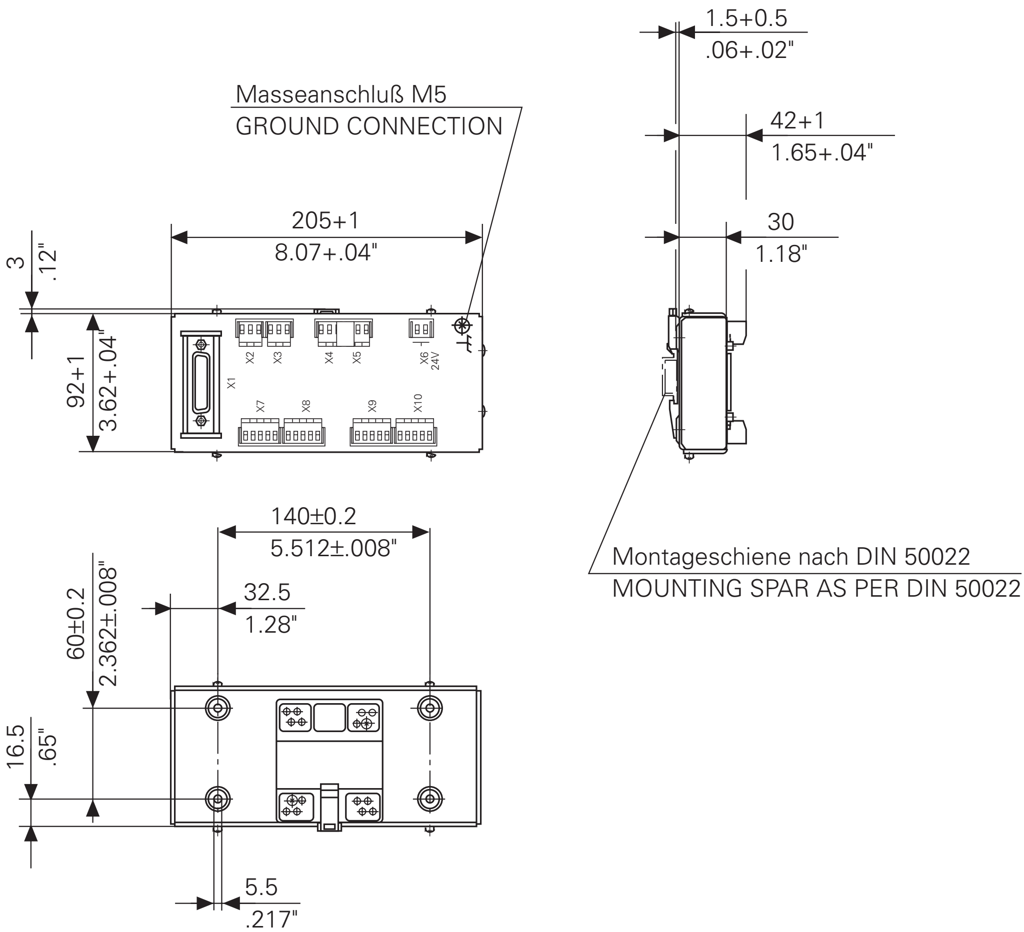


PL 410



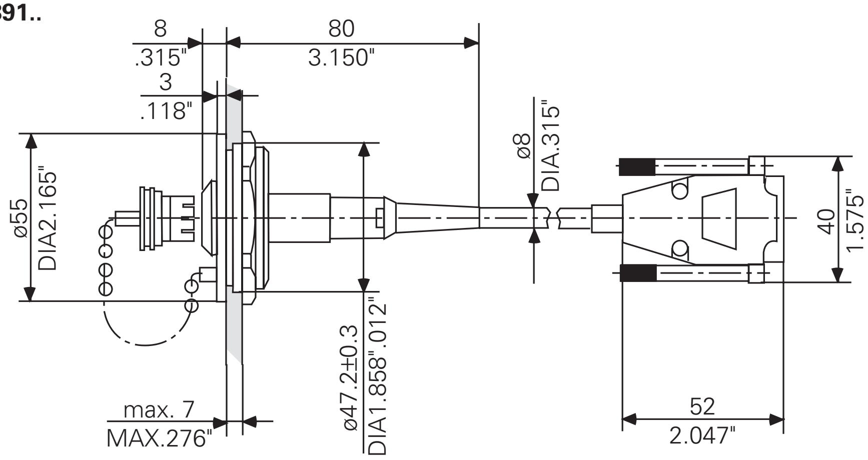
PL 410 B



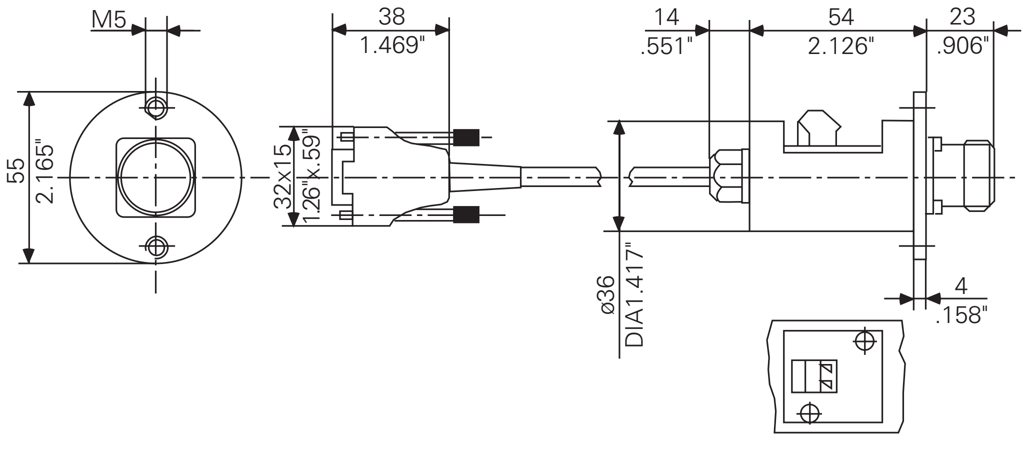

Masseanschluß M5
GROUND CONNECTION
205+1
8.07+.04"
3
.12"
92+1
3.62+.04"
X1
X2
X3
X4
X5
X6
24V
X7
X8
X9
X10
140±0.2
5.512±.008"
32.5
1.28"
60±0.2
2.362±.008"
16.5
.65"
5.5
.217"
1.5+0.5
.06+.02"
42+1
1.65+.04"
30
1.18"
Montageschiene nach DIN 50022
MOUNTING SPAR AS PER DIN 50022

16.6 Cable adapters

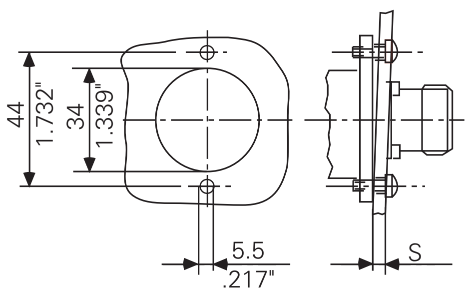
Cable adapter for TS 120
Id.-Nr. 244 891..



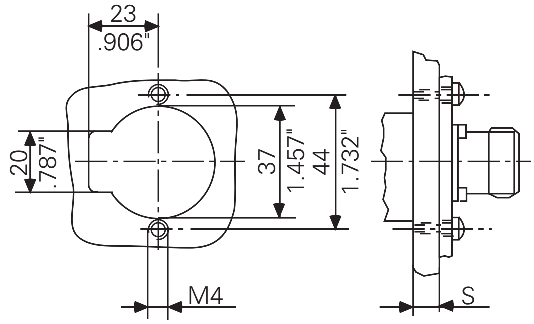
Cable adapter for HR 330
Id.-Nr. 249 889..



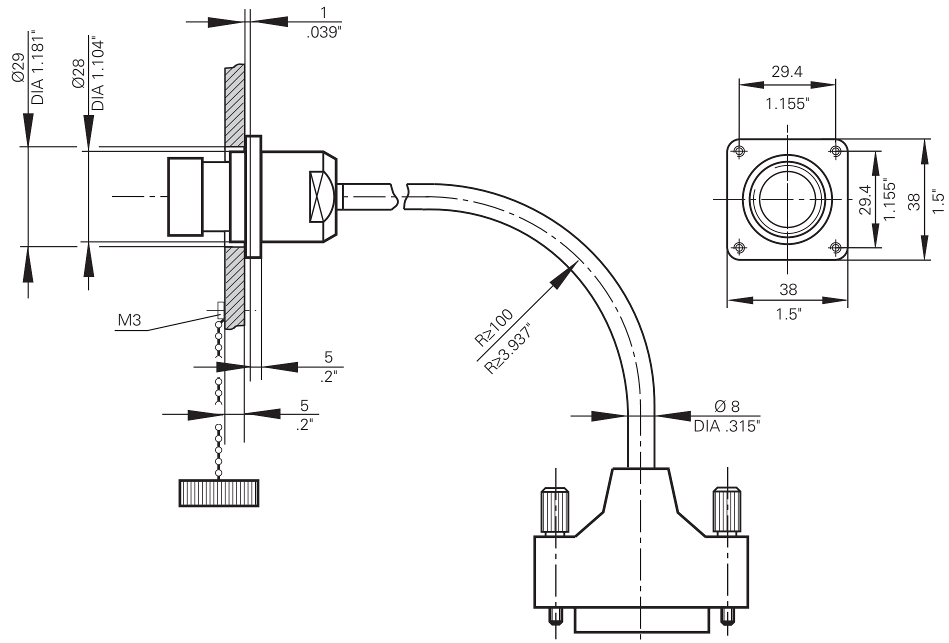
Mounting opening for wall thickness S<4



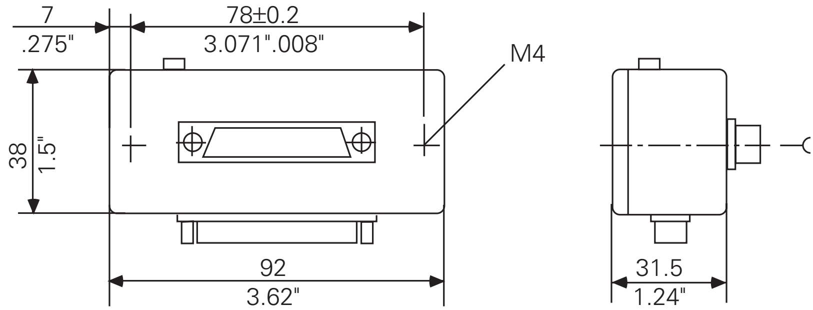
Mounting opening for wall thickness S>4



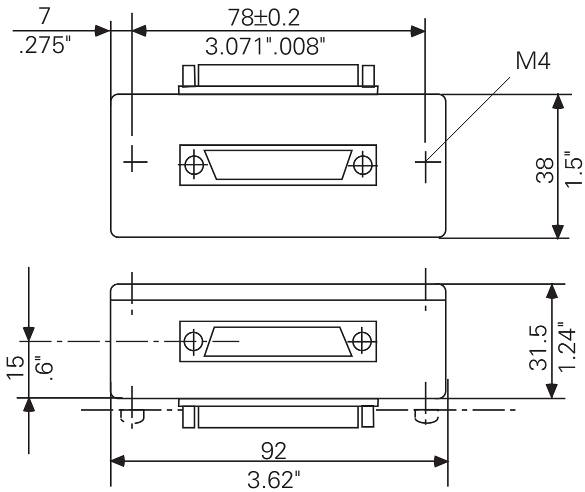
Cable adapter for TM 110 **Id.-Nr. 285 289 ..**



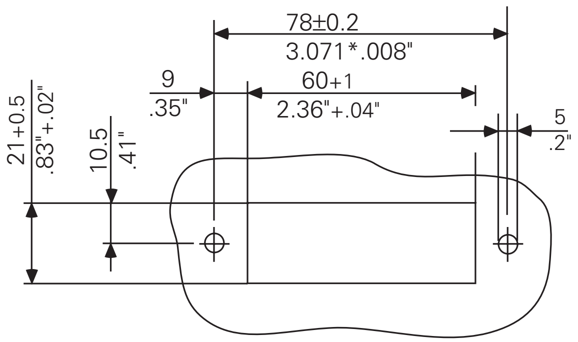
Adapter Block RS-232-C/V.24



Adapter Block RS-422/V.11

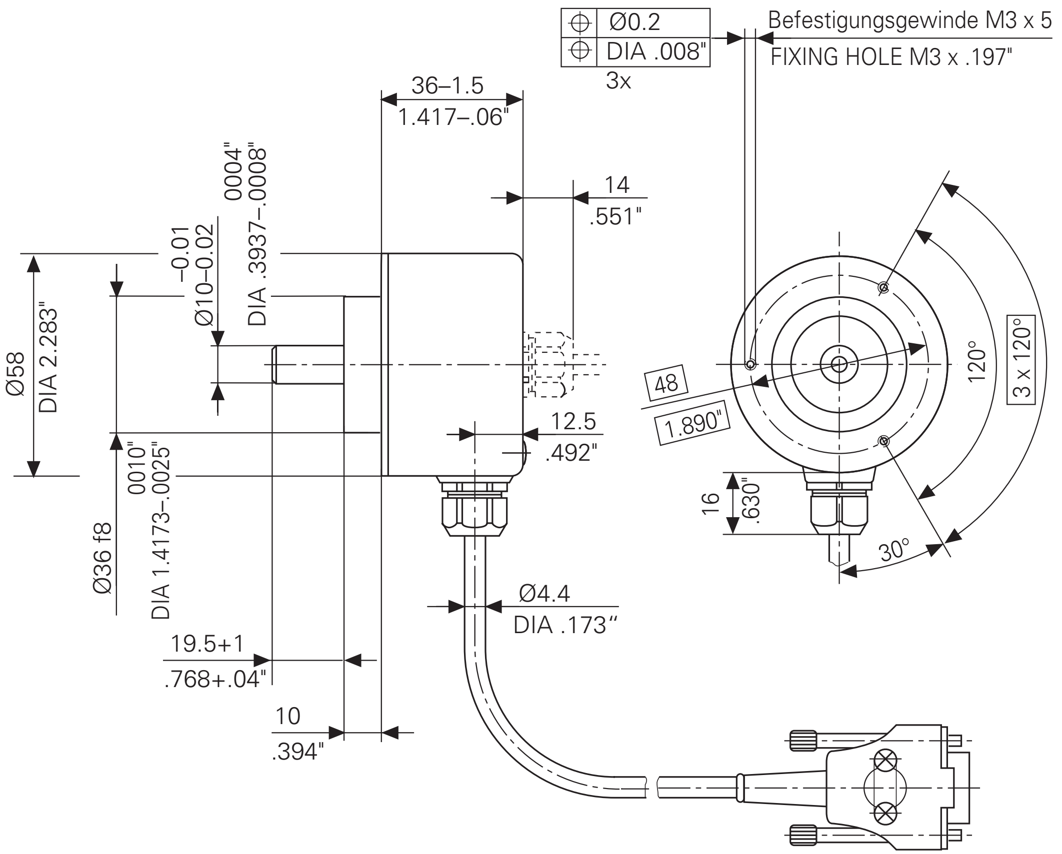


Opening for mounting the adapter

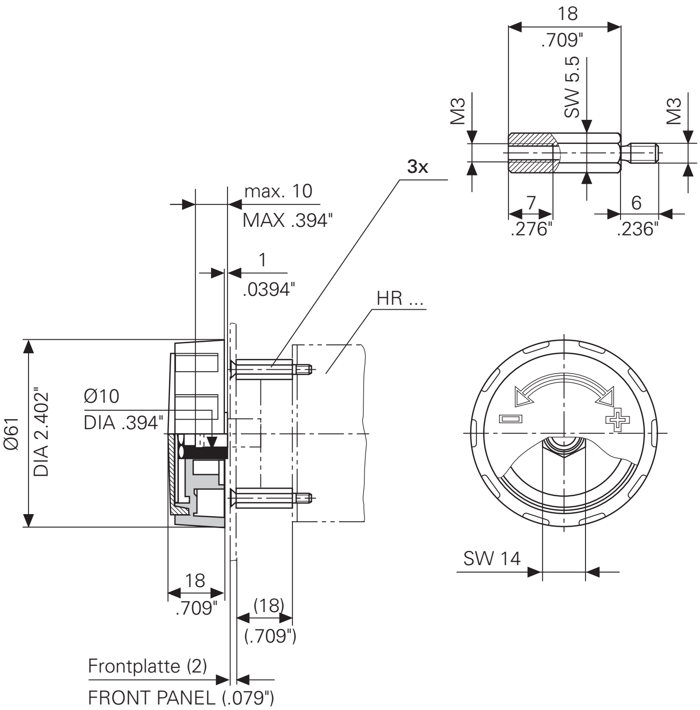


16.7 Handwheels

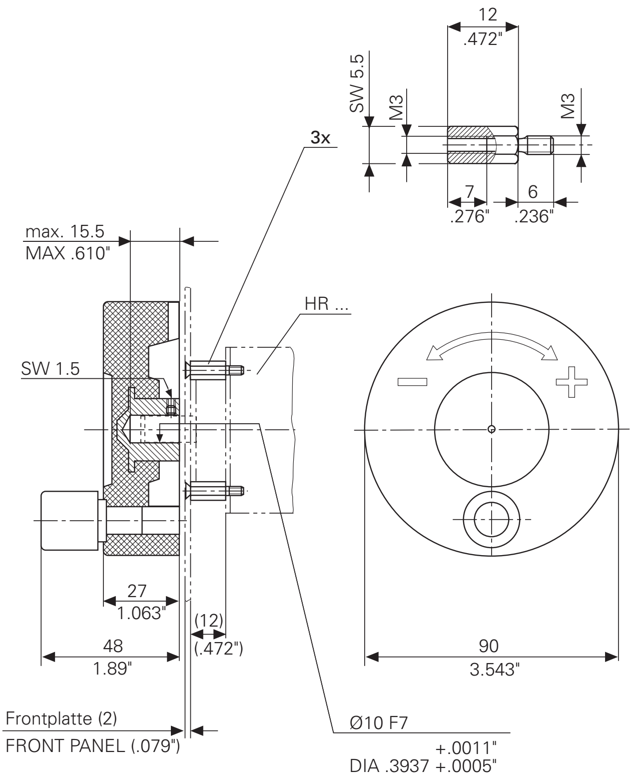
16.7.1 HR 130 integral handwheel



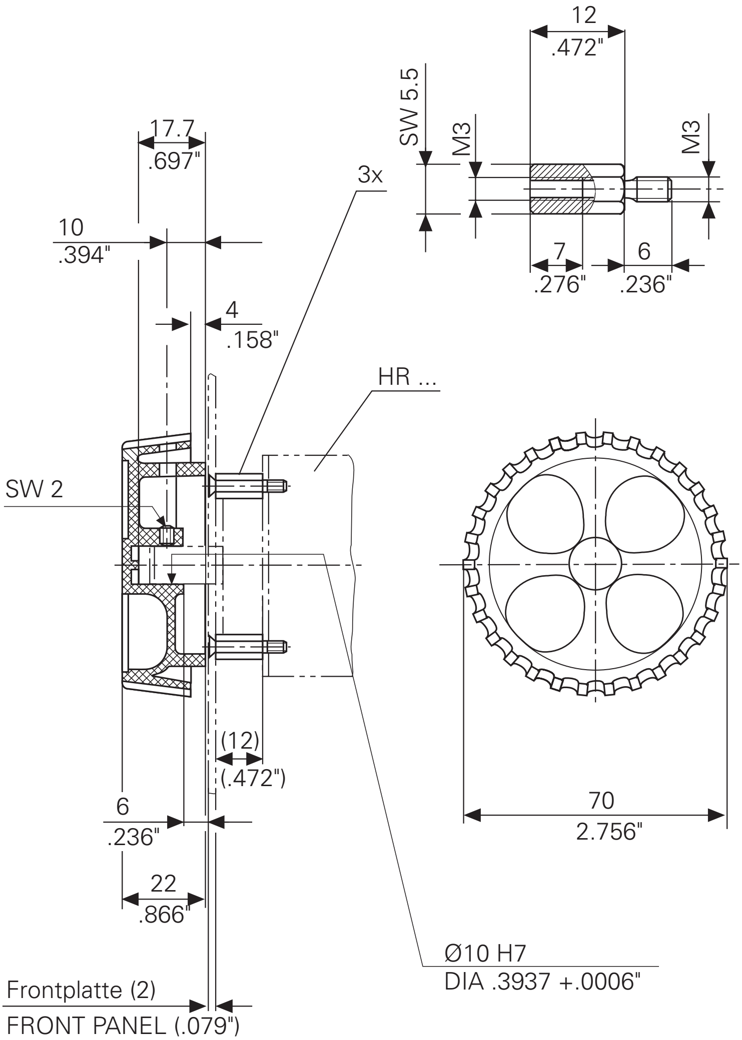
Knob, small



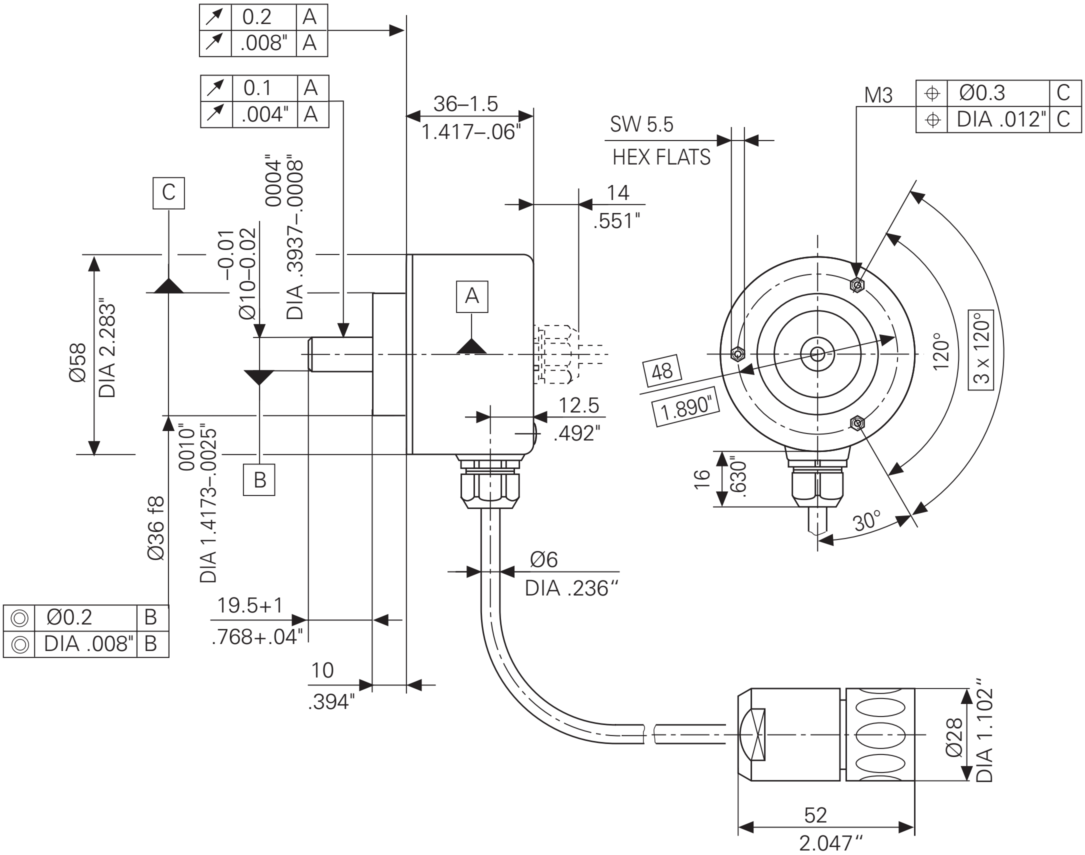
Knob large



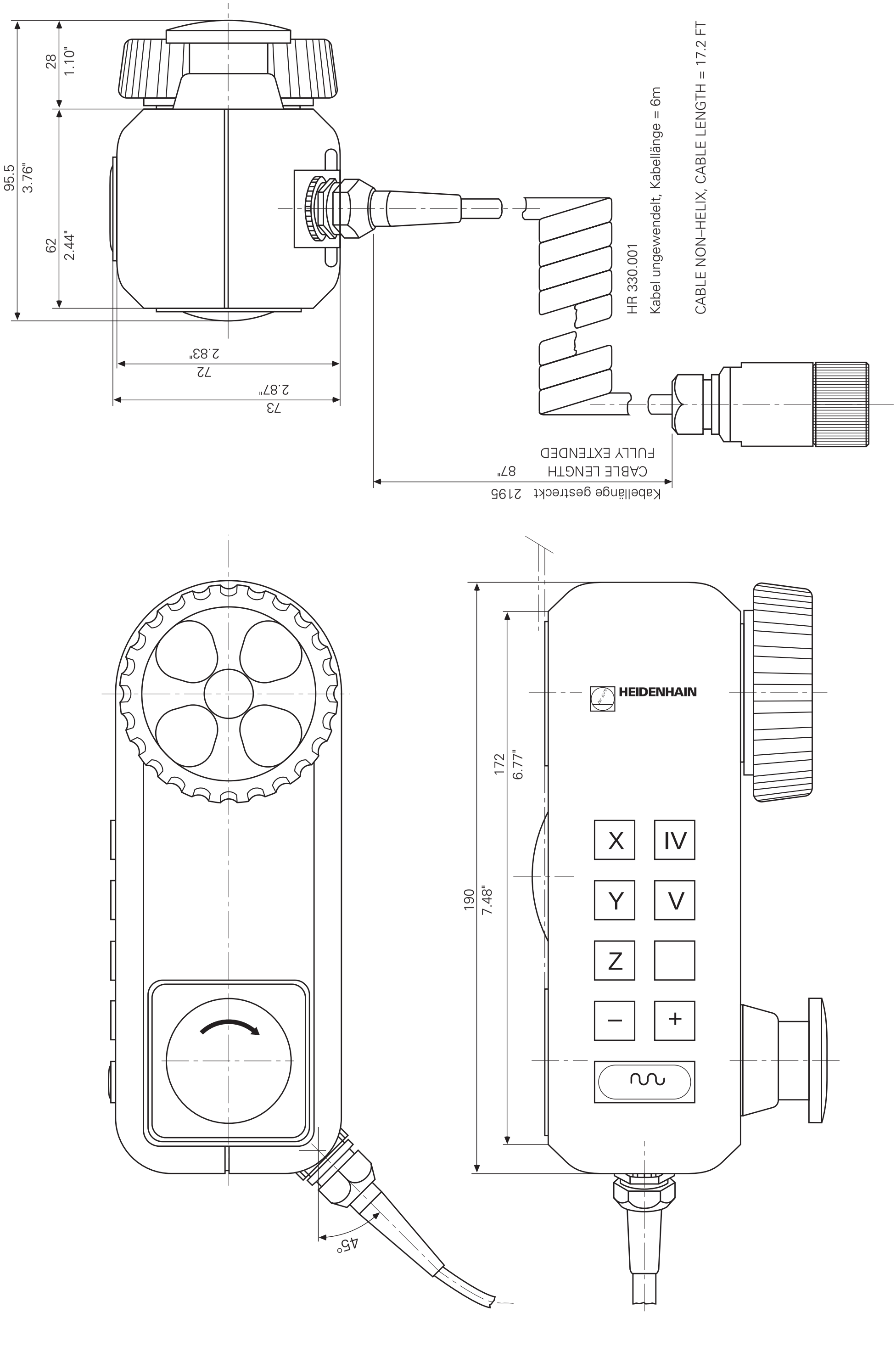
Knob, ergonomic



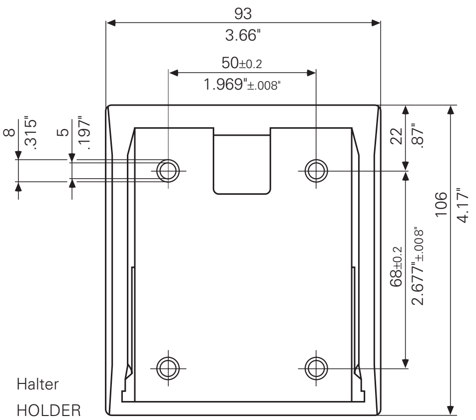
16.7.2 HR 150

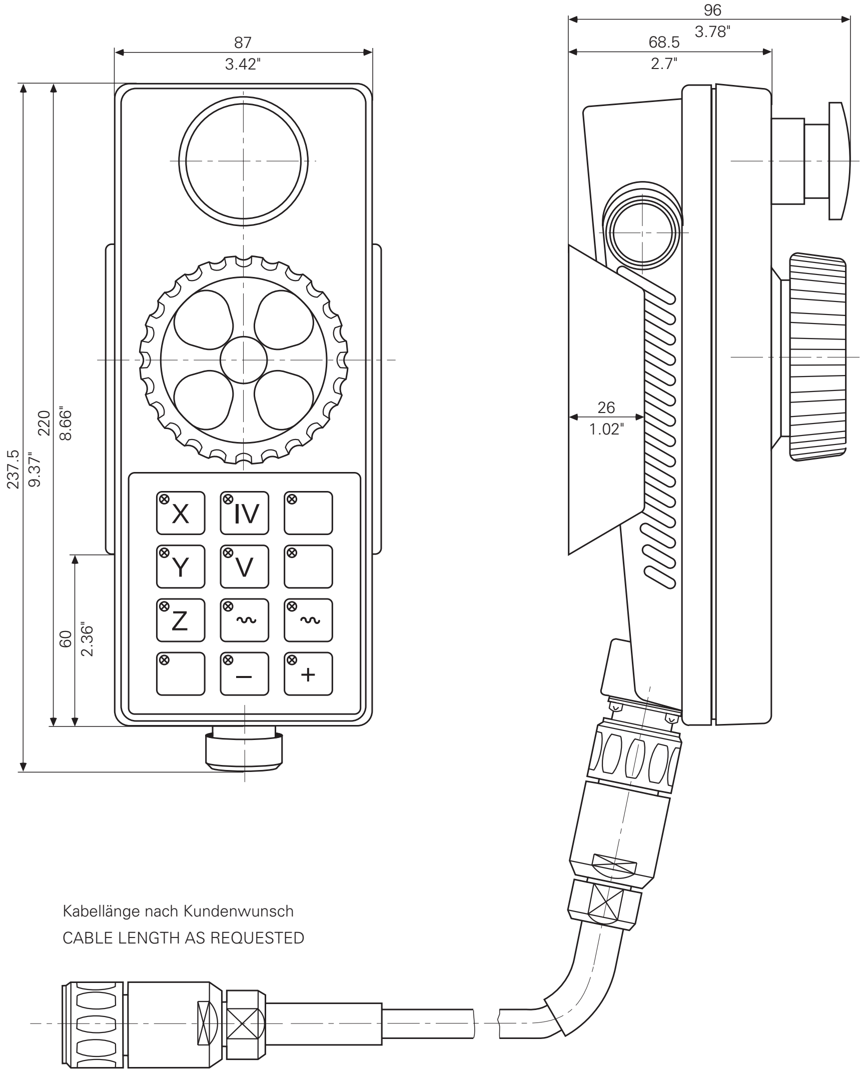

0.2 A
.008" A
0.1 A
.004" A
36–1.5
1.417–.06"
SW 5.5
HEX FLATS
M3
Ø0.3 C
DIA .012" C
14
.551"
C
A
Ø58
DIA 2.283"
–0.01
Ø10–0.02
0004"
DIA .3937–.0008"
48
1.890"
120°
3 x 120°
12.5
.492"
0010"
DIA 1.4173–.0025"
B
16
.630"
30°
Ø36 f8
Ø6
DIA .236"
Ø0.2 B
DIA .008" B
19.5+1
.768+.04"
10
.394"
Ø28
DIA 1.102"
52
2.047"

16.7.3 Portable handwheel HR 330

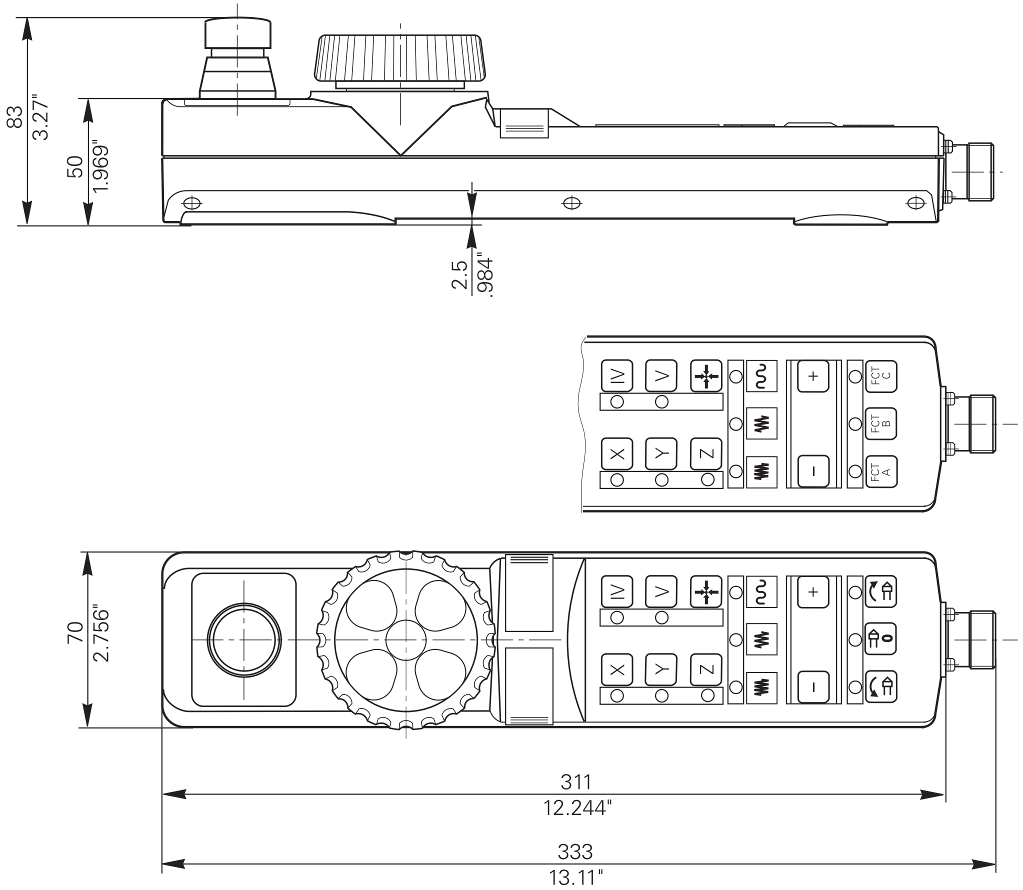


16.7.4 Portable handwheel HR 332

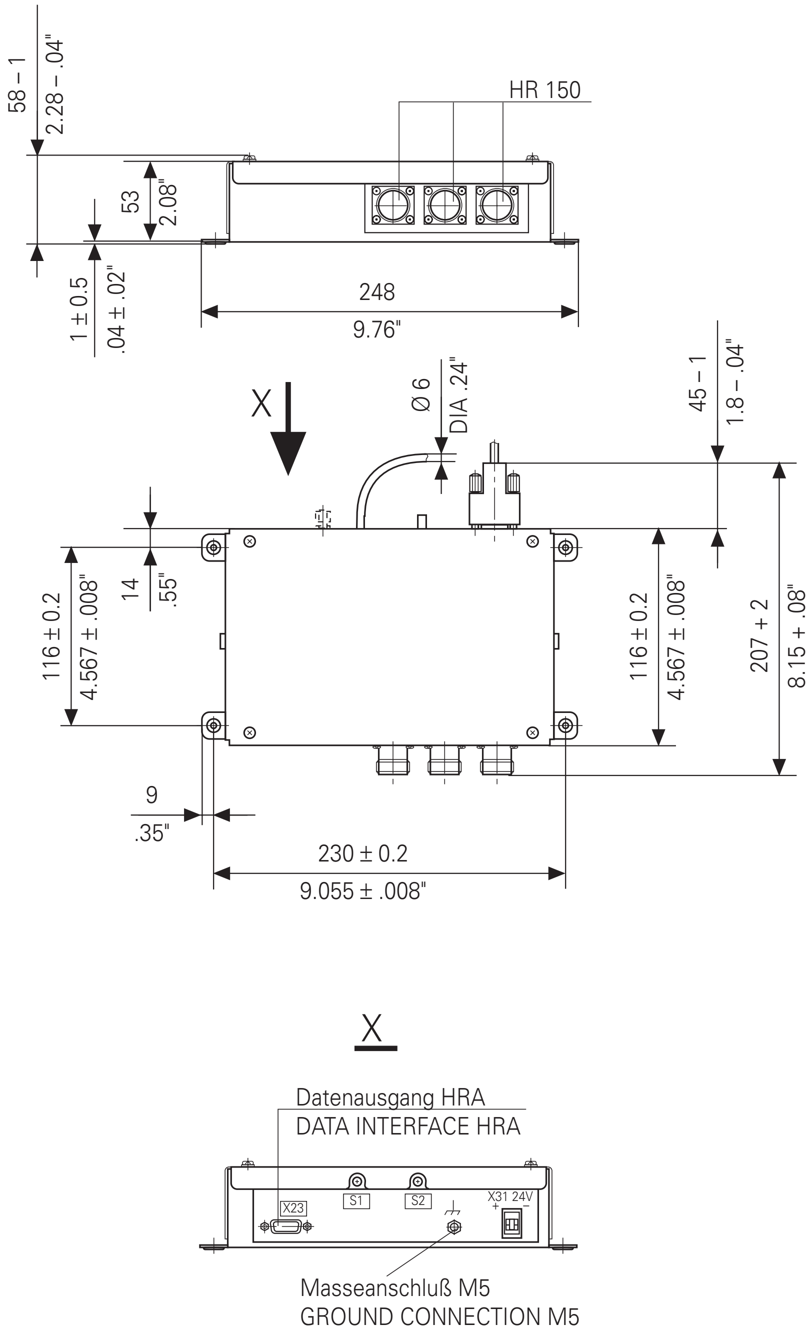




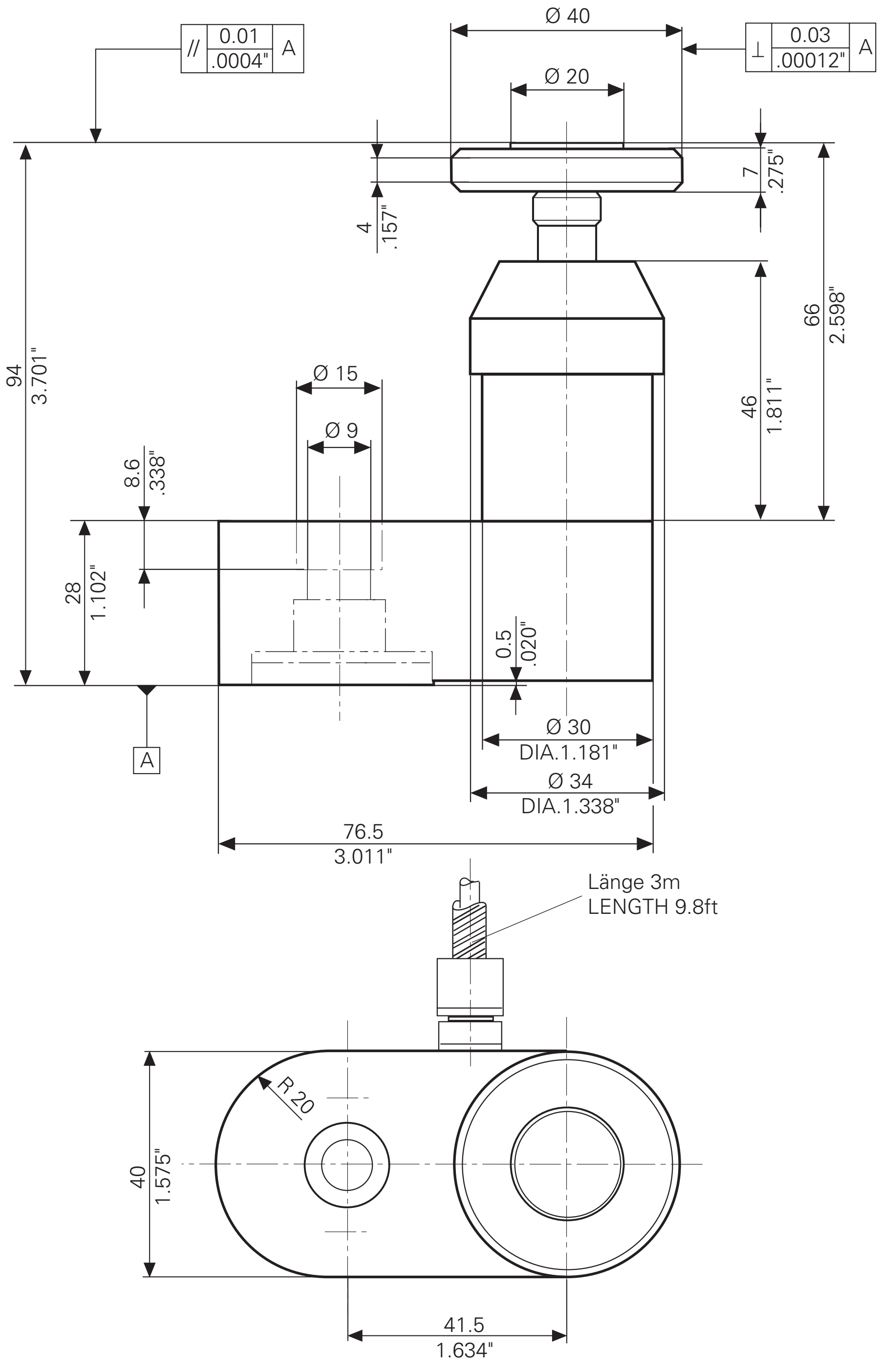
16.7.5 Portable handwheel HR 410



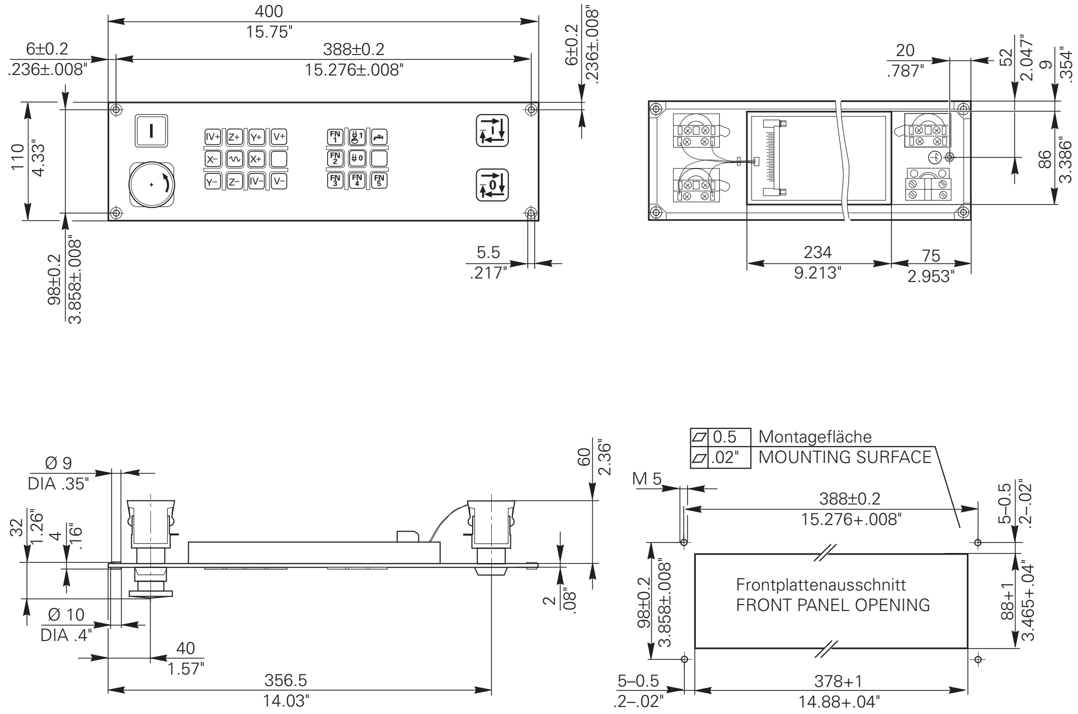
16.7.6 Handwheel adapter HRA 110 (for HR 150)



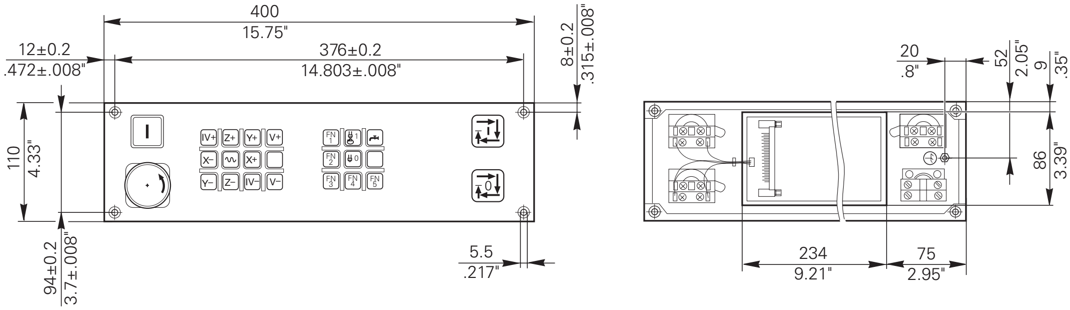
16.8 TT 110 for tool calibration

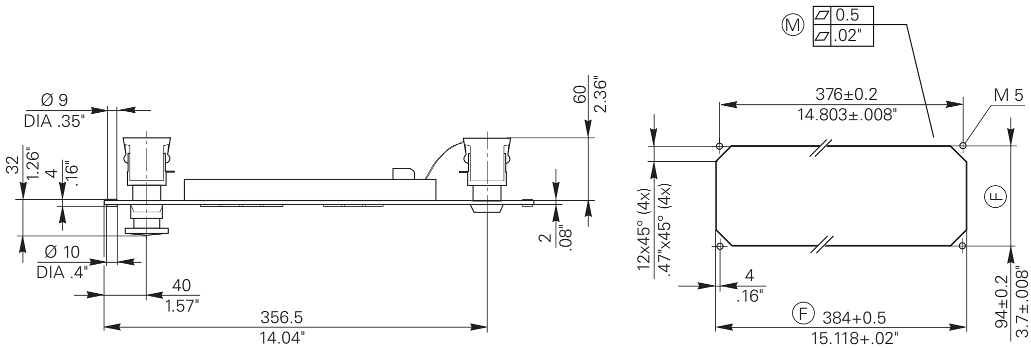


16.9 MB 410

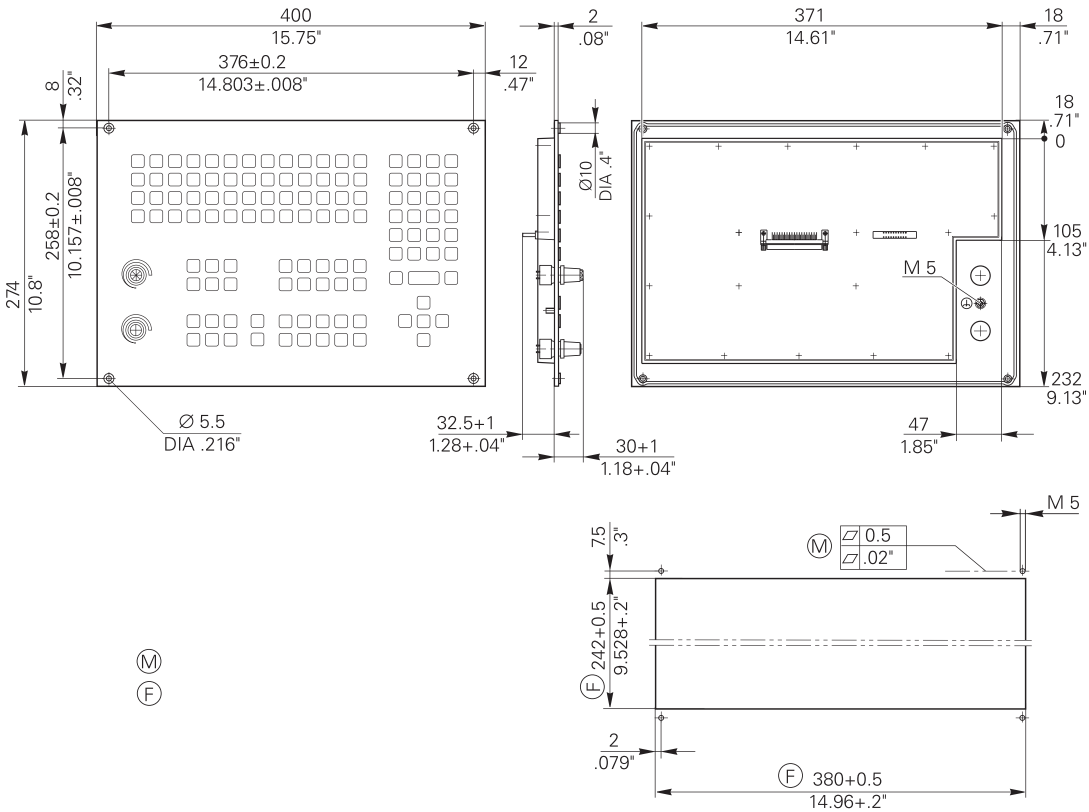

400
15.75"
6±0.2
.236±.008"
388±0.2
15.276±.008"
6±0.2
.236±.008"
110
4.33"
98±0.2
3.858±.008"
5.5
.217"
20
.787"
52
2.047"
9
.354"
86
3.386"
234
9.213"
75
2.953"
Ø 9
DIA .35"
32
1.26"
4
.16"
Ø 10
DIA .4"
40
1.57"
356.5
14.03"
60
2.36"
2
.08"
0.5 Montagefläche
.02" MOUNTING SURFACE
M 5
388±0.2
15.276+.008"
5–0.5
.2–.02"
98±0.2
3.858±.008"
Frontplattenausschnitt
FRONT PANEL OPENING
88+1
3.465+.04"
5–0.5
.2–.02"
378+1
14.88+.04"

16.10 MB 420

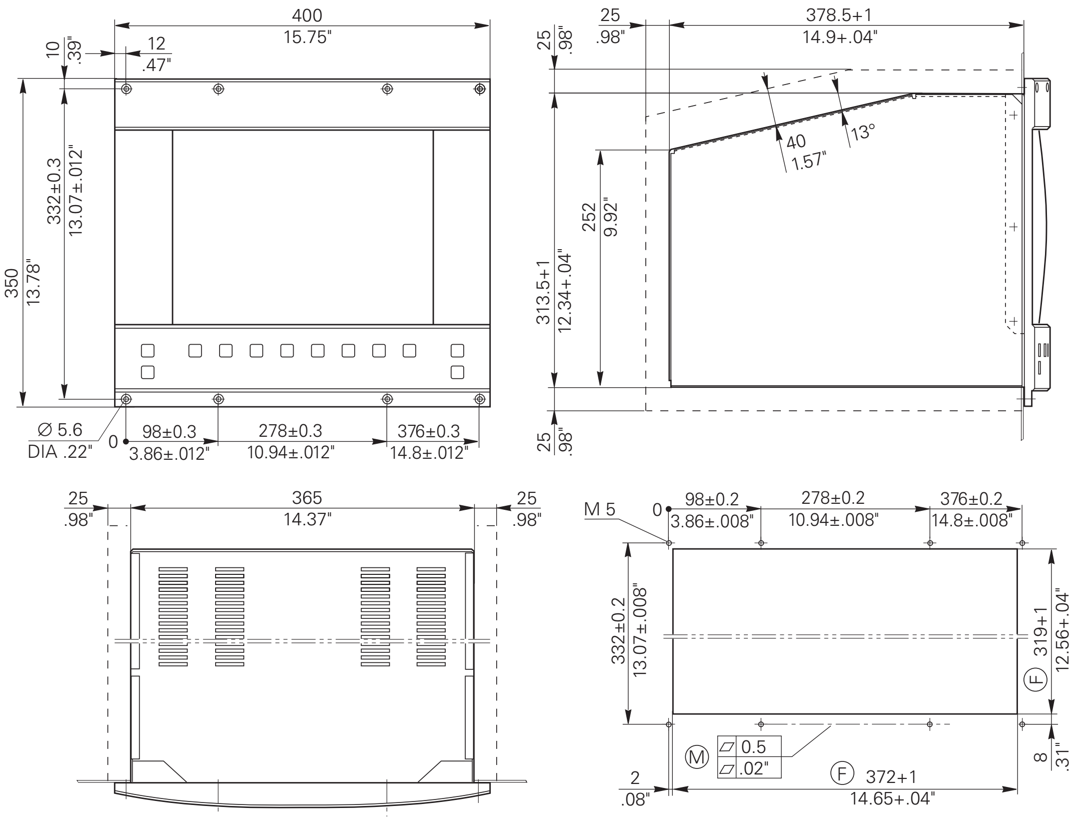

400
15.75"
12±0.2
.472±.008"
376±0.2
14.803±.008"
8±0.2
.315±.008"
110
4.33"
94±0.2
3.7±.008"
5.5
.217"
20
.8"
52
2.05"
9
.35"
86
3.39"
234
9.21"
75
2.95"


Ø 9
DIA .35"
32
1.26"
4
.16"
Ø 10
DIA .4"
40
1.57"
356.5
14.04"
60
2.36"
2
.08"
M
0.5
.02"
376±0.2
14.803±.008"
M 5
12x45° (4x)
.47"x45° (4x)
F
4
.16"
F
384+0.5
15.118+.02"
94±0.2
3.7±.008"

16.11 TE 400 B


400
15.75"
376±0.2
14.803±.008"
12
.47"
8
.32"
258±0.2
10.157±.008"
274
10.8"
Ø 5.5
DIA .216"
2
.08"
Ø10
DIA .4"
32.5+1
1.28+.04"
30+1
1.18+.04"
371
14.61"
18
.71"
18
.71"
0
105
4.13"
M 5
232
9.13"
47
1.85"
M 5
0.5
.02"
7.5
.3"
242+0.5
9.528+.2"
2
.079"
380+0.5
14.96+.2"
M
F

16.12 BC 120



400
15.75"
10
.39"
12
.47"
332±0.3
13.07±.012"
350
13.78"
Ø 5.6
DIA .22"
0
98±0.3
3.86±.012"
278±0.3
10.94±.012"
376±0.3
14.8±.012"
25
.98"
378.5+1
14.9+.04"
40
1.57"
13°
252
9.92"
313.5+1
12.34+.04"
365
14.37"
M 5
98±0.2
3.86±.008"
278±0.2
10.94±.008"
376±0.2
14.8±.008"
332±0.2
13.07±.008"
319+1
12.56+.04"
F
M
0.5
.02"
8
.31"
2
.08"
372+1
14.65+.04"

Machine integration — Contents

1 Machine axes

The HEIDENHAIN contouring controls TNC 407/TNC 415 permit the control of up to five machine axes and the main spindle (see also under "Servo positioning").

The machine parameter MP10 can be set to determine which axes should be operational on the machine. If necessary, MP10 can be used to select all the axes functions (control, display, pass over reference marks etc.)

MP10 Active axes
Entry range: %xxxxx

Bit 0	X axis	0 = not active
Bit 1	Y axis	1 = active
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

1.1 Measuring systems

Incremental measuring systems can be attached to the HEIDENHAIN contouring controls.
See also chapter "Mounting and electrical installation ".

1.1.1 Signal period

The signal period of the attached measuring system in µm or $\frac{1}{1000}^{\circ}$ is entered in the machine parameter MP330.x .

Linear measurement
For **linear measurement systems with sinusoidal output signals** the signal period is the same as the graduation period:

Signal period (~) = grating period

The standard-linear measurement systems from HEIDENHAIN have a graduation period of 20 µm (LS models; except for LS 101 and LS 405: 10 µm) and 100 µm (LB model) .
If linear measurement is performed by **rotary encoder and ballscrew,** the line count of the rotary encoder (see encoder technical data) as well as the ballscrew pitch must be considered when calculating the signal period:

$$\text{Signal period (\textasciitilde)} = \frac{\text{ballscrew pitch [mm]} \bullet \text{1000 [mm/mm]}}{\text{line count}}$$

Up to 3 decimal places can be entered in MP330.x.

For linear measurement systems in combination with the interpolation and digitising electronics EXE for the square-wave signal inputs X5 (TNC 407) and X6 (TNC 407/TNC 415) the interpolation factor of the EXE must be taken into account:

$$\text{Signal period } (\sqcap\!\sqcup) = \frac{\text{grating period}}{\text{interpolation factor}}$$

For linear measurement by **rotary encoder and ballscrew:**

$$\text{Signal period } (\sqcap\!\sqcup) = \text{ballscrew pitch [mm]} \bullet 1000\ [\mu\text{m/mm}] \quad \text{line count} \cdot \frac{1}{\text{interpolation factor}}$$

The TNC always does a 4-fold evaluation of the signals at the square-wave inputs.

If a counting step < 1 µm or $\frac{1}{1000}°$ is desired, the signal period ($\sqcap\!\sqcup$) must not be greater than 4 µm or $\frac{4}{1000}°$.

Only on the TNC 407 is it possible in MP340 to enter the interpolation factor of the external electronics (EXE) at the encoder input X5 or X6. The TNC needs this information in order to be able to determine the absolute position when encoders with distance-coded reference marks are used. The interpolation factor is entered individually for each axis.

Angular measurement: For angular measurement systems the signal period is calculated as follows:

$$\text{signal period } (\sim) = \frac{360°}{\text{line count}} \cdot 1000$$

or

$$\text{signal period } (\sqcap\!\sqcup) = \frac{360°}{\text{line count}} \cdot \frac{1}{\text{interpolation factor}} \cdot 1000$$

If the angular measurement is made by gearing up or down this must be taken into account when calculating the signal period.

MP330	Signal period
	Entry range 0.1 to 100 in [µm] or $\left[\frac{1°}{1000}\right]$
MP330.0	X axis
MP330.1	Y axis
MP330.2	Z axis
MP330.3	4th axis
MP330.4	5th axis

MP340	Interpolation factor of the EXE at X5, X6 (only TNC 407) Input: 0, 1, 5

0 = No encoder at X5, X6
or EXE without interpolation
1 = EXE without Interpolation
5 = EXE with 5-fold interpolation

MP340.0	X axis
MP340.1	Y axis
MP340.2	Z axis
MP340.3	4th axis
MP340.4	5th axis

1.1.2 Direction of traverse

The machine parameters MP210 and MP1040 determine the direction of traverse for the axes. The direction of traverse for the axes of numerically controlled machine tools are defined by DIN (see also under Sections "Axis designation" and "Commissioning and start-up procedure").

MP210 defines the counting direction for the measuring system signals. The counting direction depends on the mounting orientation of the measuring systems.

MP210	Counting direction of the measuring system signals Entry: %xxxxx

Bit 0	X axis	0 = positive
Bit 1	Y axis	1 = negative
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

Machine parameter MP1040 determines the polarity of the nominal value voltage during the positive direction of traverse.

MP1040	Polarity of the nominal value voltage for the positive direction of traverse Entry: %xxxxx

Bit 0	X axis	0 = positive
Bit 1	Y axis	1 = negative
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

The NC uses markers to tell the PLC in which direction the axes are travelling.

Marker	Function	Set	Reset
M2160 to M2164	Direction of traverse 0 = positive 1 = negative	NC	NC
M2160	X axis		
M2161	Y axis		
M2162	Z axis		
M2163	4th axis		
M2164	5th axis		

1.1.3 Measuring system monitoring

HEIDENHAIN contouring controls can monitor the signal transmissions of the measuring system. This measuring system monitoring must be activated by a machine parameter.
Three different conditions can be checked:

	Error message
The absolute position of distance-coded reference marks	C
The amplitude of the measuring system signals	A
The edge separation of the measuring system signals	B

If one of the conditions is not fulfilled, the error message "Measuring system <axis> defective A/B/C" will appear.

For sinusoidal signals the LE monitors the amplitude of the measuring system signals, for square-wave signals the LE evaluates the fault detection signal (U_{aS}) of the external electronics (EXE).

MP30 Checking the absolute position of the distance-coded reference marks
Entry: %xxxxx

Bit 0	X axis	0 = not active
Bit 1	Y axis	1 = active
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

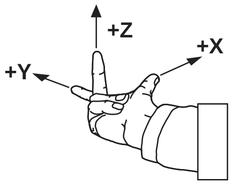
MP31	Checking the amplitude of the measuring system signals Entry: %xxxxxx	
Bit 0	X axis	0 = not active
Bit 1	Y axis	1 = active
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	
Bit 5	Axis S	

MP32	Checking the edge separation of the measuring system signals Entry: %xxxxxx	
Bit 0	X axis	0 = not active
Bit 1	Y axis	1 = active
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	
Bit 5	S-axis	

1.2 Axis designation

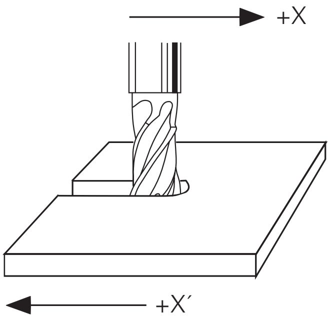
The coordinate axes and their directions of travel are standardized in ISO 841.

The directions of traverse may be simply determined by using the "Right-hand rule".



In the direction of the spindle axis the convention is:
The movement of the tool towards the workpiece is the negative direction of traverse. When creating an NC-program one proceeds as if the tool is moving and the workpiece is always stationary.

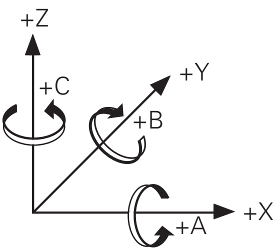
If the machine moves the workpiece, then the direction of movement and the direction of the axis are opposite to one another. The positive relative directions of movement are then designated +X', +Y' etc.



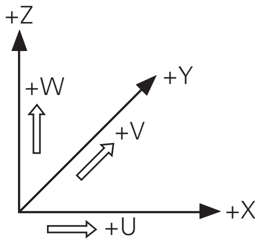
The fourth or fifth axis can be used either as an axis of rotation or alternatively as an extra linear axis.

While the three principal axes have the standard designations X, Y and Z, the designations of the fourth and fifth axes can be selected by a machine parameter.

An **axis of rotation** is designated by the letter A, B or C. The correlation with the principal axes and determination of the direction of rotation is standardized in ISO 841.



A **secondary linear axis** is designated by the letter U, V or W. The correlation with the principal axes and the direction of travel are also standardized in ISO 841.



MP410	Axis designation	
	Entry:	0 = A 1 = B 2 = C 3 = U 4 = V 5 = W
MP410.3		4th axis
MP410.4		5th axis

1.2.1 Assignment

The measuring system inputs X1 to X6 and the analogue outputs, Output 1 to Output S (on the connector X8) can be assigned to the individual axes. The assignment is determined by the machine parameters MP110 and MP120.

The assignment of measuring system input X6 and the nominal value output S to one of the five axes is only possible when the function Spindle-Orientation is not utilised.

MP110	Assignment of the measuring system inputs to the axes	
	Entry:	0 = measuring system input X1 1 = measuring system, encoder input X2 2 = measuring system, encoder input X3 3 = measuring system, encoder input X4 4 = measuring system, encoder input X5 5 = measuring system, encoder input X6

MP110.0 X axis
MP110.1 Y axis
MP110.2 Z axis
MP110.3 4th axis
MP110.4 5th axis

MP120	Assignment of the analogue outputs	
	Entry:	0 = output 1 1 = output 2 2 = output 3 3 = output 4 4 = output 5 5 = output S

MP120.0	X axis
MP120.1	Y axis
MP120.2	Z axis
MP120.3	4th axis
MP120.4	5th axis

1.2.2 Current tool axis

In the NC-block "TOOL CALL" it is determined whether the tool moves parallel to one of the principal axes X, Y, Z or parallel to the fourth axis. The fifth axis may not be defined as a tool axis. The markers M2100 to M2103 are used to show which of the four axes is currently defined as the tool axis. The appropriate marker is then set.

Marker	Function	Set	Reset
M2100	X-axis is tool axis	NC	NC
M2101	Y-axis is tool axis		
M2102	Z-axis is tool axis		
M2103	4th axis is tool axis		

1.3 VDU display

Machine parameters can be used to select which of the active axes (MP10) should be displayed in the status window.
The spindle position is displayed only when neither M03 nor M04 is active (see under "Main spindle").

MP40 VDU display
Entry: %xxxxx

Bit 0	X axis	0 = not active
Bit 1	Y axis	1 = active
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	
Bit 5	Axis S	

1.4 Traverse ranges

For all five axes, three different software traverse ranges can be defined by machine parameters (e.g. for pendulum machining). The traverse ranges are defined by so-called software limit switches.

The input values for the software limit switches are related to the machine datum (MP960.x). The momentary software limit switch range is selected by the markers (M2817, M2816) and activated by the strobe-marker (M2824).

The software limit switches for axes of rotation are not active unless a value of 0 is entered in machine parameter MP810 (see section "Display and operation").

The MOD-function "Axis-limit" can be used to enter an additional limitation for each traverse range.

MP910	Traverse ranges	
MP920	Entry range	
MP911	Linear axis:	- 99 999.9999 to + 99 999.9999 [mm]
MP921	Axis of rotation:	- 99 999.9999 to + 99 999.9999 [°]
MP912	(Values relative to the machine datum)	
MP922		

Traverse range 1
Initial values after Power-On;
Activated by PLC M2817 = 0, M2816 = 0

MP910.0	Software limit switch X+
MP910.1	Software limit switch Y+
MP910.2	Software limit switch Z+
MP910.3	Software limit switch 4+
MP910.4	Software limit switch 5+
MP920.0	Software limit switch X–
MP920.1	Software limit switch Y–
MP920.2	Software limit switch Z–
MP920.3	Software limit switch 4–
MP920.4	Software limit switch 5–

Traverse range 2
Activated by PLC M2817 = 0, M2816 = 1

MP911.0	Software limit switch X+
MP911.1	Software limit switch Y+
MP911.2	Software limit switch Z+
MP911.3	Software limit switch 4+
MP911.4	Software limit switch 5+

MP921.0	Software limit switch X–
MP921.1	Software limit switch Y–
MP921.2	Software limit switch Z–
MP921.3	Software limit switch 4–
MP921.4	Software limit switch 5–

Traverse range 3
Activated by PLC: M2817 = 1, M2816 = 0

MP912.0	Software limit switch X+
MP912.1	Software limit switch Y+
MP912.2	Software limit switch Z+
MP912.3	Software limit switch 4+
MP912.4	Software limit switch 5+

MP922.0	Software limit switch X–
MP922.1	Software limit switch Y–
MP922.2	Software limit switch Z–
MP922.3	Software limit switch 4–
MP922.4	Software limit switch 5–

MP7490 is used to select whether one or three traverse ranges can be defined with the MOD function. MP7490 is also used to select whether the datum applies for all traverse ranges or whether a separate datum can be set for each traverse range (see table below).

MP7490 Number of traverse ranges and datum points
Entry: 0 to 3

Entry	Number of traverse ranges	Number of datum points
0	1	3
1	3	3
2	1	1
3	3	1

Markers M2816 and M2817 are used to define the traverse range and/or datum point.

M2816	M2817	Traverse range/datum
0	0	Range 1
1	0	Range 2
0	1	Range 3

The change-over to the selected traverse range must be activated by the strobe-marker M2824 by the PLC. This strobe-marker is reset by the NC after the change-over has been carried out.

Marker	Function	Set	Reset
M2824	Activation of the selected traverse range (M2816/M2817)	PLC	NC

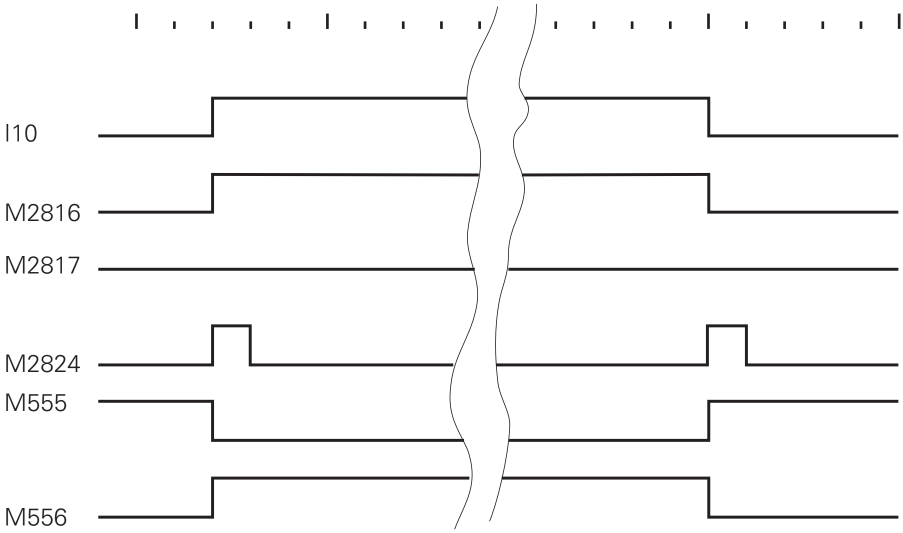
If one of the software limit switches is reached, the error message "LIMIT SWITCH ..." appears and the appropriate marker (M2624 to M2633) is set.

Marker	Function	Set	Reset
M2624	Limit switch X+	NC	NC
M2625	Limit switch X–		
M2626	Limit switch Y+		
M2627	Limit switch Y–		
M2628	Limit switch Z+		
M2629	Limit switch Z–		
M2630	Limit switch 4+		
M2631	Limit switch 4–		
M2632	Limit switch 5+		
M2633	Limit switch 5–		

Example:

PLC-program example of changing the traverse ranges. The PLC-input I10 is used as a condition for change.

```
I10 = 0         Traverse range 1
I10 = 1         Traverse range 2
.
127             LN I10          ;Traverse range 1
128             AN M555         ;already done?
129             R  M2816        ;select traverse range 1
130             R  M2817        ;select traverse range 1
131             S  M2824        ;activate change
132             S  M555         ;edge recognition traverse range1
133             R  M556         ;reset edge recognition traverse range2
134             L  I10          ;traverse range2
135             AN M556         ;already done?
136             S  M2816        ;select traverse range2
137             R  M2817        ;select traverse range2
138             S  M2824        ;activate change
139             S  M556         ;edge recognition traverse range2
140             R  M555         ;reset edge recognition traverse range1
.
```



1.5 Lubrication pulse

The PLC can control the lubrication of the guideway according to the distance travelled on each axis. In the machine parameter MP4060.X the distance after which lubrication should be performed is registered for each axis. The entry is in units of 65 536 µm.

Example:

Desired traversing distance: 100 m

$$\text{Value entered} = \frac{100\,000\,000\ \mu\text{m}}{65\,536\ \mu\text{m}} = 1\,526$$

If the stored path limit for an axis is exceeded, the NC sets a marker (M2012 to M2015, M2029) to "1".

After carrying out the lubrication the PLC must reset the accumulated traverse distance (M2548 to M2551, M2613).

MP4060	Path dependent lubrication Entry range: 0 to 65 535 (units of 65 536 µm)
MP4060.0	X axis
MP4060.1	Y axis
MP4060.2	Z axis
MP4060.3	4th axis
MP4060.4	5th axis

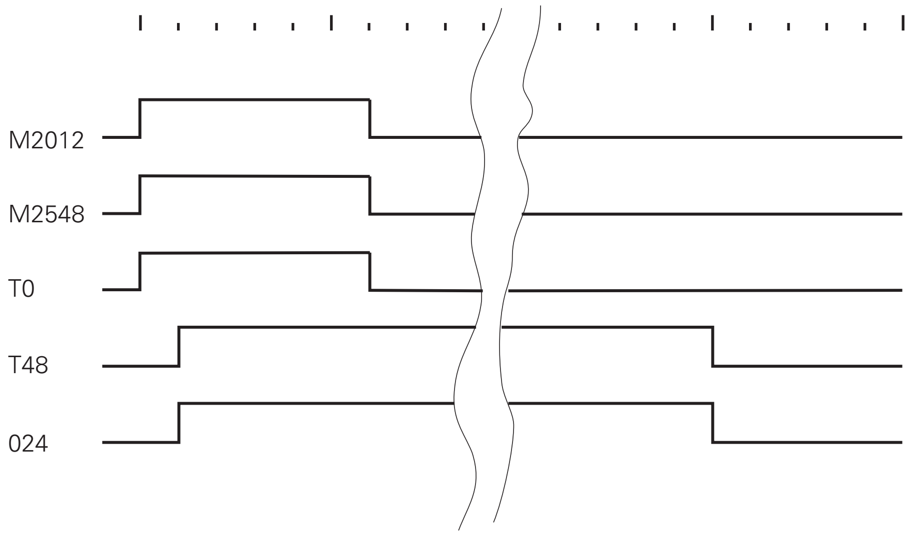
Marker	Function	Set	Reset
M2012	Lubrication pulse X axis, since value of MP4060.0 was exceeded	NC	NC
M2013	Lubrication pulse Y axis, since value of MP4060.1 was exceeded		
M2014	Lubrication pulse Z axis, since value of MP4060.2 was exceeded		
M2015	Lubrication pulse 4th axis, since value of MP4060.3 was exceeded		
M2029	Lubrication pulse 5th axis, since value of MP4060.4 was exceeded		
M2548	Reset of accumulated distance for lubrication X axis	PLC	PLC
M2549	Reset of accumulated distance for lubrication Y axis		
M2550	Reset of accumulated distance for lubrication Z axis		
M2551	Reset of accumulated distance for lubrication 4th axis		
M2613	Reset of accumulated distance for lubrication 5th axis		

Example:

PLC-program example of activating the lubrication for the X-axis.

In the machine parameter MP4060.0 the traverse distance after which the X-axis should be lubricated is entered. The duration of the lubrication is defined by the timer T0 (MP4110.0).
The PLC-output O24 is to be set for the duration of the X-axis lubrication.
In our example the lubrication is activated as soon as the marker M2012 is set. If , for some reason, the lubrication should only be activated when the axis is at rest, then this must be taken into account in the PLC-program.

```
MP4060.0 = 1 000 (approx. 65 m)
MP4110.0 =   100 (approx. 2 sec.).
.
.

45          L M2012        ;lubrication pulse X axis
46          = T0           ;start timer for duration of lubrication
47          = M2548        ;reset accumulated distance
48          L T48          ;duration of lubrication for X axis
49          = O24          ;set output for lubrication
.
.
.
```



1.6 Axis-error compensation

The HEIDENHAIN contouring control can compensate for mechanical defects in the machine.
The following axis-error compensation is possible:

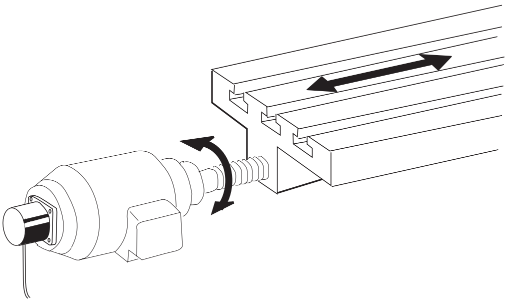
- backlash compensation,
- compensation of reversal errors in circular motion,
- linear axis-error compensation,
- non-linear axis-error compensation,
- compensation of thermal expansion.

Either the linear or the non-linear axis-error compensation can be activated. The other types of compensation can always be activated in parallel.

1.6.1 Backlash compensation

If linear traverse is measured by ballscrew and rotary encoder, a small amount of play between the movement of the table and that of the rotary encoder can result during reversals in traverse direction.

Depending on the design, the movement of the rotary encoder may be advanced or retarded relative to the table. The professional jargon describes this as positive or negative backlash. Backlash occurring outside the control loop can be automatically compensated by the TNC.



Positive backlash: rotary encoder advanced relative to the table (traverse movement of the table is too short).

Negative backlash: rotary encoder retarded relative to the table (traverse movement of the table is too long).

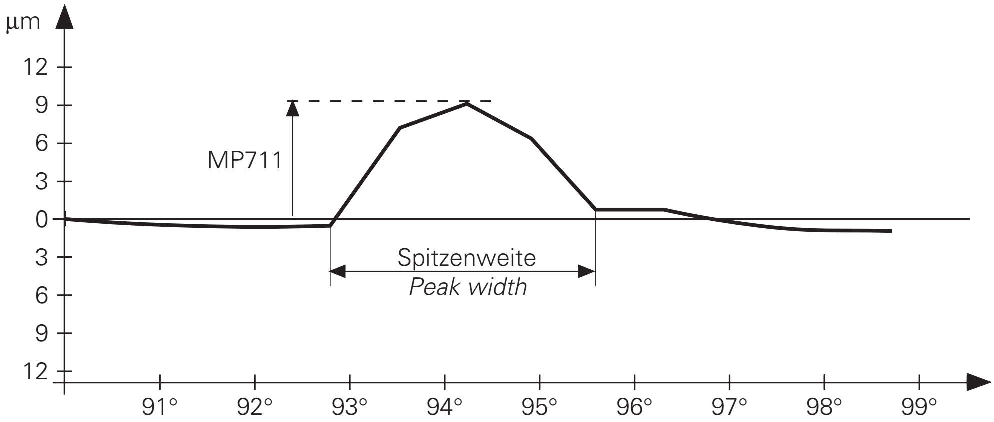
For every direction reversal the TNC subtracts or adds the value from machine parameter MP710.x to the value resulting from the measuring system signals.

MP710	Backlash compensation Entry: – 1.0000 to + 1.0000 [mm] or [°]
MP710.0	X axis
MP710.1	Y axis
MP710.2	Z axis
MP710.3	4th axis
MP710.4	5th axis

1.6.2 Compensation for reversal errors in circular motion

The stiction in the axis bearings can lead to reversal errors at the quadrant transitions during circular movements.

Commercially available devices such as the HEIDENHAIN Double Ball Bar system can diagnose these errors and determine the size and duration of their peaks.



The TNC can then compensate for these errors. The size of the reversal error which is to be compensated is entered in machine parameter MP711, and the portion of the peaks to be compensated per closed loop cycle time is entered in MP712.

The diagram from the DBB system shows the size and duration of a reversal peak.

The entry value for MP712 can be calculated as follows:

– Duration of the reversal peak

  $$t_{Pr}\,[s] = \frac{\text{Peak width}\,[°] \times 2\pi \times \text{Radius}\,[mm] \times 60}{360\,[°] \times \text{Feed rate}\,[mm/min]}$$

  the peak width is [°] indicated in the diagram;
  feed rate [mm/min] is the programmed contouring feed rate.

– Compensation per control loop cycle time (3 ms with TNC 415, 6 ms with TNC 407)

  $$\text{Compensation}\,[mm] = \frac{\text{Reversal peaks}\,[mm] \cdot \text{Control loop cycle time}\,[s]}{0.5 \cdot t_{Pr}\,[s]} \cdot 10^{-3}$$

  The compensation value is entered in MP712.

If the computed values have no effect this is because the machine dynamics are too weak.

MP711	Size of the reversal peaks in circular motion Entry: -1.0000 to +1.0000 [mm]
MP711.0	X axis
MP711.1	Y axis
MP711.2	Z axis
MP711.3	4th axis
MP711.4	5th axis

MP712	Compensation value per cycle time Entry: –0.000000 to 99.999999 (mm)
MP712.0	X axis
MP712.1	Y axis
MP712.2	Z axis
MP712.3	4th axis
MP712.4	5th axis

A second block of Kv factors can be activated with M function M105 (M106 deactivates).

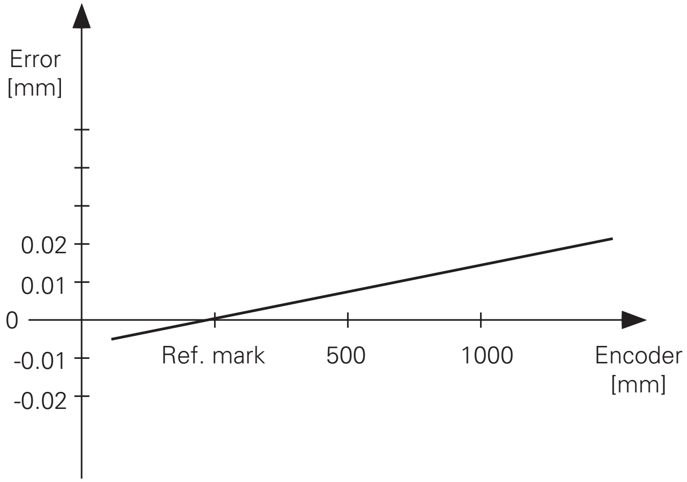
A second block of machine parameters for reversal peak compensation is also activated.
MP715 (as MP711) and MP716 (as MP712)

MP715	Height of reversal peaks in circular movements (M105) Entry: –1.0000 to +1.0000 (mm)
MP715.0	X axis
MP715.1	Y axis
MP715.2	Z axis
MP715.3	4th axis
MP715.4	5th axis
MP716	Compensation value per control loop cycle time (M105) Entry: 0.000000 to 99.999999 [mm]
MP716.0	X axis
MP716.1	Y axis
MP716.2	Z axis
MP716.3	4th axis
MP716.4	5th axis

1.6.3 Linear axis-error compensation

One linear axis-error can be compensated per axis. The axis-error is entered, with the correct sign, in machine parameter MP720. The error is positive if the table travel is too long, and negative if the travel is too short.

MP730 set the axis error compensation to linear or non-linear. Linear axis error compensation is not active for rotary axes.

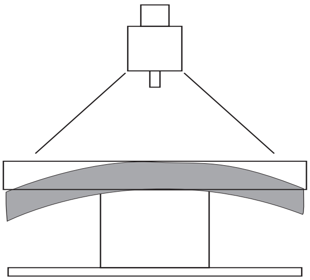


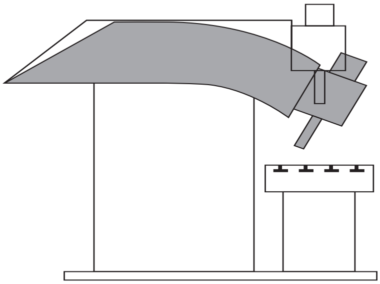
MP720	Linear axis-error compensation Entry: – 1.000 to + 1.000 [mm/m] or [1°/1000°]		
MP720.0	X axis		
MP720.1	Y axis		
MP720.2	Z axis		
MP720.3	4th axis		
MP720.4	5th axis		
MP730	Selection of linear or non-linear axis-error compensation Input: %xxxxx		
Bit 0	Axis X	0 =	Linear axis error compensation
Bit 1	Axis Y	1 =	Non-linear axis error compensation
Bit 2	Axis Z		
Bit 3	4th axis		
Bit 4	5th axis		

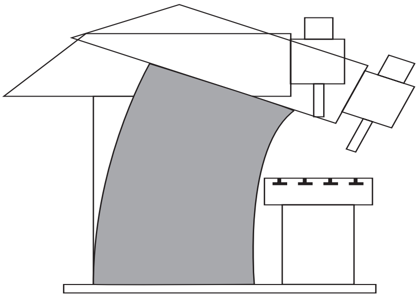
1.6.4 Non-linear axis error compensation

Depending on the design of the machine or external factors (e.g. temperature) a non-linear axis-error can occur.

Such an axis-error is usually determined by a comparator measuring instrument (e.g. HEIDENHAIN VM 101).







For example, the lead-screw pitch error for the Z axis (Z=F(Z)) or the sag as a function of the Y axis (Z=F(Y)) could be determined.

The TNC can compensate ballscrew pitch error and sag at the same time. A correction table (file extension .COM) is created for each axis in the "PLC Programming" mode (code number 80 76 67). A number of dependencies can be entered in a correction table. The number of possible compensations is limited to ten dependencies and a total of 640 points (256 points per dependency).

The following items must be defined in the table headline.

```
DATEI: ACHSE-Y                 MM      DATUM:-90          DIST:16 X Y Z U V
```

Filename	Dimensions in mm	Datum	Compensation point distance	Dependencies

The datum is the distance from the machine datum (MP960.X). The distance of the compensation points is entered as an exponent to the base 2 (e.g. enter 16 = 2^{16} = 6.5536 mm). Maximum input value is 2^{23}.

The datum and the selected distance of the compensation points must be allowed for when plotting the error curve. Only enter the kinks on the error curve. The controller performs automatic linear interpolation between the kinks.

Each axis is assigned a correction table (.COM) in the .CMA file. A number of lines with different assignments can be entered in this file. Only one line can be active at any one time. The active line is selected by soft key or with PLC module 9095.

Non-linear axis error compensation is not active until the function is activated by MP730 and there is a valid file of the .CMA type.

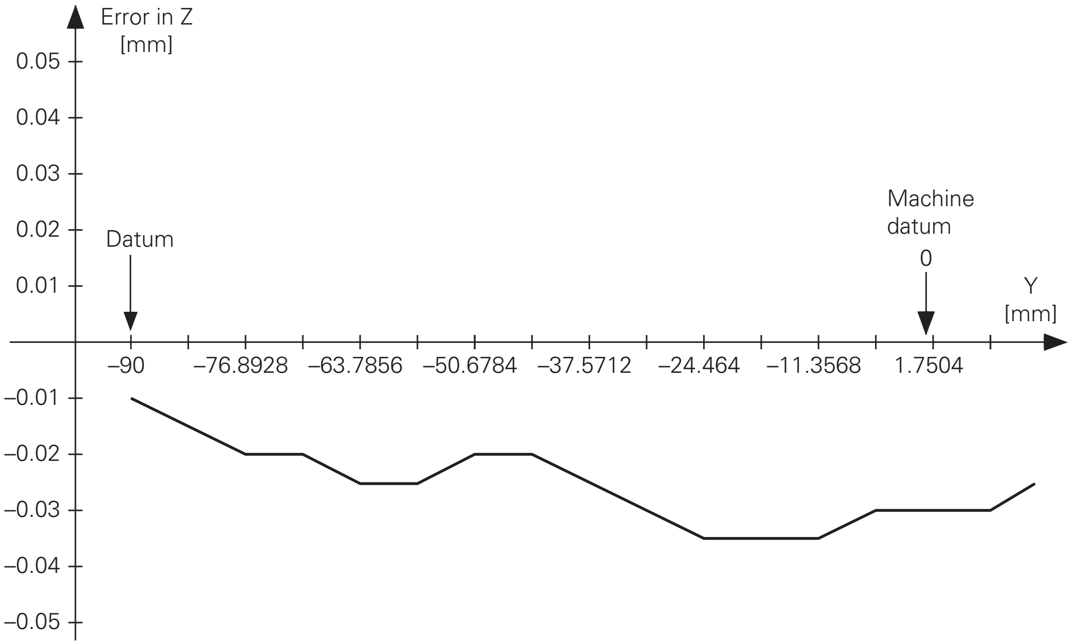
Example:

Machine with leadscrew pitch error in Z (Z = F(Z)) and Y (Y = F(Y)).
Also sag as a function of Y (Z = F(Y)). There is no compensation on the X axis.
Traverse range Z = 800 mm
Traverse range Y = 500 mm
required distance of compensation points = 7 mm
possible power of two = 2^{16} = 6.5536 mm

$$\text{Number of compensation points in Y} = \frac{500\text{ mm}}{6.5536\text{ mm}}\ 77$$

$$\text{Number of compensation points in Z} = \frac{800\text{ mm}}{6.5536\text{ mm}}\ 123$$

Datum in Y = –90
Datum in Z = –200



The sag error (Z = F(Y)) and the leadscrew pitch error (Y = F(Y)) are both entered in file AXIS-Y.COM.

MANUAL OPERATION

COMPENSATION VALUE TABLE
COMPENSATION VALUE ?

FILE: ACHSE-Y MM DATUM:-90 DIST:16 X Y Z U V

NR		Y=F( )	Z=F( )
0	-90	+0	-0,01
1	-83,4464		
2	-76,8928	+0,005	-0,02
3	-70,3392		-0,02
4	-63,7856		-0,025
5	-57,232		-0,025
6	-50,6784		-0,02
7	-44,1248		-0,02
8	-37,5712		-0,025
9	-31,0176	+0,01	
10	-24,464		-0,035
11	-17,9104		
12	-11,3568		-0,035

BEGIN TABLE	END TABLE	PAGE ⇩	PAGE ⇧	INSERT LINE	DELETE LINE	NEXT LINE	HEAD LINE

The ballscrew pitch error in Z (Z = F(Z)) is entered in file AXIS-Z.COM.

MANUAL OPERATION

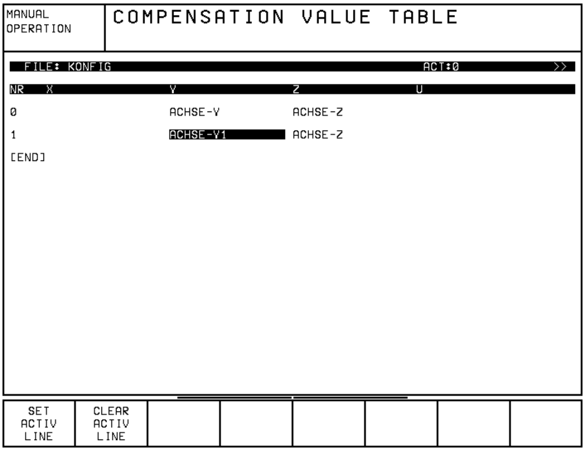
COMPENSATION VALUE TABLE
COMPENSATION VALUE ?

FILE: ACHSE-Z MM DATUM:-200 DIST:16 X Y Z U V

NR		Z=F( )
0	-200	+0
1	-193,4464	
2	-186,8928	
3	-180,3392	
4	-173,7856	
5	-167,232	+0,005
6	-160,6784	
7	-154,1248	+0,006
8	-147,5712	+0,007
9	-141,0176	
10	-134,464	
11	-127,9104	+0,01
12	-121,3568	

BEGIN TABLE	END TABLE	PAGE ⇩	PAGE ⇧	INSERT LINE	DELETE LINE	NEXT LINE	HEAD LINE

Then the files AXIS-Y.COM and AXIS-Z.COM are assigned to the Y and Z axis respectively in a file of the .CMA type (e.g. CONFIG.CMA).



A number of different assignments can be entered in the CONFIG.CMA file if required. Only one line can be active at any one time. The active line is selected by soft key. In our example this must be line 0.

Input and output of the correction tables via the data interface

The .CMA and .COM files can be uploaded and downloaded via the data interface in the "PLC Programming" mode. The .CMA file is assigned the extension .S and the .COM file the extension .V.

An axis of rotation is a special case

With an axis of rotation, the system only recognizes corrections of entries from 0 to 360.

Correction table with code number 105 296

In the interest of compatibility we have retained the entry of correction tables with code number 105 296. This was the only possible way of entering the correction tables up to software issue 250 94X 07 (243 03X 07).

The following constraints apply:

Only one dependency per axis can be programmed

$$\text{Max. correction difference} = \frac{\text{Comp. point spacing}}{64}$$

Always 64 compensation points per axis

If no .CMA file has been defined and non-linear axis error compensation is selected with MP730, then the correction tables from code number 105 296 apply.

MP730	Selection of linear or non-linear axis error compensation Entry: %xxxxx	
Bit 0	X axis	0 = Linear axis error compensation
Bit 1	Y axis	1 = Non-linear axis error compensation
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

1.6.5 Temperature compensation

Exact measurements of machine thermal behaviour (centre of expansion in the axes, amount of expansion as a function of temperature) are necessary to compensate the effects of thermal expansion.

The temperature measured through the PA 110 and Pt 100 is filed in PLC words W504 to W510 or W472 to W478 (see Section "Analog inputs"). The thermal expansion is largely proportional to the temperature value: you can directly determine the amount of expansion by multiplying the temperature value by a certain factor. This calculated expansion value must be transferred through the PLC program to the PLC word W576 to W584. As soon as the words W576 to W584 receive a value, the "lag tracking" becomes active.

"Lag tracking" means that the actual machine position changes by a certain value per PLC cycle until it has changed by the full value from the words W576 to W584. This does not change the value in the actual position display. The increment of change per PLC cycle must be defined in MP4070.

W576 to W584	Lag tracking error compensation (Compensation speed from MP4070) Input range: +32 767 to –32 768 [Error! µm]
W576	Axis X
W578	Axis Y
W580	Axis Z
W582	4th axis
W584	5th axis

MP4070	Compensation per PLC cycle for lag tracking error compensation Entry: 0.000 1 to 0.005 [mm]

Example:

The temperature value transmitted through the PA110 to the word W506 is multiplied by a compensation factor (linear expansion is assumed) and is sent to the word W578 (Y axis) for "lag tracking". The correction factor, which can only be determined through exact measurements of the machine, is entered, for example, in MP4210.44.

```
.
.
.
L       D944        ;correction factor from MP4210.44
X       W506        ;correction factor × temperature value from PA 110 (X8) = current thermal
                    expansion
=       W578        ;value for lag tracking of the Y axis
.
.
.
```

1.6.6 Stiction

Guideways with a relatively high level of stiction can cause servo lag at low feed-rates, even when using feed precontrol. A servo lag can be detected using, for example the TNC's integral oscilloscope, and the TNC can also compensate the error. This is done by entering an axis-specific factor to compensate the stiction in machine parameter 1511 (guideline factor: 5 000 to 10 000). A higher nominal value based on this factor is then output while traversing.

$$F_{add} = \frac{\Delta\ s_a}{t_C} \times k_v \times MP1511$$

F_{add} = Additional feed rate $[\frac{m}{min}]$

Δs_a = Servo lag difference after one control loop cycle [mm]

t_C = Control loop cycle time [µs]

k_v = Position loop gain $[\frac{m/min}{mm}]$

MP1511 = Factor for stiction compensation [µs]

$$V_{add} = \frac{MP1050}{MP1010} \times F_{add}$$

V_{add} = Additional analog voltage [V]

MP1050 = Analog voltage for rapid traverse [V]

MP1010 = Rapid traverse $[\frac{m}{min}]$

This increase in nominal value is limited by MP1512. If this limit is set too high then the machine will oscillate at standstill (guideline: < 50). MP 1512 limits the value Δs_a in the above formula.

$$MP1512 = \frac{\Delta s_{alimit} \times 256}{GP}$$

MP1512 =	Limit to the amount of stiction compensation [counting steps]
Δs_{alimit} =	Limit value for Δs_a [µm]
GP =	Grating period of the measuring system [µm]

Compensation may only be active at low feed rates as the increased nominal value would cause oscillation at high speed. The feed-rate limit for stiction compensation is defined in MP1513.

Stiction compensation is effective only during operation with feed precontrol. If stiction compensation is to be in effect in the manual operating modes as well, feed precontrol (MP1391) must be activated for manual operation in each axis.

MP1511	Factor for stiction compensation Entry: 0 to 16 777 215 [µs]
MP1511.0	X axis
MP1511.1	Y axis
MP1511.2	Z axis
MP1511.3	4th axis
MP1511.4	5th axis

MP1512	Limit of extent of stiction compensation Entry: 0 to 16 777 215 (counting steps)
MP1512.0	X axis
MP1512.1	Y axis
MP1512.2	Z axis
MP1512.3	4th axis
MP1512.4	5th axis

MP1513	Feed-rate limit for stiction compensation Entry: 0 to 300 000 (mm/min)
MP1513.0	X axis
MP1513.1	Y axis
MP1513.2	Z axis
MP1513.3	4th axis
MP1513.4	5th axis

MP1391 Feed precontrol in all operating modes
Entry: %xxxxx

Bit 0	X axis	0 =	Feed precontrol in the "Positioning with MDI", "Program run, single block", and "Program run, full sequence" operating modes
Bit 1	Y axis		
Bit 2	Z axis		
Bit 3	4th axis	1 =	Feed precontrol in all operating modes
Bit 4	5th axis		

1.7 PLC positioning

The five axes of the control can also be positioned by the PLC. The spindle too can be positioned with the aid of the PLC (see under "Main spindle"). The positions of the individual axes must be stored as Doublewords (D528 to D544) before activating the positioning.

The feed for positioning the individual axes is stored in W560 to W568. The transfer of the positions and the feed to the PLC is carried out, for example, by Q-parameters or machine parameters (MP4210.X, MP4220.X).

Five of 5 axes (3 of 5 axes on TNC 407) can be traversed simultaneously [simultaneous activation of all the strobe-markers (M2704 to M2708)].

The PLC positioning can be interrupted by resetting the strobe-marker (M2704 to M2708).
The NC resets the strobe-markers (M2704 to 2708) if:
- the axes have reached the given position
- there is an NC STOP in the manual or handwheel mode
- NC STOP and internal stop in the automatic operating modes
- EMERGENCY STOP is output
- an error message is released that results in a stop

Strobe markers M2704 to M2708 may only be set when no program has started or when an M/S/T strobe is present. In "Manual" mode the strobe markers may only be set when no axis direction key has been pressed.

The Doublewords D528 to D544 have a multiple usage. They have the following meaning for the PLC positioning:

Address	Function
D528	Position X axis [1/ 10 000 mm]
D532	Position Y axis
D536	Position Z axis
D540	Position 4th axis
D544	Position 5th axis

Feed for PLC-positioning

W560	Feed X axis [mm/min]
W562	Feed Y axis
W564	Feed Z axis
W566	Feed 4th axis
W568	Feed 5th axis

Marker	Function	Set	Reset
M2704	Activate PLC-positioning X axis	PLC	NC; PLC
M2705	Activate PLC-positioning Y axis		
M2706	Activate PLC-positioning Z axis		
M2707	Activate PLC-positioning 4th axis		
M2708	Activate PLC-positioning 5th axis		

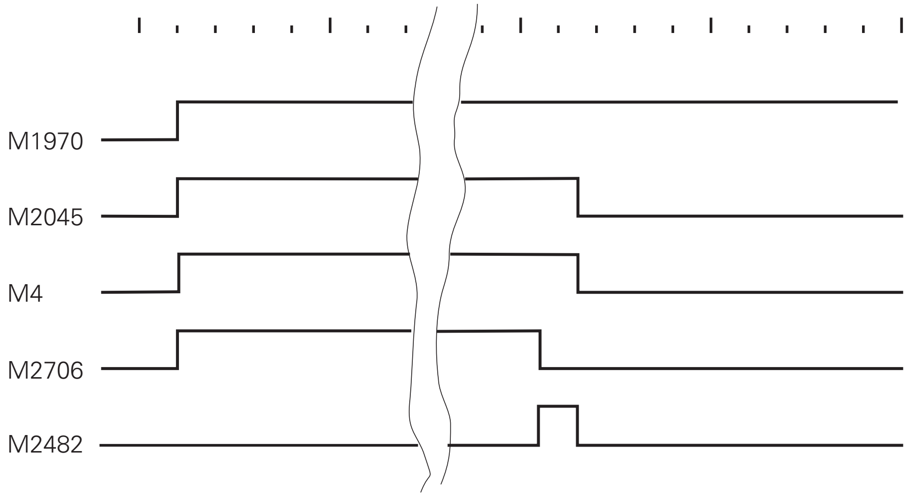
Note:

- The positions which are given are relative to the machine datum (MP960.x).
- Software limit switches are not considered.
- Tool compensations are not calculated.
- The path compensation must be terminated before a PLC positioning.
- PLC positioning is not displayed in the test graphics.

Example: PLC positioning of the Z-axis

A PLC positioning in the Z-axis is to be initiated with the M-function M70. The target position is stored in the machine parameter MP4210.2. The feed for the PLC positioning is defined in machine parameter MP4220.2

```
.
.
.
L  M0
ON M0
S  M2496      ;decoded M-code output (M1900 to M1999)
S  M2719      ;Word processing (incompatible with TNC 155/355)
L  M1970      ;M-function M70?
A  M2045      ;change signal M-function?
AN  M4        ;edge marker
CMT 110       ;load position and feed
S  M2706      ;activate PLC positioning Z axis
S  M4         ;edge marker
L  M1970      ;M-function M70?
A  M2045      ;change signal M-function?
AN M2706      ;PLC positioning Z axis done
S  M2482      ;acknowledgement M-function done
LN M2045      ;no M-function?
R  M2482      ;acknowledgement reset
R  M4         ;edge marker reset
.
.
.
EM            ;main program end
LBL 110       ;load Z position and feed rate
L  D776       ;load target position from MP4210.2
=  D536       ;target position PLC positioning Z axis
L  W964       ;load feed from MP4220.2
=  W564       ;feed PLC positioning Z axis
EM
```


M1970
M2045
M4
M2706
M2482

1.8 PLC axes

The controlled axes can be assigned to the PLC individually. On a machine with four NC axes for example, the 5th axis could be used to position the tool changer independently of the four NC axes. Machine parameter MP60 defines the axes to be controlled by the PLC.
In the PLC program the positioning of the PLC axes is controlled by modules:

Module 9120: Start positioning a PLC axis
Module 9121: Stop positioning a PLC axis
Module 9122: Interrogate status of a PLC axis
Module 9123: Traverse the reference marks of a PLC axis

These modules are described in the chapter "PLC Programming".
PLC axes are always traversed in the servo lag mode.
Several PLC axes can be started simultaneously. They are not interpolated together however.
A modulo value for the counting mode of the axes can be entered in MP810 (see section "Display and operation").

MP60 PLC auxiliary axes
Input: %xxxxx

Bit 0	Axis X	0 = NC axis
Bit 1	Axis Y	1 = PLC axis
Bit 2	Axis Z	
Bit 3	4th axis	
Bit 4	5th axis	

Example:
A tool sequence is to be pre-positioned with TOOL DEF. The tool sequence is driven by a controlled axis. The acknowledgement for the TOOL DEF block is given when positioning starts.
The feed-rate for positioning with TOOL DEF is filed in MP 4210.0 and the feed-rate for reference traversing the tool sequence is filed in MP 4210.1.

```
Machine parameters:
      MP 10     :%1XXXX   ;activate 5th axis
      MP 60     :%1XXXX   ;declare 5th axis
      MP 410.4  :   2     ;declare 5th axis as axis of rotation C
      MP 810.4  :  360    ;modulo value of tool sequence
      MP 960.4  :  XX     ;datum shift
      MP 7261   :  XX     ;number of tools with pocket number
      MP 7480.1 : 1 to 3  ;output tool and pocket no. with TOOL DEF
      MP 4210.0 :  XX     ;(D768) feedrate of PLC axis with TOOL DEF
      MP 4210.1 :  XX     ;(D772) feedrate of PLC axis with
                          reference traverse
Auxiliary addresses:
      D 168               ;desired position
      W 230               ;number of tools in sequence (=MP 7261)
      D 232               ;length of sequence (=MP 810)
      B 236               ;status of PLC axis
      B 238               ;Identifier Submit job
      M 544               ;reference traverse
      M 599               ;logic one
```

```
; Main program

   L       M2185              ;1ST PLC SCAN AFTER
                               PLC INTERRUPT
   O       M2180              ;1ST PLC SCAN AFTER POWER ON
   S       M544               ;ACTIVATE REF TRAVERSE
   L       M544
   O       M2047              ;TOOL DEF
   CMT     40
LBL     11                    ;END OF MODULE
   EM                         ;END OF MODULE

LBL     40
   L       M544               ;REF TRAVERSE ?
   CMT     41                 ;READ MP
   L       M544               ;REF TRAVERSE ?
   CMT     42
   L       M2047              ;STROBE TOOL DEF
   JPF     11                 ;END OF MODULE
   PS      K+4                ;5TH AXIS
   CM      9122               ;POLL STATUS
   PL      B236               ;STATUS
   L       B236
   BT      K+2                ;POSITIONING RUNNING
   JPT     11                 ;END OF MODULE


LBL     43                    ;POSITION CALCULATION
   L       D232               ;LENGTH OF SEQUENCE
   /[
   L       W230               ;NUMBER OF TOOLS IN SEQUENCE
   -       K+1
   ]
   X[
   L       W262               ;POCKET NUMBER
   -       K+1
   ]
   =       D168               ;POSITION

 ; POSITION SEQUENCE

   PS      K+4                ;AXIS
   PS      D168               ;POSITION
   PS      D768               ;FEED-RATE
   PS      K+0                ;ABSOLUTE
   CM      9120               ;START POSITIONING
   L       M599               ;LOG ONE
   S       M2484              ;TOOL DEF ACKNOWLEDGE
   EM
```

```
; READ MACHINE PARAMETERS

LBL     41
   RPLY    B238
   ==      K+0                    ;POLL ALREADY RUNNING
   JPF     11                     ;END OF MODULE
   SUBM    45                     ;READ MP
   =       B238                   ;STORE IDENTIFIER
   EM

LBL     45
   PS      K+810                  ;MP NUMBER (MODULO VALUE)
   PS      K+4                    ;MP INDEX
   CM      9032                   ;CALL MODULE
   PL      D232                   ;STORE (SEQUENCE LENGTH)
   PS      K+7261                 ;MP NUMBER (NUMBER OF TOOLS E
                                   WITH POCKET NUMBER
   PS      K+0                    ;INDEX
   CM      9032                   ;CALL MODULE
   PL      W230                   ;STORE (NUMBER OF
                                   TOOLS WITH POCKET NUMBER)F
   EM


  ; APPROACH REFERENCE MARKS FOR PLC AXIS

LBL     42                        ;
   PS      K+4                    ;PLC AXIS
   PS      D772                   ;FEED-RATE
   PS      K+0                    ;POSITIVE TRAVERSE DIRECTION
   CM      9123                   ;START REF TRAVERSE
   L       M599                   ;LOG ONE
   R       M544                   ;REF TRAVERSE OF PLC AXIS
   EM
```

1.9 Swivel axes

This function is not available in TNC 407.

Swivelling heads and/or swivelling tables are used for multi-faced machining on milling machines. The NC programs are either generated by a CAD system or are created direct on the TNC using the "Tilt working plane" function.

1.9.1 "Tilt working plane" function

The swivel axes are moved either manually or NC controlled.

The position of the swivel axis is defined in Cycle 19 "Tilt working plane" (e.g. A-45, B-45). When Cycle 19 is activated the TNC performs a coordinate transformation. The Z axis stays parallel to the tool axis while the X/Y plane is perpendicular to the direction vector of the tool axis. The swivel axes must first be positioned manually or with an appropriate NC block.

The status window displays the swivelled coordinate system. The machining plane can be transformed spatially at random with this procedure. However the NC program is still programmed in the normal way in the X/Y plane. The NC automatically interpolates the affected axes as it runs the program. All path functions and cycles such as "Set datum" and "Probe" can also be used in the transformed machining plane. The combination of coordinate transformation cycles must match the activate and deactivate sequence.

Tool radius compensation in the machining plane and tool length compensation parallel to the tool axis is active. The "3D ROT" soft key can be used to activate the swivelled machining plane for "Manual" and "Program Run" separately. The position of the swivel axes can be edited for manual operation.

The mechanical offset between the swivel axes is entered in machine parameters. The machine parameters that describe the machine geometry are divided into nine blocks, each block describing either a paraxial shift or a rotation.

With swivel heads the starting point is the tool datum of the machine (usually the spindle head). The shift or rotation to the next nearest swivel axis is then entered one at a time. This operation is repeated until a point is reached that is not separated from the machine by a free axis of rotation. The swivel head must be in its datum position when the shifts are being calculated (e.g. A = 0, B = 0).

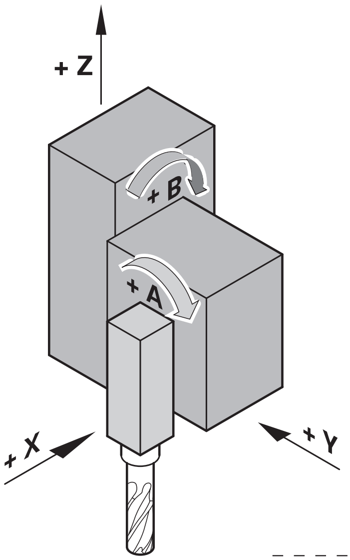
For tilting tables, the machine geometry is described starting from the centre of rotation of the first swivel axis (as seen from the workpiece) and not from tool datum. First the centre of rotation is defined in absolute coordinates — in relation to the machine datum — then the shifts and rotations are entered one at a time until a point is reached that is not separated from the machine by a free axis of rotation.

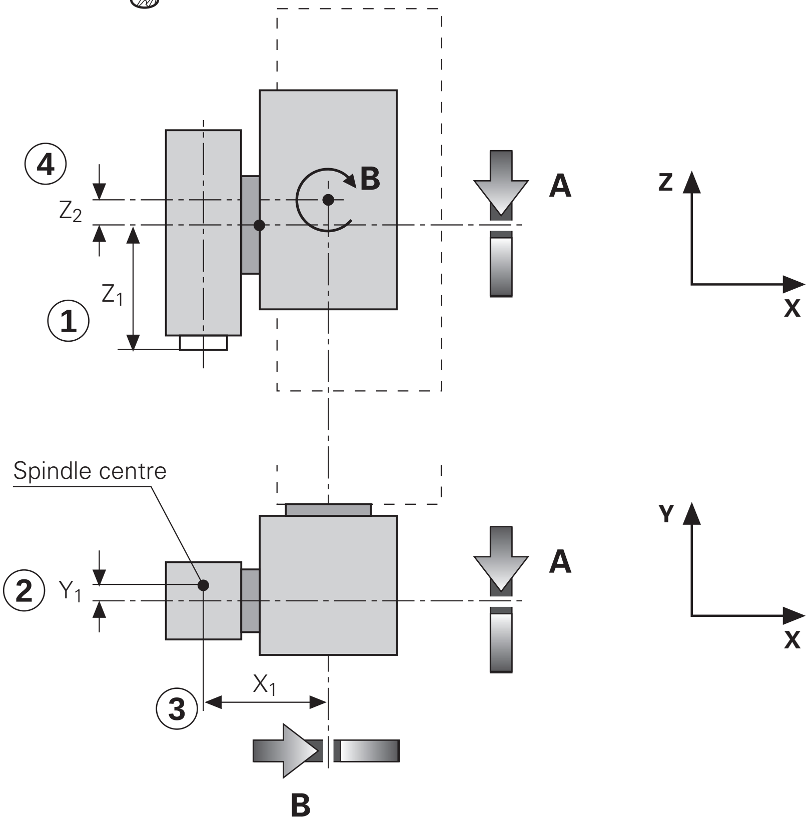
When machining with tilting tables, the coordinate system stays parallel to the machine coordinate system. The "Tilt working plane" function is enabled with MP7500. The descriptions in MP7510 to MP7592 are also used for other functions (e.g. "Cylinder interpolation").

- PLC positioning with M91 or M92 is not possible when Cycle 19 "Tilt working plane" is active
- The "Basic rotation" feature can be executed only when the tilt axes are at their 0° positions
- PLC positioning commands are always executed paraxially to the machine coordinate system. Cycle 19 therefore has no influence on PLC positioning commands.
- Datum compensation via PLC is effective also with the "Tilt working plane" function.

MP7500	"Tilt working plane" function Entry: 0 or 1 0 = inactive 1 = active	
MP75x0	Transformed axis Entry: %xxxxxx Entry 0 means end of transformation sequence	
Bit 0	X axis	
Bit 1	Y axis	
Bit 2	Z axis	
Bit 3	A axis	
Bit 4	B axis	
Bit 5	C axis	
MP75x1	Supplementary identifier for transformation Entry: %xx	
Bit 0	Swivel axis	0 = swivel head 1 = tilting table
Bit 1	Dimension in MP75x2	0 = incremental step (for swivel head) 1 = absolute related to machine datum (for tilting table)
MP75x2	Dimension for transformation Entry: –99,999.9999 to +99,999.9999 Entry 0 means free rotating axis	

Example 1: Double swivel head, right-angled

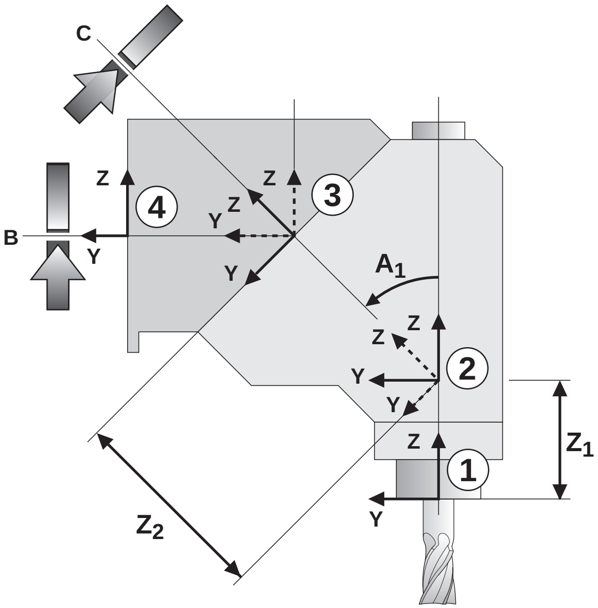




Z_1 = 200.4 mm
Z_2 = 3.1 mm
X_1 = 201.5 mm
Y_1 = 1.9 mm

```
MP 7500 : +1                ;activate function

MP 7510 : %000100           ;shift in Z axis
MP 7511 : %00               ;swivel head
MP 7512 : +200.4            ;dimension Z1

MP 7520 : %000010           ;shift in Y axis
MP 7521 : %00               ;swivel head
MP 7522 : –1.9              ;dimension Y1

MP 7530 : %001000           ;free swivel axis A
MP 7531 : %00               ;swivel head
MP 7532 : +0                ;variable dimension

MP 7540 : %000001           ;shift in X axis
MP 7541 : %00               ;swivel head
MP 7542 : +201.5            ;dimension X1

MP 7550 : %000100           ;shift in Z axis
MP 7551 : %00               ;swivel head
MP 7552 : +3.1              ;dimension Z2

MP 7560 : %010000           ;free swivel axis B
MP 7561 : %00               ;swivel head
MP 7562 : +0                ;variable dimension

MP 7570 : %000000           ;end of transformation sequence
```

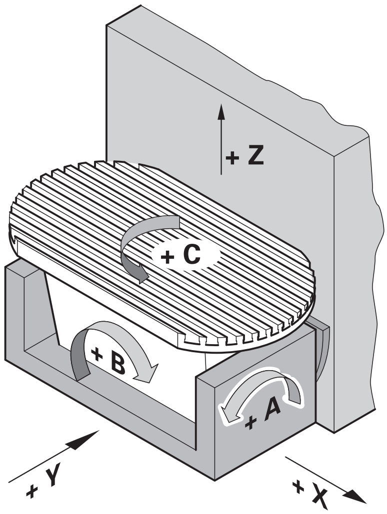
Example 2: Double swivel head, 45°

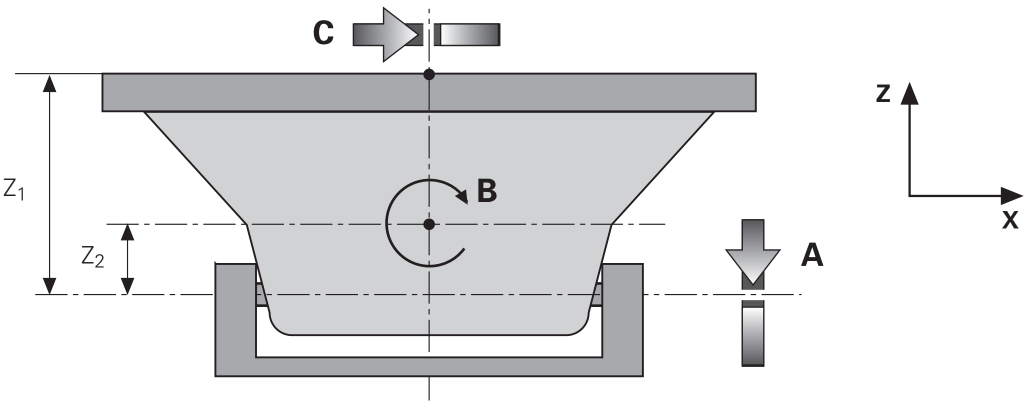


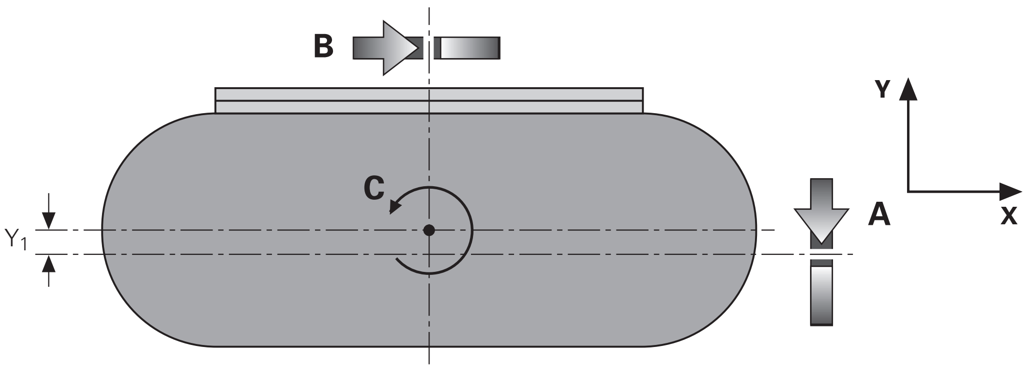
Z_1 = 150.5 mm
Z_2 = 251.5 mm
A_1 = 45°

```
MP 7500 : +1                ;activate function

MP 7510 : %000100           ;shift in Z axis
MP 7511 : %00        ;swivel head
MP 7512 : +150.5            ;dimension Z1

MP 7520 : %001000           ;rotate coordinate system about A axis
MP 7521 : %00        ;swivel head
MP 7522 : –45               ;dimension A1

MP 7530 : %000100           ;shift in Z axis
MP 7531 : %00        ;swivel head
MP 7532 : +251.5            ;dimension Z2

MP 7540 : %100000           ;free swivel axis C
MP 7541 : %00        ;swivel head
MP 7542 : +0                ;variable dimension

MP 7550 : %001000           ;rotate coordinate system about A axis
MP 7551 : %00        ;swivel head
MP 7552 : +45               ;dimension A1

MP 7560 : %010000           ;free swivel axis B
MP 7561 : %00        ;swivel head
MP 7562 : +0                ;variable dimension
MP 7570 : %000000          ;end of transformation sequence
```

Example 3: Universal table (swivel, tilt, rotate)







X_1 = 1.6 mm
Y_1 = 2.7 mm
Z_1 = 331.3 mm
Z_2 = 125.9 mm

Coordinates (related to machine datum) of the centre of rotation of rotary table C with all swivel axes in datum position:

X_R = 420.0 mm
Y_R = 151.2 mm
Z_R = –395.4 mm

```
MP 7500: +1          ;activate function

MP 7510: %000001     ;X coordinates of centre of rotation of C axis
MP 7511: %11         ;tilting table, absolute value
MP 7512: +420        ;dimension XR

MP 7520: %000010     ;Y coordinates of centre of rotation of C axis
MP 7521: %11         ;tilting table, absolute value
MP 7522: +151.2      ;dimension YR

MP 7530: %000100     ;Z coordinates of centre of rotation of C axis
MP 7531: %11         ;tilting table, absolute value
MP 7532: –395.4      ;dimension ZR

MP 7540: %100000     ;free swivel axis C
MP 7541: %01         ;tilting table
MP 7542: +0          ;variable dimension

MP 7550: %000010     ;shift in Y axis
MP 7551: %01         ;tilting table
MP 7552: –2.7        ;dimension Y1

MP 7560: %000100     ;shift in Z axis
MP 7561: %01         ;tilting table
MP 7562: –331.3      ;dimension Z1

MP 7570: %001000     ;free swivel axis A
MP 7571: %01         ;tilting table
MP 7572: +0          ;variable dimension

MP 7580: %000100     ;shift in Z axis
MP 7581: %01         ;tilting table
MP 7582: +125.9      ;dimension Z2

MP 7590: %010000     ;free swivel axis B
MP 7591: %01         ;tilting table
MP 7592: +0          ;variable dimension
```

1.9.2 Five-axis NC programs with swivel axes

The miscellaneous function M114 activates a function that automatically compensates tool lengths when running five-axis NC programs, incorporating the machine geometry values from MP7510 ff. This means that the TNC automatically compensates the offset that results from positioning swivel axes.

The tool tip is always on the programmed nominal coordinates, so the postprocessor can disregard the machine geometry. The tool radius must still be corrected by the CAD system or postprocessor however.

Unlike the "Tilt working plane" function, the coordinate system does not swivel. For machines with rotary tables therefore, remember that rotating the table also rotates the coordinate system in relation to the machine coordinate system. This is not affected by the M114 function.

The programmed feed-rate relates to the tool tip but this is only achieved when tool length compensation is provided by the TNC. If tool length is already compensated by the CAD system, the programmed feed-rate will be the feed-rate of the tool datum. Finally, function M114 is deactivated by M115 or END PGM.

1.10 Synchronized axes

With the HEIDENHAIN TNC, two controlled axes can be coupled in such a way that they can only be moved simultaneously. This facility is required, for example, for gantry axes and tandem tables, and can be activated both for operation with servo lag and in the feed precontrol mode.

In the following description the main axis and tracked axis are referred to as master and slave, respectively.

The Synchronize function is activated with MP850 which defines the master axis to which the specified slave axis must be tracked.

Example: 4th axis is the slave of the X axis

MP850.0	0
MP850.1	0
MP850.2	0
MP850.3	1
MP850.4	0

Of the five controlled axes, two pairs can be synchronized.

MP850 Synchronized axes
Entry: 0 to 5

0 = Main axis
1 = axis tracked to X axis
2 = axis tracked to Y axis
3 = axis tracked to Z axis
4 = axis tracked to 4th axis
5 = axis tracked to 5th axis

MP850.0	X axis
MP850.1	Y axis
MP850.2	Z axis
MP850.3	4th axis
MP850.4	5th axis

1.10.1 Synchronization control

The TNC monitors the synchronization of the master and slave axes. If the positions of the master and slave axes differ by a distance greater than that entered in MP855.x, the error message GROSS POSITIONING ERROR <AXIS> #A appears indicating the slave axis. The positional difference is shown in the LAG display for the slave axis. Synchronization monitoring is entered in MP855.x of the slave axis. Machine parameter MP860.x defines the datum for synchronization control.

MP860.x = 0: Datum at position upon switch-on
When the machine is switched on it is assumed that the master and slave axes are synchronized with one another. Their position upon switch-on is taken as the synchronization reference. In this mode, passing over the reference mark is only necessary for the master axis, and only if the defined references are to be reproduced. Synchrony monitoring begins immediately upon switch-on.

MP860.x = 1: Datum at reference marks (machine datum)
Both axes position to the same reference value when they have passed over their respective reference marks. The default setting can be modified with MP960 (machine datum).

In this mode, any offset between the two axes is compensated in the slave axis at the speed from MP1330 after both reference marks are traversed, and synchronization does not activate until compensation is completed. The way in which the reference marks are traversed (MP1350) must be set the same for both axes. The master axis must be defined before the slave axis in the sequence for approaching the reference marks (MP1340).

Where rotary encoders are used for linear measurement (MP1350 = 2), only one end position switch should be used for both axes because the reference end position marker for the slave axis is only used to evaluate the reference mark and not to evaluate the direction of traverse when controlling the sequence. The direction of traverse is defined by the value of the reference end position marker for the master axis.
Traversing the reference mark is completed when one reference mark has been evaluated for both axes (see also section "Reference marks").

MP855	Synchronization monitoring Entry: 0 to 100.0000 [mm] 0 = Monitoring inactive
MP855.0	X axis
MP855.1	Y axis
MP855.2	Z axis
MP855.3	4th axis
MP855.4	5th axis
MP860	Datum for synchronization control Entry: 0 or 1 0 = Datum at position upon switch-on 1 = Datum at reference marks (machine datum)
MP860.0	X axis
MP860.1	Y axis
MP860.2	Z axis
MP860.3	4th axis
MP860.4	5th axis

1.10.2 Conventions

The slave axis cannot be moved independently.

The nominal value displayed for the slave axis indicates the nominal value of the master axis.

The PLC program must ensure that the master axis does not move until the slave axis is ready (locking, feed-rate enable).

The markers for direction of traverse (M2160 ff.) and axis in motion (M2128 ff.) for the slave axis are not set.

An axis cannot be master and slave at the same time.

Master and slave must be linear axes.

Axis error compensation (both linear and non-linear) must be entered separately for both axes.

The values for rapid traverse, acceleration, software limit switches, feed-rate for passing over reference marks and manual feed-rate are confirmed from the input values of the master axis.

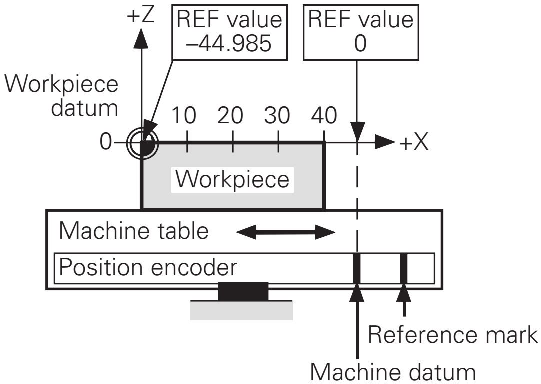
In servo lag mode the k_v factor for master and slave axis should be the same.

Both axes must be either analogue or digital (TNC 425) controlled.

2 Reference marks

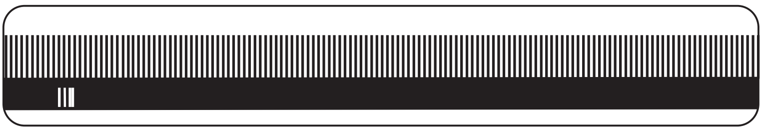
By setting a datum point, a definite positional value (coordinate) is assigned to each axis position for the machining of the workpiece. Since the actual-position value is established incrementally by the measuring system, this correlation between axis positions and positional values must be re-established after every power interruption.

The HEIDENHAIN linear measurement systems are therefore equipped with one or more reference marks. On passing a reference mark a signal is generated which identifies the particular position as a reference point. By passing over the reference marks after a power interruption, the relationships between the axis positions and positional values (and, at the same time, the fixed machine relationships) which were most recently determined by the datum point setting are re-established.

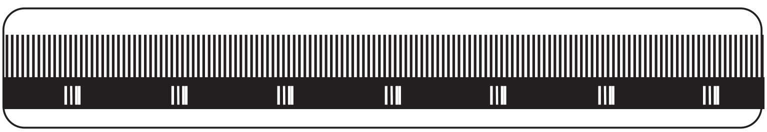


Since it is often inconvenient to re-establish the reference points by traversing large distances after switching on, HEIDENHAIN recommends the use of measuring systems with distance-coded reference marks. With this kind of measuring system the absolute position is available after crossing two reference marks.

The scale graduation consists of the line grating and a reference mark track which runs parallel to it. The distances between any two consecutive reference marks are defined differently, so that the absolute position of the machine slide can be determined from this distance.



Scale with one reference mark



Scale with distance-coded reference marks

2.1 Passing over the reference marks

The reference marks for axes must be passed after switching on the control. This can be achieved by

- pressing the external START key. The axis sequence is determined by machine parameter MP1340.X (automatic passing of the reference marks),

- pressing the external axis direction keys. The sequence is determined by the operator.

Only after passing over the reference mark

- can the software limits be activated,

- can the most recently set datum point be reproduced,

- is PLC positioning and positioning with the miscellaneous functions M91 and M92 possible,

- is the counter value set to zero for non-controlled axes.

For distance-coded measuring systems the machine datum (MP960.x) is reference to the Zero-Reference mark. In linear measurement systems the Zero-Reference mark is the first reference mark after the start of the measuring length; in angular measurement systems the Zero-Reference mark is marked.

The direction of traverse and the velocity on passing the reference marks is defined by machine parameters (MP1320, MP1330.X).

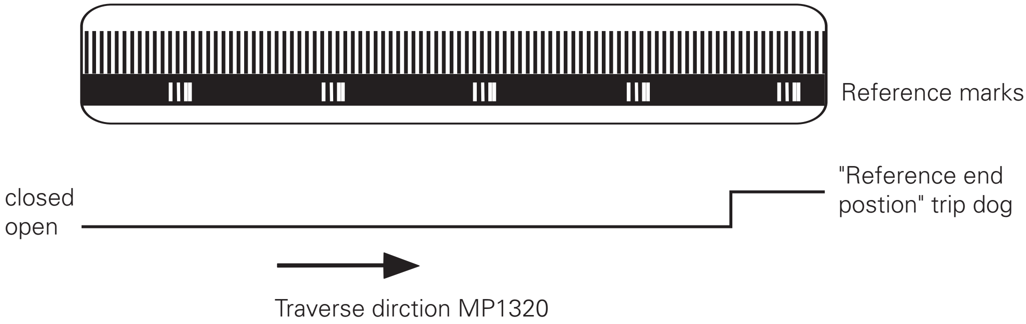
The functional sequence for passing the reference marks can be fixed specifically for the axes by machine parameters (MP1350.X).

The operating condition "PASS OVER REFERENCE MARKS" is sent to the PLC by the NC (W272). If the operating mode is changed before all reference marks have been passed, the Soft key "PASS OVER REFERENCE" will appear. The markers M2136 to M2140 inform the PLC for which axes the reference marks have not yet been crossed.

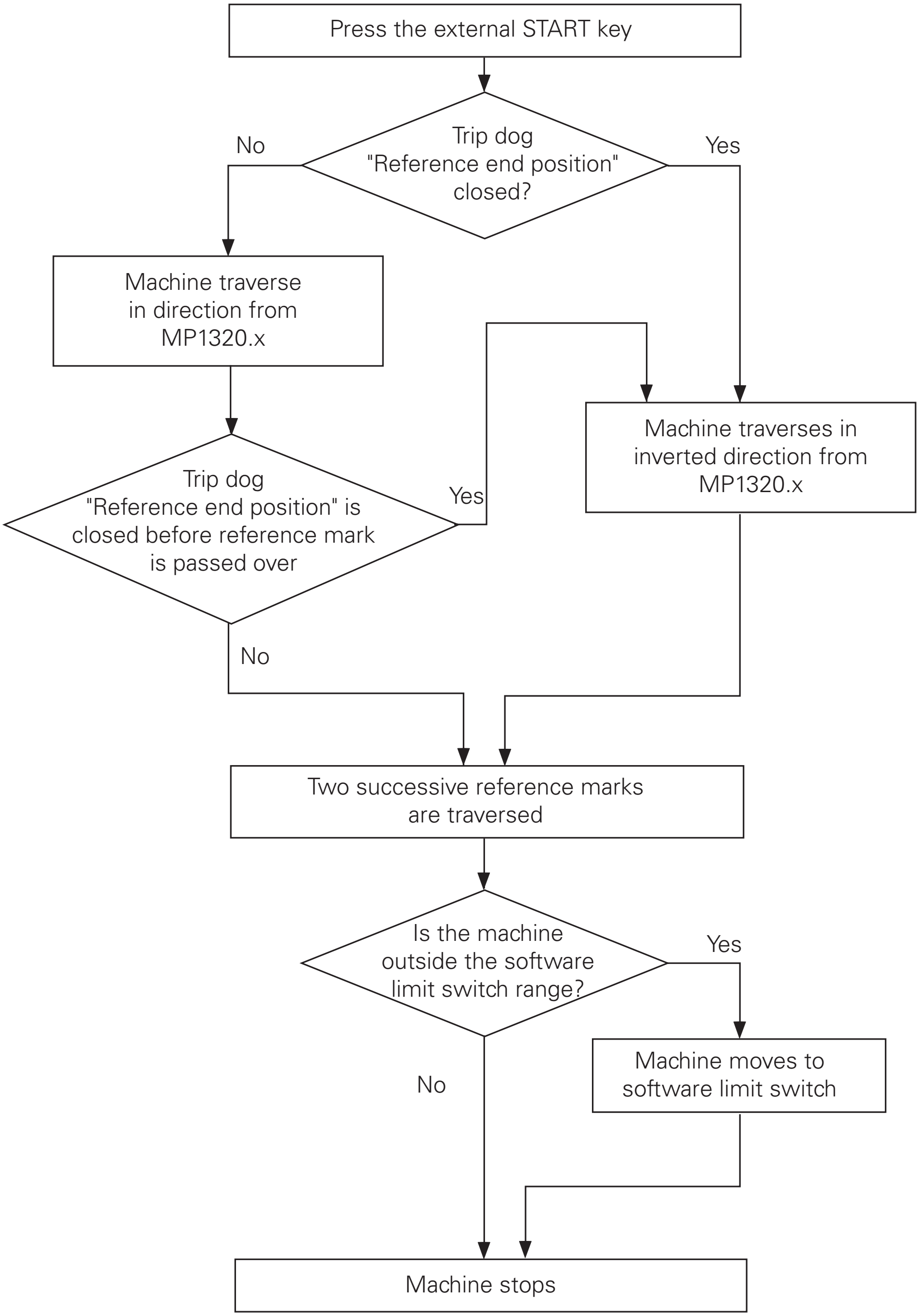
In order to avoid exceeding the traverse range when passing over the reference marks a trip dog (reference end-position) is necessary. This trip dog must be fixed at the end of the traverse range by the manufacturer of the machine. The trigger signal from the trip dog is connected to an available PLC input. In the PLC program this PLC input is combined with the markers for "Reference end-position" (M2506, M2556 to M2559).

2.1.1 Measuring systems with distance-coded reference marks

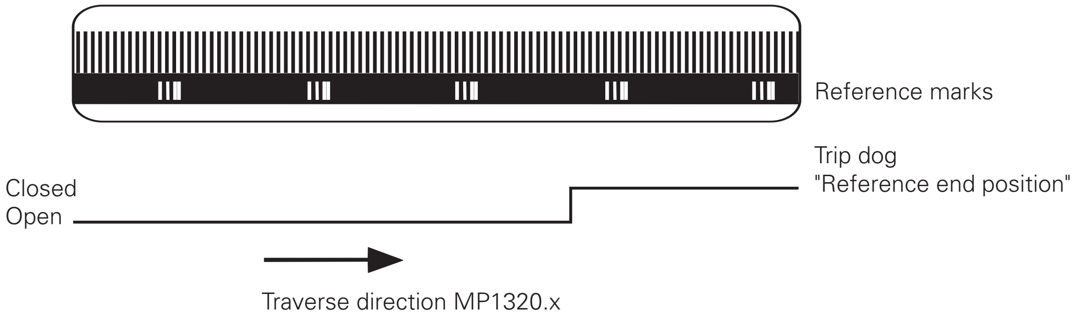
Machine parameter MP1350.x=3



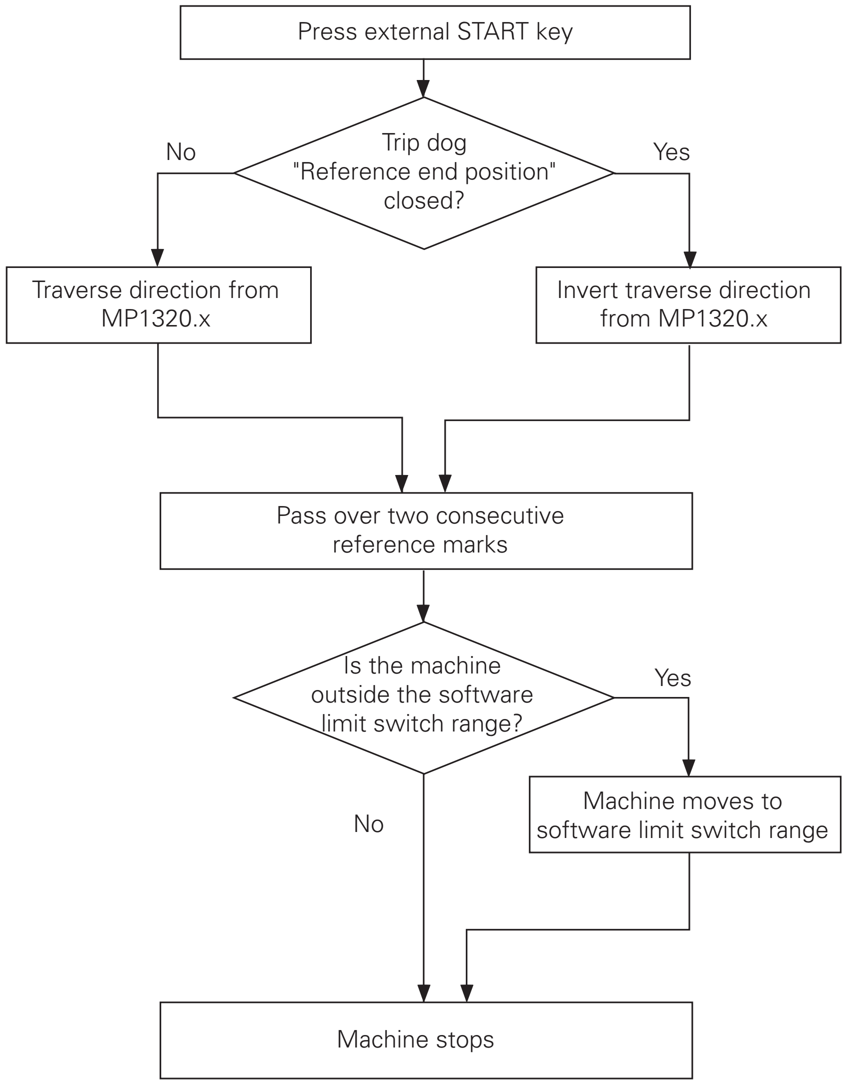
Sequence "Automatic passing over reference marks" (press the external START key).
MP1350.x = 3



Machine parameter MP1350.x=0



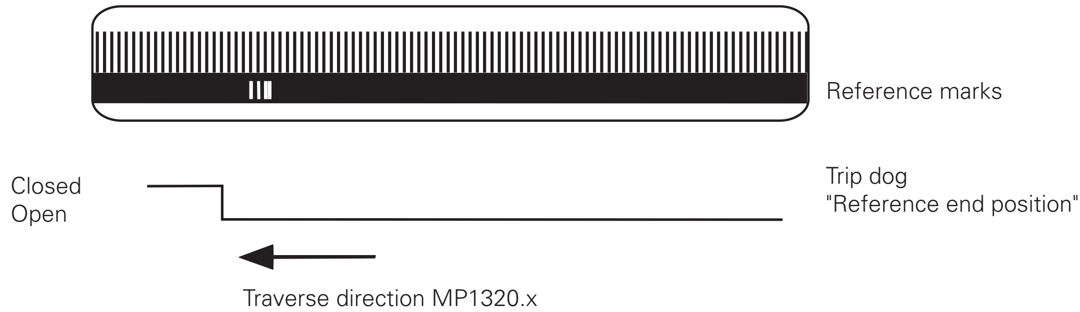
Sequence "Automatic passing over reference marks" (press the external START key).
MP1350.x = 0



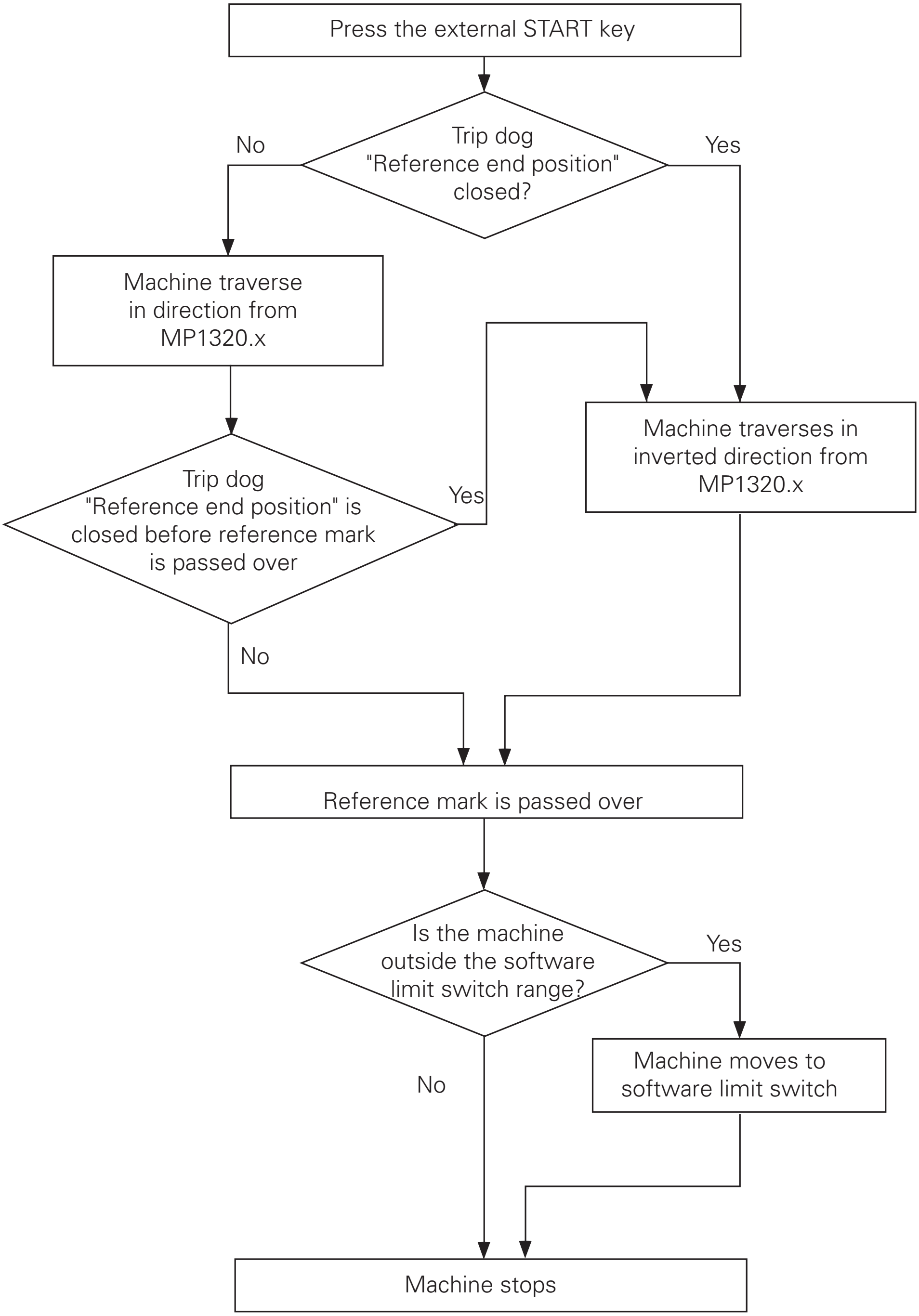
If during automatic pass-over the trip dog is not closed until it is in the "Reference end-position" range, the contouring control will ignore the signal. It is therefore necessary that there be at least two reference marks in the range of the "Reference end-position".

2.1.2 Measuring systems with one reference mark

Machine parameter MP1350.x=1



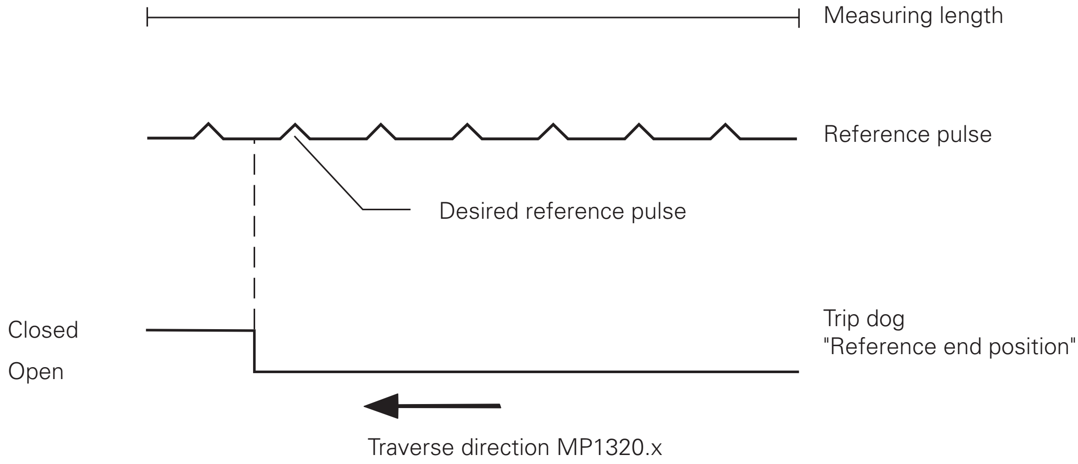
Sequence "Automatic passing over reference marks" (press the external START key).
MP1350.X = 1



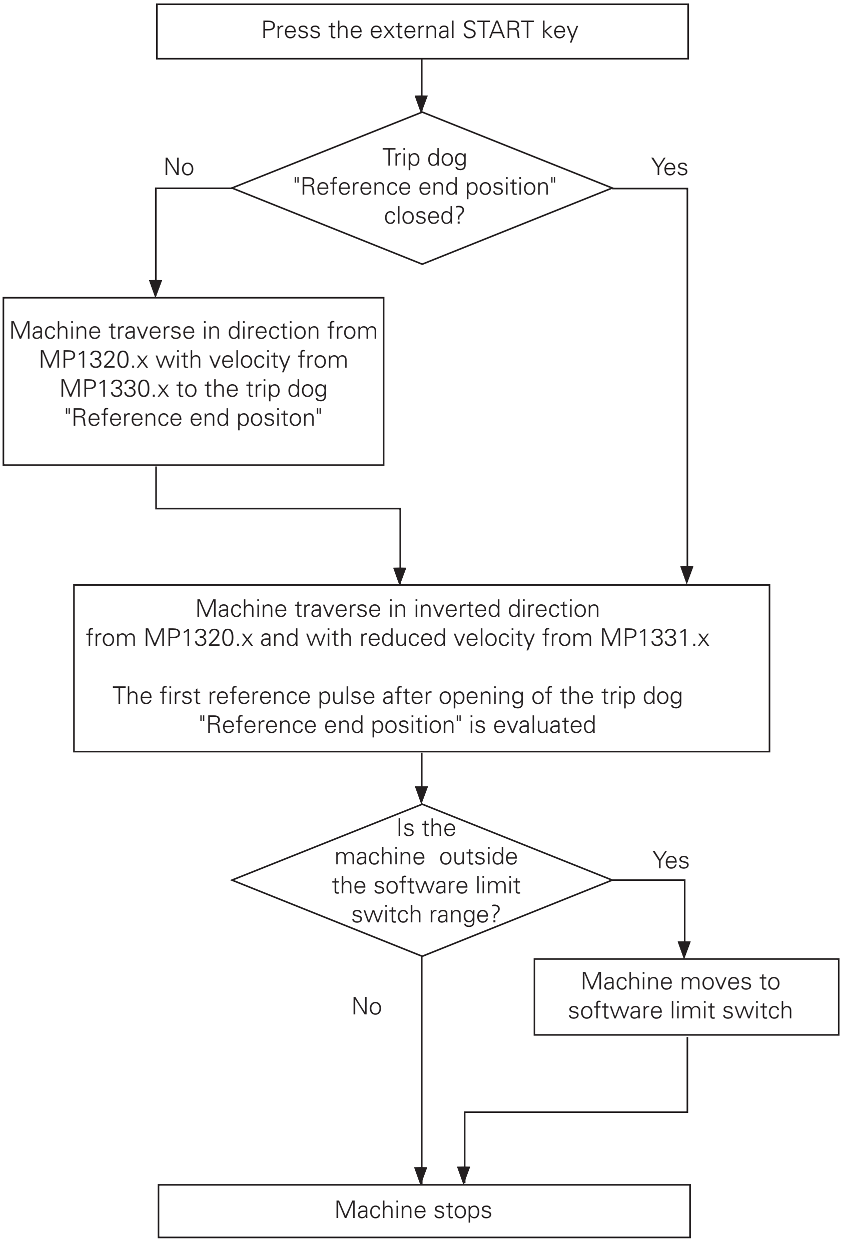
2.1.3 Linear measurement via rotary encoder

Machine parameter MP1350.x = 2

For linear measurement using a rotary encoder a reference pulse is produced on each revolution of the encoder. It must be ensured that, after switching on the machine, always the same reference pulse is evaluated. This can also be achieved by using the trip dog "Reference end-position".



Sequence "Automatic passing over reference marks" (Press the external START key).
MP1350.X = 2



MP1320		Direction for traversing the reference marks Entry: %xxxxx
Bit 0	X axis:	0 = positive
Bit 1	Y axis	1 = positive
Bit 2	Z axis	
Bit 3	Axis 4	
Bit 4	Axis 5	

MP1330	Feed rate for traversing the reference marks Entry: 80 to 30 000 [mm/min]
MP1330.0	X axis
MP1330.1	Y axis
MP1330.2	Z axis
MP1330.3	Axis 4
MP1330.4	Axis 5
MP1331	Feed rate for leaving the reference end-position (only for rotary encoders MP1350=2) Entry: 80 to 500 [mm/min]
MP1331.0	X axis
MP1331.1	Y axis
MP1331.2	Z axis
MP1331.3	Axis 4
MP1331.4	Axis 5
MP1340	Sequence for traversing reference marks

Entry: 0 = no evaluation of the reference mark
1 = X axis
2 = Y axis
3 = Z axis
4 = Axis 4
5 = Axis 5

MP1340.0	1st axis
MP1340.1	2nd axis
MP1340.2	3rd axis
MP1340.3	4th axis
MP1340.4	5th axis

MP1350	Type of reference mark approach		
	Entry: 0 = measuring system with distance-coded reference marks 1 = measuring system with one reference mark 2 = special sequence (linear measurement via rotary encoder) 3 = measuring system with distance-coded reference marks		
MP1350.0	X axis		
MP1550.1	Y axis		
MP1350.2	Z axis		
MP1350.3	Axis 4		
MP1350.4	Axis 5		

Marker	Function	Set	Reset
M2136	Reference marks X axis not yet traversed	NC	NC
M2137	Reference marks Y axis not yet traversed		
M2138	Reference marks Z axis not yet traversed		
M2139	Reference marks axis 4 not yet traversed		
M2140	Reference marks axis 5 not yet traversed		
M2556	Reference end-position for X axis	PLC	PLC
M2557	Reference end-position for Y axis		
M2558	Reference end-position for Z axis		
M2559	Reference end-position for axis 4		
M2506	Reference end-position for axis 5		

Address	Function	Set	Reset
W272	Operating mode	NC	NC
	1 = Manual operation 2 = Handwheel 3 = Positioning with manual entry 4 = Program run/single block 5 = Program run/full sequence 7 = Pass over reference points		

2.2 Machine datum

The reference mark defines a point on the measuring system. The reference points of all axes define the scale datum. MP960.x contains the distance from the scale datum to the machine datum. All REF-based displays and positioning movements refer to the machine datum (see also Section "Display and operation").

3 Servo positioning of the NC-axes

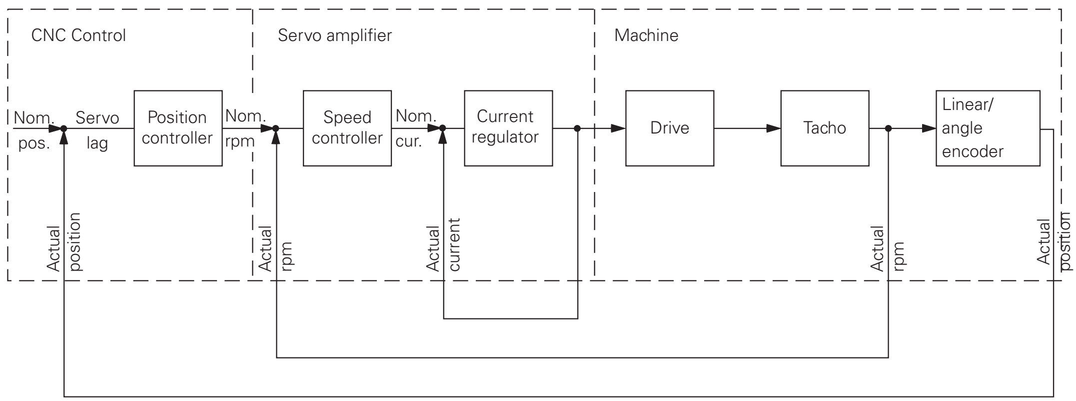
This section describes all the control functions which are important for the control and monitoring of the NC-axes .

Further parameters for the NC-axes can be found under "Machine axes".
The control of the main spindle (S-axis) is described under "Main spindle".

3.1 The position control loop of an NC-machine

In CNC machines the servo control is normally implemented as a cascade control (see following block diagram).

The motor speed control and the current control (both in the drive amplifier) are integrated into the servo position control (CNC-control). The servo controlled system consists of the motor and machine slide.



Block diagram of the position control loop, here as a cascade control

3.2 Servo positioning in TNC controls

Two control methods are possible with the control. They can be used alternatively.

1. control with lag (Section 3.2.1)
2. control with feedforward control (Section 3.2.2)

The choice of control method is determined by machine parameters MP1390 and MP1391.
The machine adjustment must always be carried out for both methods of control.

MP1390 Velocity feedforward control in the operating modes "Positioning with MDI", "Program run, single block", and "Program run, full sequence"
Input: 0 or 1
0 = operation with feedforward control
1 = operation with servo lag

MP1391 Velocity feedforward control in the operating modes "Manual" and "Handwheel"
Input: %xxxxx

Bit 0	Axis X	0 =	operation with servo lag
Bit 1	Axis Y	1 =	operation with feedforward control
Bit 2	Axis Z		
Bit 3	Axis 4		
Bit 4	Axis 5		

3.2.1 Control with servo lag

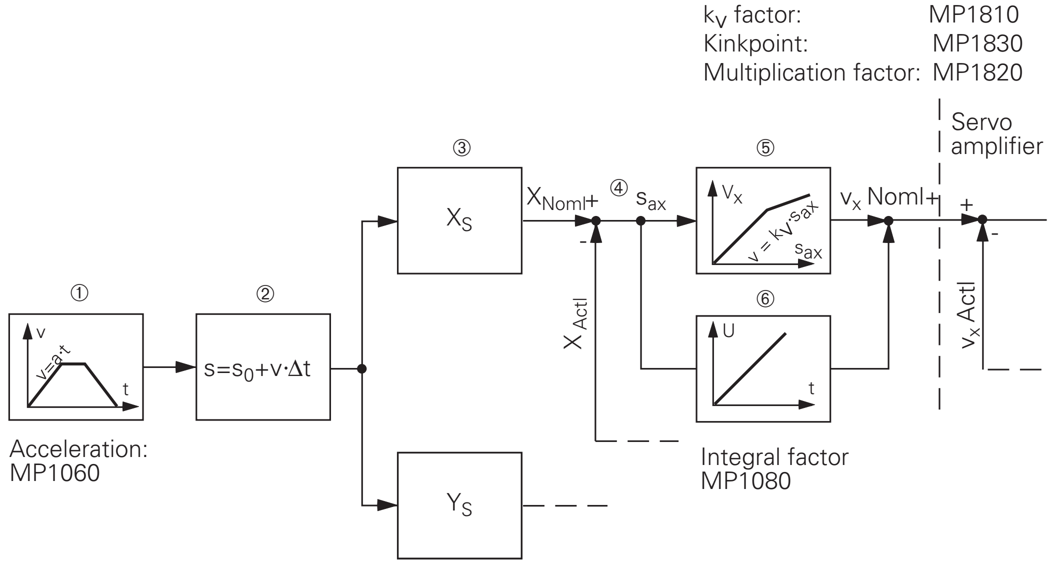
Servo lag refers to the difference (lag) between the nominal position which is defined by the NC and the actual position of the axes. Control would not be possible without this lag.

The advantage of operation with servo lag lies in the "softer" control, which is especially suitable for machining 3D-contours, since corners and radii are smoothed out, depending on the Kv factor (position loop gain factor) and the machining feed rate.

Depending on how far these "contour errors" can be tolerated, the machine operator or manufacturer must decide and choose either operation with servo lag or with feedforward control.

Operation with servo lag is depicted in a simplified form in the following block diagram for the X axis. It shows a part of the cascade control mentioned previously.

All machine parameters which influence the control characteristic are shown here.



① The control calculates a velocity value every 3 ms (TNC 415) or every 6 ms (TNC 407) from the feed rate programmed in the NC-program and the final position, allowing for the acceleration which has been stored (MP1060). The stored acceleration is valid for the rising as well as the falling slope. If several axes are traversed simultaneously, then the smallest value for acceleration is effective.

② Every 3 ms (6 ms) a nominal position value is derived from the calculated velocity.

$s = s_o + v \cdot \Delta t$	s	= Nominal path value
	s_o	= Previous nominal path value
	v	= Calculated velocity
	Δt	= Cycle time 3 ms

③ The nominal path value is resolved into the individual axis components, depending on which axes have been programmed.

④ The axis-dependent nominal path value is compared with the actual value of the positions and the lag s_a is calculated.

$s_{ax} = x_{Noml} - x_{Actl}$	s_{ax}	= Lag for X-axis
	x_{Noml}	= Nominal path value for X-axis
	$x_{Act\,l}$	= Actual path value for X-axis

⑤ The lag is multiplied by the $k_{v\,factor}$ MP1810 and passed on to the drive amplifier as a nominal velocity value (analogue voltage).

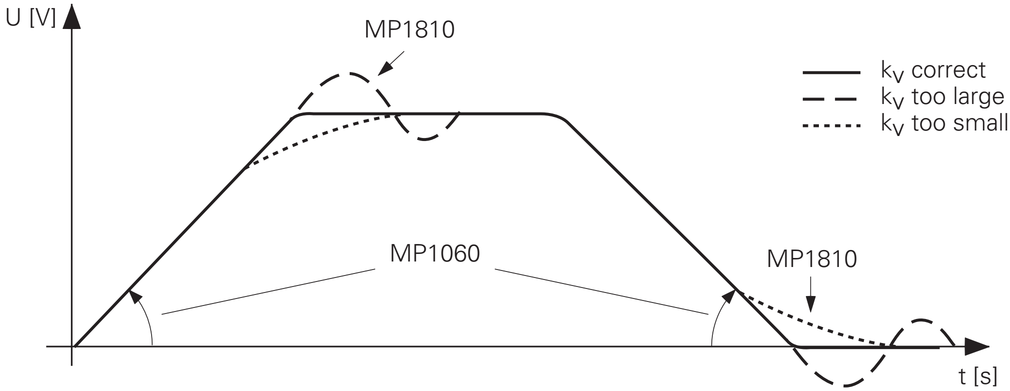
$v_x = k_v \cdot s_{ax}$	v_x	= Nominal velocity-value for X-axis

⑥ If the axes are stopped, the integral factor MP1080 is effective as well. It causes an offset adjustment (see the section "Offset adjustment").

The k_v factor (position loop gain) determines the amplification of the control loop. The optimal k_v factor must be determined by trial and error.

If you choose a very high k_v factor, the lag is very small. However, this can lead to oscillations when moving into a new position. If the k_v factor is too small, the new position will be reached too slowly. For axes that are interpolated with each other the k_v factor must be equal to prevent contour deviations.

The following diagram shows the response for various k_v factors.



The acceleration can be programmed by the machine parameter MP1060. It determines the slope of the ramp on the rising and falling edges.

For axes which are mutually interpolated the k_v factor must be the same, in order to avoid contour distortion!

MP1060	Acceleration Entry: 0.001 to 5.0 [m/s^2]
MP1060.0	Acceleration X axis
MP1060.1	Acceleration Y axis
MP1060.2	Acceleration Z axis
MP1060.3	Acceleration 4th axis
MP1060.4	Acceleration 5th axis

The following formula shows the relationship among K_v factor, feed rate and servo lag:

$$k_v = \frac{v_e}{s_a}$$

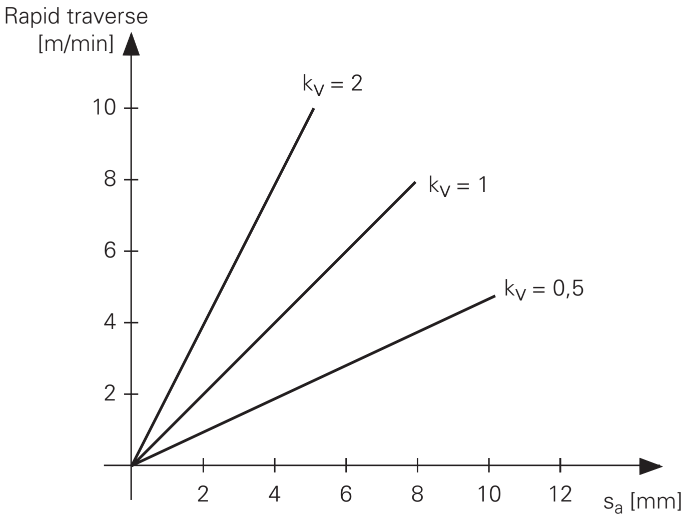
k_v = position loop gain $[\frac{m/min}{mm}]$

v_e = rapid traverse $[\frac{m}{mm}]$

s_a = servo lag [mm]

or

$$s_a = \frac{v_e}{k_v}$$

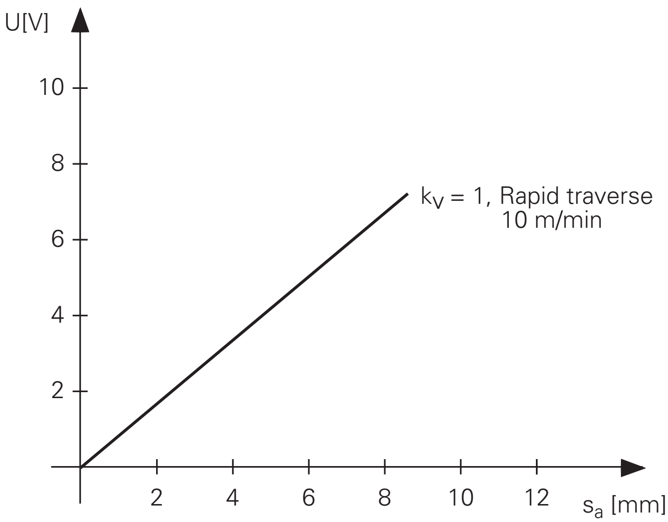


The maximum feed rate stored in machine parameter MP1010 can be reduced by the PLC.

If the PLC enters a feed value in Doubleword D596 which is lower than that in machine parameter MP1010, then the value in D596 is effective. If the value which is entered is higher, then MP1010 remains effective. After switching on the control, or an interruption of the running of the PLC, the Doubleword D596 is pre-loaded with the value 300000.

The rapid traverse (maximum traversing speed) must be adjusted by the desired analogue voltage (e.g. 9 V) on the servo-amplifier (see section "Commissioning and start-up procedure"). For each axis-specific rapid traverse there is an analogue voltage which is stored in the machine parameter MP1050.

The resulting lag error s_a thus depends on the analogue voltage.



A special feed rate for manual operation (Manual-feed) is stored in machine parameter MP1020. In general, it is significantly lower than the rapid traverse.

MP1810	K_v factor for operation with lag
	Entry: 0.100 to 20.000 [$\frac{m/min}{mm}$]
MP1810.0	k_v factor X axis
MP1810.1	k_v factor Y axis
MP1810.2	k_v factor Z axis
MP1810.3	k_v factor 4th axis
MP1810.4	k_v factor 5th axis

MP1815	K_v factor for operation with lag active after M105
	Entry: 0.100 to 20.000 [$\frac{m/min}{mm}$]
MP1815.0	X axis
MP1815.1	Y axis
MP1815.2	Z axis
MP1815.3	4th axis
MP1815.4	5th axis

MP1010	Rapid traverse
	Entry: 10 to 300 000 [mm/min]
MP1010.0	X axis
MP1010.1	Y axis
MP1010.2	Z axis
MP1010.3	4th axis
MP1010.4	5th axis

MP1050	Analogue voltage for rapid traverse
	Entry: 1.000 to 9.000 [V]
MP1050.0	X axis
MP1050.1	Y axis
MP1050.2	Z axis
MP1050.3	4th axis
MP1050.4	5th axis

MP1020	Manual feed
	Entry: 10 to 300 000 [mm/min]
MP1020.0	Manual feed X axis
MP1020.1	Manual feed Y axis
MP1020.2	Manual feed Z axis
MP1020.3	Manual feed 4th axis
MP1020.4	Manual feed 5th axis

Address	Function
D596	Rapid traverse from PLC

Servo resolution:
The internal calculating resolution of the control is always 0.1 µm. However the display step can be set with MP7290. The rounding up or down is only in display (see section "Display and operation"). Normally a servo resolution of 1 µm is sufficient, i.e. the control must be able to generate at least one voltage step per 1 µm positional deviation. For 0.1 µm servo resolution at least one potential step must be generated per 0.1 µm.

Calculation of the smallest voltage step:
The controls produce an analogue voltage 0 to 10 V. This 10 V is divided by the TNC 415 with a 16 Bit A/D-converter to give 65 536 divisions. As a result, the smallest potential step is 0.15 mV.
In the TNC 407 the 10 V analogue potential is produced by a 14 Bit A/D converter, giving 16 384 divisions. The resulting smallest potential step is 0.6 mV.

Potential steps per µm (0.1 µm) positional deviation:
As described above, moving with the rapid traverse (MP1010) results in a certain lag distance s_a. The rapid traverse rate is reached at a definite voltage (MP1050). So one can calculate a definite potential ΔU per µm of positional deviation (lag).

$$\Delta U = \frac{\text{MP1050 [mV]}}{s_a\ [\mu m]}$$

If ΔU is divided by the smallest voltage step which can be produced (0.15 mV for TNC 415 or 0.6 mV for TNC 407), the result is the number of voltage steps which are produced per µm or 0.1 µm positional deviation.

for TNC 415:

$$n = \frac{\Delta U\ [mV]}{0.15\ [mV]}$$

for TNC 407:

$$n = \frac{\Delta U\ [mV]}{0.6\ [mV]}$$

Example:

$k_v = 2\,\frac{m/min}{mm}$ rapid traverse 5 000 [mm/min], U = 9 [V]

$$s_a = \frac{Ve}{k_v} = \frac{5\,000\ [mm/min]}{2\ \frac{m/min}{mm}} = 2\,500\ [\mu m]$$

$$\Delta U = \frac{9\,000\ [mV]}{2\,500\ [\mu m]} = 3.6\ [mV/\mu m]$$

$$n_{TNC\ 415} = \frac{3.6\ [mV/mm]}{0.15\ [mV]} = 24 \text{ steps/µm positional deviation}$$

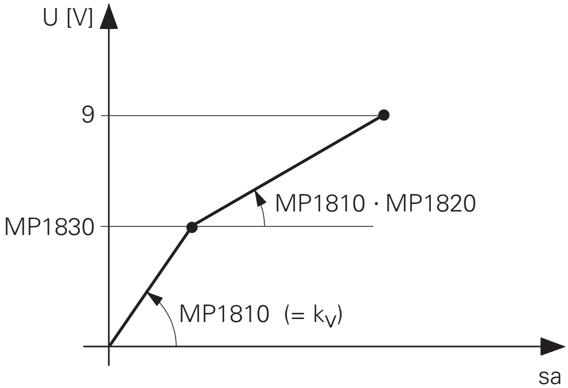
$$n_{TNC\ 407} = \frac{3.6\ [mV/mm]}{0.6\ [mV]} = 6 \text{ steps/µm positional deviation}$$

Kink point:

For machines with high rapid-traverse rates it is usually not possible to raise the K_V factor enough to give an optimal loop characteristic over the entire range of speeds (stop, machining feed rates, rapid traverse). In such cases one can introduce a kink point, which gives the following advantages:

– for lower rates, a high K_v, i.e. a larger voltage step per µm of positional deviation,
– for higher rates (above the machining rates) a lower K_v.

The position of the kink point is determined by machine parameter MP1830. In the higher range the K_V factor is multiplied by the factor from MP1820.



The kink point must lie above the range of machining feeds!
Under these conditions, the lag can be calculated as follows:

$$s_a = \frac{Ve}{k_v} \cdot \left[\frac{\text{MP1830 [\%]}}{\text{100 [\%]}} + \frac{\text{100 [\%]} - \text{MP1830 [\%]}}{\text{MP1820} \cdot \text{100 [\%]}} \right]$$

MP1820	Multiplication factor for the K_V factor Entry: 0.001 to 1.000
MP1820.0	X axis
MP1820.1	Y axis
MP1820.2	Z axis
MP1820.3	4th axis
MP1820.4	5th axis

MP1830	Kink point Entry: 0.000 to 100.000 [%]
MP1830.0	Axis X
MP1830.1	Axis Y
MP1830.2	Axis Z
MP1830.3	4th axis
MP1830.4	5th axis

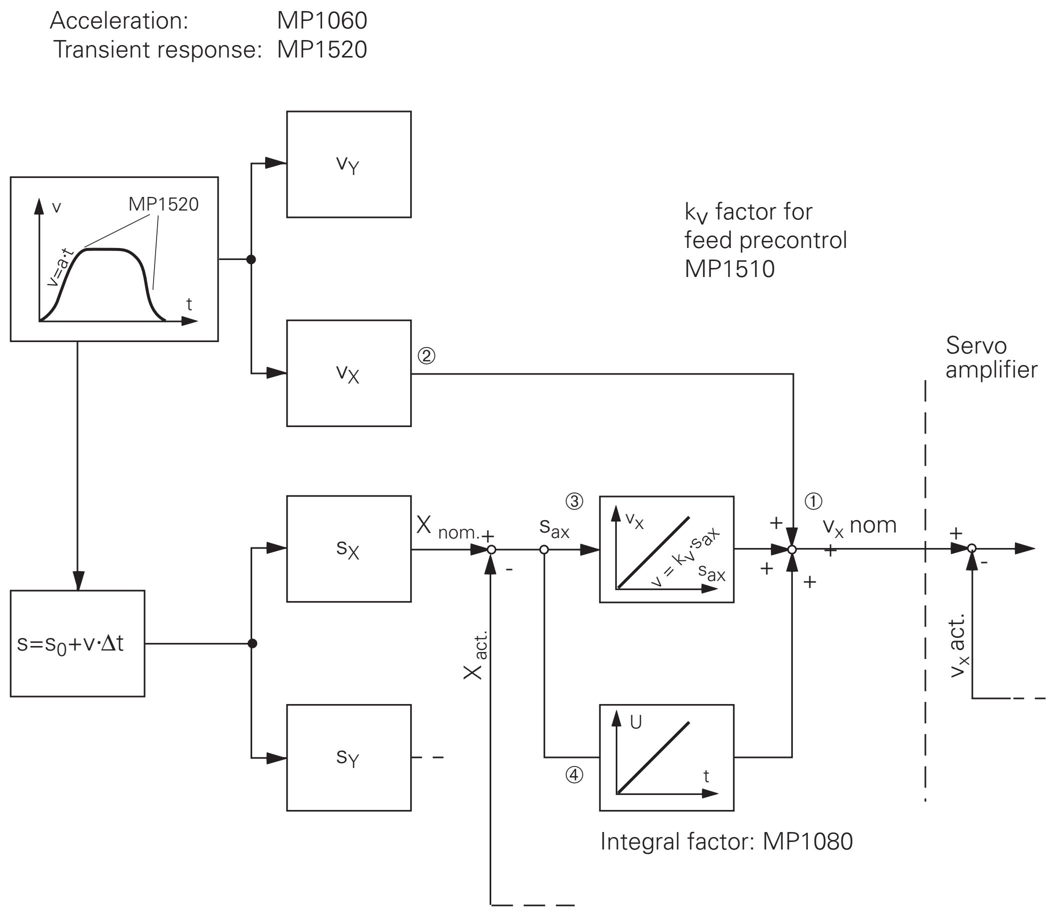
3.2.2 Feedforward control

Control with feedforward control means that the nominal velocity value for the machine is varied by a machine parameter (control element of the nominal velocity). Together with the velocity element which is calculated from the lag (servo-controlled element of the nominal velocity), this gives the final nominal velocity value.

The lag is very small with this method. Operation with feedforward control has the advantage that contours can be followed very accurately at a high speed. Normally, work will be carried out using feedforward control.

MP1390 switches feedforward control on in the "Positioning with manual data input", "Program run/single block" and "Program run/full sequence" modes. MP1391 has the same effect for the "Manual" and "Handwheel" modes.

Operation with feedforward control for the X axis can be shown in a simplified form in the following block diagram. All machine parameters which influence the servo characteristic are shown here.



① When operating with feedforward control the set velocity value (the analogue potential for the drive amplifier) is formed from three components: feedforward control, servo lag, integral component.

② The feedforward control value, which is adjusted to the dynamic characteristics of the machine by machine parameters, makes it possible to control the loop with a lag distance which is nearly "0".

③ Deviations in the actual positional value are compensated as described under "Operation with lag". The servo follows the pre-compensated curve.
The K_V factor for the feedforward control then comes into play.

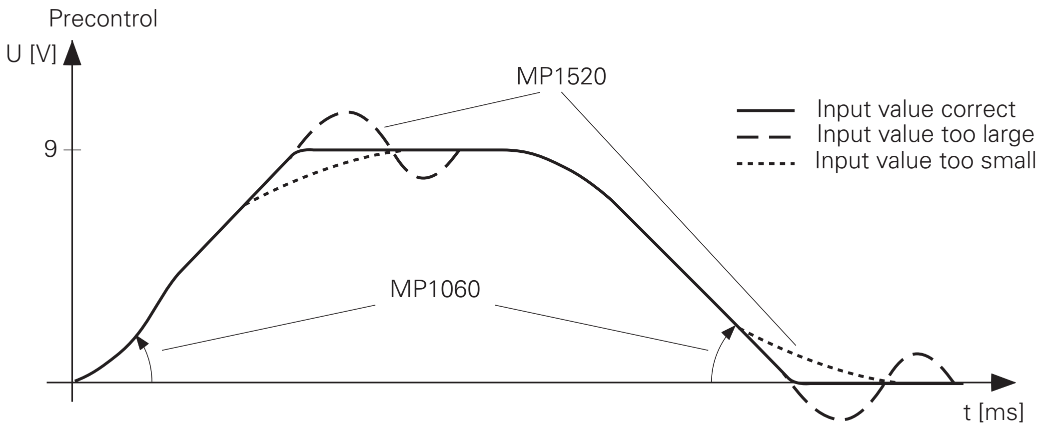
④ If the axes are stopped, the integral factor will also be effective: it results in an offset correction.

MP1390 Feedforward control in the "Positioning with manual data input", "Program run, single block" and "Program run, full sequence" modes
Entry: 0 or 1
0 = Operation with feedforward control
1 = Operation with servo lag

MP1391 Feedforward control in the "Manual" and "Handwheel" modes
Entry: %xxxxx

Bit 0	X axis	0 =	Operation with servo lag
Bit 1	Y axis	1 =	Operation with feedforward control
Bit 2	Z axis		
Bit 3	4th axis		
Bit 4	5th axis		

The feedforward control value is adjusted to the dynamics of the machine by two machine parameters.

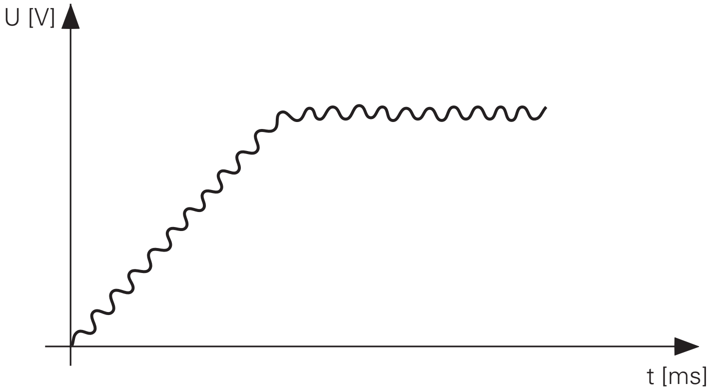


The machine parameter MP1060 determines the acceleration (= slope of the precontrolled speed curve).

MP1520 determines the transient response into the nominal position when accelerating and decelerating. The greater the value which is entered, the more the system will tend to oscillate.

The fine control of the precontrolled speed set-value is influenced by the K_V factor for feedforward control MP1510.

The transient response when accelerating is defined separately by MP1530 in software types 243 05, 259 96, 243 07 and 243 02.



If the K_V factor is too large, the system will oscillate about the precontrolled speed set-value.
The size of the K_V factor is normally the same as the K_V factor for operation with lag distance. Depending on the dynamic characteristics of the machine it may also be higher (see "Commissioning and start-up procedure").

A second set of k_v factors can be defined in machine parameter MP1515.x and selected with M function M105 (see section "Display and operation").

MP1530	*Damping factor for transient response* *(Only with software types 243 05, 259 96, 243 07, 243 02)* *Entry: 0.010 to 0.999*
MP1520	Transient response Entry: value 0.100 to 10.000
MP1510	K_V factor for feedforward control (M105) Entry: 0.100 to 20.000 $[\frac{m/min}{mm}]$
MP1510.0	X axis
MP1510.1	Y axis
MP1510.2	Z axis
MP1510.3	4th axis
MP1510.4	5th axis
MP1515	K_V factor for feedforward control active after M105 Entry: 0.100 to 20.000 $[\frac{m/min}{mm}]$
MP1515.0	X axis
MP1515.1	Y axis
MP1515.2	Z axis
MP1515.3	4th axis
MP1515.4	5th axis

3.3 Offset adjustment

The TNC 407/TNC 415 controls include several possibilities for compensating for an offset voltage which would cause the axes to drift.

The maximum permissible offset-voltage in the control is 100 mV. If this voltage is reached or exceeded, then the error message

"**GROSS POSITIONING ERROR E <AXIS> #**"

will appear.

The automatic cyclical offset-adjustment and the adjustment via integral factor must not be active simultaneously!

3.3.1 Offset adjustment by code number

An automatic offset adjustment can be activated with the code number 75 368. After entering the code number the control shows the offset values for the axes X, Y, Z, 4, 5 in the dialogue line. The values indicate the voltage in 0.15 mV units. Thus a display of 10 means 10 x 0.15 mV = 1.5 mV. The display 0 means no offset.

On pressing the ENT key or the CONTINUE soft key the offset values are automatically compensated. The control puts out an appropriate compensating voltage. In the TNC 407 compensation only takes place if the offset voltage is $\geq$ 0.6 mV. To switch off the automatic offset adjustment, enter the code number and press the NO ENT key or the QUIT soft key. If the code number has been entered but no changes are desired, press the END soft key.

The offset values are stored in the control and are non-volatile. If the control is exchanged, the code number must be entered to reactivate the offset adjustment.

3.3.2 Automatic cyclical offset adjustment

The machine parameter MP1220 can be used to program a time interval, after which an offset adjustment will be performed cyclically.
An automatic adjustment will be carried out, when the predetermined time has elapsed and the following conditions are fulfilled:

- all axes are stopped,
- the spindle is not switched on,
- the axes are not clamped.

For each adjustment cycle there will be a 1 mV correction if the offset voltage is larger than 1 mV. If the offset voltage is smaller than 1 mV then, in the TNC 415, compensation steps of 0.15 mV will be used (in the TNC 407 in steps of 0.6 mV).

MP1220 Automatic cyclical offset adjustment
Entry: 0 to 65 536 [s]

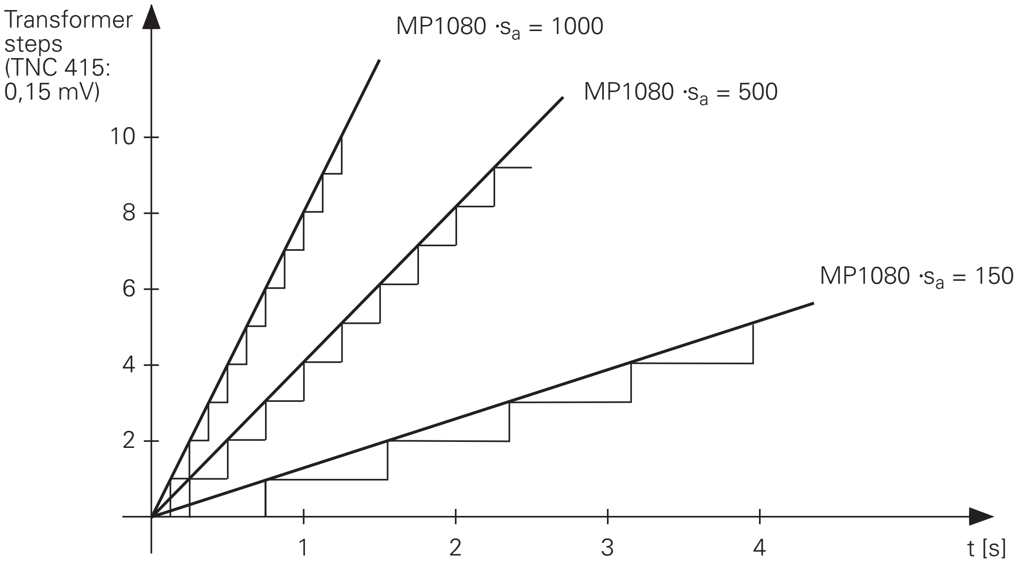
0 = no automatic adjustment

3.3.3 Offset adjustment with integral factor

The integral factor MP1080 also results in an automatic offset adjustment. It is only effective in the stop condition (see block diagram of control loop).

According to the size of the factor the offset voltage will be reduced quickly or slowly.

Even a small amount of play in the drives can lead to instability in the control loop. An integral factor of 0 is entered in this case.



MP1080	Integral factor Entry: 0 to 65 535
MP1080.0	Integral factor X axis
MP1080.1	Integral factor Y axis
MP1080.2	Integral factor Z axis
MP1080.3	Integral factor 4th axis
MP1080.4	Integral factor 5th axis

3.4 Contour behaviour

3.4.1 Radial acceleration

As well as the normal acceleration (MP1060) there is also a machine parameter for radial acceleration (MP1070).

The machine parameter limits the feed for circular movements according to the following formula:

$v = \sqrt{r\,[m] \cdot MP1070\,[m/s^2]}$

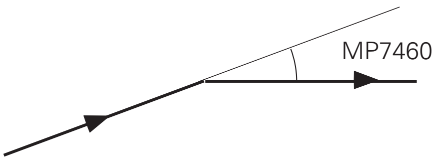
v= feed rate for circular movements [m/s]
r= radius [m] (cutter mid-point contour)

It is recommended that a value should be entered which is between the half of and the same as that in MP1060 (Acceleration). If the programmed feed is lower than that above, then the programmed feed will be used. MP1070 is effective for operation with lag and with feedforward control.

MP1070	Radial acceleration Entry: 0.001 to 5.000 [m/s^2]

3.4.2 Constant feed rate in corners

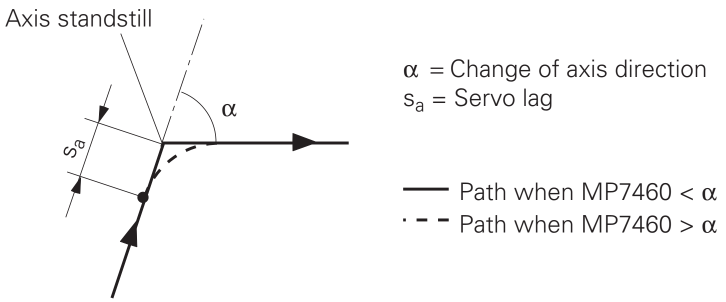
Machine parameter MP7460 defines the angle which can still be traversed with constant surface speed. This machine parameter is effective for corners without a radius compensation, for internal corners it is also effective with a radius compensation.



This machine parameter is effective for operation with feedforward control as well as for operation with lag. The permissible size of the angle depends on the drives in the machine.

Realistic values are 5° to 15°.

The resulting path is as follows:

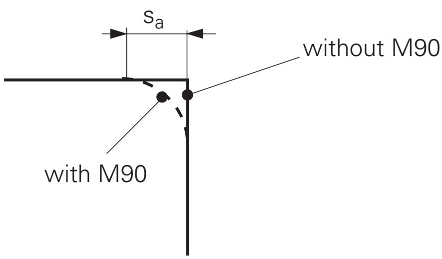


MP7460 Constant feed rate in corners
Entry value: 0.0001 to 179.9999°

3.4.3 Constant feed rate in corners with M90

The miscellaneous function M90 was introduced to enable constant surface speed to be achieved in corners without radius compensation.

This miscellaneous function is only effective in operation with lag!



3.4.4 Contour speed related to tool cutting edge

The programmed feed-rate normally relates to the path of the cutter centre.

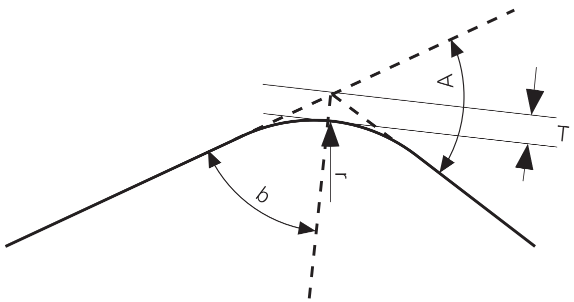
When M function M109 is active the feed-rate at radii is increased or reduced to maintain a constant feed-rate at the cutting edge of the tool.
When M function M110 is active the feed-rate is reduced only. There is no increase in feed-rate.
When M function M111 is active the programmed feed-rate is re-assigned to the path of the cutter centre.

3.4.5 Preset tolerance at corners

3D forms are often described by a progression of very short linear blocks. The control reduces the feed rate at the discontinuous transitions in order to keep the tool exactly on path.

If you are working with servo lag, depending on the feed rate the corners will be more or less "rounded" if M90 is entered or if MP7460 is set correspondingly. Since feedforward control is intended to virtually eliminate servo lag, working with feedforward control will result in sharp corners. This is not desired, however, on 3D forms.

With the M function M112 you can enter a tolerance for the deviation from the programmed contour at corners and a limit angle for activation of the tolerance.



A rounding arc is inserted at corners. The radius of the arc is a function of the entered tolerance T and the limit angle A. Three radii are calculated:

$$r_1 = \frac{T \times \sin(b)}{1 \sin(b)} \qquad r_2 = \frac{1}{2} \times l \times \tan(b) \qquad r_3 = \frac{v^2}{a}$$

T =	The tolerance entered with M112. If no tolerance was entered, the tolerance is be considered to be infinite.
b =	Half of the angle of the adjacent line segments
l =	The shorter of the two adjacent line segments
v =	Programmed feed rate × 1.5 (Feed rate override = 150%)
a =	Smaller value from the acceleration of the interpolated axes (MP1060) and the radial acceleration (MP1070).

The smallest of the 3 radii is used. r_3 is included in the calculation only if the change in direction is greater than the given angle *A*. As long as the limit angle *A* is not exceeded, the size of the inserted arc is independent of the feed rate; the largest possible arc is inserted.

If the feed rate during program run too large for the calculated rounding arc, the TNC automatically reduces the feed rate. M112 is inactivated again by M113. M112 is effective in NC blocks without radius compensation both in operation with servo lag and with feedforward control. M124 influences the point spacing for calculating the rounding arc. Refer to your User's Manual for more information.

3.5 Monitoring functions

The NC monitors the axis positions and the dynamic behaviour of the machine. If the fixed values in the machine parameters are exceeded, an error message is displayed and the machine is stopped. If monitoring is not wanted, it can be deactivated for the following using M2688 to M2692:

- Position monitoring
- Standstill monitoring
- Movement monitoring
- Monitoring of the analog voltage

Marker	Function	Set	Reset
M2688	No monitoring X axis	PLC	PLC
M2689	No monitoring Y axis		
M2690	No monitoring Z axis		
M2691	No monitoring 4th axis		
M2692	No monitoring 5th axis		

The monitoring can be reactivated by resetting the markers in the PLC.

The monitoring functions can also be switched off conditionally. In the machine parameter MP4130 a PLC-input is defined, which is interrogated with the same cycle time as the control loop (TNC 415: 3 ms; TNC 407: 6 ms). The condition for activating this input is stored in MP4131. This function must be activated from the PLC by setting Bit 0 in Word W522.

If monitoring functions are switched off, safe operation of the machine cannot be guaranteed. Uncontrolled movements of the axes are not recognized.

MP4130 Fast PLC input to suppress the monitoring functions
Entry: 0 to 255 (No. of the PLC input)

MP4131 Activation condition for fast PLC input from MP4130
Entry: 0 or 1

0 = activation for Low level
1 = activation for High level

Address	Function	Set	Reset
W522 Bit 0	Monitoring functions suppressed if PLC input from MP 4130 is activated	PLC	PLC

3.5.1 Position monitoring for operation with lag

The machine parameters MP1710 and MP1720 determine the ranges for the continuous position monitoring in the machine (lag monitoring). The monitoring is active as soon as the axes are under the control of the position control loop.

If the limits of parameter MP1710 are exceeded, the error message

"Positioning error"

will appear and the machine stops. The error message can be cancelled by the "CE" key.

If the limit of parameter MP1720 is exceeded, the flashing error message

"Gross positioning error A"

appears.

This error can only be cancelled by switching off the control. An entry value of approx. 1 to 1.4 times lag for rapid traverse is realistic. MP1720 is larger than MP1710.

MP1710	Position monitoring for operation with lag (cancellable) Entry: 0.0000 to 300.0000 (mm)
MP1710.0	Axis X
MP1710.1	Axis Y
MP1710.2	Axis Z
MP1710.3	4th axis
MP1710.4	5th axis

MP1720	Position monitoring for operation with lag (EMERGENCY STOP) Entry: 0.0000 to 300.0000 (mm)
MP1720.0	Axis X
MP1720.1	Axis Y
MP1720.2	Axis Z
MP1720.3	4th axis
MP1720.4	5th axis

3.5.2 Position monitoring for operation with feedforward control

In operation with feedforward control the ranges for continuous position monitoring are determined by MP1410 and MP1420 lag monitoring. The monitoring is active as soon as the axes are under the control of the position control loop.

If the limits of MP1410 are exceeded, the error message

"Position error"

will appear, which can be cancelled with the "CE" key. If the limit of parameter MP1420 is exceeded, the flashing error message

"Gross positioning error A"

appears. This error can only be cancelled by switching off the control.

If clamped axes cause the error message "Position error", a residual voltage can remain at the nominal value output, since the machine axes can no longer be traversed. MP1150 can be used to program a time after which the residual voltage becomes zero. After this period a nominal/actual value transfer is carried out. If the error message is cancelled by "CE" before the time has elapsed the nominal/actual value transfer is carried out and the residual voltage is switched off.

The machine parameter functions in operation with lag as well as with feedforward control.

MP1410	Position monitoring in operation with precontrol (cancellable) Entry: 0.0010 to 30.0000 (mm)
MP1410.0	Axis X
MP1410.1	Axis Y
MP1410.2	Axis Z
MP1410.3	4th axis
MP1410.4	5th axis
MP1420	Position monitoring in operation with precontrol (EMERGENCY STOP) Entry: 0.0010 to 30.0000 (mm)
MP1420.0	Axis X
MP1420.1	Axis Y
MP1420.2	Axis Z
MP1420.3	4th axis
MP1420.4	5th axis
MP1150	Delay time before switching off the residual voltage on error message "Position error" Entry: 0 to 65.535 (s)

3.5.3 Monitoring the analogue voltage

Monitoring of the analogue voltage is only possible in operation with feedforward control.
If it exceeds 10 V, the flashing error message

"Gross positioning error B"

is displayed.

3.5.4 Movement-monitoring

The movement-monitoring functions in operation with feedforward control as well as with lag.

During movement monitoring, the actual path travelled is compared with a nominal path calculated by the NC at short intervals (several servo cycles). If, during this period, the actual path travelled differs from the calculated path, the flashing error message

"Gross positioning error C"

is displayed.

In machine parameter MP1140 a voltage can be stored, below which the movement-monitoring is out of action.

If 10 [V] is entered in this machine parameter, the movement-monitoring becomes inactive. It is not possible to operate the machine safely without movement monitoring.

MP1140	Movement monitoring Entry: 0.030 to 10.000 (V)
MP1140.0	Axis X
MP1140.1	Axis Y
MP1140.2	Axis Z
MP1140.3	4th axis
MP1140.4	5th axis

3.5.5 Standstill monitoring

The Standstill monitoring operates with feedforward control and with lag. The monitoring begins when the axes have reached the positioning window. The range in which the axes may move is defined in MP1110.

As soon as position deviation is greater than the value which is stored in MP1110, the flashing error message "`GROSS POSITIONING ERROR D`" is displayed. The message also appears if, on running-in to a position, an overshoot occurs which is larger than the value in MP1110, or if the axis moves in the opposite direction on beginning a positioning movement.

MP1110	Standstill monitoring Entry: 0.0010 to 30.0000 (mm)
MP1110.0	X axis
MP1110.1	Y axis
MP1110.2	Z axis
MP1110.3	4th axis
MP1110.4	5th axis

3.5.6 Positioning window

The positioning window defines the limits within which the control considers that a position has been reached. After reaching the position the control starts the execution of the next block. The size of the positioning window is defined in machine parameter MP1030.

During feedforward control the positioning window is not evaluated until the velocity falls below the value in MP1525*. This MP does not exist on old software versions, in which the maximum velocity for positioning window evaluation is fixed at 0.5 mm/min.

If the value which is entered is too small, the run-in time and therefore the time between one program block and the next will be lengthened.

If the axes have reached the positioning window after a movement, the markers M2008 to M2011 and M2017 are set (see section "Axis in position").

MP1030	Positioning window Entry: 0.0001 to 2.0000 (mm)
MP1030.0	X axis
MP1030.1	Y axis
MP1030.2	Z axis
MP1030.3	4th axis
MP1030.4	5th axis

MP1525	Maximum velocity for checking the positioning window Entry: 0.100 to 10.000 [mm/min] Recommended value: 0.5 [mm/min]

* As of software versions 280 54x 02, 280 56x 02 and 280 58x 02

3.6 Controlled axes

The machine parameter MP50 determines which of the five NC-axes should be controlled.

MP50		Controlled axes Entry: %xxxxx
Bit 0	X axis	0 = not controlled
Bit 1	Y axis	1 = controlled
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

Further parameters for the NC-axes may be found in the section "machine axes".

The PLC functions which are described in the following sections are only effective for controlled axes.

3.6.1 Axis-enable, feed rate enable

After switching on the control voltage the "Axis-enable markers" are automatically set by the NC, so that the machine axes can be held in closed position loops by the control .

The axis-enable markers can be reset by the NC if the control loop is opened by the PLC, see section "Open control loop").

Marker	Function	Set	Reset
M2000	Axis enable X	NC	NC
M2001	Axis enable Y		
M2002	Axis enable Z		
M2003	Axis enable 4		
M2016	Axis enable 5		

The feed rate enable must be specified for all axes by M2451 and complement marker M2467 or through byte B520 before the axes can be moved. If the feed rate enable is removed, the analogue voltage is output as 0 V and the axes immediately stop moving. The letter "F" is then highlighted in the status display.

Marker	Function	Set	Reset
M2451	Feed rate enable for all axes	PLC	PLC
M2467	Complement feed rate enable		

Address	Function		
B520	Axis-specific feed rate enable		
	Bit 0	+1	Feed rate enable Axis X
		+0	No Feed rate enable Axis X
	Bit 1	+2	Feed rate enable Axis Y
		+0	No Feed rate enable Axis Y
	Bit 2	+4	Feed rate enable Axis Z
		+0	No Feed rate enable Axis Z
	Bit 3	+8	Feed rate enable 4th axis
		+0	No Feed rate enable 4th axis
	Bit 4	+16	Feed rate enable 5th axis
		+0	No Feed rate enable 5th axis

3.6.2 Axes in position

If the axes have reached the defined positioning window (MP1030, see section "Positioning window") after a movement, the "axis in position" markers are set by the NC. This also happens after switching on the control voltage.

As soon as a positioning is started, the markers are reset by the NC. This is also valid when passing over the reference marks. In the "Electronic handwheel" mode of operation the markers M2008 to M2011 and M2017 are reset.

For contours which can be machined with constant feed rate the "Axis in position" markers are not set.

Marker	Function	Set	Reset
M2008	X axis in position	NC	NC
M2009	Y axis in position		
M2010	Z axis in position		
M2011	4th axis in position		
M2017	5th axis in position		

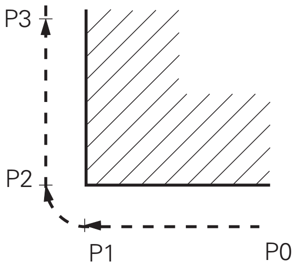
3.6.3 Axes in motion

If the axes are in motion the appropriate "Axis in motion" markers are set by the NC.

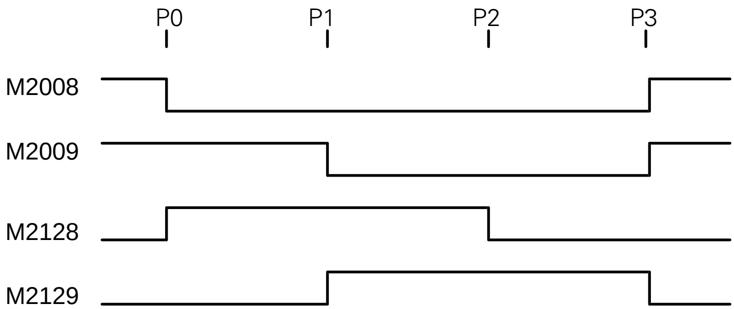
Note: The markers are only set in the operating modes "Positioning with manual entry", "Single block" and "Full sequence".

Marker	Function	Set	Reset
M2128	X axis in motion	NC	NC
M2129	Y axis in motion		
M2130	Z axis in motion		
M2131	4th axis in motion		
M2132	5th axis in motion		

Example for markers "Axis in position" and "Axis in motion":




```
.
.
.
L X80  Y20  RL
L X0        RL
L X80       RL
.
.
.
```



3.6.4 Open control loop

After Marker M2830 is set, the control loops of each axis and of the spindle are opened, and an NC stop is performed. This makes it possible, for example, for the control loops to be opened at the same time as the drives are switched off. If Marker M2827 is set, no external EMERGENCY STOP (Input I3 “Control is operational”) is sent to the NC; instead, the spindle and the position controllers are opened in all axes, and an NC stop is carried out.

If, after the execution of an NC block, the control loop for a particular axis is opened and this axis is clamped, then it is necessary to delay this opening to give the clamp sufficient time to operate. The markers M2492 to M2495 and M2500 serve this purpose. If one of these markers is set, and the appropriate axis is in position, then the next NC block will only then be processed when the "Open control loop" marker (M2544 to M2547, M2507) has been set. After the control loops are opened the markers M2000 to M2003 and M2016 are reset by the NC. As soon as the "Axis in position" markers are reset, the control loop must be closed again so that the axis can be moved.

Marker	Function	Set	Reset
M2544	Open control loop X axis	PLC	PLC
M2545	Open control loop Y axis	PLC	PLC
M2546	Open control loop Z axis	PLC	PLC
M2547	Open control loop 4th axis	PLC	PLC
M2507	Open control loop 5th axis	PLC	PLC
M2492	Await open control loop X axis	PLC	PLC
M2493	Await open control loop Y axis	PLC	PLC
M2494	Await open control loop Z axis	PLC	PLC
M2495	Await open control loop 4th axis	PLC	PLC
M2500	Await open control loop 5th axis	PLC	PLC
M2827	Suppress EMERGENCY STOP, open the control loop, NC stop	PLC	PLC
M2830	NC stop and open the control loop	PLC	PLC

3.6.5 Actual - nominal value transfer

If the markers M2552 to M2555 and M2505 are set, then the actual positional value will be transferred to the nominal value.

Actual value transfer is only possible in the "Manual" and Electronic handwheel" modes of operation.

Marker	Function	Set	Reset
M2552	Actual - nominal value transfer X axis	PLC	PLC
M2553	Actual - nominal value transfer Y axis	PLC	PLC
M2554	Actual - nominal value transfer Z axis	PLC	PLC
M2555	Actual - nominal value transfer 4th axis	PLC	PLC
M2505	Actual - nominal value transfer 5th axis	PLC	PLC

4 Main Spindle

The spindle is controlled by the PLC.

The spindle speed (S output) can be either an analogue voltage output or a coded output. The spindle speed is programmed in rpm in the NC-program.

The machine parameter MP3010 determines whether the spindle speed is a coded output or an analogue output, and whether the spindle orientation should be active.

Up to eight gear ranges can be defined by machine parameters.

Tapping cycles are available for tapping with and without floating tap holder.

MP3010 Spindle-speed output, Spindle orientation
Entry: 0 to 8

0 = No spindle-speed output

1 = Coded output of the spindle speed,
only if the spindle speed changes

2 = Coded output of the spindle-speed at every TOOL CALL

3 = Analogue output of the spindle speed and gear-change signal,
only when the gear range is changed

4 = Analogue output of the spindle speed and gear-change signal
at every TOOL CALL

5 = Analogue output of the spindle speed and no gear-change signal

6 = As for Entry value 3, but with controlled spindle for orientation

7 = As for Entry value 4, but with controlled spindle for orientation

8 = As for Entry value 5, but with controlled spindle for orientation

The functions are described in detail in the following sections.

4.1 Analogue output of the spindle speed

For analogue output of the spindle speed (S-analogue) spindle speeds can be programmed from 0 to 99 999.999 rpm. The programmed spindle speed is available as an analogue DC-voltage of 0 to 10 V on connector X8 of the logic unit. In addition, gear change signals can be assigned to any PLC-outputs by the PLC program.

For S-analogue the value 991 must be entered in the machine parameter MP3020!

If the control is operating with S-analogue, marker M2042 is set. The programmed spindle speed is stored in D356 in 1/1 000 [rpm]. The nominal spindle speed is stored in Word W320 in rpm. The actual spindle speed is stored in Word W322 in rpm.

Marker	Function	Set	Reset
M2042	Analogue output of the spindle speed	NC	NC
D356	Programmed spindle speed	NC	NC
W320	Nominal spindle speed	NC	NC
W322	Actual spindle speed	NC	NC

4.1.1 Direction of spindle rotation

The polarity of the S-analogue voltage is defined by machine parameter MP3130. The count direction of the measuring system signals for the spindle is defined in MP3140. The direction of spindle rotation can be reversed by setting marker M2489, i.e. polarity is inverted. This can become necessary, for example, when shifting spindle transmission between horizontal and vertical.

The S-analogue voltage appears at the output as soon as the marker M2485 for M03 or M2486 for M04 is set. At the same time the miscellaneous function is displayed in the status window of the control (see section "Display and operation"). Marker M2487 turns off the analogue voltage and displays M05.

If more than one of the markers M2485 to M2487 is set at the same time, this will release the flashing error message

"Error in PLC-program 1Q"

Marker M2608 can be used to remove the spindle-enable, at the same time M03, M04 or M05 is displayed inversely. The S-analogue voltage will then go to 0 V.

MP3130 Polarity of the S-analogue voltage
Entry: 0 to 3

0 = M03 positive voltage
M04 negative voltage
1 = M03 negative voltage
M04 positive voltage
2 = M03 and M04 positive voltage
3 = M03 and M04 negative voltage

MP3140 Count direction of the measuring system signals for the spindle

0 = positive count direction with M03
1 = negative count direction with M03

Marker	Function	Set	Reset
M2489	Change direction of spindle rotation	PLC	PLC
M2485	Status display and output of the analogue voltage for M03	PLC	PLC
M2486	Status display and output of the analogue voltage for M04	PLC	PLC
M2487	Status display M05 and spindle-stop	PLC	PLC
M2608	Inverse display of M03, M04, M05 and S-analogue = 0 V [no spindle-enable)	PLC	PLC

4.1.2 Gear range

With the aid of the machine parameters MP3510 and MP3210 up to eight gear ranges can be defined. The nominal speed for S-override 100 % and a matching S-analogue voltage is entered for each gear range.
If not all gear ranges are required, then the remaining machine parameters are set to zero.

Machine parameter MP3515 can be used to define a maximum achievable spindle speed for each gear range which cannot be exceeded by S-override.

The voltage range can be restricted by machine parameter MP3240.1.

MP3120 can determine whether zero spindle speed is permitted, in spite of a minimum producable analogue output voltage.

If an S-analogue voltage output 0 V is produced, marker M2005 will be set. This is also the case when the potential has been turned down to zero with the override potentiometer.

The ramp slope for the S-analogue voltage on output of M03, M04 is stored in MP3410.0.

If the spindle voltage is on a rising or falling ramp, marker M2004 will be set. This also happens when the spindle voltage is altered with the override potentiometer, the voltage changes very quickly, and the ramp of MP3410.0 cannot be followed.

MP3510	Spindle speed for gear ranges Entry: 0 to 99 999.999 [rpm]
MP3510.0	Spindle speed for gear range 1
MP3510.1	Spindle speed for gear range 2
MP3510.2	Spindle speed for gear range 3
MP3510.3	Spindle speed for gear range 4
MP3510.4	Spindle speed for gear range 5
MP3510.5	Spindle speed for gear range 6
MP3510.6	Spindle speed for gear range 7
MP3510.7	Spindle speed for gear range 8
MP3515	Maximum spindle speed Entry: 0 to 99 999.999 [rpm]
MP3515.0	Maximum spindle speed, gear range 1
MP3515.1	Maximum spindle speed, gear range 2
MP3515.2	Maximum spindle speed, gear range 3
MP3515.3	Maximum spindle speed, gear range 4
MP3515.4	Maximum spindle speed, gear range 5
MP3515.5	Maximum spindle speed, gear range 6
MP3515.6	Maximum spindle speed, gear range 7
MP3515.7	Maximum spindle speed, gear range 8
MP3210	S-analogue voltage for gear range Entry: 0 to 9.999 [V]
MP3210.0	S-analogue voltage, gear range 1
MP3210.1	S-analogue voltage, gear range 2
MP3210.2	S-analogue voltage, gear range 3
MP3210.3	S-analogue voltage, gear range 4
MP3210.4	S-analogue voltage, gear range 5
MP3210.5	S-analogue voltage, gear range 6
MP3210.6	S-analogue voltage, gear range 7
MP3210.7	S-analogue voltage, gear range 8
MP3240.1	Minimum S-analogue voltage output Entry: 0 to 9.999 [V]
MP3120	Zero spindle speed permitted Entry value 0:S = 0 permitted Entry value 1:S = 0 not permitted
MP3410.0	Ramp slope for spindle for M03, M04, M05 Entry: 0 to 1.9999 [V/ms]

Marker	Function	Set	Reset
M2004	S-analogue voltage not on the ramp	NC	NC
M2005	S-analogue voltage = 0 V	NC	NC

4.1.3 S-Override

The spindle speed can be altered within certain limits by the S-override potentiometer . These limits are fixed by machine parameter MP3310. However, the upper spindle speed-limit from MP3515 cannot be exceeded by S-override.

The %-factor which is adjusted by S-override is kept in Word W492 and W764. However, W764 be changed by the PLC. As soon as a different value is entered here it will be adopted by the NC. In this way the %-factor can be influenced by the PLC .

If W764 is not changed by the PLC, the normal override-function remains active.

MP7620 bit 3 determines whether the override is effective in 1% steps or according to a non-linear characteristic curve. In the non-linear characteristic, the lower range changes in 0.01% steps, and in the range over 2.5% the step is 0.75%. The range of override values in W492 and W764 lies between 1 and 150 in 1% steps for the linear characteristic, and between 0 and 15 000 for the non-linear characteristic.

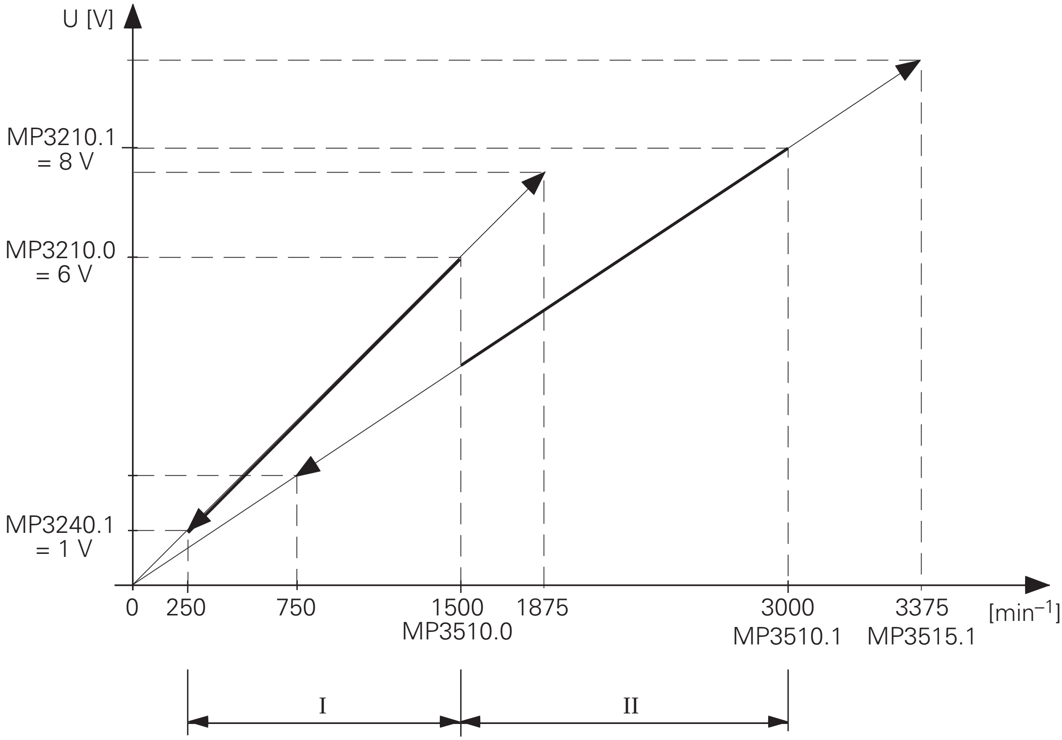
MP3310.0 Limit with S-override (maximum)
Entry: 0 to 150 [%]

MP3310.1 Limit with S-override (minimum)
Entry: 0 to 150 [%]

Word	Function	Set	Reset
W492	%-factor - spindle-override (NC → PLC)	NC	NC
W764	%-factor - spindle override (PLC → NC)	NC; PLC	NC; PLC

MP7620 Feed-rate and spindle override
Entry: %xx0x
Bit 3 Override in 1% steps or non-linear
0 = 1% steps
1 = non-linear

The following diagram explains the relationships:



Gear range I: 1500 rpm at 6V (MP3510.0, MP3210.0)
Gear range II: 3000 rpm at 8V (MP3510.1, MP3210.1)
Upper S override limit: 125% (MP3310.0)
Lower S override limit: 50% (MP3310.1)
Max. output speed for gear range II: 3375 rpm (MP3515.1)
Minimum output voltage: 1V (MP3240.1)

4.1.4 Gear change

Gear change is controlled by the PLC-program. Up to eight gear ranges are available, coded in Word W256. The NC enters the gear code for the gear ranges one to eight in Word W256 according to the programmed spindle speeds in the NC-program (see MP3510).
After setting the gear-code, the marker M2043 is set by the NC as a change signal.

The spindle speed which is programmed in the NC-program as TOOL CALL S is stored in Doubleword D356 and D756 in 1/1000 [rpm] by the NC.

If a different spindle speed (from the spindle speed fixed by the NC) is to be activated by the PLC, the speed must be entered by the PLC in D756, this can then be activated by marker M2814 .

A different gear range from that which is selected by the NC can be activated by setting Word W256 and marker M2814 with the PLC. Word W256 remains unchanged until the next gear change signal. Marker M2814 is reset by the NC after the gear change .

Check that the spindle speed selected by the PLC is within the spindle-speed limits of the selected gear range.

The PLC-program must transfer the gear-code to the machine. The NC waits for the acknowledgement signal "Gear change completed" (M2480) before continuing the program . As soon as marker M2480 is set by the PLC the NC resets the strobe signal M2043. The marker M2480 may only be set in one PLC-cycle.

Marker	Function	Set	Reset
M2043	Gear-code change signal	NC	NC
M2480	Acknowledgement "Gear change completed"	PLC	PLC
M2814	Activation of a gear range and spindle speed via PLC	PLC	NC
D356	Programmed spindle speed	NC	
D756	Set spindle-speed from the PLC; programmed spindle speed	NC, PLC	
W256	G-code for S-analogue	NC	

An alternating S-analogue voltage can be produced for changing gear. For this purpose the marker M2490 and the marker M2491 must be alternately set and reset by the PLC. This may be achieved by interrogating a particular timer in the PLC-program. The markers M2490 and M2491 are reset to stop the spindle. The output voltage is defined by MP3240.2.

Marker	Function	Set	Reset
M2490	Spindle rotation left (for gear change)	PLC	PLC
M2491	Spindle rotation right (for gear change)	PLC	PLC

MP3240.2 Jog-voltage for gear change
Entry: 0.000 to 9.999 (V)

The direction of spindle rotation after the gear change is determined by the PLC-program with the markers M2485 to M2487 (see "Direction of spindle rotation").

Machine parameter MP3030 can be used to determine whether an axis-halt should occur on a TOOL CALL with only a spindle speed output.

If an axis-halt is not permitted, then no PLC-positioning, datum-correction, spindle-orientation or change of limit switch range may be performed during the S-code output.

MP3030 Axis-halt on TOOL CALL with only a spindle speed output.
1 = No axis-halt on TOOL CALL
0 = Axis-halt on TOOL CALL

If a non-permissible spindle speed is programmed, the marker M2092 will be set by the NC, and simultaneously the error message "Wrong spindle speed" is displayed.

Marker	Function	Set	Reset
M2092	Prohibited spindle speed	NC	NC

PLC example for gear change and jog-voltage output:

```
.       .           .
.       .           .
.       .           .
L       M2043       ;Gear-code change signal for S-analogue
CMT     50          ;Module-call (output gear-code )
.       .           .
.       .           .
.       .           .
L       M2043       ;Gear-code change signal for S-analogue
AN      M968        ;Buffer marker for T13
AN      T13
AN      T14
AN      T61         ;Timer 13 for spindle-left, running
AN      T62         ;Timer 14 for spindle-right, running
=       T13         ;Set timer 13  (value from MP4110.13)
S       M968        ;Set buffer marker for T13
L       M2043
AN      M969        ;Buffer marker for T14
AN      T13
AN      T14
AN      T61
AN      T62
=       T14         ;Set timer 14 (value from MP4110.14)
S       M969        ;Set buffer marker for T14
L       T61         ;Timer 13 running
=       M2490       ;Spindle-left for gear change
L       T62         ;Timer 14 running
=       M2491       ;Spindle-right for gear change
LN      T13
AN      T61
R       M968        ;Reset buffer marker
LN      T14
AN      T61
R       M968        ;Reset buffer marker
.       .           .
.       .           .
.       .           .
L       M2043
A       I5          ;Input for acknowledgement signal
S       M2480       ;Acknowledgement - gear changed
.       .           .
.       .           .
EM                  ;End main program


LBL50               ;Module start
L       M2043       ;Gear-code change signal
L       W256        ;Gear-code for S-analogue
B=      015         ;Outputs 15, 16 and 17 for gear change are activated
EM                  ;Module end
```

4.1.5 Offset adjustment

Spindle orientation is followed by automatic offset adjustment. To give the spindle sufficient time to recover, offset adjustment does not start until the spindle has been in position for at least two seconds. Offset is then compensated cyclically by 0.152 mV at one second intervals.

4.2 BCD-coded output of spindle speed

For coded output of the spindle speed an S-code is entered in Word W258 by the NC. At the same time strobe marker M2044 is set by the NC. The signals are static. The NC delays the continuation of program run until marker M2481 gives the acknowledgement that the new spindle speed has been activated. On acknowledgement the strobe marker M2044 is reset. The S-code is not altered until the next S-output.

A spindle speed is entered in the TOOL CALL-block with a maximum of four figures in rpm, if necessary, rounded off by the NC to the nearest standard value. Spindle speeds of 0 to 9 000 are possible. The spindle speed which has been entered is put out by the control as S-Code according to DIN 66025, as shown in the list of S-Codes.
With MP3020 the minimum and maximum spindle speeds and the desired spindle-speed steps can be defined.

The entry value is a 5-figure number, derived as follows:

Minimum spindle speed	00 to 99	(see S-code-table)
Maximum spindle speed	00 to 99	(see S-code-table)
Spindle-speed step	1 to 9	

Example:

The minimum spindle speed is to be 1 rpm (S-code 20), the maximum spindle speed is to be 1000 rpm (S-code 80). Only every second spindle speed should be programmable. This gives an entry value of 20802 for MP3020. An entry value of 991 would mean no limit.
The S-code is stored in the PLC in Word W258. The minimum spindle speed from MP3020 is stored in Word W1008.

Marker	Function	Set	Reset
M2044	S-code change signal	NC	NC
M2481	S-code acknowledgement	PLC	PLC
W258	S-code	NC	NC
W1008	S-code for minimum spindle speed	NC	

MP3020 Definition of the spindle speed range
Entry: 1 to 99,999

S-code table

S-function code	rpm
S 00	0
S 01	0.112
S 02	0.125
S 03	0.14
S 04	0.16
S 05	0.18
S 06	0.2
S 07	0.224
S 08	0.25
S 09	0.28
S 10	0.315
S 11	0.355
S 12	0.4
S 13	0.45
S 14	0.5
S 15	0.56
S 16	0.63
S 17	0.71
S 18	0.8
S 19	0.9
S 20	1
S 21	1.12
S 22	1.25
S 23	1.4
S 24	1.6
S 25	1.8
S 26	2
S 27	2.24
S 28	2.5
S 29	2.8
S 30	3.15
S 31	3.55
S 32	4
S 33	4.5
S 34	5
S 35	5.6
S 36	6.3
S 37	7.1
S 38	8
S 39	9
S 40	10
S 41	11.2
S 42	12.5
S 43	14
S 44	16
S 45	18
S 46	20
S 47	22.4
S 48	25
S 49	28

S-function code	rpm
S 50	31.5
S 51	35.5
S 52	40
S 53	45
S 54	50
S 55	56
S 56	63
S 57	71
S 58	80
S 59	90
S 60	100
S 61	112
S 62	125
S 63	140
S 64	160
S 65	180
S 66	200
S 67	224
S 68	250
S 69	280
S 70	315
S 71	355
S 72	400
S 73	450
S 74	500
S 75	560
S 76	630
S 77	710
S 78	800
S 79	900
S 80	1000
S 81	1120
S 82	1250
S 83	1400
S 84	1600
S 85	1800
S 86	2000
S 87	2240
S 88	2500
S 89	2800
S 90	3150
S 91	3550
S 92	4000
S 93	4500
S 94	5000
S 95	5600
S 96	6300
S 97	7100
S 98	8000
S 99	9000

4.3 Spindle-orientation

In order to use the TNC on machines with automatic tool-changers an orientation of the main spindle (S-axis) is necessary. This requires a controllable spindle. The orientation is carried out by a miscellaneous function M19 or another M-function, and must be initiated by the PLC-program. The spindle orientation functions asynchronously to the NC positioning movements. The PLC must not acknowledge the oriented stop until M2007 is set. MP3010 determines whether the control operates with or without spindle-orientation (see Section 4, "Main spindle").

There are three ways in which the PLC can orient the spindle:

- with Module 9171
- with Marker M2712
- through initiator with M2501

Spindle orientation with Module 9171
Module 9171 makes it possible to define the velocity, angle and the direction of rotation for spindle orientation. Marker M2712 is set as long as the spindle is being oriented. See the description of Module 9171 in the "PLC-Programming" chapter.

Spindle orientation with Marker M2712
If spindle orientation is activated with M2712, the nominal position is taken from D592. The nominal position is based on the reference point and can, for example, be set in machine parameter MP4210.x. It must be copied into the Doubleword D592. The nominal position can also be take from the "Orientation" cycle (CYCL DEF 13). In this case, MSB of D592 = 1 and the other bits are set to 0. This must be realized through the PLC program. Marker M2408 is set during execution of Cycle 13. The speed of spindle orientation is taken from MP3520.1.

For the orientation of a stationary spindle, the direction is taken that has the shortest path to the stop position, provided that at the start the distance between the nominal and actual positions is not greater than the positioning window (MP3420). If the distance is greater than the positioning window, the spindle will be positioned according to Marker M2656 with M03 or M04.

Sequence of spindle orientation with Marker M2712 or Module 9171
The spindle speed is reduced along the ramp from MP3410.1 to the speed for spindle orientation. Once this speed has been reached the spindle is in the control loop and is oriented to the nominal position along the ramp from MP3410.1.

If the spindle does not have to remain in the control loop after reaching the nominal position, marker M2499 must be set. The spindle is not free of the control loop until this marker has been set. If M2499 remains set, the control loop is opened after each orientation as soon as the positioning window is reached.

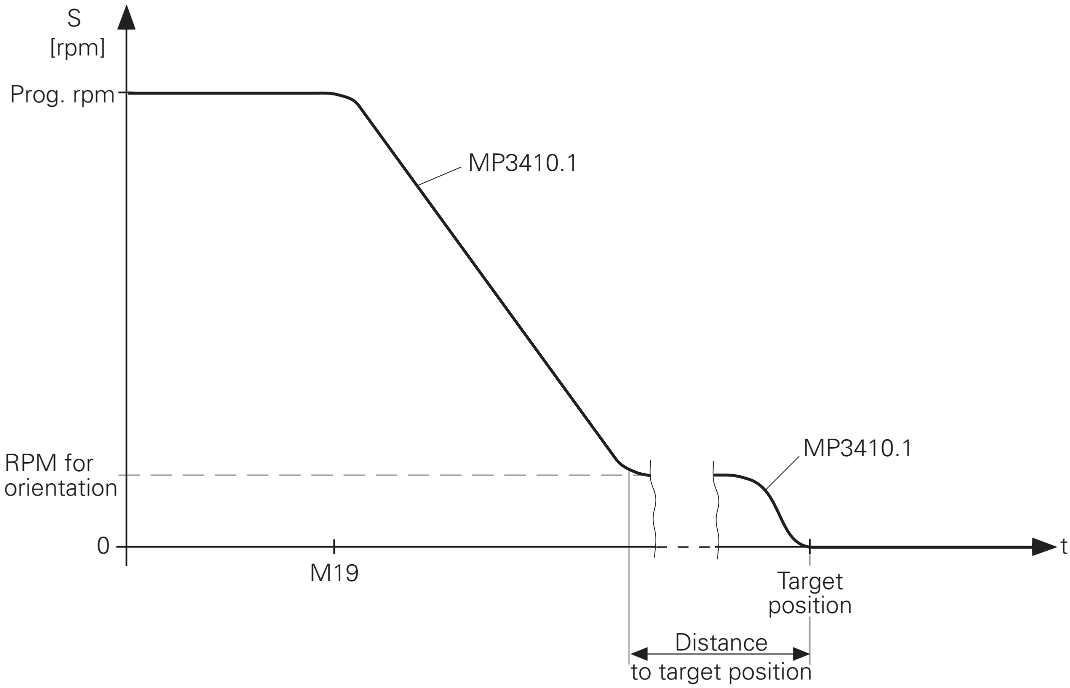
The spindle can be oriented down to 0.1 degree. This value results from the line number of the installed rotary encoder (ROD 426.xxx8 with 1024 lines) and the 4-fold evaluation in the TNC. A positioning window is defined with MP3420. Marker M2007 is set once the spindle is in the positioning window after orientation.

Machine parameter MP3430 was introduced to make it possible to easily compensate rotary encoder misalignment resulting from imprecise mounting. The offset of the nominal and actual position of the reference mark is entered here for compensation during spindle orientation.

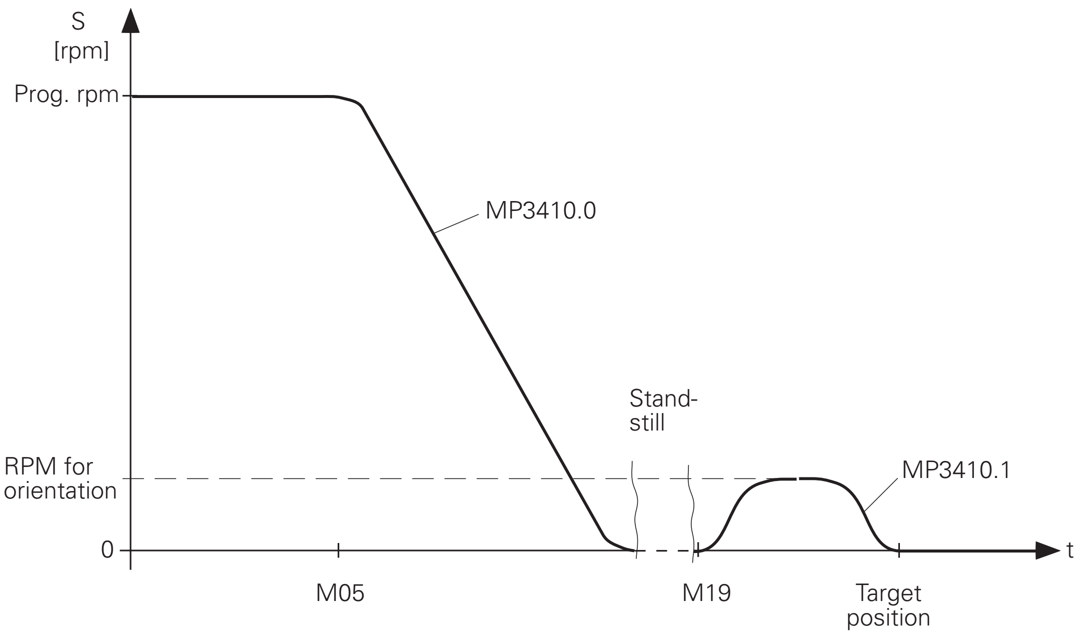
The reference mark is immediately evaluated by the NC the first time the spindle is switched on. The reference mark can be evaluated once again for special applications if Marker M2615 is set.

A separate k_v-factor for each gear range can be entered in MP3440 for a more exact control of the spindle orientation. A display of the spindle position is activated with MP40. The spindle position is displayed only if neither M03 nor M04 is active. It is displayed as a value below 360 degrees.

Orientation from rotation:



Orientation from standstill:



Spindle orientation through initiator with Marker M2501

The spindle can also be oriented through an initiator. This requires that Marker M2501 be set by the PLC. This marker activates a spindle rotation at the speed defined in machine parameter MP3520.0 and in the direction of rotation defined in Marker M2656. If Marker M2501 is reset by the PLC (e.g. through initiator), the spindle stops. The position value is shown in the status window. This method can also be used to a enable a jog-mode spindle orientation.

MP3410.1 Spindle ramp slope for spindle orientation
Entry: 0.0000 to 1.9999 [V/ms]

MP3420 Positioning window for spindle
Entry: 0 to 65 535 [Increments]
1 Increment represents approx. 0.088 degrees

$$= \frac{360 \text{ degrees}}{1024 \text{ lines} \times \text{fourfold evaluation}}$$

MP3430 Deviation of the reference mark from the desired position (spindle preset)
Entry: 0 to 360 [degrees]

MP3440 k_v-Factor for spindle orientation
Entry: 0.1 to 10 [$\frac{1000°/\text{min}}{°}$]

MP3440.0 k_v-Factor 1st gear range
MP3440.1 k_v-Factor 2nd gear range
MP3440.2 k_v-Factor 3rd gear range
MP3440.3 k_v-Factor 4th gear range
MP3440.4 k_v-Factor 5th gear range
MP3440.5 k_v-Factor 6th gear range
MP3440.6 k_v-Factor 7th gear range
MP3440.7 k_v-Factor 8th gear range

MP3520.0 Spindle speed activated by Marker M2501
Entry: 0.000 to 99 999.999 [rpm]

MP3520.1 Spindle speed for spindle orientation
Entry: 0 to 99 999.999 [rpm]

Marker	Function	Set	Reset
M2007	Spindle in Position	NC	NC
M2127	Spindle in motion (controlled spindle)	NC	NC
M2408	Cycle 13 is being executed	NC	PLC
M2499	Open the spindle control loop	PLC	PLC
M2501	Activates spindle-speed MP3520.0 and direction of rotation from Marker M2656	PLC	PLC
M2615	Re-evaluation of the reference marks for spindle orientation	PLC	NC
M2656	Spindle orientation from stop 0 = Orientation with M03 1 = Orientation with M04	PLC	PLC
M2712	Activate PLC positioning for spindle orientation	NC, PLC	NC
D592	Nominal position for spindle orientation (Strobe M2712)		

Example of a PLC-program for spindle-orientation with M19/M20

Program description:
Using the M-Function M19 to activate a spindle-orientation to the value which is determined by the spindle-orientation cycle.

Using the M-Function M20 to activate a spindle-orientation to the value which is stored in MP4210.47.

```
;Main program
LN      M1          ;Generate logic ONE
S       M1          ;Buffer marker which is continuously ONE
.       .           .
.       .           .
.       .           .
LN      M2045       ;M-strobe inactive
R       M12         ;Spindle-orientation cycle active
R       M2482       ;Acknowledgement of M-code
```

Activation of the orientation to the value from the HEIDENHAIN cycle

```
L       M1919       ;Decoded M-function 19
A       M2045       ;M-code change signal
AN      M12         ;Orientation-cycle active
CMT     180         ;Activation of the Orientation from the cycle
S       M12         ;Orientation-cycle active
```

Activation of the orientation to the Value of MP4210.47

```
L       M1920       ;Decoded M-function 20
A       M2045       ;M-code change signal
AN      M12         ;Orientation-cycle active
CMT     181         ;Activation of the orientation from the MP
S       M12         ;Orientation-cycle active
```

Acknowledgement of the orientation

```
L       M1919       ;Decoded M-function 19
O       M1920       ;Decoded M-function 20
A       M2045       ;M-code change signal
A       M2007       ;Spindle in position
AN      M2712       ;Strobe for spindle-orientation
S       M2482       ;Acknowledgement of M-code
.       .           .
.       .           .
.       .           .
EM                  ;Main program - end
```

LBL 180; Transfer positional value from the spindle-orientation cycle

```
L       K+O         ;Load the constant value ZEROl
BS      K+31        ;Set Bit 31 to ONE
=       D592        ;Position spindle-orientation
.       .           .
.       .           .
.       .           .
L       M1          ;Buffer marker which is continuously ONE
S       M2712       ;Strobe for spindle-orientation
.       .           .
.       .           .
.       .           .
EM
```

LBL 181; Transfer positional-value from the machine parameter

```
L       D956        ;MP4210.47 PLC-positional-value
=       D592        ;Position spindle-orientation
.       .           .
.       .           .
.       .           .
L       M1          ;Buffer marker which is continuously ONE
S       M2712       ;Strobe for spindle-orientation
.       .           .
.       .           .
.       .           .
EM
```

4.4 Tapping

There are two tapping cycles available in the control:

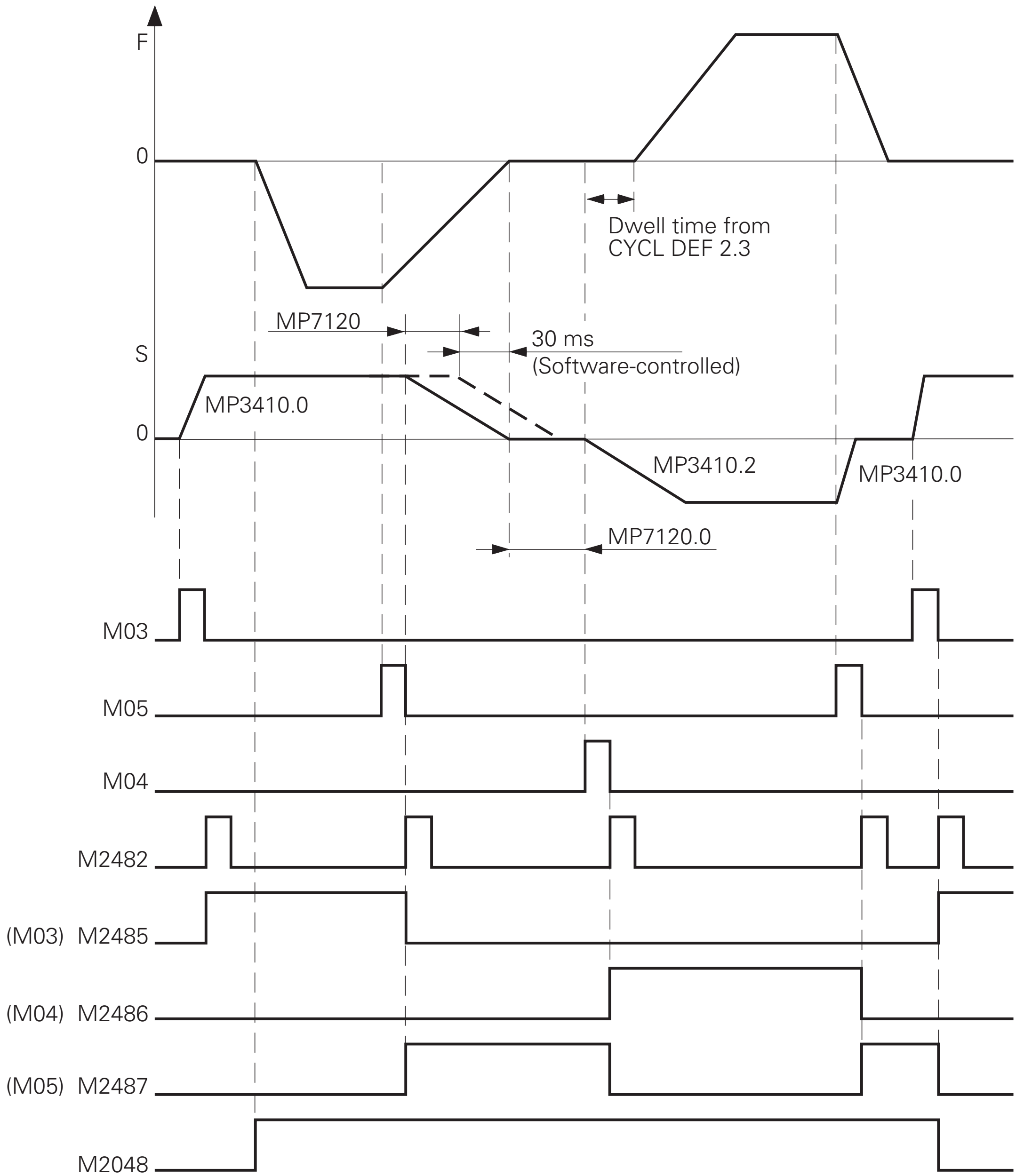
- tapping with floating tap holder and
- rigid tapping.

Tapping with floating tap holder is possible for both analogue and BCD-coded spindle-speed output. Rigid tapping is only possible for analogue spindle-speed output.

The tapping cycle is defined in the NC-program and can be called with CYCL CALL (M03). Tapping is adjusted to the dynamic behaviour of the machine by machine parameters.

4.4.1 Tapping with floating tap holder for analogue spindle-speed output

The following diagram shows the sequence of events in the cycle.



If the tapping cycle is called, marker M2048 is set by the NC.

After switching on the spindle with M03 and successful acknowledgement (marker M2482) the set spindle-speed should have been reached when the feed commences. Upon switch-on, the spindle follows the ramp from MP3410.0, on switching off it follows the ramp from MP3410.2. Switch-off is initiated by the NC with M05. The M functions that are output must be acknowledged.

However, if the feed-ramp is flatter than the spindle-ramp, then the spindle follows the flatter feed-ramp. Switch-off can be delayed by MP7120.2.

Example:
If spindle speed s = 1000 [rpm] = 1.8 [V] and the thread ramp is 0.05 [V/ms], then:
1.8 [V]: 0.05 [V/ms] = 36 ms
The spindle is decelerated 36 ms before reaching the total hole depth. If the spindle deceleration is delayed by the spindle slow-down time, then it is only possible to delay up to 30 ms before reaching the total hole depth. Any greater delay will be ignored. In this example a maximum slow-down time of 6 ms is in effect.

The restarting of the Spindle with M04 can be delayed by MP7120.0. The ramp follows MP3410.2 again. The restarting of the feed can be delayed with the dwell-time which is programmed in the cycle. The dwell-times permit an optimal adjustment of the floating tap holder.

The switch-off of the spindle is again performed by the NC with M05. The switch-off ramp follows MP3410.0. After this, the spindle is switched on again with M03.

The feed-override can only be effective within limits when tapping, otherwise the floating tap holder may be damaged. MP7110 was introduced for this limit.

MP3410.2 Ramp slope of the spindle for tapping
Entry: 0.0000 to 1.9999 (V/ms)

MP7120.0 Dwell-time for change of direction of spindle rotation
Entry: 0.0000 to 65.535 (s)

MP 7120.2 Spindle run-on time after reaching total hole depth
Entry: 0.0000 to 65.535 (s)

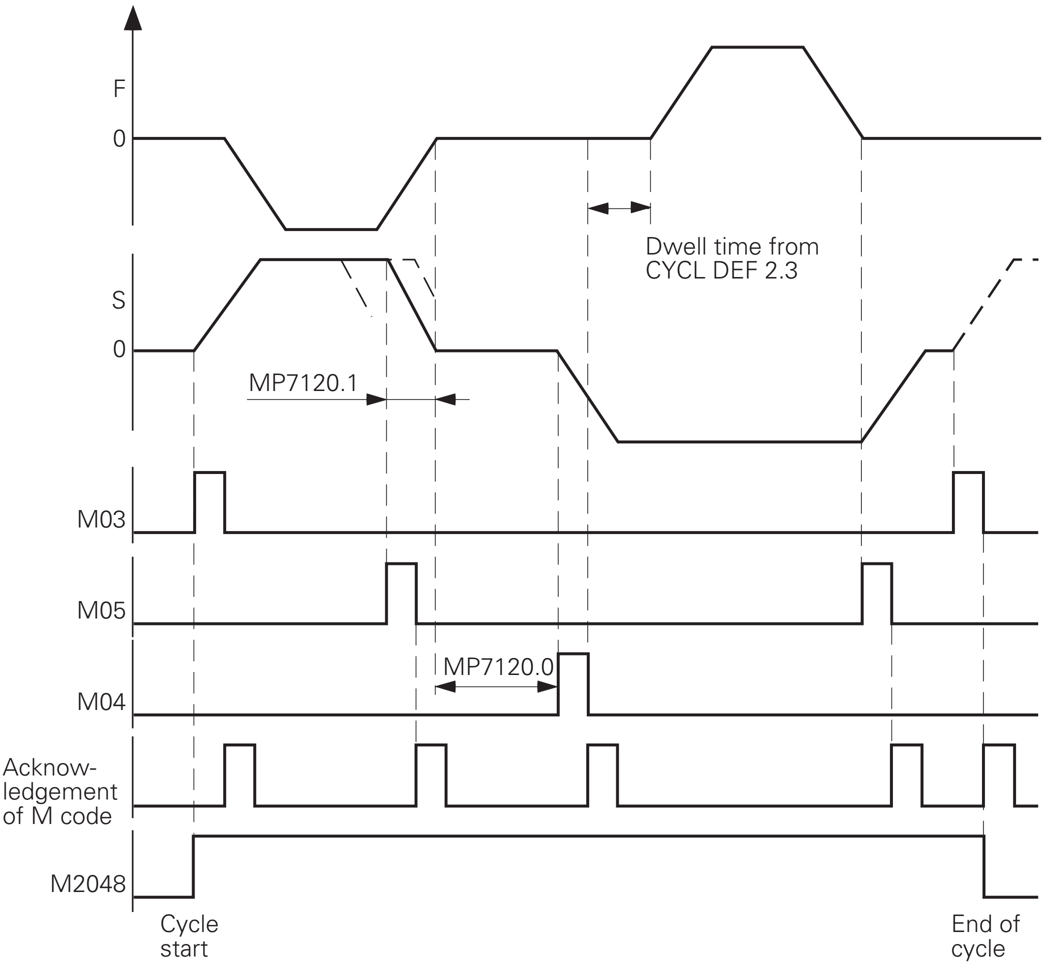
MP7110.0 Minimum for feed-override when tapping
Entry: 0 to 150 [%]

MP7110.1 Maximum for feed-override when tapping
Entry: 0 to 150 [%]

Marker	Function	Set	Reset
M2048	Tapping cycle called	NC	NC

4.4.2 Tapping with floating tap holder and BCD-coded spindle-speed output

The following diagram shows the sequence of events in the cycle:



Since, with BCD-coded spindle speed output, the spindle- and feed ramps cannot be synchronized by the NC, a machine parameter MP7120.1 was introduced to enable the spindle to be switched off early.

The machine parameter MP7120.0 (dwell-time for change of direction of rotation) and the programmable dwell-time are just as effective as for analogue spindle-speed output.

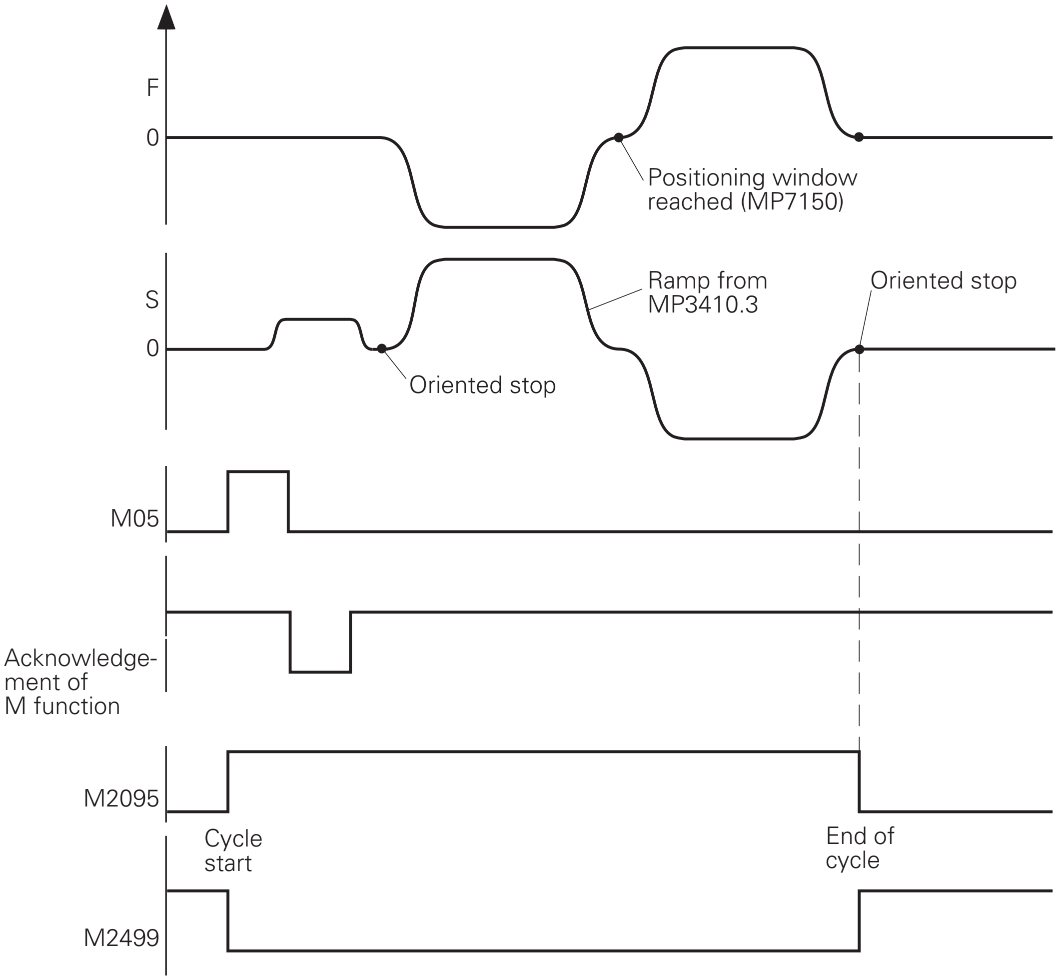
MP7120.1 Pre-cut out time for the spindle when tapping with BCD-coded output
Entry: 0.0000 to 65.535 (s)

4.4.3 Rigid Tapping

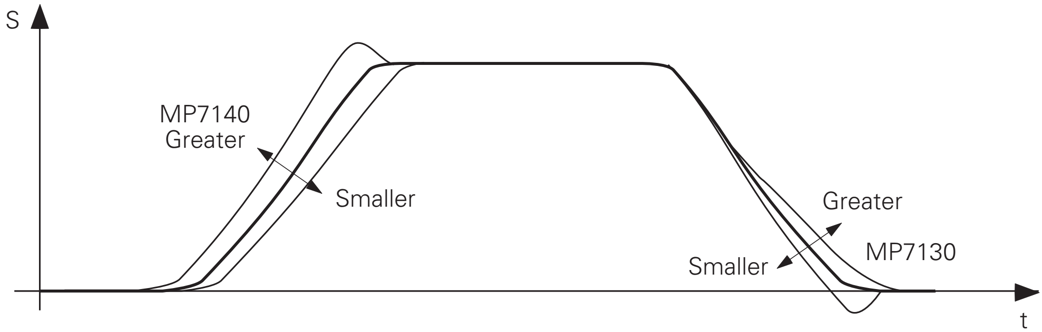
Cycle 17

During rigid tapping the spindle position control loop is open. The machine tool operator defines rigid tapping with Cycle 17 in the NC part program "CYCL DEF 17". While Cycle 17 is being run the TNC automatically switches to operation with velocity feedforward. You define the dynamic behavior of the spindle and the tool axis in machine parameters. The tool axis tracks the actual position of the spindle during tapping. Before tapping begins, the axes, for example Z and S, are synchronized by means of an oriented spindle stop. This means the every Z position is assigned to a corresponding spindle position. The NC carries out the oriented spindle stop. M2127 is set by the NC and in the PLC the spindle position control loop must be closed (M2499).

Synchronization makes it possible to cut the same thread more than once. The permanently assigned spindle position depends on the thread pitch entered in the cycle. To save machining time you can deselect this function with MP7160. Then, however, it is no longer possible to cut the thread more than once. M2095 and M2048 are set while the cycle is running.



The positioning window from MP7150 is effective for the tool axis while Cycle 17 is being run. The input value must be less than or equal to MP1030.x. You define the spindle acceleration and deceleration process in MP3410.3, MP7130 and MP7140. Spindle overshoot is to be avoided during acceleration and approach.



Marker	Function	Set	Reset
M2048	Cycle 2 or cycle 17 active	NC	NC
M2095	Cycle 17 or cycle 18 active	NC	NC
M2499	Open control loop S-axis	PLC	PLC

MP3410.3 Ramp slope of the spindle during rigid tapping
Entry: 0.0000 to 1.9999 (V/msec)

MP7130 Spindle run-in characteristic
Entry: 0.001 to 10.000 [°/min](matching MP1520)

MP7140 Transient response of spindle on acceleration
Entry: 0.001 to 1.000 (matching MP1520)

MP7150 Positioning window for tool axis during rigid tapping
Entry: 0.0001 to 2.0000 [mm]

MP7160 MP7160 Spindle orientation with Cycle 17
Entry: 0 or 1
0 = spindle orientation before Cycle 17
1 = no spindle orientation with Cycle 17

Cycle 18

With Cycle 18 the tool axis tracks the actual position of the spindle. The starting position is the current position. The target position is the total hole depth. Approaching and departing movements must be programmed separately. During Cycle 18 marker M2095 is set. M2499 must be reset so that the cycle is run. MP7130, MP7140 and MP7150 have the same function as for Cycle 17.

5 EMERGENCY STOP-routine

A PLC-input (X42/4) and a PLC-output (X41/34 and in the PL410 B: X8/16) with the designation "control is ready" are available in the control for the EMERGENCY STOP-routine.

If a malfunction is recognized in the control, the TNC switches the control-is-ready output signal off, a flashing error message appears on the VDU-screen and the PLC-program is stopped. This error message can not be cancelled. After removing the fault the switch-on routine must be gone through again.

If the control-is-ready signal is switched off by an event outside the control, the error message EXTERNAL EMERGENCY STOP is displayed and the NC sets the marker M2190 and M2191. This error message can only be cancelled after the control voltage is switched on again.

The external EMERGENCY STOP is evaluated by the control as an external stop. If the external EMERGENCY STOP is triggered during an axis movement, the moving axis is stopped in a controlled manner. If the drive amplifiers are blocked by the external EMERGENCY STOP, the nominal-value output may exceed the position-monitoring fixed by the machine parameter. In this case, the error message POSITIONING ERROR or GROSS POSITIONING ERROR <AXIS> will be displayed.

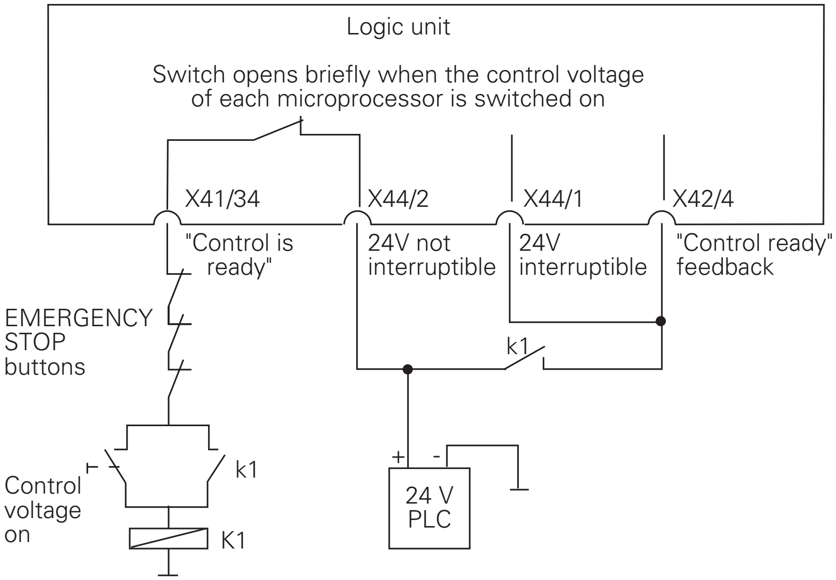
EMERGENCY STOP can be suppressed with Marker M2827. If Marker M2827 is set, an external EMERGENCY STOP is not reported to the NC and, instead, all control loops are opened and the NC stops.

Marker	Function	Set	Reset
M2190	Cancellable error message is displayed	NC	NC
M2191	Error message "External EMERGENCY STOP" is displayed	NC	NC
M2827	Suppress EMERGENCY STOP, open the control loop, stop NC	PLC	PLC

5.1 Connection diagram

Under fault conditions the control-is-ready output should switch off the 24-volt supply. Because of the enormous importance of this function this output is tested by the control every time the mains power is switched on.

Basic circuit diagram:

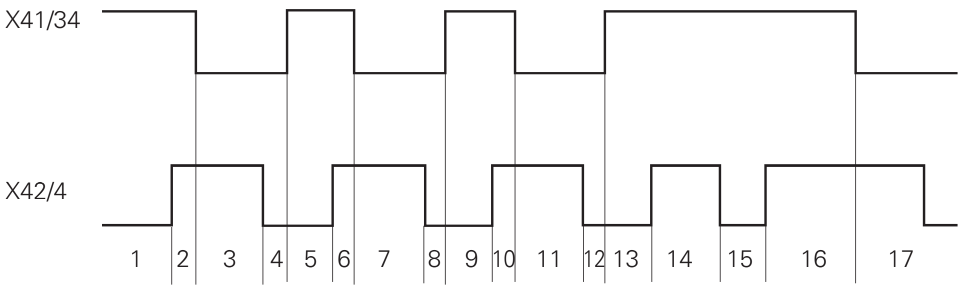


This is merely a suggestion for switching. The machine manufacturer must ensure that all necessary safety specifications are met.

5.2 Flow-diagram

The external electronics must meet the specified conditions. In particular, the acknowledgement for "Control ready" must reach the TNC 415 after a maximum of 114 ms (for the TNC 407: 146 ms) .

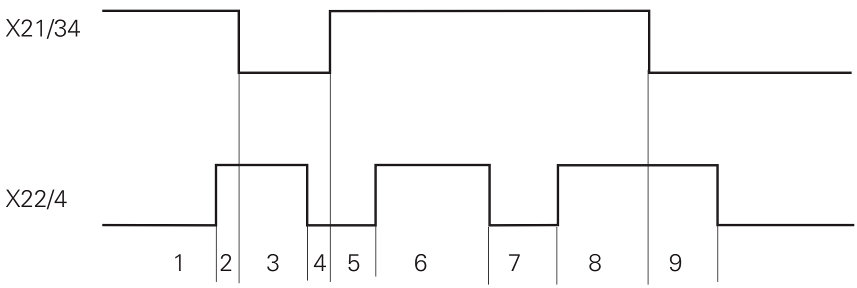
5.2.1 TNC 415



		VDU display
1.	Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
2.	Recognize the control voltage on X42/4 and switch-off of the control-is-ready signal on X41/34 by the first microprocessor (t < 66 ms).	
3.	Maximum time during which the acknowledgement of Control readiness on X42/4 must go to 0 (t < 114 ms). If exceeded	EMERGENCY STOP DEFECTIVE
4.	Recognize the acknowledgement and set X41/34 (t < 20 ms).	
5.	Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
6.	Recognize the control voltage on X42/4 and switch-off of the control-is-ready signal on X41/34 by the second microprocessor (t < 66 ms).	
7.	Maximum time during which the acknowledgement of control readiness on X42/4 must go to 0. (t < 114 ms). If exceeded	EMERGENCY STOP DEFECTIVE
8.	Recognize the acknowledgement and set X41/34 (t < 20 ms).	
9.	Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
10.	Recognize the control voltage on X42/4 and switch-off of the control-is-ready signal on X41/34 by the third microprocessor (t < 66 ms).	

11.	Maximum time during which the acknowledgement of control readiness on X42/4 must go to 0. (t < 114 ms). If exceeded	EMERGENCY STOP DEFECTIVE
12.	Recognize the acknowledgement and set X41/34 (t < 20 ms).	
13.	Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
14.	Normal control operation. Output and control-is-ready acknowledgement signals are high.	
15.	Control voltage is switched off externally.	EMERGENCY STOP
16.	After switching on the control voltage again, the error message can be cancelled, followed by normal control operation.	
17.	The control switches off the control-is-ready output (X41/34) after recognizing a fault.	Flashing error-message

5.2.2 TNC 407

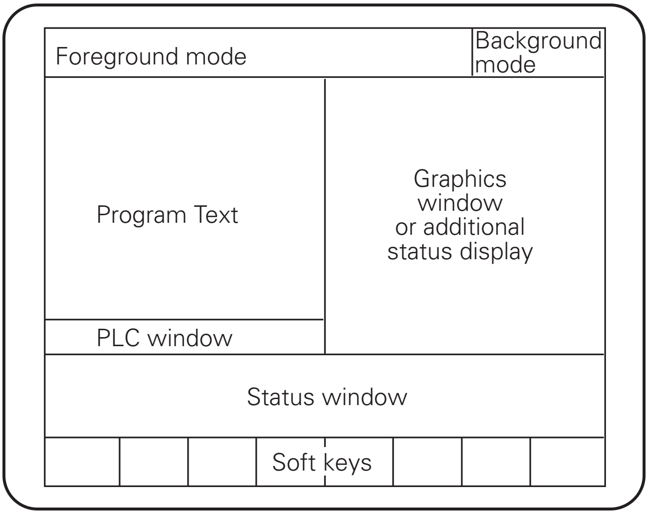


		VDU-display
1.	Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
2.	Recognize the control voltage on X42/4 and reset the control-is-ready output on X41/34 (t < 70 ms).	
3.	Maximum time during which the control-is-ready on X42/4 must go to 0 (t < 146 ms). If exceeded	 EMERGENCY STOP DEFECTIVE
4.	Recognize the acknowledgement and set output X41/34 (t < 24 ms)	
5.	Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
6.	Normal control operation. Output and control-is-ready acknowledgement signals are high.	
7.	Control voltage switched off externally.	EMERGENCY STOP
8.	After switching on the control voltage again, the error message can be cancelled, followed by normal control operation.	
9.	The control switches off the control-is-ready output (X41/34) after recognizing a fault.	Flashing error-message

6 Display and operation

Machine parameters and PLC-markers can be used to influence the control behaviour for certain functions. All machine parameters and PLC-markers which influence the display and the operation of the control, and for which there is no separate section in this manual, are described in the section "Display and operation" .

The display is divided into several "Windows" for optimum depiction of the information on the VDU-screen.



The VDU-screen displays can be changed by machine parameters and PLC-markers.

6.1 Machine datum

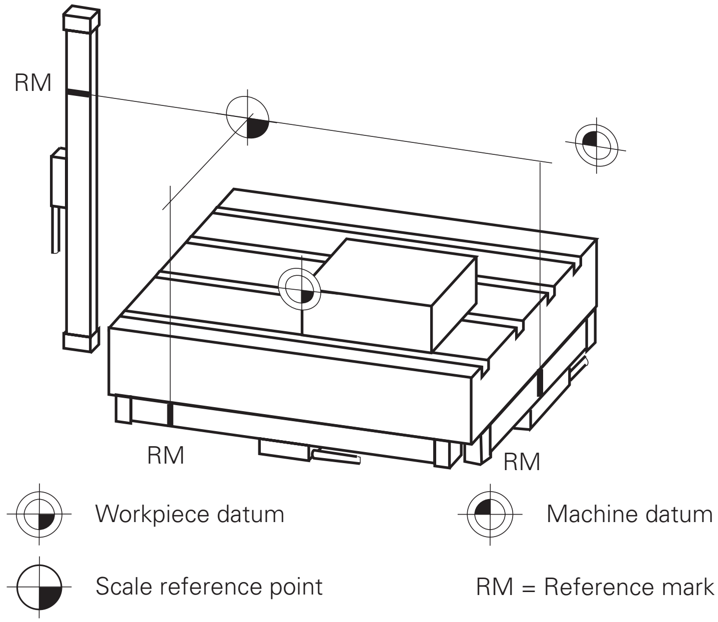
In the operating modes "Manual operation" and "Electronic handwheel", a workpiece datum can be defined. NC positioning blocks normally refer to this datum. If a positioning block should be referenced to the machine datum instead of the workpiece datum, this must be programmed in M91. The distance of the machine datum to the scale datum is entered in machine parameter MP960.x. All REF-referenced displays and positioning movements are referenced to the machine datum.

If the NC positioning blocks should always be referenced to the machine datum, then the "datum setting" can be disabled for the individual axes through MP7295.

With MP950.x you can define a further machine-referenced position. If you wish to reference to this position in a positioning block, you must program it in with M92. MP950.x defines the distance from this machine-referenced position to the machine datum.

M91 and M92 are non-modal.

MP7296 can be used to select whether a new datum point can be set with the "DATUM SET" soft key or additionally with the axis keys.



MP950	Datum point for positioning blocks with M92 Entry: –99 999.999 to +99 999.999 [mm] or [°] Values referenced to the machine datum
MP950.0	X axis
MP950.1	Y axis
MP950.2	Z axis
MP950.3	4th axis
MP950.4	5th axis

MP960	Machine datum Entry: –99 999.999 to +99 999.999 [mm] or [°] Values referenced to the measuring system datum
MP960.0	X axis
MP960.1	Y axis
MP960.2	Z axis
MP960.3	4th axis
MP960.4	5th axis

MP7295	Disable "set datum point" Entry: %xxxxx

Bit 0	X axis	0 = not disabled
Bit 1	Y axis	1 = disabled
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

MP7296	Set datum point with axis keys Entry: 0 or 1 0 = datum point can be set with axis keys and soft key. 1 = datum point can only be set with soft key.

6.2 Colour adjustment

The BC 110 is a 14 inch colour-graphics screen with a resolution of 640 x 490 pixels. The colours in the screen-display can be selected by machine parameter. For example, the colours can be adjusted to suit the corporate image of the machine tool builder image or the design of the machine.
The following colour adjustments cannot be altered by machine parameters:

– HEIDENHAIN company logo after switching on the machine (GREEN),
– flashing error messages (RED),
– error message for invalid machine parameter (RED),
– plan view in the graphics display (BLUE),
– cursor (always inverse).

The entry values for colour adjustment are byte-oriented. The preferred entry is hexadecimal.

Colour	Red		Green		Blue	
HEX Ranges	0 to 3	0 to F	0 to 3	0 to F	0 to 3	0 to F
Adjustment	coarse	fine	coarse	fine	coarse	fine
Entry for yellow: $0....	3	9	3	9	0	0

Since it is possible to make mistakes when setting the colours (e.g.. red error messages on red background), HEIDENHAIN recommends a standard colour-adjustment. This standard colour setting is the setting generally used by HEIDENHAIN and is prompted by the control system when creating the MP list.

The standard colour-adjustment is given in the following list.

Machine parameter	Colour for ...	Standard colour-adjustment
MP7350	Window frame	$030200C
MP7351	Error messages	$03F3F0F
MP 7352	Operating-mode display "Machine"	
MP7352.0	Background	$0000000
MP7352.1	Text for operating mode	$0342008
MP7352.2	Dialogue	$03F3828
MP7353	Operating-mode display "Programming"	
MP7353.0	Background	$0000000
MP7353.1	Text for operating mode	$0342008
MP7353.2	Dialogue	$03F3828
MP7354	Program-text display "Machine"	
MP7354.0	Background	$0080400
MP7354.1	General program-text	$038240C
MP7354.2	Current block	$038341C
MP7354.3	Background not active window	$00C0800

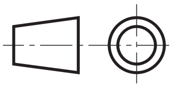
MP7355	Program-Text-display "Programming"	
MP7355.0	Background	$0080400
MP7355.1	General program-text	$038240C
MP7355.2	Current block	$038341C
MP7355.3	Background not active window	$00C0800
MP7356	Status- and PLC-window	
MP7356.0	Background	$00C0800
MP7356.1	Axis positions in the status-display	$03F2C18
MP7356.2	Status-display, except axis positions	$03F280C
MP7357	Soft key display "Machine"	
MP7357.0	Background	$0000000
MP7357.1	Symbols	$03F3828
MP7358	Soft key display "Programming"	
MP7358.0	Background	$0000000
MP7358.1	Symbols	$03F3828
MP7360	Graphics: 3D-view	
MP7360.0	Background	$0000000
MP7360.1	Top surface	$0203038
MP7360.2	Front face	$00C1820
MP7360.3	Text-display in graphics window	$03F3F3F
MP7360.4	Side face	$0102028
MP7361	Graphics: view in three planes (and Oscilloscope)	
MP7361.0	Background	$0000000
MP7361.1	Plan (Grid)	$0203038
MP7361.2	Front and side view (not selected channel)	$0203038
MP7361.3	Cross-axes and text in graphics-display (Cursor, data, screen window)	$03F3F3F
MP7361.4	Cursor (selected channel)	$03F0000
MP7362	Additional status-display in graphics window	
MP7362.0	Background graphics window	$0080400
MP7362.1	Background status display	$00C0800
MP7362.2	Status symbols	$038240C
MP7362.3	Status values	$03F2C18
MP7363	Interactive graphics	
MP7363.0	Background	$0000000
MP7363.1	Resolved contour	$03F3F3F
MP7363.2	Subprograms and frame for zoom	$0003F00
MP7363.3	Alternative solutions	$0003F00
MP7363.4	Unresolved contour	$03F0000

6.3 Graphics window

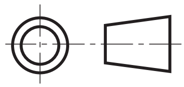
The simulation-graphics (TNC 407 and TNC 415) or parallel-graphics (TNC 415 only) are depicted in the graphics window. It is possible to choose between three different graphics display modes. With the aid of a Soft key the operator can select an additional status-display instead of the graphics (see User's Manual). The graphics-display on the VDU-screen can be altered by machine parameters.

6.3.1 View in three planes

The 3-plane display mode can be produced according to either European or American convention.

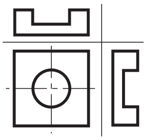


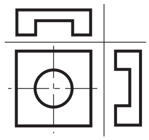
German preferrred



U.S. preferred

Example
:

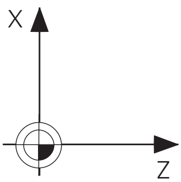




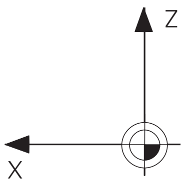
The cursor position can be shown in the 3-plane display mode. This function must be activated with machine parameter MP7310, bit 3.

6.3.2 Rotating the coordinate system

The coordinate system can be rotated by +90° in the machining plane
This is useful when, for instance, the Y-axis is fixed as the tool axis.



No rotation



90° rotation

6.3.3 BLK form and datum shift

A number of BLK forms can be programmed in sequence in an NC program. Machine parameters are used to select whether, after a Cycle 7 "Datum Shift", the definition of the next BLK form is also shifted.

MP 7310	Graphics display Entry: %xxxxx	
Bit 0	Switch-over display in three planes	0 = German preferred 1 = U.S. preferred
Bit 1	Rotate coordinate system in machining plane by +90°	0 = no rotation 1 = coordinate system rotated through +90°
Bit 2	BLK form after datum shift	0 = BLK form will not shift 1 = BLK form will shift
Bit 3	Show cursor position in 3-plane display	0 = not shown 1 = cursor position shown

6.4 Status-window

The status of the control (axis positions, tools, feed, M-functions etc.) is displayed on the VDU-screen in the status window. A Soft key can be used to activate an additional status-display in the graphics window instead of the graphics.

The colours for the status-window display can be altered by machine parameters.

6.4.1 Position display

The input resolutions for the TNCs are:
- TNC 415 B: 0.0001 mm or 0.0001°
- TNC 415 F and TNC 407: 0.001 mm or 0.001°

The display-step for the axis positions can be selected for specific axes by machine parameter. Regardless of this selection the TNC always attempts to position precisely to 0.0001 mm or 0.0001° (regardless of the grating period of the measuring system and the smallest voltage increment).

The most recently selected axis is displayed inversely. The PLC is informed (by a marker) which axis has an inverse display. This information can be evaluated, for example, in connection with hand-control equipment.

Machine parameter MP7285 can be used to select whether, in the positional-display, the tool axis takes account of the tool length or not.

MP7285	Calculation of the tool length in the position-display of the tool axis Entry: 0 or 1 0 = tool length ignored 1 = tool length taken into account
MP7290	Position-display step Entry: 0 to 6 0 = 0.1 mm or 0.1° 1 = 0.05 mm or 0.05° 2 = 0.01 mm or 0.01° 3 = 0.005 mm or 0.005° 4 = 0.001 mm or 0.001° 5 = 0.0005 mm or 0.0005° TNC 415 B only 6 = 0.0001 mm or 0.0001° TNC 415 B only

The following input values apply for software types 259 96 (TNC 415 A) and 243 02 (TNC 407):

0 = 0.001 mm or 0.001^{c}
1 = 0.005 mm or 0.005^{c}
2 = 0.0001 mm or 0.0001^{c} *TNC 415 A only*

MP7290.0	X axis
MP7290.1	Y axis
MP7290.2	Z axis
MP7290.3	4th axis
MP7290.4	5th axis

Marker	Function	Set	Reset
M2096	X Key last pressed	NC	NC
M2097	Y Key last pressed		
M2098	Z Key last pressed		
M2099	Key IV last pressed		
M2148	Key V last pressed		

6.4.2 Position display with rotary axes

Display mode and range of traverse

MP810 is used to set whether the range of traverse limits (software limit switches) are active for rotary axes; it also defines the modulo value for the counting mode. An entry of 0 means that the software limit switches are active and the display ranges from -99,999.999 to +99,999.999.

Any other entry defines the modulo value for the position, target position and non-linear axis error correction. The software limit switches are not active for all entries other than 0.

MP810 Display mode for rotary axes and PLC axes
Entry: 0.0000 to 9,999.9999 [°]

0 = display ± 99,999.9999; software limit switches active
≠0 = modulo value for display; software limit switches not active

MP810.0	Axis X
MP810.1	Axis Y
MP810.2	Axis Z
MP810.3	4th axis
MP810.4	5th axis

Contrary to the above, the following applies for software types 259 36. .., 259 97. .., 243 02. .. and 243 03., 259 93., 259 94, up to version 08: MP810 is not available but MP7470 is, with the following meaning:

MP7470 Position display and software limit switches for rotary axes
Entry: 0 or 1

0 = 0 to + 359.9999° (no software limit switches)
1 = - 30,000.0000° to +30,000.0000° (software limit switches active)

–**"Free rotation"** :

A free rotation of the rotary axis can be activated by the PLC. "Free rotation" means that the axis of rotation can be turned as often as required (with a display of 0 to 360°) without being affected by software limit switches.

The free rotation function is a PLC-function, i.e. the PLC-program must be created by the manufacturer of the machine in the proper form. The function could, for example, be activated by M-functions. The choice of axes and the direction of traverse are made by specific bytes. The feed for free rotation is the same as the feed for PLC-positioning of the axes 4 and 5 (W566 and W568).

The max. feed is 300 000 °/min. The feed is not displayed in the status-window. The feed can be continuously varied by the PLC with an override-percentage (W754), e.g. by copying W494 (current feed-override) to W754. The free rotation function is activated and deactivated with marker M2720. If M2720 is set by PLC, the NC takes the information from B518 and B519 and resets M2720.

Marker	Function	Set	Reset
M 2720	Activate/deactivate "Free-rotation" of rotary axis	PLC	NC

Address Function

B518 0 = "Free-rotation" cancelled
8 = "Free-rotation" for 4th axis
16 = "Free-rotation" for 5th axis

B519 Definition of the direction of traverse

–	+	Entry value
	4th and 5th axes	0
4th axis	5th axis	8
5th axis	4th axis	16
4th and 5th axes		24

W754 Percentage for feed-override "free rotation"
(0 to 300 %)

6.4.3 Feed rate display

In the operating modes "Program run/single block" and "Program run/full sequence" the programmed contour feed-rate is displayed when the feed-potentiometer is in the 100 % position. The Feed-Potentiometer can be used to vary this programmed feed-rate from 0 to 150 %.

If rapid traverse was programmed, this is indicated in the display with FMAX and marker M2151 is set.

With machine parameter MP7620 you can set whether the feed rate override is active in 1% steps or according to a non-linear curve. The non-linear override curve permits fine a resolution in the lower range of the potentiometer.

The override values are shown in the PLC in word W766 and W494. If the PLC program overwrites a value in W766, it will become active regardless of the potentiometer setting. For 1% steps the value range is 0 to 150, for a non linear curve it is 0 to 15 000, corresponding to 0 to 150%.

In the manual operating modes the axis feed-rate is displayed, not the contour feed-rate.
There is a choice of two display modes:

- The axis-specific feed-rate from machine parameter MP1020.X is only displayed after pressing an axis-direction key. If two axis-direction keys are pressed simultaneously no feed-rate will be shown.
- Even when none of the axis-direction keys is operated one feed-rate will be displayed, which can also be adjusted by the feed-potentiometer. The smallest feed-rate from MP 1020.X is valid for all axes. The axis feed-rate will also be shown if several axis-direction keys are pressed simultaneously.

MP7270 Display of the feed-rates in manual operating modes (Manual operation, Electronic handwheel)
Entry: 0 or 1

0 = display of the axis feed-rate only when an axis-direction key is pressed (axis-specific feed from MP1020.X)

1 = display of the axis feed-rate before operating an axis-direction key (smallest value from MP1020.X for all axes)

MP7620 Feed rate and spindle override
Entry: 0 to 11

Bit 3 Characteristic curve for linear or non-linear override
+0 = Override linear in 1% steps
+8 = Override non-linear

Address	Function
W494	%-factor feed rate override (NC → PLC)
W766	%-factor feed rate override (PLC → NC)

Marker	Function	Set	Reset
M2151	Rapid traverse programmed (FMAX)	NC	NC

Feed-rate for rotary axes

The TNC interprets the programmed feed-rate for a rotary axis in degrees/min. The contour feed-rate is therefore governed by the distance of the tool centre from the centre of rotation of the rotary axis. When M function M116 is activated the contour feed-rate is interpreted in mm/min., i.e. the feed-rate is now independent of the distance of the tool centre from the centre of rotation of the rotary axis.

M116 is automatically cancelled by PGM END. M116 is only active when the centre of rotation of a rotary axis is defined in machine parameters MP7510 ff.

6.4.4 Display of the M-functions

The miscellaneous functions for control of the spindle (M03, M04, M05) and the coolant (M08, M09) are displayed in the status window. The display of these M-functions is controlled by the PLC, i.e. the manufacturer of the machine must take this into account when creating the PLC-program. The markers M2485 and M2486 also change the polarity of the analogue voltage for the spindle. M2608 switches off the analogue output for the Spindle. The programmed spindle-speed is, however, still displayed (see section "Main spindle").

The display of other M-functions can be achieved in the PLC-window (see chapter "PLC-description", section "Modules")

Marker	Function	Set	Reset
M2485	Status display and sign of S-analogue for M03	PLC	PLC
M2486	Status display and sign of S-analogue for M04		
M2487	Status display for M05 and spindle stop		
M2608	Status display M03, M04, M05 inverse and S-analogue output = 0V		

M2508	M2657	Display
0	0	M09
0	1	M07
1	0	M08
1	1	MK

Marker	Function
M2609	Status-display M07, M08, M09 inverse

6.4.5 Control is in operation

If the control is in operation, i.e. a positioning or M-function is performed, the status window displays a large asterisk "*". If a current NC-program is interrupted with the external stop key, the "*" will flash in the status-display. This information is transmitted to the PLC through the markers M2183 and M2184. M2183 and M2184 are effective in the operating modes "Positioning with MDI", "Program run/single block" and "Program run/full sequence".

Marker	Function	Set	Reset
M2183	Program interruption (display "Control operational" flashes)	NC	NC
M2184	Control operational (display "Control operational" on or flashes)		

6.4.6 Cancel status-display

A machine parameter can be used to decide whether with the M-functions M02 and M30, as well as with NC-block "END PGM" the status-display should be reset and the contents of the Q-parameters erased. All the programmed values in the status-display, e.g. tool number, tool length, tool radius, scaling factor, datum-shift, feed-rate etc. will then be reset. The Q-parameters will all be set to 0.

MP7300 Cancel Status-display and Q-Parameters
Entry: 0 to 7

0 = Cancel status display and Q parameters and tool data when program is selected
1 = Cancel status display, Q parameters and tool data with M02, M30, END PGM and when program is selected
2 = Cancel status display and tool data when program is selected
3 = Cancel status display and tool data when program is selected and with M02, M30, END PGM
4 = Cancel status display and Q parameters when program is selected
5 = Cancel status display and Q parameters when program is selected and with M02, M0, END PGM
6 = Cancel status display when program is selected
7 = Cancel status display when program is selected and with M02, M30, END PGM

6.5 PLC-Window

The displays in the PLC-Window are formatted by the PLC. Any ASCII-text can be displayed in two lines of 38 characters. In the left half of the line a bar chart can also be displayed. Text and bar chart can be mixed.
The display in the PLC-Window is activated by PLC-modules. These modules are explained under "PLC-description".

PLC-module 9070	String-address
PLC-module 9071	String-length
PLC-module 9080	Cancel display
PLC-module 9081	Interrogate PLC-window
PLC-module 9082	Describe PLC-window
PLC-module 9083	Display bar chart

Marker	Function	Set	Reset
M3171	Error message in PLC-modules	NC	NC

6.6 Error messages

Under certain conditions error messages from the NC or the PLC will be displayed on the screen under the display for the operating mode. Non-flashing error messages can be cancelled with the CE-key. In the event of a flashing error message the machine must be switched off and the fault corrected. If a non-flashing error message is displayed on the screen, marker M2190 will be set.

The manufacturer of the machine determines the conditions under which the PLC will produce error messages. Up to 100 different PLC error messages can be generated.

The dialogues for PLC error messages can be determined by the manufacturer of the machine. Please contact HEIDENHAIN about this. The standard version has dialogues with the reserved designations "PLC: ERROR 0" to "PLC: ERROR 99". These PLC error messages can be activated by the PLC-markers M2924 to M3023.

The error messages can be cancelled either by pressing the CE-key or by resetting the appropriate markers.

If several PLC error messages are activated simultaneously, they can be read out one after another by pressing the CE key.

If the program run is to be stopped on output of a PLC error message, this must be explicitly programmed in the PLC-program (NC STOP).

In order to show a PLC error message as a flashing message, the marker M2815 must also be set. If marker M2815 is set, but none of the 100 PLC error messages is activated, then the flashing error message "EMERGENCY STOP PLC" will be displayed.

If, for some reason, the display of the PLC error message is not wanted in the field under the operating-mode display, the message can also be displayed in the PLC-window.

Marker	Function	Set	Reset
M2815	Flashing PLC error message	PLC	PLC
M2190	Non-flashing error message is displayed	NC	NC
M2924	PLC error message 0	PLC	NC; PLC
M2925	PLC error message 1		
M2926	PLC error message 2		
.	. .		
.	. .		
.	. .		
M3022	PLC error message 98		
M3023	PLC error message 99		

6.7 Cycles

The HEIDENHAIN contouring controls permit calling HEIDENHAIN standard-cycles within the NC program (e.g.. pecking, tapping, pocket milling etc.). In addition, the manufacturer of the machine can program manufacturer-cycles and store them in the control (see "OEM-cycles"). The sequence of some cycles can be altered by machine parameters and PLC-markers. The description of the "Tapping" and "Spindle-orientation" cycles can be found in the section "Main spindle".

6.7.1 Cycle inhibit

Machine parameter MP7245 can be used to selectively inhibit the HEIDENHAIN standard-cycles.

MP7245.0 Inhibit the HEIDENHAIN Cycles 1 to 15
Entry: 0 to 65 535
$0 to $FFFF

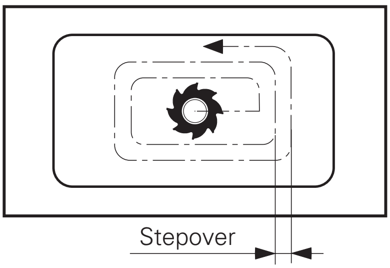
Bit 1	Cycle 1	0 = do not disable
Bit 2	Cycle 2	1 = disable
Bit 3	Cycle 3	
Bit 4	Cycle 4	
Bit 5	Cycle 5	
Bit 6	Cycle 6	
Bit 7	Cycle 7	
Bit 8	Cycle 8	
Bit 9	Cycle 9	
Bit 10	Cycle 10	
Bit 11	Cycle 11	
Bit 12	Cycle 12	
Bit 13	Cycle 13	
Bit 14	Cycle 14	
Bit 15	Cycle 15	

MP7245.1 Inhibit the HEIDENHAIN standard-Cycles 16 to 30
Entry: 0 to 65 535
$0 to $FFFF

Bit 0	Cycle 16	0 = do not disable
Bit 1	Cycle 17	1 = disable
Bit 2	Cycle 18	
Bit 3	Cycle 19	
Bit 4	Cycle 20	
Bit 5	Cycle 21	
Bit 6	Cycle 22	
Bit 7	Cycle 23	
Bit 8	Cycle 24	
Bit 9	Cycle 25	
Bit 10	Cycle 26	
Bit 11	Cycle 27	
Bit 12	Cycle 28	
Bit 13	Cycle 29	
Bit 14	Cycle 30	
Bit 15	Cycle 31	

6.7.2 Pocket milling

The overlap-factor for clearing out a rectangular or circular pocket (Cycle 4 and Cycle 5) can be altered by machine parameters.



Stepover = (MP7430) x cutter radius

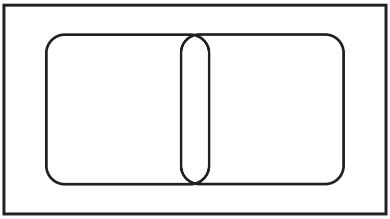
MP7430 Overlap-factor for pocket milling
Entry: 0.100 to 1.414

6.7.3 Milling-cycles for pockets with free-programmed contours

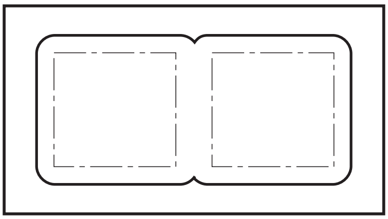
The sequence for milling-cycles for pockets with free-programmed contours (Cycles 6, 14, 15, 16) can be changed by machine parameters.

The choices are:

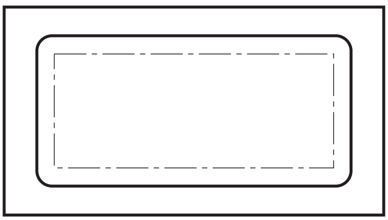
- Whether a channel should first be milled around the contour and the pocket cleared out afterwards, or the pocket cleared out first, and then a channel milled round the contour.
- Whether the channel should be milled in a clockwise or anti-clockwise direction.
- Under which conditions programmed pockets should be merged. The choice is between: merging programmed pockets when the programmed contours overlap, or when the tool centre paths intersect.
- Whether the channel-milling should be performed in one operation for all stepovers, or whether for each peck depth channel-milling and clearing out should be performed alternately.
- Whether after the pocket has been machined, the tool should be moved back to the initial position before the cycle was called or should merely be moved to setup clearance.



The programmed contours of two pockets have a small overlap.



MP7420 Bit 2 = 0 (entry +0):
The control clears out the pockets separately, since the tool centre paths do not intersect. Material will remain in inside corners.



MP7420 Bit 2 = 1 (entry +4):
The control clears out the pockets jointly, since the programmed contours overlap. No material will remain in inside corners.

MP7420 Cycles for milling pockets with free-programmed contours.
Entry: %xxxx

Bit 0	Channel-Milling direction for	0 =	Anti-clockwise channel-milling of the pocket contours, clockwise for islands
		1 =	Clockwise channel-milling of the pocket contours, anti-clockwise for islands
Bit 1	Sequence for	0 =	First channel-milling, then clear out pocket clearing out and -
		1 =	First clear out pocket , then channel-milling channel-milling
Bit 2	Merge programmed contours	0 =	Contours merged only if the tool centre paths intersect
		4 =	Contours merged if the programmed contours overlap
Bit 3	Clearing out and channel-milling to pocket depth, or for each infeed depth	0 =	Clearing out and channel-milling performed in one operation for all depths
		1 =	For each peck, first perform channel-milling, and then clearing out (depending on Bit 1) before next peck
Bit 4	Position after cycle has been executed	0 =	Move tool to position that was approached before cycle was called
		1 =	TNC moves tool to "clearance height"

6.7.4 Scaling factor

Machine parameters can be used to decide whether Cycle 11 "Scaling factor" only operates in the machining plane or also parallel to the tool axis.

MP7410 "Scaling factor" cycle in two or three axes
Entry: 0 or 1

0 = Cycle "Scaling factor" operates in all three principal axes
1 = Cycle "Scaling factor" only operates in the machining plane

6.7.5 Cylinder interpolation

A contour can be machined on a cylindrical surface with Cycle 27 "Cylinder surface" (see User's Manual).

This requires the centre of rotation of a rotary axis to be defined in machine parameters MP7510 ff. (see section "Swivel Axes"). The same reference position must apply for a description of the machine geometry via machine parameters MP7510 ff. and for any datum shift.

6.8 Return to the contour

With the HEIDENHAIN contouring control it is possible to resume an interrupted program, or to make a block scan up to a predetermined block number (see User's Manual).

These functions must be enabled by machine parameters and the PLC program must be changed accordingly.

Markers inform the PLC about the individual conditions during the resumption of the program or block scan. Depending on these markers, certain functions can be enabled in the PLC-program (e.g. operating the axis-direction keys in "Manual operation").

M2018 is set if the "MANUAL OPERATION" soft key is pressed.

M2019 is set if the "Return to contour" function is activated with the "RESTORE POSITION" soft key.

M2059 is set if the "RESTORE AT N" soft key is pressed. M2059 is reset if the "RESTORE POSITION" or "INTERNAL STOP" soft key is pressed.

During the block scan the PLC positioning commands are only included in the calculation if they are also executed. The TOOL CALL block usually releases PLC positioning commands for the tool change position. If these movements are to be offset in the block scan, the absolute position must be entered referenced to the machine datum in MP951.x and the offset must be activated for specific axes with MP7450.

When using random selection in the central tool memory (see section "Tool change"), an alteration of the position numbers in the tool memory must be avoided during block scan. This is transmitted from the PLC to the NC by marker M2612.

Machine parameter MP7680 can be used to select whether the block scan may be interrupted by a programmed STOP or M06 and whether the programmed dwell time should be taken into account during the block scan.

The feed for returning to the contour is specifically defined in machine parameter MP4220.X. This machine parameter is also used to set a number in the PLC (see "PLC description.")

MP4220	Feed for "Return to the contour"
	Entry: 10 to 30 000 $\left(\frac{mm}{min}\right)$
MP4220.0	X axis
MP4220.1	Y axis
MP4220.2	Z axis
MP4220.3	4th axis
MP4220.4	5th axis

MP7680	Machine parameter with multiple function Entry: %xxxxxxx	
Bit 1	Return to the contour	0 = inactive 1 = active
Bit 2	Block scan	0 = inactive 1 = active
Bit 3	Interruption in block scan on STOP or M06	0 = interruption 1 = no interruption
Bit4	Allow for the programmed dwell time during block scan	0 = allow for dwell time 1 = ignore dwell time
Bit 5	Start calculation for block scan	0 = Start from cursor 1 = Start from program start

MP951	Simulated tool change position for TOOL CALL during block scan Entry: –99 999.9999 to +99 999.9999 [mm] or [°]
MP951.0	Axis X
MP951.1	Axis Y
MP951.2	Axis Z
MP951.3	4th axis
MP951.4	5th axis

MP7450 Calculate the tool change position from MP951 in block scan
Entry: %xxxxx

Bit 0	Axis X	0 = Calculate
Bit 1	Axis Y	1 = Do not calculate
Bit 2	Axis Z	
Bit 3	4th axis	
Bit 4	5th axis	

Marker	Function	Set	Reset
M2018	Soft key "Manual operation" pressed	NC	NC
M2019	"Restore position" is active	NC	NC
M2612	No change in position number in central tool memory	PLC	PLC
M2059	Block scan is active	NC; PLC	NC; PLC

6.9 Files

With the TNC it is possible to process different types of file with the file management. The types of file are designated by an extension after the names.

.H	= HEIDENHAIN program
.I	= DIN/ISO program
.T	= Tool table
.D	= Datum table
.P	= Pallet table
.A	= Text file
.HLP	= Help file
.PNT	= Point table

The file name can be up to 16 characters long (letters and digits). The maximum length of the file name is defined in MP7222.

MP7222	Length of file names Entry: 0 to 2 0 = 8 characters 1 = 12 characters 2 = 16 characters

6.9.1 File types – disable and protect

The individual types of file can be selectively disabled and protected. Only file types which are not disabled are visible to the file management in the control. If a file type is locked, all files of this type are cleared. The individual file types can also be protected so that they cannot be edited or changed. Protected files are displayed with the colours defined in MP7354.1 or MP7355.1 in the file directory.

MP7224.0	Disable file type Entry: %xxx xxxxx	
Bit 0	HEIDENHAIN programs	0 = do not disable
Bit 1	ISO programs	1 = disable
Bit 2	Tool-tables	
Bit 3	Datum-tables	
Bit 4	Pallet-tables	
Bit 5	Text files	
Bit 6	Help files	
Bit 7	Point tables	

MP7224.1	Protect file type Entry: %xxxxxxx	
Bit 0	HEIDENHAIN programs	0 = not protected
Bit 1	ISO programs	1 = protected
Bit 2	Tool-tables	
Bit 3	Datum-tables	
Bit 4	Pallet-tables	
Bit 5	Text files	
Bit 6	Help files	
Bit 7	Point tables	

6.9.2 Block numbers - step size for ISO programs

The block-number step size for ISO programs can be fixed by a machine parameter.

MP7220	Block-number step size for ISO programs Entry: 0 to 150

6.9.3 Table size

The size of the pallet- and datum-tables can be fixed by machine parameters. For further information about the tool-table see section "Tool changer", for pallet-table see "PLC description".

MP7226.0	Size of pallet-tables Entry: 0 to 255 [lines]
MP7226.1	Size of datum-tables Entry: 0 to 255 [lines]

6.9.4 Datum point for values in datum table

MP7475 is used to define whether the values in the datum table refer to the set workpiece datum or the machine datum (MP960).

MP7475	Datum in datum table Entry: 0 or 1 0 = Datum point is workpiece datum 1 = Datum point is machine datum

6.10 User-parameters

With the MOD-function up to 16 different machine parameters can be made accessible to the machine operator as User-parameters. The machine manufacturer determines in machine parameter MP7330.x which machine parameters are to be defined as User-parameters. If, for example, you wish to define MP5030.1 as the first user-parameter, you must enter the value 5030.01 in MP 7330.0.

If a User-parameter is selected by the operator, a dialogue appears on the screen. Machine parameter MP7340.X determines which dialogue should be displayed. A line number from the PLC dialogues is defined with an entry value between 0 and 4095 (see "PLC description").

MP7330	Determination of the User-Parameters Entry range: 0 to 9999.00 (No. of the desired machine parameters).
MP7330.0	User-parameter 0
MP7330.1	User-parameter 1
.	.
.	.
MP7330.14	User-parameter 14
MP7330.15	User-parameter 15
MP7340	Dialogues for User-parameter Entry:0 to 4095 (line number of the PLC dialogs)
MP7340.0	Dialogue for User-parameter 0
MP7340.1	Dialogue for User-parameter 1
.	.
.	.
MP7340.14	Dialogue for User-parameter 14
MP7340.15	Dialogue for User-parameter 15

6.11 Code-numbers

The MOD functions can be used to enter code-numbers for the control. These code numbers can be used to activate certain control functions.
The following code-numbers have a fixed meaning:

Code-number	Function
95 148	Select machine parameter list
807 667	Select PLC mode
105 296	Correction tables for the non-linear axis-error compensation
86 357	Remove program protection
75 368	Automatic offset adjustment
123	Select the user-available list of machine parameters
531 210	Erase markers M1000 to M2000 and byte 0 to byte 127 Special function: Switch on control and press MOD key at the same time. Enter code number. The entire RAM memory is now erased (MPs, NC programs, PLC program, all markers and bytes).
688 379	Oscilloscope

The value of the code-number which is entered is stored in Doubleword D276. The machine manufacturer can evaluate this code with the aid of the PLC and define his own functions for code numbers or disable the preset code numbers.

Address	Function
D276	Value of the code-number most recently entered by MOD

6.12 Programming station

Machine parameters can be used to set the control so that it can be utilized as a programming station, without the machine. In this setting only the operating modes "Programming and editing" and "Program test" function. It is possible to select whether the PLC should be active or not in the "programming station" setting.

MP7210 Programming station
Entry values: 0, 1, 2

Entry:
0 = Control and programming
1 = Programming station, "PLC active"
2 = Programming station, "PLC inactive"

6.13 Dialogue language

The HEIDENHAIN contouring controls are available in ten different dialogue languages, see chapter "Introduction". The dialogue language can be altered by a simple software exchange. English, as a basic language, is stored in every control as a second language and can be selected by machine parameter.

If the basic language English has been selected, then marker M2041 is set by the NC.

MP7230 Change dialogue language
Entry: 0 or 1

0 = First dialogue language
1 = Basic language - English

Marker	Function	Set	Reset
M 2041	Basic language - English is selected	NC	NC

6.13.1 Decimal sign

The decimal sign can be selected by machine parameter.

MP7280	Decimal sign Entry: 0 or 1 0 = Decimal comma 1 = Decimal point	

6.14 Memory test

A machine parameter can be used to select if the RAM and the EPROM memory areas should be tested on switching on the control.

MP7690	Memory test at switch-on Entry: %xx	
Bit 0	RAM-test	0 = memory test at switch-on
Bit 1	EPROM-test	1 = no memory test at switch-on

6.15 End of program

In the operating modes "Program run/single block" or "Program run/full sequence", if the end of the program is reached the NC sets the marker M2061. This marker is only reset at the start of the next program.
The information "End of program" can be evaluated by the PLC. This is necessary when operating, for instance, with a pallet-changer.

Marker	Function	Set	Reset
M2061	END PGM, M02 or M30 has been executed	NC	NC

6.16 Overwrite Q-parameters

The values in the Q-Parameters Q100 to Q107 can be overwritten by the PLC. In this way information from the PLC can be transferred to the machining program. The value which is to be transferred is stored in Doubleword D528. The Q-parameter which is to be overwritten is defined in Word W516. The transfer is initiated by the strobe-marker M2713 during an M/S/T strobe.

The Doubleword D528 has a multiple usage. See also the sections "PLC positioning" and "Datum shift correction".

Address	Function
W516	Number of the Q-parameter to be overwritten (Q100 to Q107 = 0 to 7)
D528	Value to be transferred to the Q-Parameter

Marker	Function	Set	Reset
M2713	Activate the transfer of the value from D528 to the Q-Parameter defined in W516	PLC	NC

6.17 Arc end-point tolerance

The control uses the NC-data which is entered to calculate the deviation in the arc radius between the beginning and end of the arc. If the tolerance defined in MP7431 is exceeded, the error message "Arc end-point in correct" will appear.

MP7431 Arc end-point tolerance
Entry: 0.0001 to 0.016 [mm]

6.18 Radius compensation R+, R-

A path to be traversed can be increased or decreased by the tool radius by entering "R+" or "R-". The input dialogue is not initiated with the "L" key but directly with the orange axis direction key. For reasons of compatibility this function has been retained for point and straight cut controls.

Example:	X + 20	R+		Dialogue programming
	G07	X + 20	G49	DIN/ISO programming

Paraxially corrected positioning blocks (R+/R-) and radius-corrected positioning blocks (RR/RL) must not be entered one after another.
To avoid erroneous entries MP7246 can be used to disable the input of paraxial positioning blocks.

MP7246 Disable paraxial positioning blocks
Entry: 0 or 1

0 = paraxial positioning blocks enabled
1 = paraxial positioning blocks disabled

6.19 "POWER INTERRUPTED" Message

When the control voltage is disconnected the TNC issues the message "POWER INTERRUPTED".
The PLC is not active until this message is reset with the CE key.
The "POWER INTERRUPTED" message does not appear if MP7212 is set to 1.

MP7212 "POWER INTERRUPTED"
Entry: 0 or 1

0 = Reset "POWER INTERRUPTED" message with CE key
1 = No "POWER INTERRUPTED" message

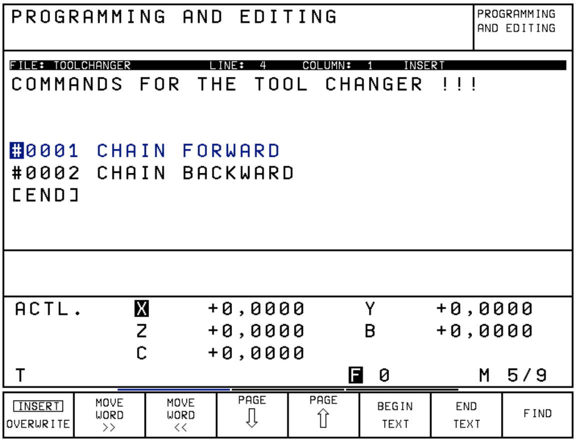
6.20 Help files

If files of the .HLP type are stored in either the PLC EPROM or the NC memory, the HELP soft key will appear after the MOD key is activated.

The machine tool manufacturer creates the HELP file in the "PLC Programming" mode. Help texts (information) or machine commands can be edited in the HELP file. With machine commands a numerical value in the format "#xxxx" is edited at the start of line.

As soon as the cursor is moved on a line with a numerical value, this value is available in W270 of the PLC, and the corresponding command is evaluated in the PLC program. If the cursor is moved on a line without a valid numerical value, then the value -2 is entered in word W270. If no HELP file is selected the value in W270 is –1.

Example:



Several HELP files can be created. The user selects the desired file with the PGM NAME key after operating the HELP soft key.

If a HELP file is selected in the foreground and background modes then the message "Background programming not possible" will appear.

HELP files are stored externally with the identifier "J".

Address	Function	
W270	Help file line number	
	–1 =	No help file selected
	–2 =	No valid numerical value
	0 to 9999 =	Line number

7 M-functions

Up to 100 miscellaneous functions (M-functions) can be programmed in HEIDENHAIN contouring controls. The code for these M-functions is transferred to the PLC either before or after execution of the NC-block. A number of these M-functions have a fixed meaning for the NC. These M-functions are marked with * in the following table. The other M-functions are freely available.

	M-function	Effective at: Beginning of block-	End of block
*	M 00		●
	M 01		●
*	M 02		●
*	M 03	●	
*	M 04	●	
*	M 05		●
*	M 06		●
	M 07	●	
*	M 08	●	
*	M 09		●
	M 10		●
	M 11	●	
	M 12		●
*	M 13	●	
*	M 14	●	
	M 15	●	
	M 16	●	
	M 17	●	
	M 18	●	
	M 19		●
	M 20	●	
	M 21	●	
	M 22	●	
	M 23	●	
	M 24	●	
	M 25	●	
	M 26	●	
	M 27	●	
	M 28	●	
	M 29	●	
*	M 30		●
	M 31	●	
	M 32		●
	M 33		●
	M 34		●
	M 35		●
	M 36	●	
	M 37	●	
	M 38	●	
	M 39	●	
	M 40	●	
	M 41	●	
	M 42	●	
	M 43	●	
	M 44	●	
	M 45	●	
	M 46	●	
	M 47	●	
	M 48	●	
	M 49	●	
	M 50	●	
	M 51	●	
	M 52		●
	M 53		●
	M 54		●
	M 55	●	
	M 56	●	
	M 57	●	
	M 58	●	
	M 59	●	
	M 60		●
	M 61	●	
	M 62	●	
	M 63		●
	M 64		●
	M 65		●
	M 66		●
	M 67		●
	M 68		●
	M 69		●
	M 70		●
	M 71	●	
	M 72	●	
	M 73	●	
	M 74	●	
	M 75	●	
	M 76	●	
	M 77	●	
	M 78	●	
	M 79	●	
	M 80	●	
	M 81	●	
	M 82	●	
	M 83	●	
	M 84	●	
	M 85	●	
	M 86	●	
	M 87	●	
	M 88	●	
[1]	M 89		●
*	M 90	●	
*	M 91	●	
*	M 92	●	
	M 93	●	
*	M 94	●	
	M 95		●
	M 96		●
*	M 97		●
*	M 98		●
*	M 99		●

[1] Function is dependent on machine parameter M7440

The evaluation of the M-function must be programmed in the PLC. When transferring an M-Function to the PLC the code for the M-function is stored in Word W260 and the strobe-marker M2045 is set.

The execution of the M-function must be signalled to the NC by setting the markers M2482. The next NC-block is only processed when the signal is acknowledged and the marker M2045 (strobe signal for M-function) is reset by the NC. The M-functions M00 to M99 can also be decoded and transferred to the markers M1900 to M1999. This function is activated by the marker M2496. The decoded output is retained for reasons of compatibility. However, HEIDENHAIN recommends M-code evaluation using Word W260.

Address	Function	Set	Reset
W260	Code for M-function	NC	

Marker	Function	Set	Reset
M2045	Strobe signal for M-function	NC	NC
M2482	Acknowledgement of M-function	PLC	PLC
M2496	Enable-marker for the decoded M-code-transfer to markers M1900 to M1999	PLC	PLC
M1900	Miscellaneous function M00	NC	NC
M1901	Miscellaneous function M01	NC	NC
M1902	Miscellaneous function M02	NC	NC
.	.		
.	.		
.	.		
M1999	Miscellaneous function M99	NC	NC

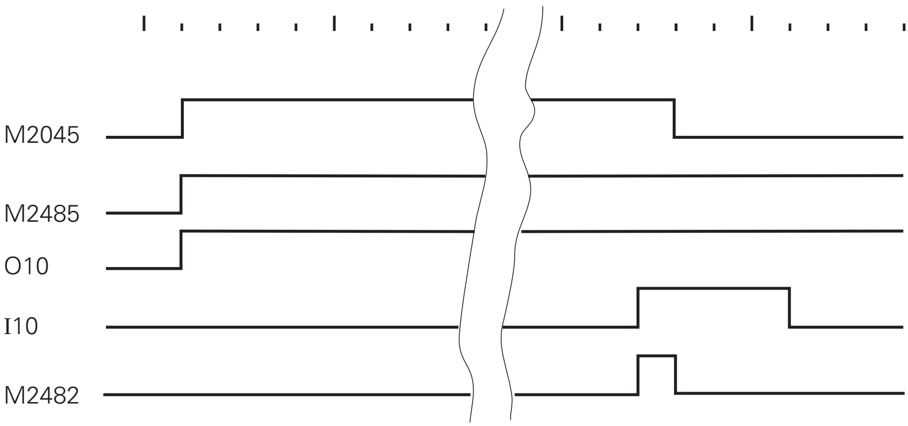
All M-functions over 99 are not transferred to the PLC but have a specific significance for the NC to activate functions (see User's Manual).

Example:

Evaluation of the miscellaneous function M03 in the PLC.

PLC-output: O10 = Spindle ON/OFF
PLC-input: I10 = Acknowledgement of M-function

```
199      L  M2045        Change signal for M-function
200      RN  M2482       Reset acknowledgement of M-function
201      CMT  77         Evaluation of M-function
.
.
.
901      EM
902      LBL 77
903      CASE W260       M-code?
904      CM0
905      CM1
906      CM2
907      CM3
.
.
.
930      ENDC
931      EM
.
.
.
1170     LBL 3           M-function M03
1171     L  M1
1172     S  M2485        Status-display M03, sign of S-analogue
1173     R  M2486        Reset status-display M04, M05
1174      R  M2487
1175     S  010          Spindle-ON
1176     L  I10          Acknowledgement of M-function?
1177     S  M2482        Acknowledgement of M-function
1178     EM
```



7.1 Program-run interruption with M-functions

Normally, when an M-function is produced, the program run in the operating modes "Program run/full sequence" and "Program run/single block" is interrupted until the PLC acknowledges that the M-function has been performed.

For some applications this can be disadvantageous (e.g. laser cutting-machines, DNC-operation). In such applications the program should be executed continuously and not wait for the acknowledgement of the M-function. This function can be selected by machine parameter MP7440, Bit 2. If this function is selected then PLC-positioning, datum-correction, spindle-orientation or limit switch range-change are all not permitted during the output of the M-Function.

This function must not be used with milling machines or boring machines.

7.2 Program-run interruption with M06

According to ISO 6983, the M-Function M06 means a tool change. Machine parameter MP7440, Bit 0 can be used to select whether on transferring M06 to the PLC the program should halt. If the control is set so that a program-run interruption occurs on M06 then the program must be restarted after the tool change. This can also be carried out directly by the PLC.

7.3 Modal cycle call M89

The M-Function M89 can be used for the modal cycle-call.

The possibilities for calling a cycle are:

– With the NC-block "CYCL CALL".
– With the miscellaneous function M99. M99 is only effective for a single block and must be reprogrammed for each execution.
– With the miscellaneous function M89 (depending on the machine parameter).
  M89 as a cycle-call is modally effective, i.e. for every following positioning block there will be a call of the last-programmed machining-cycle. M89 is cancelled by M99 or a CYCL CALL-block.

If **M89 is not defined as a modal Cycle-call** by machine parameters,
then M89 will be transferred to the PLC as a normal M-function at the beginning of the block.

7.4 Reduced feed-rate of tool axis with M103

The entry M103 F... can be used to reduce the contour feed-rate for movements of the tool axis in the negative direction. The feed-rate element of the tool axis is limited to a value that the TNC computes from the last programmed feed-rate.

$F_{max} = F_{prog} * F_{\%}$
F_{max} = Maximum feed-rate in negative direction of tool axis
F_{prog} = Last programmed feed-rate
$F_{\%}$ = Programmed factor after M103 in %
M103 F... is cancelled by a new entry for M103 without a factor.
The function M103 F... is enabled by MP7440, Bit 4.

7.5 Selecting Kv factors with M105/M106

A second set of Kv factors is selected with M-function M105. These Kv factors are defined in machine parameters MP1515.x (feed precontrol) and MP1815.x (lag mode). Selecting a higher Kv factor can selectively enhance the contour accuracy. M105 influences the compensation of reversal spikes during circular movement. M105 activates the machine parameters MP715 and MP716. The original set of Kv factors is restored by the M-function M106. M-functions M105/M106 are enabled by MP7440, Bit 3.

MP7440	Activating M-functions Entry: %xxxxx	
Bit 0	Program halt at M06	0 = program halt at M06 1 = no program halt at M06
Bit 1	Modal cycle call M89	0 = normal code transfer from M89 at start of block 1 = modal cycle call M89 at end of block
Bit 2	Program run hold with M-functions	0 = program run hold until feedback of M-function 1 = no program run hold (no waiting for feedback)
Bit 3	Select Kv factors with M105/M106	0 = function not active 1 = function active
Bit 4	Reduced feed-rate in the tools with M103 F...	0 = function not active 1 = function active

8 Key-simulation

The entry to the HEIDENHAIN contouring controls is by the keys on the TNC-keyboard (TE 400) and the manufacturer's own machine-control panel. The two control panels are joined to connectors X45 and X46 on the logic-unit by a connecting cable (see "Assembly and electrical installation ").

The key-code from the TNC-keyboard is directly evaluated by the NC. PLC inputs and outputs for the machine-control panel are available on connector X46 . These PLC inputs and outputs must be evaluated by the PLC and the appropriate information passed to the NC.

8.1 TNC-keyboard (TE 400)

The key-code from the TNC-keyboard is directly evaluated by the NC.

The keys on the TNC-keyboard and the soft keys on the BC 110 B can be inhibited by the PLC. With M2876 the entire alphabetic keyboard can be inhibited. M2877 inhibits the soft-key row and M2878 inhibits the changeover keys to the right of the screen. All other keys can be inhibited selectively with M2854 to M2923. If an inhibited key is pressed, the NC sets the marker M2182 and deposits the key-code for the key which was operated in word W274. The PLC must reset the marker M2182 after evaluating this information.

The keys on the TNC-keyboard and the soft keys on screen can also be simulated by the PLC. To achieve this the appropriate key-code is entered in Word W516 and activated by the strobe-marker M2813. After execution of the key-code the NC resets the strobe-marker M2813.

A fixed code has been introduced for certain soft key functions. As with key simulation, this function is executed by entering the required code in W516 and activating with M2813. The appropriate soft key function must be displayed in the foreground or background mode for this. With the trailing edge of M2813 the PLC is told whether the function was properly executed.

Address	Function	Set	Reset
W272	Operating mode 1 = Manual operation 2 = Electronic handwheel 3 = Positioning with manual entry 4 = Program run/single block 5 = Program run/full sequence 7 = Pass over reference point	NC	NC
W274	Key-code for the operated, inhibited key, Signal via M2182	NC	NC
W516	Word with multiple-function Key-code for simulating TNC-key activated by M2813	PLC	PLC

Marker	Function	S	R
M2182	Inhibited key was operated	NC	PLC
M2187	Soft key function not executed	NC	NC
M2813	Activate the key from W516	PLC	NC
M2876	Inhibit the alpha keyboard	PLC	PLC
M2877	Inhibit the soft-key row below the screen	PLC	PLC
M2878	Inhibit the changeover keys at right of screen	PLC	PLC

Marker	Function		Key-code	Set	Reset
M2854	CHF	inhibit	58	PLC	PLC
M2855	PGM NAME	inhibit	59	PLC	PLC
M2856	L	inhibit	60	PLC	PLC
M2857	RND	inhibit	61	PLC	PLC
M2858	CC	inhibit	62	PLC	PLC
M2859	C	inhibit	63	PLC	PLC
M2860		inhibit	64	PLC	PLC
M2861		inhibit	65	PLC	PLC
M2862	MOD	inhibit	66	PLC	PLC
M2863	P	inhibit	67	PLC	PLC
M2864	I	inhibit	68	PLC	PLC
M2865	PGM CALL	inhibit	69	PLC	PLC
M2867	CR	inhibit	71	PLC	PLC
M2868		inhibit	72	PLC	PLC
M2869		inhibit	73	PLC	PLC
M2870		inhibit	74	PLC	PLC

Marker	Function		Key-code	Set	Reset
M2871		inhibit	75	PLC	PLC
M2872		inhibit	76	PLC	PLC
M2873		inhibit	77	PLC	PLC
M2874	TOUCH PROBE	inhibit	78	PLC	PLC
M2875	V	inhibit	79	PLC	PLC
M2880	TOOL DEF	inhibit	84	PLC	PLC
M2881	TOOL CALL	inhibit	85	PLC	PLC
M2882		inhibit	86	PLC	PLC
M2883		inhibit	87	PLC	PLC
M2884	↑	inhibit	88	PLC	PLC
M2885	←	inhibit	89	PLC	PLC
M2886	→	inhibit	90	PLC	PLC
M2887	CYCL DEF	inhibit	91	PLC	PLC
M2888	CYCL CALL	inhibit	92	PLC	PLC
M2889	LBL SET	inhibit	93	PLC	PLC
M2890	LBL CALL	inhibit	94	PLC	PLC

Marker	Function		Key-code	Set	Reset
M2891	NO ENT	inhibit	95	PLC	PLC
M2892	STOP	inhibit	96	PLC	PLC
M2893	EXT	inhibit	97	PLC	PLC
M2894	CL PGM	inhibit	98	PLC	PLC
M2895	DEL □	inhibit	99	PLC	PLC
M2896		inhibit	100	PLC	PLC
M2897	ENT	inhibit	101	PLC	PLC
M2898	GOTO □	inhibit	102	PLC	PLC
M2899	↓	inhibit	103	PLC	PLC
M2901	CE	inhibit	105	PLC	PLC
M2902	IV	inhibit	106	PLC	PLC
M2903	Z	inhibit	107	PLC	PLC
M2904	Y	inhibit	108	PLC	PLC
M2905	X	inhibit	109	PLC	PLC
M2906	Q	inhibit	110	PLC	PLC

Marker	Function		Key-code	Set	Reset
M2907	0	inhibit	111	PLC	PLC
M2908	1	inhibit	112	PLC	PLC
M2909	4	inhibit	113	PLC	PLC
M2910	7	inhibit	114	PLC	PLC
M2911	.	inhibit	115	PLC	PLC
M2912	2	inhibit	116	PLC	PLC
M2913	5	inhibit	117	PLC	PLC
M2914	8	inhibit	118	PLC	PLC
M2915	END	inhibit	119	PLC	PLC
M2916	APPR DEP	inhibit	120	PLC	PLC
M2921	3	inhibit	125	PLC	PLC
M2922	6	inhibit	126	PLC	PLC
M2923	9	inhibit	127	PLC	PLC

Key-code for the alphabetic keyboard:

xx 50 (Hex)
└→ ASCII key-code (see "Appendix")

Key-code for the soft-key row on the screen:

xx51 (Hex)

xx results as follows:

◁									▷
08	00	01	02	03	04	05	06	07	0A

Key code for changeover keys at right of screen:

xx52 (Hex)

xx results as follows:

	00
GRAPHICS TEXT SPLIT SCREEN	01

Code for soft key functions:

0000 (Hex): INTERNAL STOP
0100 (Hex): M (M-function)
0200 (Hex): S (S-function)
0300 (Hex): TOUCH PROBE
0400 (Hex): PASS OVER REFERENCE
0500 (Hex): RESTORE POSITION

Example:

If the "Positions-transfer" key is pressed in the operating mode "Positioning with manual entry" , a linear NC-block with all three principal coordinates (X, Y, Z) is to be generated.

```
66          CASE W272          ;Interrogate - operating mode
 .             .                   .
70          CM 3               ;Positioning with manual entry
 .             .                   .
75          ENDC
 .             .                   .
1102        EM                 ;End main program

1103        LBL 3              ;Operating mode: Positioning with manual entry
1104        L M10              ;Key simulation active?
1105        SN M2896           ;No, then disable "Positions-transfer" key
1106        L M2182            ;Disabled key operated?
1107        CMT 31             ;Yes, then call key simulation
1108        EM

1109        LBL 31             ;Key simulation
1110        L M10              ;Key simulation active?
1111        R M2896            ;Yes, then enable "Positions-transfer" key
1112        SN M10             ;Otherwise set key simulation active
1113        CASE B200          ;Perform single-step
1114        CM 130             ;Key L(line)
1115        CM 131             ;Key X
1116        CM 132             ;Key "Position-transfer"
1117        CM 133             ;Key Y
1118        CM 132             ;Key "Position-transfer"
1119        CM 134             ;Key Z
1120        CM 132             ;Key "Position-transfer"
1121        CM 135             ;Key "END-BLOCK"
1122        CM 141             ;Reset key simulation
1123        ENDC
1124        EM

1125        LBL 130            ;L(ine)
1126        L K60              ;Key-code for L(ine)
1127        = W102
1128        CM 136             ;Simulate key
1129        EM

1103        LBL 131            ;X
1131        L K109             ;Key-code for X
1132        = W102
1133        CM 136             ;Simulate key
1134        EM

1135        LBL 132            ;"Position-transfer"
1136        L K100             ;Key-code for "Position-transfer"
1137        = W102
1138        CM 136             ;Simulate key
1139        EM
```

```
LBL 133             ;Y
L K108              ;Key-code for Y
= W 102
CM 136              ;Simulate key
EM

LBL 134             ;Z
L K107              ;Key-code for Z
= W102
CM 136              ;Simulate key
EM

LBL 135             ;"END BLOCK"
L K119              ;Key-code for "END BLOCK"
= W102
CM 136              ;Simulate key
EM

LBL 136             ;Key simulation
L M2813             ;Strobe - key transfer from W516
JPT 137             ;Still set, then wait
L B200              ;Case byte
+ K+1
= B200              ;Increment case byte
L W102              ;Buffered key-code
= W516              ;To NC
LN M2813
S M2813             ;Set strobe (activate simulation)
EM

LBL 137             ;Return marker
EM

LBL 141             ;End key simulation
L M2813             ;Simulation performed?
JPT 137             ;No, then wait
L K+0
= B200              ;Reset step counter
L M10
R M10               ;Reset flag "Key simulation active"
R M2182             ;Reset marker "Disabled key operated"
EM
```

8.2 Machine-control panel

A manufacturer's specific machine-control panel can be connected to the HEIDENHAIN contouring controls. See under "Assembly and electrical installation".

25 PLC inputs (I 128 to I 152) and 8 PLC outputs (O 0 to O 7) are available on the female connector X46 for the evaluation of the keys on the machine-control panel. The evaluation of the signals from the machine-control panel must be performed in the PLC-program. The appropriate markers will be set thereby. For safety reasons a complement-marker must be reset when some functions are activated. This is especially so for keys with several contacts. If the complement-marker is not properly set or reset, the flashing error message "Error in PLC-program" will appear. The displayed code identifies the marker where the error has occurred.

An axis-direction key which has been operated can be stored by marker M2450 (Complement-marker M2466) . This means that the axis will be traversed until NC STOP occurs. This memory function must be activated by machine parameter MP7680 Bit 0.

MP7680 - Machine parameter with multiple-function

	Bit 0 - Memory function for axis-direction keys	0 = Not stored 1 = Stored

Marker	Function	Error message	Set	Reset
M2448	NC-start (edge evaluation)	1A	PLC	PLC
M2464	Complement NC-start			
M2449	Rapid traverse	1B		
M2465	Complement - rapid traverse			
M2488	NC-stop ("0" signifies stop)			
M2450	Memory function for axis-direction keys	1C		
M2466	Complement - memory function for axis-direction keys			
M2451	Feed-release	1D		
M2467	Complement - feed-release			
M2456	Manual traverse X+	1I		
M2472	Complement - manual traverse X+			
M2457	Manual traverse X-	1J		
M2473	Complement - manual traverse X-			
M2458	Manual traverse Y+	1K		
M2474	Complement - manual traverse Y+			
M2459	Manual traverse Y-	1L		
M2475	Complement - manual traverse Y-			

Marker	Function	Error-message	Set	Reset
M2460	Manual traverse Z+	1M	PLC	PLC
M2476	Complement - manual traverse Z+			
M2461	Manual traverse Z-	1N		
M2477	Complement - manual traverse Z-			
M2462	Manual traverse 4+	1O		
M2478	Complement - manual traverse 4+			
M2463	Manual traverse 4-	1P		
M2479	Complement - manual traverse 4-			
M2524	Manual traverse 5+	2M		
M2540	Complement - manual traverse 5+			
M2525	Manual traverse 5-	2N		
M2541	Complement - manual traverse			

Marker for spindle control: see section "Main spindle".

Example:
NC-start key with two contacts I 128 and I 129
axis-direction key X+ with one contact I 130

```
        .
71      L I128        ;First contact NC-start key
72      = M2448       ;NC-start
73      LN I129       ;Second contact NC-start key
74      = M2464       ;Complement - NC-start
        .
100     L I130        ;Axis-direction key X+
101     = M2456       ;Manual traverse X+
102     LN I130
103     = M2472       ;Complement - manual traverse X+
        .
```

9 Touch probe

The following 3D touch probes can be connected:
The triggering touch probes

- TS 120 with cable transmission and integrated interface electronics,
- TS 511 with infrared transmission of the switch signal and connected by a separate interface electronics unit (APE)
- TT 110 for tool calibration

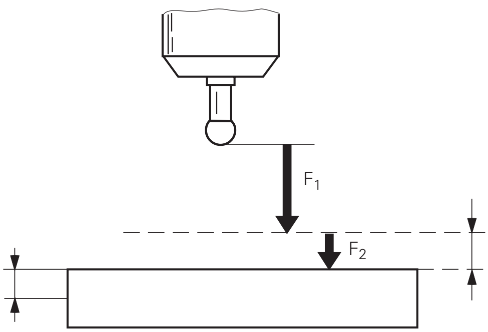
- The TNC 415/425 also supports the TM 110 measuring touch probe.

The chapter "Mounting and Electrical Installation" contains instructions for connecting the touch probes. Machine parameters MP6010, MP6200 and MP6500 determine which touch probes are connected. The machine tool builder must ensure that the spindle locks as soon as the touch probe has been inserted.

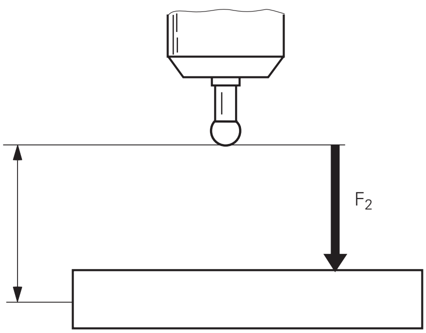
9.1 Standard touch probe cycles

The touch probe can be controlled either with the probing cycles in the "Manual" and "Electronic Handwheel" modes or by the "Touch Probe" function in the NC program (see User's Manual). The touch probe is interfaced to the measuring conditions using machine parameters.

Touch probe function "Reference Plane"



Touch probe cycles in the "Manual" and "Electronic Handwheel" operating modes



F1 = Rapid traverse during probing cycle:	MP6150 for triggering touch probe MP6200 = 0 MP6361 for measuring touch probe MP6200 = 1
F2 = Probing feed rate:	MP6120 for triggering touch probe MP6200 = 0 MP6360 for measuring touch probe MP6200 = 1

An error message "Touch point inaccessible" appears if the maximum measuring range (MP6130) is exceeded. Machine parameters MP6140 and MP6150 have no function with the probing cycles in the "Manual" and "Electronic Handwheel" modes.

The probing sequence must be enabled by the PLC with marker M2503. This marker is set by the NC when a probing cycle starts and the NC waits until the PLC resets marker M2503 before executing the probing function. A number of conditions are transferred to the PLC with markers M2022 to M2027. This information can be processed further in the PLC program. The probing function is controlled entirely from the NC.

In all modes if the stylus is deflected and marker M2502 is set, the controller stops the machine. If M2502 is not set, the controller only detects stylus deflection if the probing function has started. This is why HEIDENHAIN recommend setting marker M2502 as soon as the touch probe is in the spindle. This recommendation does not apply to the TS 511, however, since a stylus deflection is only recognised when the system is not in standby mode.

If M2502 is set, then the maximum feed-rate is limited by MP6150 in addition to MP 1010 and MP1020. MP7411 selects whether the tool data (length, radius, axis) from the last TOOL CALL block or from the calibrated data of the probe system are used in a touch probe block. The centre offset of the probe system can be determined when calibrating. This centre offset is then automatically corrected in all probing operations (see User's Manual). MP6160 defines whether spindle orientation for a rotation by 180° will be performed directly by the NC or through the PLC. If the NC orients the spindle directly, the PLC need only reset Marker M2499. If the PLC orients the spindle, the number of the M function will be entered in MP6560. For the triggering touch probe the rotation is activated by pressing a soft key. For the measuring touch probe the rotation is automatically activated during calibration. This can be deselected with MP6321. During every spindle orientation the Marker M2127 is set.

MP6010	Select touch probe Input: 0 or 1 0 = TS 120 1 = TS 511
MP6200	Selection between triggering or measuring touch probe Entry: 0 or 1 0 = Triggering touch probe (e.g. TS 120) 1 = Measuring touch probe (e.g. TM 110)
MP6120	Probing feed rate (triggering touch probe) Entry: 10 to 3000 [mm/min]
MP6360	Probing feed rate (measuring touch probe) Entry: 10 to 3000 [mm/min]
MP6130	Maximum measuring range Entry: 0.001 to 99 999.9999 [mm]
MP6140	Setup clearance above measuring point Entry: 0.001 to 99 999.9999 [mm]
MP6150	Rapid traverse in probing cycle (triggering touch probe) Entry: 10 to 10 000 [mm/min]

MP6321 Measuring the centre offset when calibrating the TM 110
Entry:0 or 1
0 = Calibration with measuring the centre offset
1 = Calibration without measuring the centre offset

MP6361 Rapid traverse in the touch probe cycle (measuring touch probe)
Entry: 10 to 10 000 [mm/min]

MP6160 Spindle orientation 180° rotation
Entry: –1 to 88
–1 = Spindle orientation directly through NC
0 = Function not active
1 to 88 = Number of the M function for spindle orientation by PLC

MP7411 Tool data in touch probe block
Entry: 0 or 1
0 = With the touch probe block the current tool data are overwritten with the calibrated data of the probe system.

1 = The current tool data are retained even with a touch probe block.

Marker	Function	Set	Reset
M2502	NC STOP with deflected stylus in all modes (not TS 511)	PLC	PLC
M2503	Enable marker for probing functions	NC	PLC
M2022	Touch probe not ready (no standby signal at connector X12 or signals from TM 110 are faulty)	NC	NC
M2023	Stylus deflected before start of probing cycle	NC	NC
M2025	Stylus deflected (probing sequence is executed)	NC	PLC
M2026	Probing sequence ended or interrupted	NC	NC
M2027	Low battery voltage (battery warning at connector X12); only evaluated during the probing sequence)	NC	NC
M2499	Open the spindle control loop	PLC	PLC
M2127	Spindle in motion	NC	NC

Example:
Enable of probing function in the PLC.

```
.
.
.
LN      I9      ;I9 = 0 => TS not in holder
=       M2502   ;stop if stylus deflected
;
L       M2503
AN      I9
R       M2503   ;acknowledge probing cycle
;
;
;safety function
;M03/M04 deactivate if TS not in holder
;
L       M2485   ;M03 activated?
O       M2486   ;M04 activated?
AN      I9      ;touch probe not in holder
S       M2487   ;display M05
R       M2485   ;deactivate M03
R       M2486   ;deactivate M04
R       M922    ;clear buffered marker M03
R       M923    ;clear buffered marker M04
R       M2488   ;NC stop
```

9.2 Digitizing with TS 120

Digitizing is possible with all HEIDENHAIN triggering touch probes. However, because the digitizing process can take several hours to complete, it is advisable to use the TS 120 touch probe with signal transmission by cable. The TS 511 touch probe with its infrared signal transmission can be continuously operated for up to 8 hours on one battery charge.

Technical Requirements

- "Digitizing with TS 120" is possible with TNC 415 using software types 259 96 and 259 97 and with TNC 407 using type 243 02 (see also the section "Software" in the chapter headed "Introduction",).
- Installing the software module "Digitizing with the TS 120". The ID. number of the logic unit indicates whether the software module is already installed (see the section "EPROM Locations" in the chapter headed "Introduction", and the section "Hardware Components" in the chapter "Installation and Electrical Connection").
  If the module is already installed, the software number of the option will appear beneath the NC and PLC software number, when the system is switched on or when the MOD key is pressed.
- Interfaced touch probe TS 120.
- The machine must be optimized for "Trailing error operation".

The digitizing sequence is optimized by the machine parameters.

Machine parameter MP6210 is the value for the oscillations executed by the touch probe as it scans the form. It is governed by the dynamic response of the machine.
The dynamic response is in turn determined by the Kv factor (trailing error operation). The greater the Kv factor, the greater the number of oscillations.

Machine parameter MP6210 determines the maximum probing feed rate in conjunction with the programmed probe point interval, PP.INT from the scanning cycles "Meander" and "Contour Lines":

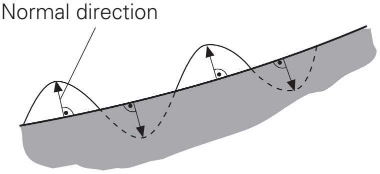
F[mm/min] = PP.INT [mm] x oscillations [1/s] x 60 [s/min]

This relation gives the formula for calculating the input value of MP6210:

$$\text{Oscillations [1/s]} = \frac{\text{optimized F [mm/min]}^{1)}}{\text{PP.INT [mm]} \times 60 \text{ [s/min]}}$$

1) The optimized probing feed rate depends on the feed rate in the normal direction (MP6230).

The feed rate in the normal direction (MP6230) is the resultant velocity at which the touch probe is fed from the non-deflected to the deflected condition perpendicular to the contour and vice versa.



Apart from the oscillation amplitude, the feed rate in the normal direction also determines the maximum scanning feed rate. If MP6230 is too low, the machine dynamics will be under-utilized and the scanning feed rate will also be low. If MP6230 is too high the oscillation amplitude will be too high. In this case the stylus will be seen to lift off from the contour (the stylus "taps" the surface of the workpiece) and the scanning feed rate no longer increases.

The maximum stylus deflection is defined by machine parameter MP6240. MP6240 will depend on the length of the stylus used. MP6240 defines the maximum travel by which the stylus retracts on inside corners. If the touch probe is not "cleared" after the stylus backs out by the travel defined in MP6240, the touch probe axis (e.g. Z) is retracted in the positive direction. Digitizing continues as soon as the touch probe is cleared.

Machine parameter MP6260 defines whether an M90 is appended to each NC block in the output digitized data (see also "Constant contour speed at corners with M90").

Machine parameter MP6270 defines the output format of the digitized data, i.e. the number of decimal places to which the coordinates are output.

MP6210 Number of oscillations in normal direction per second.
Input range 0 to 65.535 [1/s]

MP6230 Feed rate in normal direction
Input range: 0 to 1000 [mm/min]

MP6240 Maximum stylus deflection
Input range: 0 to 10.000 [mm]

MP6260 Output of M90 for NC blocks with digitized data
Input value: 0 = no M90 output
1 = M90 output in each NC block

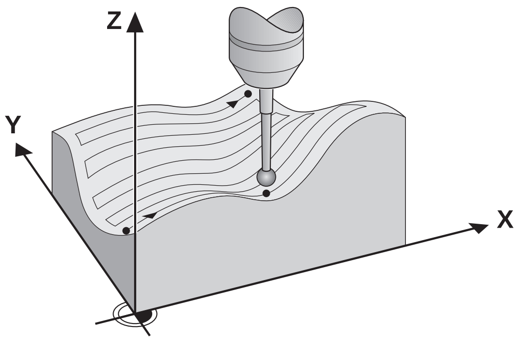
MP6270 Rounding decimal places
Input value: 0 = output 0.001 mm steps (1µm)
1 = output 0.01 mm steps (10 µm)
2 = output 0.0001 mm steps (0.1 µm)

9.2.1 Scanning cycles

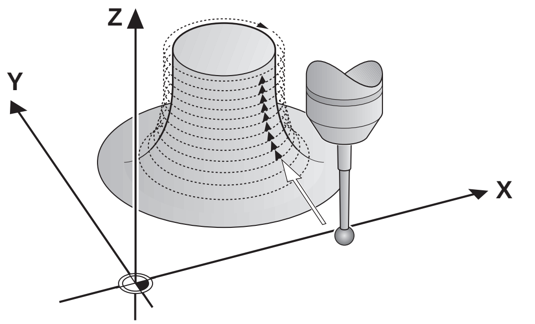
Direct access to the position control loop of the TNC controller allows the rapid recording of measured values (3 to 5 values per second). With a programmed probe point interval of 1 mm, this produces a scanning feed rate of 180 to 300 mm/min. Three scanning cycles are used for digitizing: "Range", "Meander" and "Contour Lines".

The "Range" cycle defines the cuboid scanning range and the file where the digitized data are stored.

The "Meander" cycle digitizes a 3-D form meander-wise (line by line) in the pre-defined range.



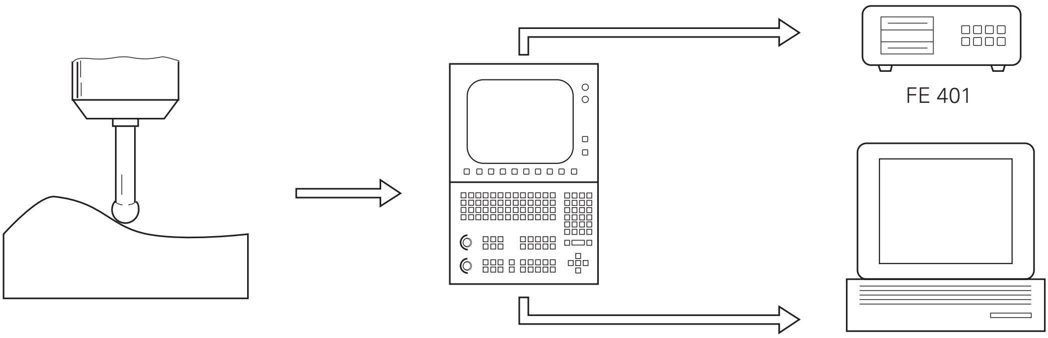
The "Contour Lines" cycle digitizes a 3-D form level-by-level in contour lines within a pre-defined range. Level-by-level digitizing is mainly used for forms with steep edges.



During meander digitizing a very flat surface may cause little movement in the probe axis. This can result in a lack of lubrication in the probe axis. MP6220 and MP6221 make it possible to lubricate the axis at the end of the lines.

MP6220	Traverse for lubricating the probe axis at end of line Entry: 0.000 to 999.999 [mm]
MP6221	Time after which the probe axis must be lubricated Entry: 0 to 65 535 [min]

The acquired data can be stored in the controller's program memory, on the FE 401 floppy disk unit or in a PC.



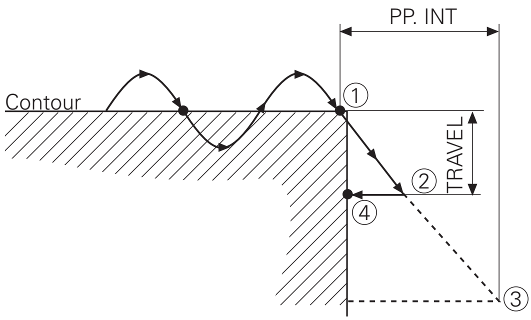
Further particulars about scanning cycles will be found in the TNC User's Manual.

9.2.2 Response of the digitizing sequence at corners

The scanning sequence responds differently for inside and outside corners. Here the two parameters PP.INT (maximum probe point interval) and TRAVEL from the "Meander" and "Contour Lines" scanning cycles operate like a travel limiter. Depending on the values that are entered for these parameters, either the travel or the probe point interval are limited.

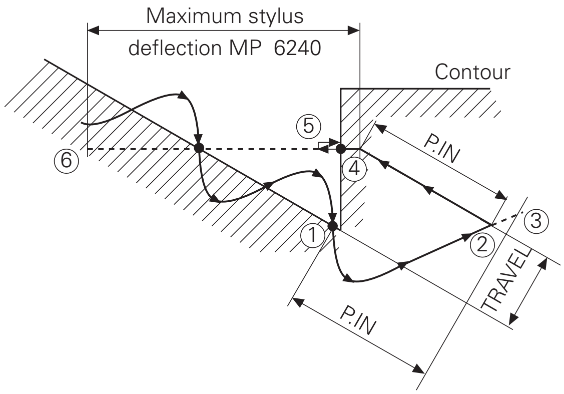
The travel is also responsible for geometrical accuracy at the corners. The smaller the stroke, the greater the accuracy of corner resolution. If too small a stroke is defined however, it may affect clearance at acute inside corners (minimum travel = 0.1 mm).

Outside Corners



On outside corners, having probed the last point ① the touch probe moves down the resultant straight line until it either makes workpiece contact again or hits one of the two limits TRAVEL ② or PP.INT ③. In the illustrated example TRAVEL is the limit, and the touch probe returns to the contour ④ in the inverse scanning direction. The new scanning direction is defined by the probed points ① and ④.

Inside Corners



On inside corners, having probed the last point ① the touch probe continues to oscillate in the same scanning direction but changes direction because it cannot get clear. It then moves down the resultant straight line until it either gets clear or hits one of the limits TRAVEL ② or PP.INT. ③

The touch probe moves in inverse scanning direction to get clear. If the programmed probe point interval PP.INT ④ is too small for the probe to clear, it travels in negative direction by up to the value of MP6240 (maximum stylus deflection). As soon as the touch probe is clear it returns to the contour ⑤ in the inverse travel direction. The new scanning direction is defined by the probed points ① and ⑤. If the touch probe has failed to clear even after it has backed out by the value of MP6240 ⑥, the touch probe axis (e.g. Z) is cleared in the positive direction. If the stylus is still deflected after it reaches the "clearance height" (see "Range" scanning cycle) the scanning sequence is aborted and an error message is displayed.

9.2.3 Optimizing the digitizing sequence

The following preparations should be made before optimizing machine and controller.

- Set up the flat workpiece with vertical face and plane surface in the machining plane (e.g. XY plane).

- Probe the surface with probing function "Surface = Datum" ("MANUAL" or "EL. HANDWHEEL MODE") and enter the reference plane as +0 mm.

- Basic setting of the relevant machine parameters for "Digitizing"

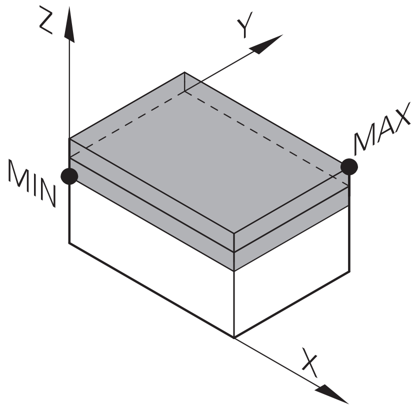
MP6210	=	5 [1/s]	Oscillations in normal direction
MP6230	=	30 [mm/min]	Feed rate in normal direction
MP6240	=	5 [mm]	Maximum stylus deflection

- The interface must be configured in the "PROGRAMMING AND EDITING" mode, depending on the type of desired memory for the digitized data (internal controller RAM, FE 401 or PC with "Block Transfer" program). See TNC 407/TNC 415 Operating Manual.
- Connect FE 401 or PC to the serial data interface if necessary.

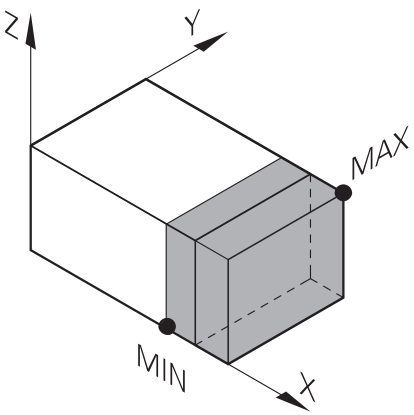
- Enter NC program with the scanning cycles "Range" and "Meander" and the scanning direction X and the probe point interval 1 mm, e.g.

0	BEGIN PGM OPTIDIGI MM	
1	BLK FORM 0.1 Z X+0 Y+0 Z-10	Blocks 1 to 4 are needed
2	BLK FORM 0.2 X+100 Y+100 Z+10	for the parallel graphics
3	TOOL DEF 1 L+0 R+4	of the TNC 415
4	TOOL CALL 1 Z S1000	
5	TCH PROBE 5.0 RANGE	Define the range
6	TCH PROBE 5.1 PGM NAME: DIGIDAT	to be digitized with
7	TCH PROBE 5.2 Z X+0 Y+0 Z-10	the program name for
8	TCH PROBE 5.3 X+100 Y+100 Z+10	the digitized data and
9	TCH PROBE 5.4 HEIGHT: +25	clearance height (absolute)
10	TCH PROBE 6.0 MEANDER	"Meander" scanning in X
11	TCH PROBE 6.1 DIRECTN:X	direction with probe point
12	TCH PROBE 6.2 TRAVEL:0.5 L.SPAC:1 P:P.INT:1	and line interval and the
13	END PGM OPTIDIGI MM	stroke (for clearing steep edges).

Optimize the X and Y axes by defining the range such that only the level surface of the component is scanned.



Optimize the Z axis by defining the range such that mainly the vertical face is scanned.



Procedure for optimizing machine parameters

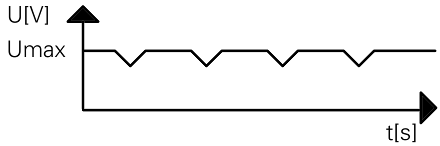
Connect a storage oscilloscope to the nominal command input of the servo amplifier for the X axis.

↓

In the "PROGRAM RUN/FULL SEQUENCE" mode, select OPTIDIGI program and operate the external START KEY.
Machine runs, digitizing starts.

↓

Increase the feedrate [1] with the override potentiometer until X nominal command attains. maximum value and shows only minor voltage dips (axis should just still traverse at even speed).



↓

Are oscillations in normal direction visible (stylus tip is seen to lift off workpiece surface)?

no → Increment MP6230 (e.g. by 10 mm/min) → back to "In the "PROGRAM RUN/FULL SEQUENCE" mode…"

yes ↓

Decrement Mp6230 again (e.g. by 10 mm/min)

↓

Repeat optimizing sequence for Y axis.
Switch the NC program to scanning direction Y [2].

↓

Repeat optimizing sequence for Z axis.
Change the scanning range accordingly.
Pay particular attention to vertical face.

[1] Keep a note of the maximum feedrate that is possible in the worst axis.

[2] The optimizing sequence is geared to the worst axis. This is why the feedrate must not be further increased with the override potentiometer and normal feedrate.

Calculation of possible oscillations in normal direction

$$\text{MP 6210 [1/s]} = \frac{\text{optimised F}^{\,1)}\ \text{[mm / min]}}{\text{PP.INT}^{\,2)}\ \text{[mm]} \times 60\ \text{[s / min]}}$$

When the calculated machine parameter MP 6210 is entered the feed override potentiometer is trimmed to the "attained feed rate".

1) Maximum possible feed rate noted during optimizing

2) PP.INT is the programmed maximum probe point interval from the "Meander" scanning cycle (in the example, PP.INT = 1 mm).

9.3 Digitising with the TM 110

The TM 110 measuring touch probe permits high digitising speeds up to 3 m/min (1181 ipm). The stylus deflection is measured in every direction directly by integral measuring systems and evaluated in the TNC.

Technical prerequisites

– Interfaced TM 110 touch probe
– "Digitising with TM 110" software protection module
– TNC 415 or TNC 425 as of software version 280 54 or 280 56
– PC for saving the surface data

The machine must be prepared for the use of the TM 110 measuring touch probe. The ballscrew must be mechanically clamped and it must be ensured that the spindle drive cannot be started while the probe is in use. The TM 110 must be at right angles to the machine axes (use a dial gauge if necessary). The square cover on the TM 110 can be used as a datum surface.

The "Digitising with TM 110" software module also enables digitising with the TS 120 touch trigger probe. The machine parameters for digitising with the TM 110 and TS 120 become available when the software module has been installed.

9.3.1 Interfacing the TM 110

MP6200 defines whether the measuring or the triggering touch probe is to be used.

Danger of breakage!
If you wish to use both the triggering and the measuring touch probe, you must make quite sure that the type of touch probe in use is entered in MP6200.

The counting direction of the measuring system signals must match the counting direction of the measuring systems for the machine tool axes (MP210). In the "Positioning with MDI" mode the machine can be positioned by pressing the PNT soft key. The machine must move in the direction in which the stylus was deflected. If this is not the case, the counting directions must be changed in MP6320.

MP6322 assigns the touch probe axes (the measuring systems in the probe) to the machine axes. For machines with swivel heads the user must enter the respective mounting position of the touch probe in MP6322. If the touch probe is in a horizontal position, the undefined deflection resulting from the stylus's own weight makes it impossible to find the centre of the stylus tip. It is therefore not possible to use the TM in a horizontal attitude to locate a workpiece. In a horizontal attitude the TM can be used only for digitizing.

Danger of breakage!
The mounting position of the touch probe must be entered correctly in MP6322, otherwise the calculation of the maximum deflection from MP6330 may be incorrect.

If the stylus is deflected by a distance greater than the value defined in MP 6330, the blinking error message "Stylus deflection exceeds max" is output. The maximum permissible deflection for the TM 110 is 4 mm.

MP 6310 defines the mean constant deflection depth during digitising. On standard parts an entry value of 1 mm has proven to be useful. On parts with sharp changes in directions (steep edges) that are scanned at high speed the deflection depth must be increased. The probing for can also be adjusted with the deflection depth (approx. 4 N per mm deflection).

After the "Meander" or "Contour line" cycle has been started, the probe moves at the feed rate defined in MP6361 to the clearance height, and then in the working plain to the point above the starting point. It then moves at the feed rate defined in MP6350 to the MIN point. If no touch point is reached, the probe moves to the first touch point in the direction defined in the cycle at the feed rate given in MP6350.

MP6360 (probing feed rate) and MP6361 (rapid traverse in the probe cycle) are effective in the standard touch cycles (Chapter 9.1).

MP6362 makes it possible to automatically reduce the probing feed rate if the ball tip moves too far from the path.

During contour-line scanning the probe sometimes ends the contour line at a point located near but not exactly at the starting point. MP6390 defines a target window within which the probe is considered to have returned to the starting point. The target window is a square. The entry value is half the length of one side of the square.

MP6200	Selection of triggering of measuring touch probe Entry: 0 or 1 0 = Triggering touch probe (e.g. TS 120) 1 = Measuring touch probe (e.g. TM 110)
MP6310	Stylus deflection depth (measuring touch probe) Entry: 0.1000 to 2.0000 [mm]
MP6320	Counting direction of the measuring system signals (measuring touch probe) Entry: %xxx Bit 0 = X axis 0 = Positive Bit 1 = Y axis 1 = Negative Bit 2 = Z axis
MP6321	Measuring the centre offset while calibrating the TM110 Entry: 0 or 1 0 = Calibrate and measure centre offset 1 = Calibrate without measuring centre offset

MP6322 Assignment of touch probe axes to the machine axes

Entry: 0 to 2

0 = Touch probe axis X

1 = Touch probe axis Y

2 = Touch probe axis Z

MP6322.0 Machine axis X

MP6322.1 Machine axis Y

MP6322.2 Machine axis Z

MP6330 Maximum stylus deflection (measuring touch probe)

Entry: 0.1 to 4.000 [mm]

MP6350 Feed rate for positioning to the MIN point and contour approach (measuring touch probe)

Entry: 10 to 3000 [mm/min]

MP6360 Probing feed rate (measuring touch probe)

Entry: 10 to 3000 [mm/min]

MP6361 Rapid traverse in scanning cycle (measuring touch probe)

Entry: 10 to 10 000 [mm/min]

MP6362 Feed rate reduction if stylus of the TM 110 is deflected away from its path

Entry: 0 or 1

0 = Feed rate reduction not active

1 = Feed rate reduction active

MP6390 Target window for contour-line end point

Entry: 0.1000 to 4.0000 [mm]

9.4 Tool calibration with the TT 110

The HEIDENHAIN TT 110 touch probe serves for measuring and inspecting tools. HEIDENHAIN provides standard cycles for automatic tool measurement and calibration with the TT 110 (see User's Manual).

Technical Prerequisites:

- TT 110
- TNC 4xx as of software version 280 58, 280 54, 280 56
- Central tool file TOOL.T must be active (via machine parameter)
- The machine must feature a controlled spindle (for spindle orientation)

9.4.1 Interfacing the TT 110

MP6500 enables the cycles for tool measurement. The cycles should only be enabled when a TT 110 is mounted and interfaced.

MP6505 defines the probing direction for tool radius measurement. For tool radius measurement, MP6530 defines the distance from the tool bottom to the top of the probe contact. In the tool table an additional tool-specific offset is entered in the field L-OFFS. MP6540 defines a safety zone around the probe contact of the TT 110. When a cycle for tool measurement is started the tool moves automatically from the "clearance height" entered in the cycle at the feed rate defined in MP6550 to the limit of the safety zone.

The probe disk diameter or the cube edge length is entered in MP6531. The coordinates of the probe contact centre referenced to the machine datum is entered in MP6580.

For calibrating and measuring individual teeth, the spindle must be brought to a stop at a defined angular orientation. MP6560 defines whether the spindle is oriented directly through the NC or through the PLC. If the spindle is oriented by the NC, the PLC need only reset the Marker M2499. If the PLC orients the spindle the number of the M function is entered in MP6560. The respective positions are then transferred as for the "Spindle Orientation" cycle. Marker M2127 is set during every spindle orientation.

Probing feed rate and spindle speed

For tool measurement of a non-rotating tool the probing feed rate is taken from MP6520.

For tool measurement of a rotating tool the probing feed rate and the spindle speed are automatically calculated by the TNC. The rotational speed is calculated from the maximum permissible surface cutting speed (MP6570) and the tool radius from the tool table. The rotational speed is limited to a maximum of 1000 rpm.

$$n = \frac{MP6570}{2 \times \pi \times r \times 10^{-3}}$$

n =	Rotational speed [rev./min]
MP6570 =	Maximum permissible surface speed of the tool edge [m/min]
r =	Tool radius [mm]

The probing feed rate is calculated from the calculated rotational speed and the measuring tolerance given in MP6510.

$v = \text{Meas. tolerance} \times n$	Meas. tolerance =	Meas. tolerance [mm] depending on MP6507
	n =	Rotational speed [rpm]

MP6507 defines the method of calculating the probing feed rate:

MP6507=0: Calculation of the feed rate with constant tolerance
This setting guarantees that the measuring tolerance remains constant regardless of the tool radius (MP6510). If the tool is very large, however, the necessary probing feed rate comes so close to zero that it falls below the lowest programmable increment. The smaller the maximum surface cutting speed and the permissible measuring error, the sooner this effect becomes noticeable.

MP6507=1: Calculation of the feed rate with variable tolerance
In this setting the permissible measuring tolerance changes depending on the tool radius. This ensures that there is a probing feed rate even for large tool radii. The measuring tolerance changes according to the following table:

Tool radius	Measuring tolerance
up to 30 mm	MP6510
30 to 60 mm	2 × MP6510
60 to 90 mm	3 × MP6510
90 to 120 mm	4 × MP6510

etc.

MP6507=2: Constant probing feed rate
The probing feed rate remains constant regardless of the tool. The absolute measuring error grows linearly with increasing tool radius.

$$\text{Meas. tolerance} = \frac{r}{5\,[\text{mm}]} \times \text{MP6510}$$

r =	Tool radius [mm]
MP6510 =	Max. permissible measuring error [mm]

$$v = \text{Meas. tolerance} \times n$$

$$v = \frac{\text{MP6570} \times \text{MP6510}}{10 \times \pi \times 10^{-3}}$$

v =	Probing feed rate [m/min]
MP6570 =	Maximum permissible surface speed at the cutting edge [m/min]

Markers in the PLC
Marker M2390 is set when a tool measuring cycle is started. Marker M2391 indicates whether a cycle for tool measurement or tool inspection was activated. If inspection shows that one of the entered tolerances is exceeded, the tool is inhibited and Marker M2392 or M2393 is set.

Markers M2502, M2503, M2022, M2023, M2025 and M2026 function as in the standard probing cycles. The cycles for tool measurement must therefore also be released by the PLC with Marker M2503.

If the spindle is oriented directly by the NC (MP6560 = –1), Marker M2499 must be reset by the PLC.

MP6500 Tool Calibration with TT 110
Entry: 0 or 1
0 = Cycles for tool calibration inhibited
1 = Cycles for tool calibration not inhibited

MP6505 Probing direction for tool measurement
Entry: 0 to 3
0 = Positive probing direction in the angle reference axis (0°axis)
1 = Positive probing direction in the +90°-axis
2 = Negative probing direction in the angle reference axis (0° axis)
3 = Negative probing direction in the +90°-axis

MP6507 Calculation of the probing feed rate
Entry: 0 to 2
0 = Calculation of the probing feed rate with constant tolerance
1 = Calculation of the probing feed rate with variable tolerance
2 = Constant probing feed rate

MP6510 Max. permissible measuring error for tool measurement with rotating tool
Entry: 0.002 to 0.999 [mm]

MP6520 Probing feed rate for tool measurement with non-rotating tool
Entry: 10 to 3000 [mm/min]

MP6530 Distance from tool lower edge to probe contract upper edge for tool radius measurement
Entry: 0.001 to 99.9999 [mm]

MP6531 Diameter or edge length of the TT 110 probe contact
Entry: 0.001 to 99 999.9999 [mm]

MP6540 Safety zone around the probe contact TT 110 for pre-positioning
Entry: 0.001 to 99 999.9999 [mm]

MP6550 Rapid traverse in the probing cycle for TT 110
Entry: 10 to 10 000 [mm/min]

MP6560 Spindle orientation for measuring individual teeth
Entry: –0 to 88
–1 = Spindle orientation directly via NC
0 = Function inactive (error message)
1 to 88 = Number of the M function for spindle orientation via PLC

MP6570 Max. permissible surface cutting speed at the tool edge
Entry: 1.0000 to 120.0000 [m/min]

MP6580 Coordinates of the TT 110 stylus centre referenced to the machine datum
Entry: –99 999.9999 to 99 999.9999 [mm]

MP6580.0 Axis X
MP6580.1 Axis Y
MP6580.2 Axis Z

Marker	Function	Set	Reset
M2390	Cycle for tool calibration started	NC	NC
M2391	0 = Tool measurement 1 = Tool inspection	NC	NC
M2392	Wear tolerance exceeded	NC	NC
M2393	Breakage tolerance exceeded	NC	NC
M2499	Open the control loop for the spindle	PLC	PLC
M2127	Spindle in motion	NC	NC

10 Electronic handwheel

The following handwheels can be connected to the HEIDENHAIN control systems:
- one integral handwheel HR 130, or
- one portable handwheel HR 330, or
- one portable handwheel HR 332, or
- up to three integral handwheels HR 150 using handwheel adapter HRA 110 (see also chapter "Mounting and Electrical Installation"). The operation of the electronic handwheel is described in the User's Manual.

Machine parameter MP7640 defines which handwheel is connected to the control. If a value greater than zero is entered but no handwheel is connected, the error message "Handwheel?" appears.

Shock and vibration can cause a slight movement of the handwheel and thus lead to an unwanted traverse movement. In order to avoid this a threshold sensitivity for the electronic handwheel is entered in machine parameter MP7660.

Traversing with the handwheel is disabled by setting marker M2826, i.e. handwheel pulses are suppressed.

A interpolation factor can be selected in the "Handwheel" operating mode. This interpolation factor determines the traverse distance per turn. In order to ensure that the rapid traverse rate fixed by the machine parameter MP1010.x is not exceeded the NC determines the minimum entry value for the interpolation factor. The NC control goes by the smallest value which was entered, i.e. according to the slowest axis.

MP7641 defines whether the interpolation factor can be entered direct at the TNC control panel or via PLC module 9036 (see also Chapter 7, section "PLC Modules").

Interpolation factor	Traverse distance per turn [mm]	Effective from rapid traverse rate: MP1010.x [mm/min.]
0	20	12 000
1	10	6 000
2	5	3 000
3	2.5	1 500
4	1.25	750
5	0.625	80
6	0.312	80
7	0.156	80
8	0.078	80
9	0.039	80
10	0.019	80

Machine parameter MP7670 can be used to select a higher limit than that calculated by the NC.

Initializing parameters for the handwheel are defined in machine parameters MP7645.x. These initializing parameters are currently only evaluated by HR 332 and HRA 110. The functions are described in the corresponding sections.

MP7640	Handwheel Entry: 0 to 6 0 = no handwheel 1 = HR 330 (all keys evaluated by NC) 2 = HR 130 or HR 330 3 = HR 330 (+, - and "rapid traverse" keys evaluated by PLC) 4 = HR 332 (all keys evaluated by PLC) 5 = up to three HR 150 via HRA 110 6 = HR 410
MP7641	Entry of interpolation factor Entry: 0 or 1 0 = entry at TNC control panel 1 = entry via PLC module 9036
MP7645	Initializing parameters for handwheel Entry: 0 to 255

MP7645.0 to MP7645.7 currently evaluated only by HR 332, HR 410 or HRA 110 (see following description).

MP7650	Count direction for handwheel Entry: 0 or 1 0 = negative count direction 1 = positive count direction
MP7660	Threshold sensitivity for electronic handwheel Entry range: 0 to 65 535 [increments]
MP7670	Interpolation factor for handwheel Entry range: 0 to 10
MP7670.0	Interpolation factor for low speed
MP7670.1	Interpolation factor for medium speed (HR 410 only)
MP7670.2	Interpolation factor for high speed (HR 410 only)

Marker	Function	Set	Reset
M2826	Suppress handwheel pulses	PLC	PLC

10.1 Integral handwheel HR 130

MP7640 = 2

When the axis keys are activated the corresponding highlight and the on-screen handwheel symbol move together.

10.2 Portable handwheel HR 330

MP7640 = 1: **HR 330 (all keys evaluated by NC)**

The axis keys on the keyboard are used to move the highlight for actual-value transfer or for setting datum point. The axis keys on the HR are used to move the handwheel symbol on the screen. The +, - and rapid traverse keys are evaluated directly by the NC and inputs I160 to I162 are also set accordingly.

MP7640 = 2: **HR 330 (all keys evaluated by NC)**

The axis keys on the keyboard and on the HR are used to move both the highlight and the on-screen handwheel symbol.
The +, - and rapid traverse keys are evaluated directly by the NC and inputs I160 to I162 are also set accordingly.

MP7640 = 3: **HR 330 (+, - and rapid traverse keys evaluated by PLC)**

The axis keys on the keyboard are used to move the highlight for actual-value transfer or for setting datum point. The axis keys on the HR are used to move the handwheel symbol on the screen. The +, - and rapid traverse keys must be evaluated by the PLC. Inputs I160 to I162 are set accordingly.

Assignment of keys to PLC inputs

Key on HR	PLC input
+	I160
-	I161
"rapid traverse"	I162

10.3 Portable handwheel HR 332

MP7640 = 4: **HR 332 (all keys evaluated by PLC)**

The axis keys on the keyboard are used to move the highlight for actual-value transfer or for setting datum point. The axis keys on the HR are used to move the handwheel symbol on the screen. MP7645.0 defines whether all 12 keys and their LEDs can be addressed by the PLC or whether the axis select keys and their LEDs are excluded.

10.3.1 Assignment of keys and LEDs to the PLC inputs and outputs

The 12 keys of handwheel HR 332 are assigned to different PLC inputs and the 12 LEDs to different outputs depending on the input value of machine parameter MP7645.0.
Pressing a key sets the corresponding PLC input. The matching LED lights up when the PLC output is set.

When MP7645.0 = 0, the X, Y, Z, IV and V keys and their LEDs are assigned to the NC. The other keys are assigned to PLC inputs I164 to I170. The LEDs are assigned to the PLC outputs O100 to O106.

X	IV	O100 I164
Y	V	O102 I166
Z	O103 I167	O104 I168
O101 I165	O105 I169	O106 I170

When MP7645.0 = 1, all 12 keys are assigned to PLC inputs I160 to I171 and all LEDs to PLC outputs O96 to O107.

O96 I160	O97 I161	O98 I162
O99 I163	O100 I164	O101 I165
O102 I166	O103 I167	O104 I168
O105 I169	O106 I170	O107 I171

MP7645 Initializing parameters for handwheel
MP7645.0 has the following meaning when an HR 332 is connected:

MP7645.0 Assignment of HR 332 handwheel keypad
Entry: 0 to 255
0 = all keys and LEDs are freely addressable with the PLC except for axis selection keys and their LEDs
1 = all 12 keys and LEDs are freely addressable with the PLC.

MP7645.1 to MP7645.7 are not assigned

10.3.2 PLC program example

In the following example the currently selected axis is evaluated so as to be traversed with the +/- keys. (MP7640 = 4, MP7641 = 0 and MP7645.0 = 0).

Key on HR 322	PLC input/output
+ key	I170
- key	I169
+ LED	O105
- LED	O106

```
;Main program
LBL 1
PS        K+9      Interrogate selected handwheel axis with module 9035
CM        9035
PL        B0
L         K+0      Delete the 8 markers for "Manual
B=        M2456    Traverse" of axis X to IV
L         K$FF     Set the 8 complement markers
B=        M2472
L         MO       PLC outputs O100
ON        MO       to O106 are deleted.
R         O100     All LEDs are
R         O101     turned off except the
R         O102     LED of the selected axis.
R         O103
R         O104
R         O105
R         O106
Case      BO       Evaluation of selected handwheel axis
CM        10       X-axis
CM        11       Y-axis
CM        12       Z-axis
ENDC
EM

LBL 10             X-axis
LN        M2457    Unless X- is active
A         I170     and + key is pressed,
S         M2456    traverses in
R         M2472    X+ direction.
S         O106     LED on - key lights up.
EM

LBL 11    Y-axis
.
EM
LBL 12    Z-axis
.
EM
LBL 13    Axis IV
EM
```

10.4 HR 410 Portable Handwheel

MP7640 = 6
With MP7645.0 you define whether the keys on the handwheel are evaluated by the NC or the PLC.

MP7645.0 = 0
Keys are evaluated by NC

X		IV
Y		V
Z		ACTUAL-POSITION CAPTURE
FEED RATE SLOW	FEED RATE MEDIUM	FEED RATE FAST
–		+
O109 I173	O110 I174	O111 I175

All keys except for the functions keys A, B, and C, are evaluated by the NC. MP7670.x defines the interpolation factor for the slow, medium and fast settings. MP7671.x defines the speed for the slow, medium and fast settings. The speed is entered as a percentage of the manual feed rate (MP1020.x).

MP7645.0 =1
Keys are evaluated by PLC

O96 I160		O97 I161
O98 I162		O99 I163
O100 I164		O103 I167
O104 I168	O105 I169	O106 I170
I171		I172
O109 I173	O110 I174	O111 I175

All keys are evaluated by the PLC. Module 9036 sets the handwheel's axis and interpolation. With W766 you can influence the feed rate by pressing the direction keys.

MP7645	Initializing parameters for handwheel When an HR 410 is installed, MP7645.0 has the following meaning:	
MP7645.0	Evaluation of HR 410 handwheel keypad	
Input:	0 =	Keys evaluated by NC
	1 =	Keys evaluated by PLC

MP7645.1 to MP7645.7 are without function

MP7670	Interpolation factor for handwheel
Input:	0 to 10
MP7670.0	Interpolation factor for low speed
MP7670.1	Interpolation factor for medium speed (only HR 410)
MP7670.2	Interpolation factor for high speed (only HR 410)

MP7671	Manual feed rate in handwheel mode with HR 410
Input:	0 to 1000 [% of MP1020]
MP7671.0	Low speed
MP7671.1	Medium speed
MP7671.2	High speed

10.5 Integral handwheels HR 150 with handwheel adapter HRA 110

MP7640 = 5

If the step switch is used for the selection of the interpolation factor (S1) then inputs I160 to I167 must be evaluated in the PLC and the result must be displayed with the aid of PLC module 9036.

The third handwheel can be assigned to any desired axis. MP7645.2 defines whether the axis for the third handwheel is selected with the axis selector switch (switch S2, see MP7645.0) or is permanently assigned in machine parameter MP7645.1.

MP7645 Initializing parameters for handwheel

MP7645.0 to MP7645.2 have the following meaning when an HRA 110 is connected:

MP7645.0 Third handwheel assigned by axis selector switch
Entry: 0 to 255

Entry	Switch position	Handwheel #3
0	1 (left stop)	Z axis
	2	4th axis
	3	Axis 5
1	1 (left stop)	X axis
	2	Y axis
	3	Z axis
	4	4th axis
	5	Axis 5
2	3	Z axis
	4	4th axis
	5	Axis 5

If no axis selector switch is connected (see also MP7645.2) the third handwheel is assigned an axis according to the input value of MP7645.1.

MP7645.1 Handwheel #3 assigned by machine parameter
Entry: 0 to 255
1 = X axis
2 = Y axis
4 = Z axis
8 = 4th axis
16 = Axis 5

MP7645.2 Handwheel #3 assigned by axis selector switch or MP7645.1
Entry: 0 to 255
0 = Assigned by axis selector switch according to MP7645.0
1 = Assigned by MP7645.1
The axis selector switch only transmits data to PLC inputs I168 to I175.
The axis selector switch can therefore be used for other functions.

MP7645.3 to MP7645.7 are not assigned

10.5.1 Assignment of switch positions to the PLC inputs

The tables below give the assignment of the switch positions of S1 and S2 to PLC inputs I160 to I175.

The two switches operate with a 0 V logic, e.g. if switch S1 is in position 3 then input I162 is logic 0 and inputs I160, I161 and I163 to I167 are logic 1.

Step switch S1

Step switch for selecting interpolation factor

Switch position	PLC input
1 (left stop)	I160
2	I161
3	I162
4	I163
5	I164
6	I165
7	I166
8 (right stop)	I167

Step switch S2

Step switch for selecting axis

Switch position	PLC input
1 (left stop)	I168
2	I169
3	I170
4	I171
5	I172
6	I173
7	I174
8 (right stop)	I175

10.5.2 PLC program example

In the following example the positions of switch S1 for selecting the interpolation factor are assigned by evaluating inputs I160 to I167 from the lowest permissible interpolation factor from MP7670 to the maximum interpolation factor = 10 and transferred from the PLC to the NC (MP7640 = 5 and MP7641 = 1).

In this PLC example the minimum interpolation factor is read from MP7670 and processed with PLC module 9032.

```
LB      I160        Assignment of inputs I160 to I167
=       B1          to byte (0V logic, negative logic)

==      K0          if all inputs are 0, then
JPT     103         error message

LN      B1          Inversion byte 1 (positive logic)
=       B2

==      K0          if all bits are 0, then
JPT     103         error message

LBL     100         Loop for bit test of "inverted inputs"
L       B2          I160 to I167 in byte 2
BT      B3

JPT     101         Jump when bit set, i.e. input Ixxx = 0V
L       B3          prepare next bit for polling, i.e.
+       K+1         read input if input Ixxx was not 0V.
=       B3
JP      100

LBL     101

PS      K+7670      Read lowest interpolation factor from
PS      K+0         MP7670 with PLC module 9032
CM      9032

PL      B4          Result is in byte 4

L       B3          Current switch position (0 to 7) plus
+       B4          minimum interpolation factor from MP7670
=       B5          is new interpolation factor

<=      K+10        if this is greater than 10, then
JPT     102
L       K+10        the new interpolation factor is set
=       B5          equal to 10.

LBL     102

PS      K+4         Transfer new interpolation factor
PS      B5          to all axis subdivision
```

CM	9036	factors (PLC-<NC).
PL	B6	Read error code
L	B6	if transfer fails,
>	K0	then error message
JPT	203	
EM		
LBL	1032	Error message subprogram
.		

Example:

Switch position 4:

Minimum interpolation factor: MP7670 = 5

The results of the PLC example are as follows:

Inputs: I160 to I162 and I164 to I167 are logic 1
Input: I163 is logic 0
Byte 1: 11110111
Byte 2: 00001000
Byte 3: 3
Byte 4: 5
Byte 5: 8

Step switch S2 is configured as follows according to the above example:

Switch position	Interpolation factor
1 (left stop)	5
2	6
3	7
4	8
5	9
6	10
7	10
8 (right stop)	10

11 Analogue inputs and outputs

11.1 Analogue inputs

Eight analogue inputs are available on the input/output extensions PL 410, PL 410 B (Id.-Nr 263 371 02) and PA 110. Of these, four are analogue inputs for Pt 100 thermistors (range 0° to 100°C; resolution 0.5°) and four are analogue inputs for ± 10 V DC (resolution 100 mV). The analogue values are converted to digital values in the PL or PA and transferred to PLC words.

The content of the words can be further processed in the PLC for a variety of purposes (e.g. compensating for thermal expansion, see section "Thermal compensation").

The word addresses in the PLC depend on whether the analogue inputs are on the first or second extension. Machine parameter 4410 defines the extension on which the analogue inputs are found.

On the PL 410 the analogue inputs must be activated by a DIL switch. See chapter "Installation and electrical connection", section "PLC Inputs/Outputs".

MP4410 Activate analogue input
Entry: %xx

Bit 0 =	0	No analogue inputs on extension #1
	1	Analogue inputs on extension #1
Bit 1 =	0	No analogue inputs on extension #2
	1	Analogue inputs on extension #2

Assignment of word addresses to the analogue inputs:

Input PA	Input PL	First extension	Second extension	Function	Set	Reset
X2	X15	W496	W464	Voltage input 0	NC	NC
X3	X16	W498	W466	Voltage input 1		
X4	X17	W500	W468	Voltage input 2		
X5	X18	W502	W470	Voltage input 3		
X7	X19	W504	W472	Temperature input 0	NC	NC
X8	X20	W506	W474	Temperature input 1		
X9	X21	W508	W476	Temperature input 2		
X10	X22	W510	W478	Temperature input 3		

Internal value range:

Voltage input: –100 to +100 (–10V to +10V)
Temperature input: 0 to 200 (0°C to 100°C)

11.2 Analogue outputs

The logic unit has six analogue outputs. Five of these are always used to output the nominal value of the axes. The sixth analogue output is usually used as a nominal value output for the spindle.
If the nominal value for the spindle is not an analogue output, or if the spindle is not needed (e.g. water jet and laser cutting machines), then an alternative function can be defined for analogue output S (X8, Pin 8) in MP3011.

See also Chapter 17 "Special functions for laser cutting machines".

12 Increment positioning

In the "Electronic handwheel" mode, the function "Incremental jog positioning" can be activated by the PLC program.

The function "Increment positioning" is enabled by marker M2498. In the "Electronic Handwheel" mode the "Jog Increment" dialogue is displayed in addition to the "Interpolation factor" dialogue.

For increment positioning, the start marker and the corresponding complement marker for the desired axis must be activated. If start and complement markers are not set correctly then the message ERROR IN PLC-PROGRAM XX appears.

In the PLC program the activation of increment positioning is usually combined with the axis direction keys (see following PLC example). The jog increment can be limited with Module 9036.

Marker	Function and entry	Error message	Set	Reset
M2498	Enable increment positioning		PLC	PLC
M2512	Start increment positioning X+	2A	PLC	PLC
M2528	Complement increment positioning X+			
M2513	Start increment positioning X-	2B		
M2529	Complement increment positioning X-			
M2514	Start increment positioning Y+	2C		
M2530	Complement increment positioning Y+			
M2515	Start increment positioning Y-	2D		
M2531	Complement increment positioning Y-			
M2516	Start increment positioning Z+	2E		
M2532	Complement increment positioning Z+			
M2517	Start increment positioning Z-	2F		
M2533	Complement increment positioning Z-			
M2518	Start increment positioning 4th axis+	2G		
M2534	Complement increment positioning 4+			
M2519	Start increment positioning 4-	2H		
M2535	Complement increment positioning 4-			
M2520	Start increment positioning 5+	2I		
M2536	Complement increment positioning axis 5+			
M2521	Start increment positioning 5-	2J		
M2537	Complement increment positioning axis 5-			

PLC example:

Axis direction key X+ with one contact I138
Axis direction key X- with one contact I133

```
L          M2052      Interrogate mode
CMT        10         Electronic handwheel
.
.
EM                    Main program end
.
.
LBL 10                Increment positioning
L          M2052
S          M2498      Enable increment positioning

R          M2512      Delete marker for increment positioning in X axis
R          M2513
S          M2528
S          M2529

L          I138       Increment positioning in X+ direction
AN         I133
S          M2512
R          M2528

L          I133       Increment positioning in X- direction
AN         I138
S          M2513
R          M2529

EM
.
```

13 Hirth coupling

A frequent method of locking rotary axes and swivel heads is the so-called Hirth coupling in which more or less finely splined plates engage to form a rigid connection.

The Hirth functions are implemented in the PLC program. The only operation in the NC is a rounding off according to the grid spacing in MP430 when setting the datum point. Repositioning is initiated as PLC positioning or an error message is displayed depending on whether the axis is automatically or manually positioned with the Hirth coupling. MP420 defines the "Hirth coupling" functions and MP430 the grid spacing. The positioning accuracy and other parameters are defined in free machine parameters.

13.1 Positioning in Manual or Electronic Handwheel mode

The Axis in Position marker (M2011) is reset when selecting the 4th or 5th axis which is traversed with the electronic handwheel. This can be used to initiate unlocking of the Hirth coupling. The same applies to traversing with the axis direction key. Setting the Axis in Position marker starts a check of the nominal position compared with grid. Repositioning is executed as PLC positioning by the shortest route, i.e. to the nearest grid point.

13.2 Positioning in controlled mode

NC Start also resets the Axis in Position marker. This in turn initiates an internal check of the target position. If the target position is not in the grid then an error message is displayed.

13.3 Program example

The positioning of the 4th axis and locking by the Hirth coupling is described in the following example. The following machine parameters and markers are required for the example:

MP420.3		Activate Hirth coupling axis 4 1 = active, 0 = not active
MP4310.1	W978	Direction for PLC positioning for repositioning
MP430.3		Hirth grid axis 4
MP960.3		Shift Hirth grid against reference mark (machine datum)
MP4210.3	D776	Positioning window for PLC positioning
MP4201.3	D780	Input resolution for controlled mode
MP4220.3	W966	Feed rate for PLC positioning
	W540	PLC Position for axis 4
	W566	Feed rate for PLC position
	D12	Actual value (Ref. related)
	D20	Trailing error
	D36	Hirth grid left of nominal value
	D40	Hirth grid right of nominal value
	D44	Nominal grid value
	D28	Hirth grid

Hirth positioning in controlled mode

```
L       M0       ;create logic zero
R       M0
LN      M1       ;create logic one
S       M1
```

Activate initializing after switch-on (Module 290)

```
L       M2180    ;first PLC cycle after switch-on
O       M2185    ;first PLC cycle after interruption

CMT     290      ;read MP420.3
L       M301     ;Hirth function active
A       M304     ;Submit 1 terminated
CMT     292      ;read Hirth grid MP430.3

L       M2011    ;axis 4 in position
A       M305     ;Submit 2 terminated
S       M302     ;Switch-on delay terminated
```

Interrogate operating mode

```
L       M2053    ;positioning with manual input
O       M2054    ;program run/single block
O       M2055    ;full sequence
=       M5       ;controlled mode
```

Check nominal position

```
LN      M2011   ;axis 4 not in position
AN      M300
A       M305
A       M5
CMT     370     ;nominal value in grid spacing?
L       M4      ;1 = not in grid
A       M5      ;controlled mode
AN      M2011   ;axis 4 not in position
S       M3018   ;error message "Nominal position not attainable
                 in controlled mode"
```

Activate Hirth module

```
LN      M2011   ;axis 4 not in position
A       M301    ;Hirth status 1 = activated
A       M302    ;switch-on delay terminated
A       M305
S       M300    ;set marker for memory
L       M300    ;reset by subprogram 300
AN      M3018   ;nominal position not attainable in controlled mode
CMT     300     Hirth positioning
EM
LBL     290     ;interrogate Submit
RPLY    B128
<>      K + 0
EMT
SUBM    291
=       B128
EM
LBL     291     ;read MP420.3
PS      K + 420 ;MP420
PS      K + 3   ;Index 3
CM      9032
PLW             ;value 1/10 000
<>      K + 0
=       M301    ;1 = active, 0 = not active
L       M1
S       M304    ;Submit 1 ready
L       M1      ;M1 = 1
S       M2719   ;start word memory processing
S       M2495   ;open activation control loop, axis 4
S       M2547   ;open control loop, axis 4
EM
LBL     292     ;interrogate Submit
RPLY    B128
<>      K + 0
EMT
SUBM    293
=       B128    ;Job identifier
EM
```

```
LBL     293      ;read MP430.3
PS      K + 430
PS      K + 3
CM      9032
PL      D28      ;Hirth grid
L       M1
R       M302
S       M305     ;Submit 2 ready
EM
LBL     300      ;Subprogram for Hirth positioning
L       M2719    ;word memory processing started
SN      M3021    ;PLC error message: word memory not opened
L       M2011    ;axis 4 in position
AN      M2707    ;strobe PLC positioning axis 4 not active
CMT     30       ;calculate nominal grid value left and right
CM      360      ;actual axis 4 in grid spacing
```

Locking and unlocking

```
LN      M2011    ;axis 4 in position not fulfilled
S       O3       ;reset lock
L       M2       ;axis in grid spacing
A       M2011    ;axis 4 in position
R       O3       ;lock
```

Close control loop

```
LN      M2011    ;axis 4 in position
A       I4       ;lock released
AN      I6       ;axis not locked
R       M2547    ;close control loop axis 4
```

Open control loop

```
L       M2       ;axis in grid spacing
A       M2011    ;axis in position
A       I6       ;axis locked
AN      I4       ;axis not locked
S       M2547    ;open control loop axis 4
R       M300     ;reset activation Hirth UP
```

Close control loop

```
LN      I4       ;lock released
AN      I6       ;axis not locked
AN      M2011    ;axis in position
AN      M2547    ;control loop axis 4 closed
S       M2547    ;open control loop if lock not released
```

Positioning on grid

```
L       M2051   ;manual operation mode
O       M2052   ;electronic handwheel mode
AN      M2      ;axis 4 not in grid spacing
A       M2011   ;axis 4 in position
AN      M2707   ;axis 4 PLC positioning not active
CMT     340     ;start PLC positioning axis on grid spacing
```

Grid not reached

```
L       M5      ;controlled operation mode
AN      M2      ;axis 4 not in grid spacing
AN      M2011   ;axis 4 not in position
S       M3023   ;PLC error message: error on "Automatic" mode
A       M2      ;axis 4 in grid
A       M2011   ;axis 4 not in position
R       M3023   ;PLC error message reset
EM
```

LBL 330 Adjacent grid spacing relate to nominal axis position

```
PS      K + 8   ;target address
PS      K + 3   ;trailing error
CM      9041    ;upload coordinate value
PS      K + 0   ;target address
PS      K + 2   ;REF value
CM      9041    ;upload coordinate value
```

Calculate left and right grid

```
L       D12     ;REF value axis 4
+       D20     ;trailing error axis 4
=       D44     ;nominal grid value
```

Left grid

```
L       D44
PSW             ;load to stack
<       K + 0
CMT     331     ;decrement by Hirth grid
PLW             ;load to ACCU
/       D28
x       D28     ;suppress residual spacing
=       D36     ;left grid point
```

Right grid

```
L       D36
+       D28
=       D40     ;right grid point
```

Save direction of traverse

```
L       M2163   ;direction of traverse axis V
=       M3      ;save to buffered marker
EM
```

```
LBL     331
PLW
-       D28
PSW
EM
```

PLC Positioning to adjacent grid spacing

```
LBL     340      ;PLC positioning on grid spacing
CM      350      ;nearest grid spacing
PLW
=       D540     ;new nominal position
L       W966     ;MP4220.3 read feed rate
=       W566     ;feed rate axis 4 (MP4220.3)
L       M1       ;M1 =1
S       M2707    ;start PLC positioning axis 4
EM
LBL     350      ;calculate nearest grid spacing
LN      M2208    ;programmable with MP4310.1
JPT     351
L       D44      ;nominal grid value (ref value)
-       D36      ;left grid point
PSW              ;left difference
CM      390      ;absolute amount
L       D44
-       D40      ;right grid point
PSW              ;right difference
CM      390      ;absolute amount
PLW
=       D0       ;right difference
PLW              ;left difference
>       D0       ;compare
JPT     353      ;load right grid point
JPF     354      ;load left grid point

LBL     351      ;determine direction with preset value from MP
L       M3       ;traverse direction for PLC positioning
XO      M2209    ;programmable with MP4310.1
JPF     353      ;load right grid point
JPT     354      ;load left grid point
EM
LBL     353      ;load right grid spacing for PLC positioning
PS      D40      ;right nominal grid value
EM
LBL     354      ;load left grid spacing for PLC positioning
PS      D36      ;left nominal grid value
EM
```

```
LBL     360     ;actual axis in grid spacing
PS      K0      ;target address
PS      K2      ;REF value
CM      9041    ;upload coordinates
L       D36     ;coordinate left grid point (ref value)
-       D12     ;current ref position
PSW
CM      390     ;absolute amount
PL      D0
L       D40     ;right grid point (ref value)
-       D12     ;current ref position
PSW             ;load to data stack
CM      390     ;absolute amount
PLW             ;distance to right grid point
<       D776    positioning window from MP4210.2 for Hirth positioning
O[
L       D0      ;distance to left grid point
<       D776
]
=       M2      ;axis in grid position
EM
LBL     370     ;check, nominal value = grid spacing
PS      K + 0   ;target address
PS      K + 4   ;distance to go
CM      9041    ;upload coordinates
PS      D12     ;save distance to go to stack
PS      K + 8   ;;target address
PS      K + 3   ;trailing error
CM      9041    ;upload coordinates
PS      K + 0   ;target address
PS      K + 2   ;ref value
CM      9041    ;calculate new nominal value
PLW             ;distance to go
+       D12     ;ref value axis
+       D20     ;trailing error
MOD     D28     ;Hirth grid
PSW             ;enter data in stack memory for absolute amount
CM      390     ;absolute amount
PLW             ;read absolute amount
=       D0
L       D28     ;Hirth grid axis 4
-       D0      ;absolute amount
=       D4
L       D0
=       D4
JPT     371
L       D0
=       D4
EM
```

```
LBL     371
L       D4
>       D780        ;MP4210.3 programming accuracy axis 4 for Hirth
=       M4          ;== 1 scale graduation cannot be approached
EM
LBL     390         ;convert to absolute amount
PLW                 ;load ACCU
PSW                 ;load to data stack
BT      K + 31      ;write bit 31 to L accu, bit test
JPF     391         ;0 = positive, jump if accu = 0
PLW                 ;load ACCU
X       K - 1       ;sign
PSW                 ;write data to data stack
EM
LBL     391         ;jump label
EM
```

14 Datum correction

The PLC datum correction function is used to shift the zero or datum point with the PLC program.

Each axis (X, Y, Z, 4, 5) is assigned a double word (D528 to D544) for the correction value. The datum correction is activated by Marker M2716 during a strobe. The correction is computed in the actual value display; the display then relates to the shifted coordinate system.

Example:

Actual position display for X axis without correction = 50
Correction in D528 = +20

Strobe marker M2716 set, i.e. correction is active
New actual position display X= +70

The corrections can be transferred to double words, D528 to D544, from different memory areas:

- Enter values in MP4210.0 to MP4210.47 and they will also be in D768 to D956; now copy values via PLC program into D528 to D544.

- Define corrections in the NC program in function FN19 and they will also be in D280 or D284 (see description of FN19); now copy values via PLC program into D528 to D544.

A description of the machine geometry via machine parameters MP7510 and following (e.g. cylinder surface) must have the same reference position as any datum shift.

Address	Function
D528	Datum correction for X axis
D532	Datum correction for Y axis
D536	Datum correction for Z axis
D540	Datum correction for IV axis
D544	Datum correction for V axis

Marker	Function
M2716	Strobe marker for datum correction

PLC example:

Datum correction with M20 activated, with M21 deactivated

```
.
LN      M2045
S       M10

L       M1920
A       M2045     ;M20 activated
A       M10       ;Buffered marker for strobe marker 2716
CMT     200       ;Datum correction call
R       M10
L       M1921
A       M2045     ;M21 activated
S       M10       ;Buffered marker for strobe marker 2716
C       201       ;Deactivate datum correction call
R       M10
.
.
L       M1920
O       M1921
A       M2045
AN      M2716
S       M2482     ;Feedback for M20 and M21
EM                ;End of main program
LBL 200           ;Activate module for datum correction
L       D896      ;Value from MP4210.32
=       D528      ;Shift X axis
L       K0        ;Do not shift Y, Z, 4th and 5th axis
=       D532
=       D536
=       D540
=       D544
L       M10
S       M2716     ;Activate datum compensation
EM
LBL 201           ;Deactivate module for datum correction
L       D900      ;Value from MP4210.33
=       D528      ;Shift X axis
L       K0        ;Do not shift Y, Z, 4th and 5th axis
=       D532
=       D536
=       D540
=       D544
L       M10
S       M2716     ;Activate datum correction
EM
```

15 Tool changer

A tool changer can be operated with the PLC of the HEIDENHAIN contouring control. If the tool changer is to have servo-controlled axes this must be done using a positioning module (see chapter "Positioning Module"). However it is possible to control a tool changer by simple proximity switches.

Tool data are stored in the tool table and complete tool management (replacement tool, tool life etc.) is handled by the NC. The NC supplies the PLC with the information it needs to control the tool changer via markers and words.

15.1 Tool table, Pocket table

The operator is able to edit the tool table in the "Program run" mode.

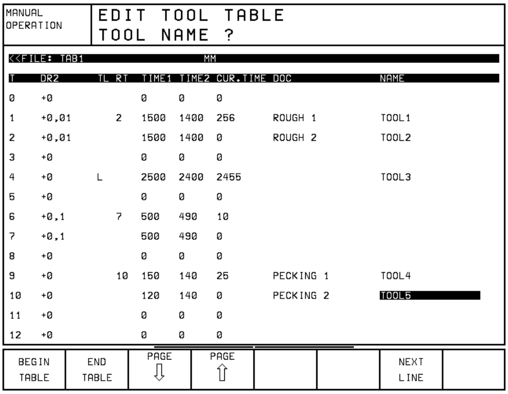
The tool table is called with the "TOOL TABLE" soft key. From here the operator calls the pocket table with the "POCKET TABLE" soft key (see User's Manual).

When the operator is in the tool table or pocket table editor, he can upload or download the tool or pocket table with the EXT key.
Identifiers T and R are assigned for the tool table and pocket table respectively on the external memory.

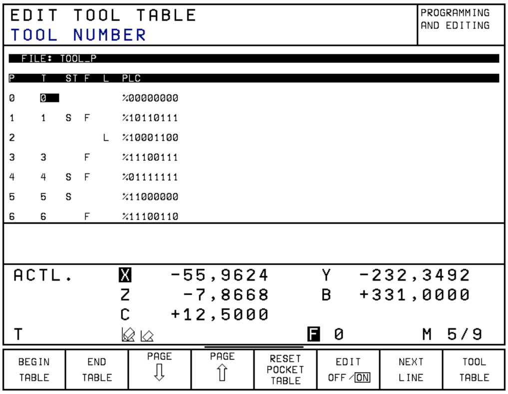
The following fields can be edited in the tool table:

NAME :	16 character alphanumeric tool name
L :	Tool length
R :	Tool radius
R2 :	Tool radius 2 for toroidal cutters
DL :	Oversize for tool length
DR :	Oversize for tool radius
DR2 :	Oversize for tool radius 2
TL :	Tool locked?
RT :	Replacement tool
TIME1 :	Max. tool life (M2094)
TIME2 :	Max. tool life (TOOL CALL)
CUR. TIME:	Current tool life
DOC :	Comment on the tool
CUT :	Number of tool cutting edges
LTOL :	Wear tolerance for tool length
RTOL :	Wear tolerance for tool radius
DIRECT :	Cutting direction of the tool
PLC :	Additional information for PLC (Module 9093)
TT: L-OFFS:	Tool length offset
TT: R-OFFS:	Tool radius offset
LBREAK:	Breakage tolerance for tool length
RBREAK:	Breakage tolerance for tool radius

The elements in the tables and the sequence in which they appear are defined in machine parameters MP7266.x and MP7264.x



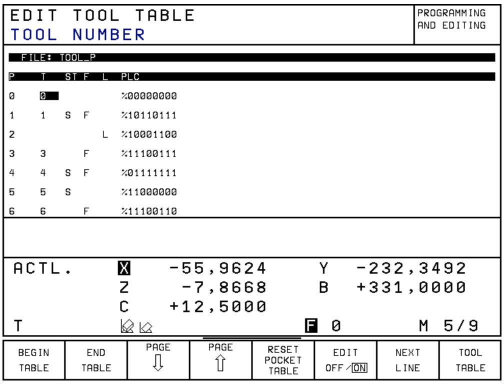
Left side of the tool table



Right side of the tool table

The following fields can be edited in the pocket table:

T	:	Tool number
ST	:	Special tool
F	:	Fixed pocket
L	:	Locked pocket
PLC	:	Additional information for PLC (module 9093)



The tool and pocket tables cannot be edited if they are disabled or protected by MP7224 (see section "Display and operation").

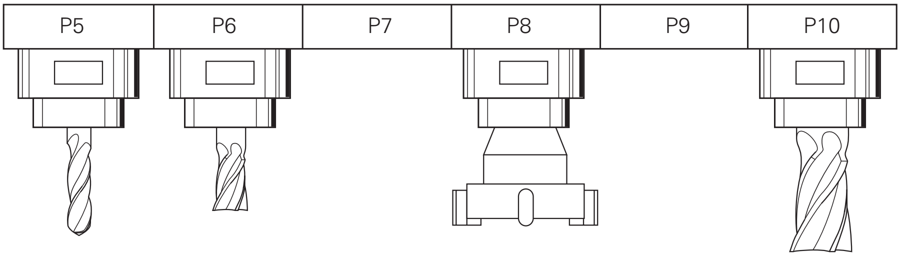
The number of tools in the tool table is defined in machine parameter MP7260. If MP7260 contains a zero value, then the system will run without a tool table ("TOOL.T" does not exist). Tool length and tool radius are programmed in the NC program with TOOL DEF (see User's Manual). Automatic tool management is not possible when operating with no tool table.

The number of pockets in the tool magazine is defined in machine parameter MP7621. No pocket table is generated if MP7621 is defined as zero. The elements displayed in the tables and their sequence are defined in machine parameters MP7266.x and MP7264.x. The PLC can read and overwrite the tool and pocket tables with modules 9092, 9093 and 9094 (see chapter "PLC Programming"). The current tool data appear in the additional status display (see section "Display and Programming").

MP7224.0	Disable file types	
Bit 2	Tool tables 0 = not disabled 1= disabled	
MP7224.1	Protect file types	
Bit 2	Tool tables 0 = not protected 1= protected	
MP7260	Number of tools in tool table Entry: 0 to 254	
MP7261	Number of pockets in pocket table Entry: 0 to 254	
MP7266	Elements of the tool table Entry: 0 to 99 0 = no display 1 to 99 = position in tool table	
MP7266.0	Tool name	(NAME)
MP7266.1	Tool length	(L)
MP7266.2	Tool radius	(R)
MP7266.3	Tool radius 2	(R2)
MP7266.4	Oversize tool length	(DL)
MP7266.5	Oversize tool radius	(DR)
MP7266.6	Oversize tool radius 2	(DR2)
MP7266.7	Tool locked?	(TL)
MP7266.8	Replacement tool	(RT)
MP7266.9	TIME 1	
MP7266.10	TIME 2	
MP7266.11	CURRENT TIME	
MP7266.12	Comment on the tool	(DOC)
MP7266.13	Number of tool cutting edges	
MP7266.14	Wear tolerance for tool length	
MP7266.15	Wear tolerance for tool radius	(RTOL)
MP7266.16	Cutting direction of the tool	(DIRECT)
MP7266.17	PLC status	(PLC)
MP7266.18	Tool length offset	(TT: L-OFFS)
MP7266.19	Tool radius offset	(TT: R-OFFS)
MP7266.20	Breakage tolerance for tool length	(LBREAK)
MP7266.21	Breakage tolerance for tool radius	(RBREAK)
MP7267	Elements of pocket table Entry: 0 to 99 0 = no display 1 to 99 = position in pocket table	
MP7267.0	Tool number	(T)
MP7267.1	Special tool	(ST)
MP7267.2	Fixed pocket	(F)
MP7267.3	Locked pocket	(L)
MP7267.4	PLC-Status	(PLC)

15.1.1 Special Tools

In the pocket table the "ST" field can be used to define tools as special tools. If these are oversize tools that are too big for one pocket in the tool magazine, then pockets must be reserved on either side of that pocket. In the pocket table these pockets are locked by the "L" field.



Variable pocket coding can be restricted by the "F" field (fixed pocket). For special tools, variable pocket coding can also be restricted with marker M2601. If marker M2601 is set then all special tools are returned to their original pocket despite "variable pocket coding" being selected. This function can also be defined for specific normal tools in the "F" field.

Marker	Function	Set	Reset
M2601	Special tool to original pocket despite variable pocket coding	PLC	PLC

In software types 243 05, 259 91, 259 96, 259 97, 243 07 and 243 02 the special tools are identified by "S" in the tool table. The number of reserved pockets has been defined in machine parameter MP7264. MP7264 is no longer available.

MP7264   *Number of reserved pockets next to special tool*
Entry: 0 to 3

15.1.2 Tool life, replacement tool

The operator is able to specify two different tool lives (TIME1 and TIME2) and a replacement tool (RT) for every tool in the tool table.

If on TOOL CALL the current tool life (CUR.TIME) is greater than TIME2, the pocket number or tool number (MP7480) of the replacement tool is automatically downloaded. If TIME2 is greater than zero and no replacement tool is defined, a TOOL CALL for the replacement tool provokes the error message "MAX. TOOL LIFE EXPIRED" when this time elapses.

If the current tool life is greater than TIME1, the NC sets flag M2094. Further action can be defined by the machine tool manufacturer (e.g. output a PLC error message).

With the M function M101, the tool can be replaced automatically after a certain service life (TIME 1 or TIME 2). The replacement tool is not installed exactly when tool life ends but can vary by several NC blocks depending on processor capacity. A T strobe M2046 is sent to the PLC and marker M2404 is set. With automatic replacement tool installation (M101) the same radius must be defined for the replacement tool as for the original tool in standard NC programs (NC blocks with RR, RL or R0).

No radius correction is defined in NC blocks with surface-normal vectors. A delta value (DR, DL) for length and radius can be defined for each tool in the tool table. The TNC uses these delta values in NC programs with surface-normal vectors. If the radius of the replacement tool differs from that of the original tool, then this must be defined in the "DR" field in the tool table. The delta value must always be negative. If a positive delta value is defined then the error message "Tool radius too large" will appear. This error message can be reset by M-function M107. M107 is reset by M108.

MP7680 can define whether the oversize for the tool radius (DR2) is considered when calculating tool length for NC blocks with surface-normal vectors.

The current tool life is only counted from TOOL CALL to TOOL CALL in the automatic modes ("Program run/Single block" and "Program run/Full sequence"). It does not matter whether the spindle is turning or the machine is traversing. The tool life counter does not stop until the program is interrupted with "Internal Stop", M02, M30 or END PGM.

The tool life counter does not run in the manual modes ("Manual Operation", "El. Handwheel" and "Positioning with manual input").

The operator can reset the current tool life by entering zero.

Marker	Function	Set	Reset
M2094	Maximum tool life elapsed (TIME1 in tool table)	NC	NC; PLC

MP7680 Machine parameters with multiple function
Entry: %xxxxxxx

Bit 6 Tool length for blocks with surface normal vector
0 = Without DR2 from the tool table
1 = With DR2 from the tool table

15.2 Controlling the tool changer

Controlling the tool changer, i.e. positioning the changing arm and carousel and the complete tool-change sequence is performed by the PLC. The NC handles tool management, i.e. tool life, tool pocket assignment and evaluation of the TOOL DEF and TOOL CALL blocks.
NC and PLC communicate by markers and words.

When a TOOL CALL BLOCK is executed the tool geometry of the defined tool is taken from the tool table. With Marker M2717 the PLC can activate the tool geometry of the tool defined in W264. M2717 can only be activated together with a strobe or when the axis is stationary (∗ off). With this marker the PLC can ensure that the current tool geometry is active even when the tool changing process is interrupted.

The TOOL DEF block can be used to pre-position the tool changer. After a tool has been changed, the next tool is programmed with TOOL DEF. The PLC evaluates the tool and pocket number and pre-positions the tool changer at the follow-up tool.

Example of NC program:

```
.
.
.
TOOL CALL 1 Z S500
TOOL DEF 2
L Z+20 R0 F500 M03
.
.
.
TOOL CALL 2 Z S1000
TOOL DEF 3
.
.
.
```

Variable and Fixed Pocket Coding

The system can be operated with either variable or fixed tool pocket coding.

Machine parameter MP7480 defines whether the tool number or the pocket number is transferred to the PLC. For variable pocket coding the pocket number must be transferred to the PLC (MP7480 = 3 or 4). With fixed pocket coding it is advisable to work with the tool number (MP7480 = 1 or 2).

When executing a TOOL CALL or TOOL DEF block, depending on the setting of MP7480, the NC transfers either only the tool number or the pocket number and the tool number of the programmed tool to the word W262 or W262 and W264 respectively. The strobe marker M2046 (TOOL CALL) or M2047 (TOOL DEF) is set. The NC does not reset these strobe markers until the PLC sets marker M2483 (TOOL CALL) or M2484 (TOOL DEF) after processing the tool or pocket number. The machining program is resumed when strobe markers M2046/M2047 are reset.

If tool number 0 is executed, then NC sets marker M2400 and this is not reset until there is a TOOL CALL for another tool.

MP7480.0 Output tool or pocket number with TOOL CALL block
Entry: 0 to 6

0 = No output
1 = Tool number output only when tool number changes (W262)
2 = Tool number output with every TOOL CALL block (W262)
3 = Output of pocket number (W262) and tool number (W264) only when tool number changes
4 = Output of pocket number (W262) and tool number (W264) with every TOOL CALL block
5 = Output of pocket number (W262) and tool number (W264)only when tool number changes. Pocket table does not change.
6 = Output of pocket number (W262) and tool number (W264) with every TOOL CALL block. Pocket table does not change.

MP7480.1 Output tool or pocket number with TOOL DEF block
Entry: 0 to 4

0 = No output
1 = Tool number output only when tool number changes (W262)
2 = Tool number output with every TOOL DEF block (W262)
3 = Output of pocket number (W262) and tool number (W264) only when tool number changes
4 = Output of pocket number (W262) and tool number (W264) with every TOOL DEF block

Word	Function
W262	Pocket number if MP7480 = 3 , 4, 5 or 6 Tool number if MP7480 = 1 or 2
W264	Tool number if MP7480 = 3, 4, 5 or 6

Marker	Function	Set	Reset
M2046	Strobe signal T code (P code) with TOOL CALL	NC	NC
M2047	Strobe signal T code (P code) with TOOL DEF	NC	NC
M2483	Feedback T code (P code) with TOOL CALL	PLC	PLC
M2484	Feedback T code (P code) with TOOL DEF	PLC	PLC
M2400	Tool number 0 programmed	NC	NC
M2717	Geometry of the tool from W264	PLC	NC

15.2.1 Output of tool number (fixed pocket coding)

Evaluating the tool number is adequate for fixed tool pocket coding. MP7480 is used to select whether the tool number should be transferred to the PLC with every TOOL CALL (TOOL DEF) block or only when the tool number changes (input values for MP7480 = 2 or 1). With this setting the tool number is transferred to the word W262 when a TOOL CALL or TOOL DEF block is executed. W264 is not used.

If MP7261 > 0, then pocket numbers are displayed in the pocket table. Entering 5 or 6 in MP7480.0 transfers the pocket number to W262 and the tool number to W264. Unlike the setting with variable pocket coding (MP7480 = 3 or 4) the assignment of tool and pocket numbers in the pocket table does not change.

15.2.2 Output of pocket number (variable pocket coding)

With variable pocket coding, the NC must transfer the pocket number of the called tool to the PLC (MP7480 = 3 or 4). In this setting, the pocket number is filed in word W262. In addition to the pocket number, the NC also transfers the current tool number in W264. Variable pocket management (the assignment of tool number to pocket number in the tool table) is handled by the NC. The PLC can use marker M2612 to tell the NC that the pocket number in the tool table should not be updated during "Block scan".

The number of tools with a pocket number is defined in machine parameter MP7261. The input value for MP7261 matches the number of pockets in the tool magazine. This means that more tools can be defined in the tool table than there is room for in the tool magazine [(MP7260) > (MP7261)]. If a tool number is programmed and no pocket is defined for it, pocket number 255 (W262) is transferred on TOOL CALL, and marker M2402 is set.

Only the tool number and the pocket number are transferred when TOOL DEF is programmed. A TOOL DEF for a manual tool has no relevance in the PLC.

A fixed pocket can be defined in the field ' F' of the pocket table. Tools for which a fixed pocket is defined are returned to the same pocket despite variable coding.

Marker	Function	Set	Reset
M2093	Another T code (P code) follows with TOOL CALL 0 = Normal tool follows normal tool (N → N) or manual tool follows manual tool (M → M) or special tool follows special tool (S → S) when M2601 = 0 1 = special tool follows manual tool (M → S) or special tool follows special tool (S → S), when M2601 = 1 or manual tool follows special tool (S → M) or manual tool follows normal tool (N → M) or normal tool follows manual tool (M → N) or normal tool follows special tool (S → N) (see marker M2600)	NC	NC

Marker	Function	Set	Reset
M2600	Transfer sequence of tool numbers or pocket numbers (M2093 = 1) 0 = first number for old tool then number for new tool (single changing arm) 1 = first number of new tool then number of old tool (double changing arm)	PLC	PLC
M2401	Tool programmed with pocket number (active only when MP7480 = 3 or 4 and TOOL CALL)	NC	NC
M2402	Tool programmed without pocket number (active only when MP7480 = 3 or 4 and TOOL CALL)	NC	NC
M2403	Special tool called (TOOL CALL)	NC	NC
M2404	TOOL CALL at end of tool life 0 = programmed TOOL CALL 1 = TOOL CALL at end of tool life	NC	NC
M2601	Special tool to original pocket despite variable pocket coding	PLC	PLC
M2612	Do not update pocket number in pocket table	PLC	PLC

A variety of tool types can be called from the machining program. The following definitions are used in the examples that follow:

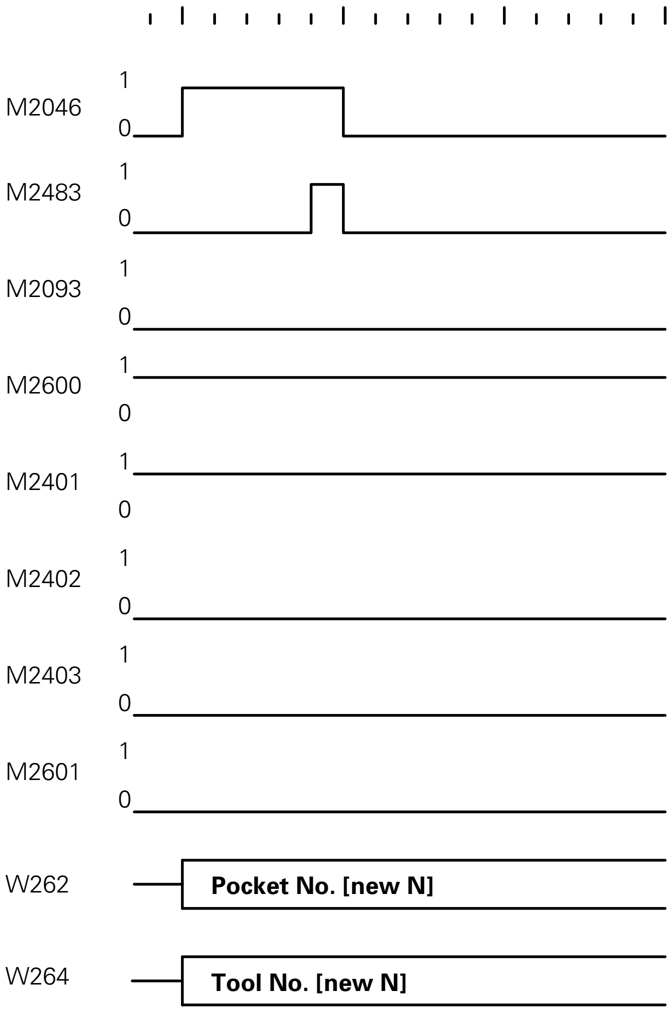
N = Tool for which a pocket number is defined in the tool table. (**N**ormal).
M = Tool for which no pocket number is defined in the tool table. These tools must be changed by hand. (**M**anual).
S = **S**pecial tool (defined in tool table).

Nine different combinations of tool-change sequence are therefore possible. For many tool-change sequences, for the tool magazine to be controlled, two pocket numbers (tool numbers) must be output in succession for one TOOL CALL. This is reported to the PLC by markers M2093 and M2600. The PLC must evaluate and acknowledge both pocket numbers (tool numbers).

The logic diagrams for the nine different tool-change sequences are shown on the following pages (activated by TOOL CALL).

N → N: Normal tool follows Normal tool

The pocket number and tool number of the called tool are transferred.



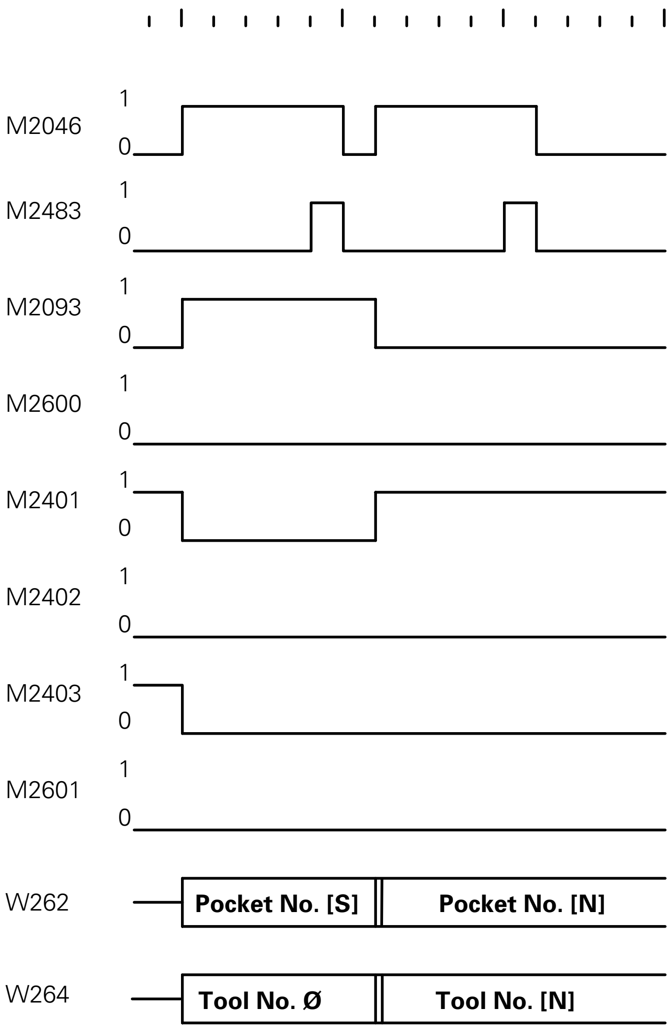
S → N: Normal tool follows Special tool

With this toolchange, two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

With M2600 the PLC can determine the sequence in which the pocket numbers are transferred. The decision will depend on whether a single or double changing arm is in use.

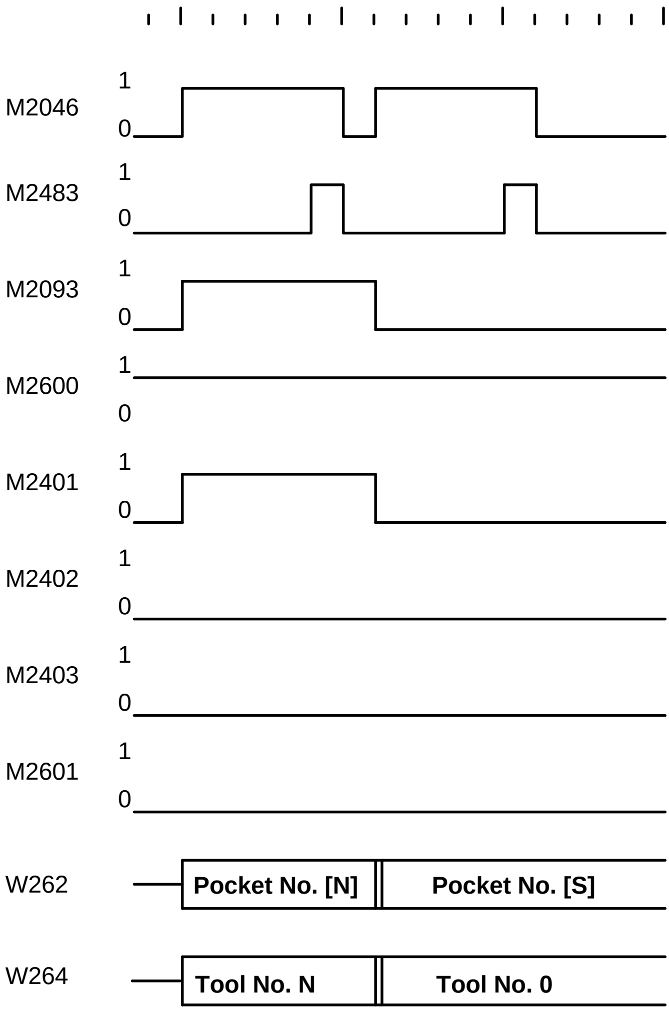
S → N, Single Changing Arm (M2600 = 0):

The pocket number of the old tool and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. After acknowledging with M2483 the pocket number and tool number of the new tool are transferred.



S → N, Double Changing Arm (M2600 = 1):

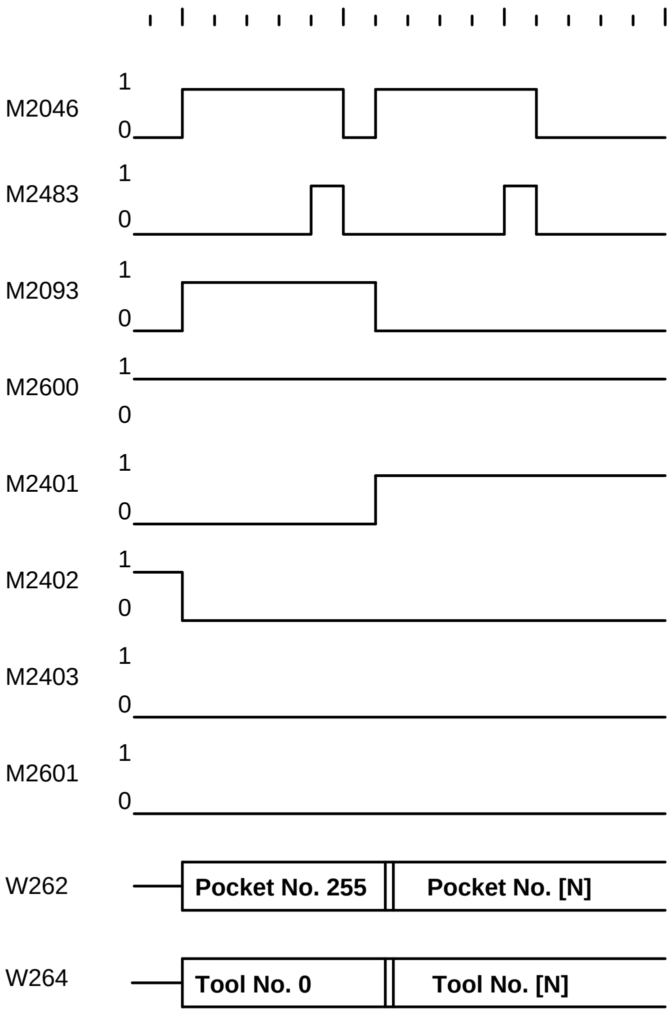
The pocket number and tool number of the new tool are transferred first. After acknowledging with M2483 the pocket number of the old tool and tool number 0 are transferred. Tool number 0 tells the PLC to clear the spindle.



M → N: Normal tool follows Manual tool

With this tool-change sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

Irrespective of flag M2600, pocket number 255 and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. Pocket number 255 means that there is no pocket in the tool magazine for the called tool. After acknowledging with M2483, the pocket number and tool number of the new tool (called tool) are transferred.



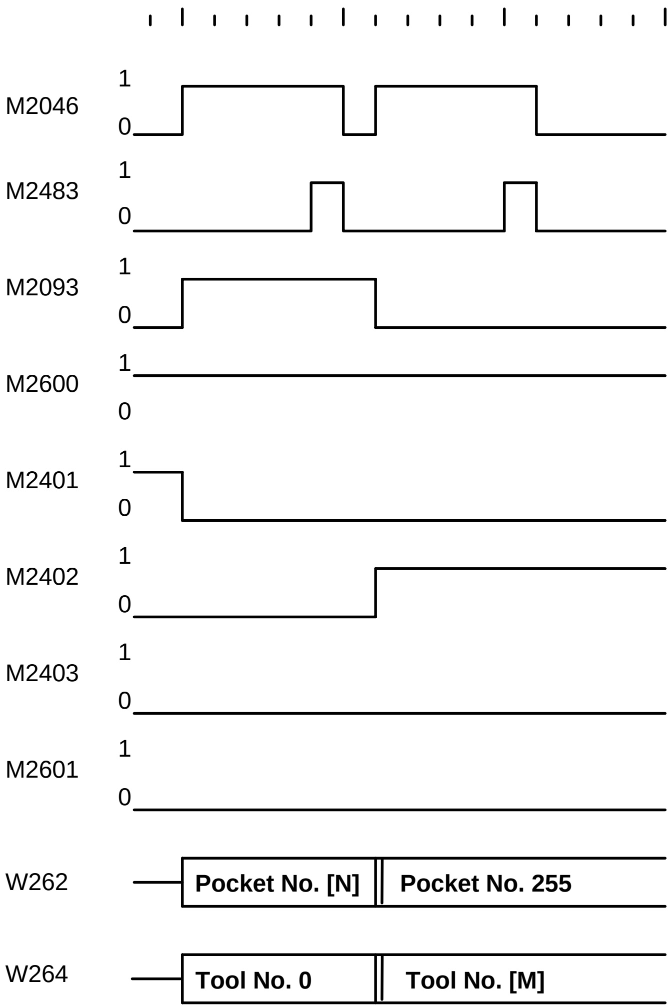
M → M: Manual tool follows Manual tool

The pocket number 255 tells the PLC that there is no pocket in the tool magazine for the called tool.

N → M: Manual tool follows Normal tool

With this tool-change sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

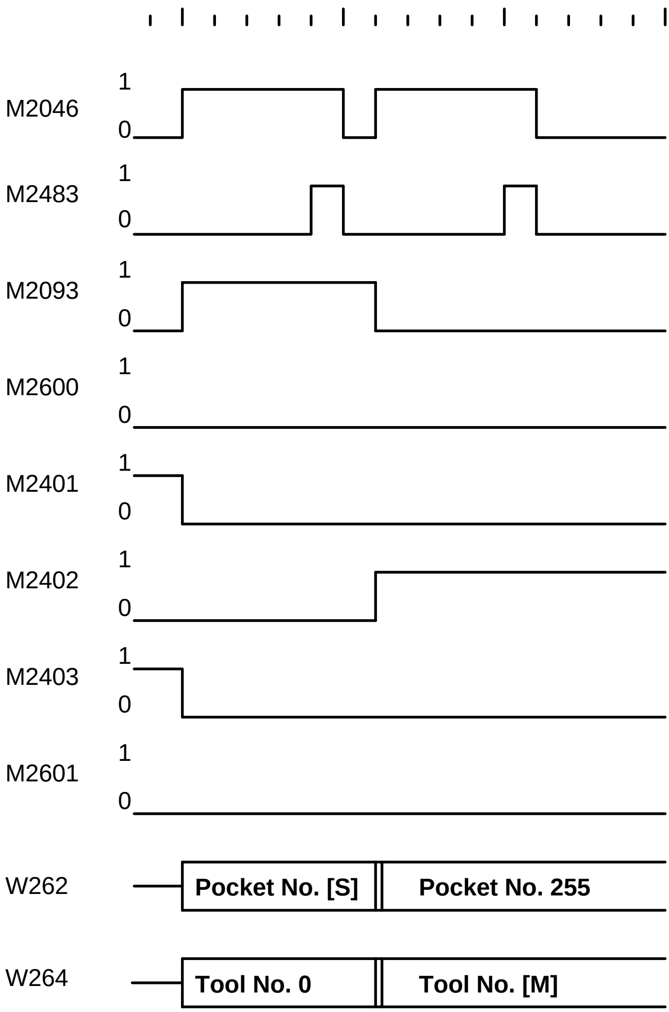
Irrespective of flag M2600, the pocket number of the old tool and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. After acknowledging with M2483, pocket number 255 and the tool number of the called tool are transferred. Pocket number 255 tells the PLC that there is no pocket in the tool magazine for the called tool.



S → M: Manual tool follows Special tool

With this tool-change sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

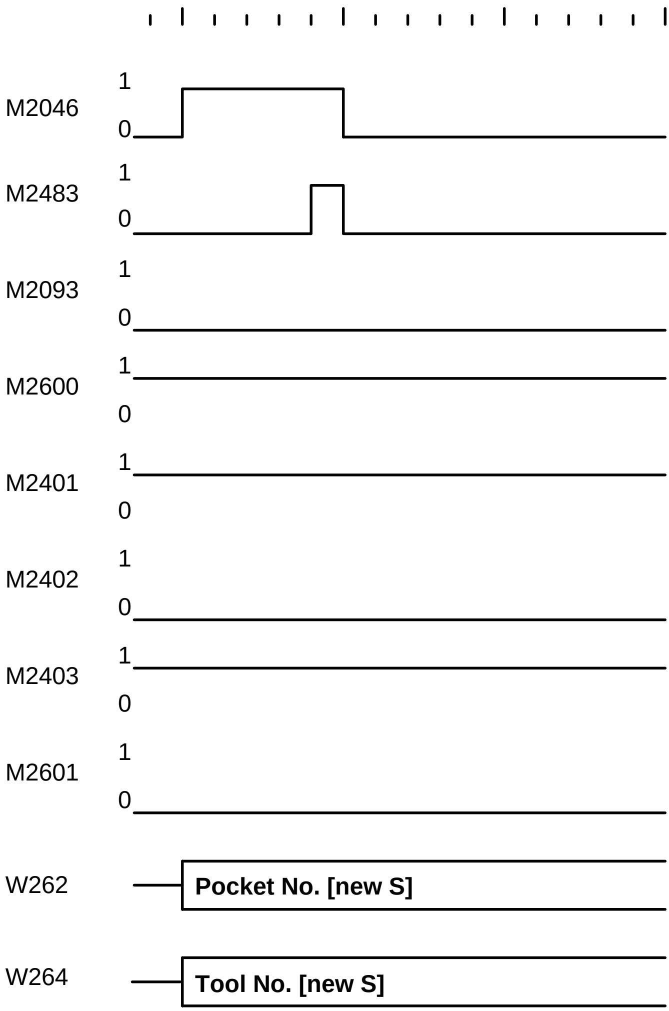
Irrespective of flag M2600, the pocket number of the old tool and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. After acknowledging with M2483, pocket number 255 and the tool number of the called tool are transferred. Pocket number 255 tells the PLC that there is no pocket in the tool magazine for the called tool.



S → S: Special tool follows Special tool

Flag 2601 or field "F" in the pocket table can be used to set whether the special tool is to be returned to its original pocket despite variable pocket coding

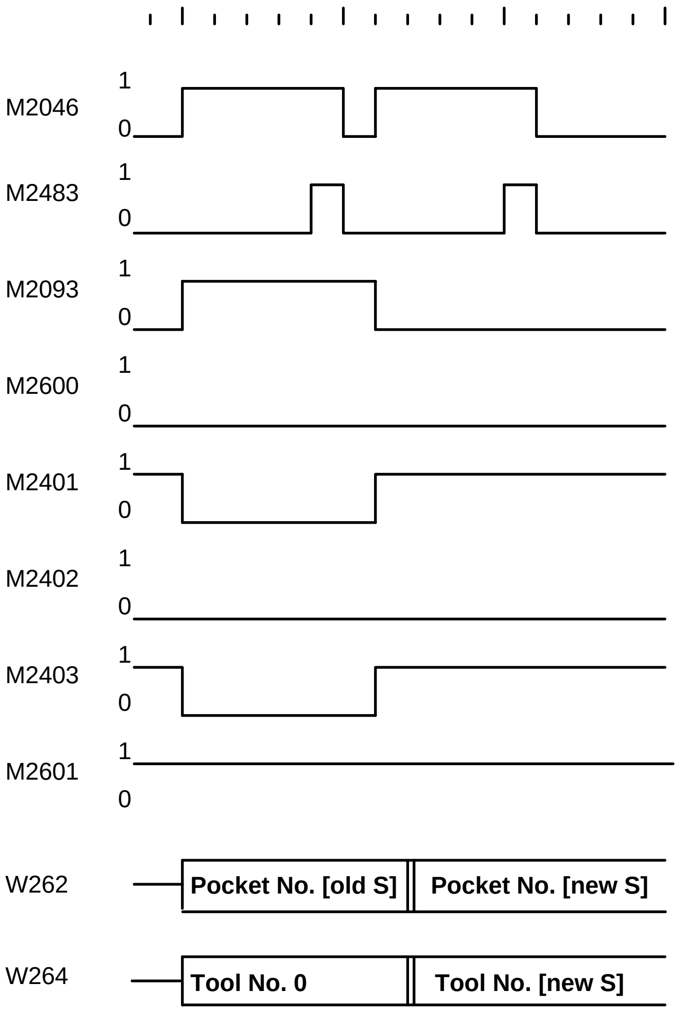
The same logic diagram applies to single and double changing arms (M2600 = 0 and 1) with variable pocket coding for special tools (M2601 = 0).



If the special tool is to return to its original pocket despite variable pocket coding (M2601 = 1), there is a different pocket number transfer sequence for single and double changing arms (M2600).

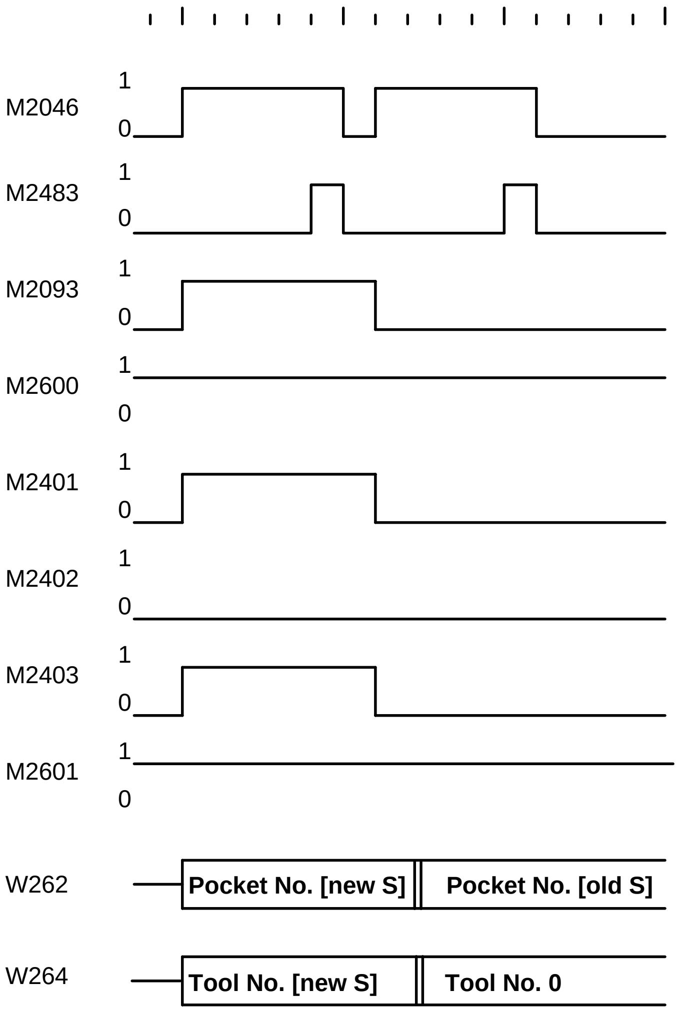
S → S, Single Changing Arm (M2600 = 0)

The pocket number of the old tool and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. After acknowledging with M2483, the pocket number and tool number of the new tool are transferred.



S → S, Double Changing Arm (M2600 = 1)

The pocket number and tool number of the new tool are transferred first. After acknowledging with M2483, the pocket number of the old tool and tool number 0 are transferred. Tool number 0 tells the PLC to clear the spindle.



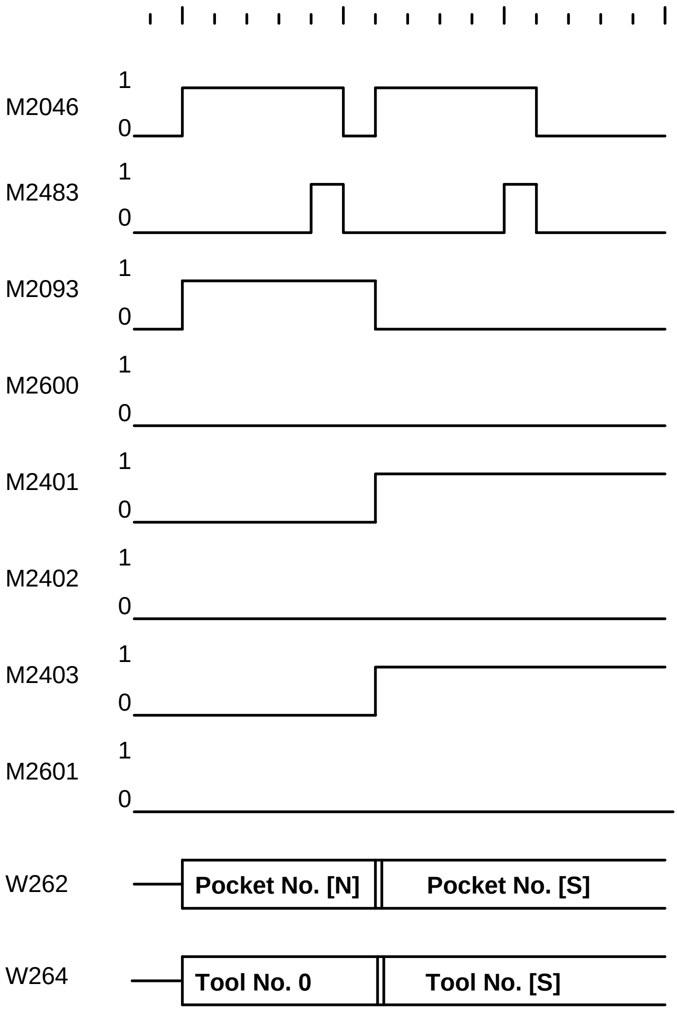
N → S: Special tool follows Normal tool

With this tool-change sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

There is a different pocket number transfer sequence depending on M2600 (single/double changing arm). M2601 is not relevant.

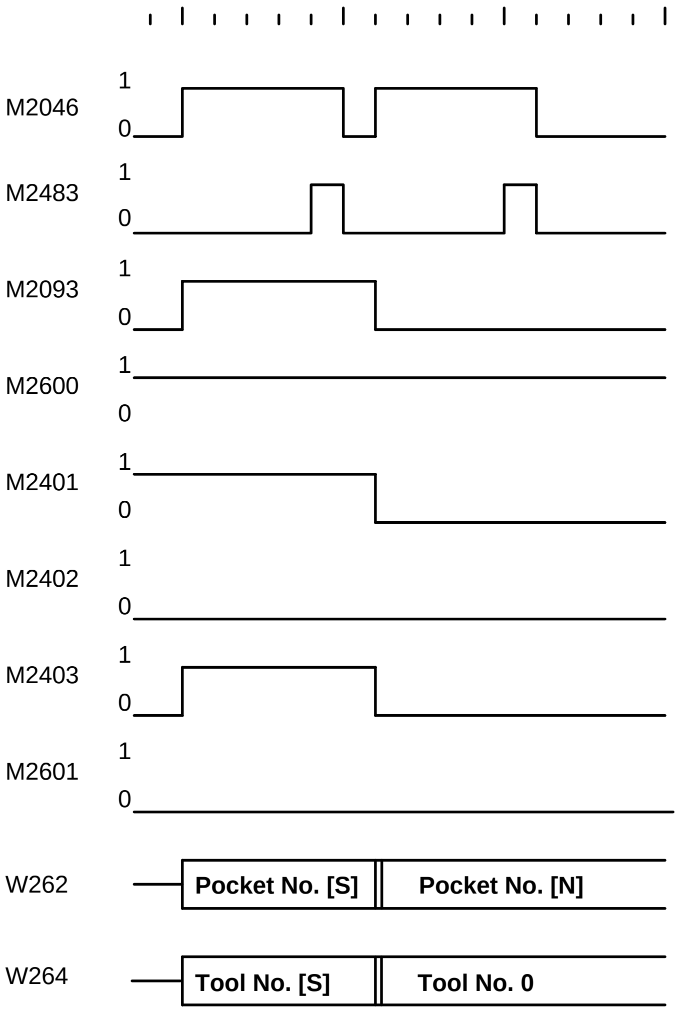
N → S, Single Changing Arm (M2600 = 0)

The pocket number of the old tool and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. After acknowledging with M2483, the pocket number and tool number of the new tool are transferred



N → S, Double Changing Arm (M2600 = 1)

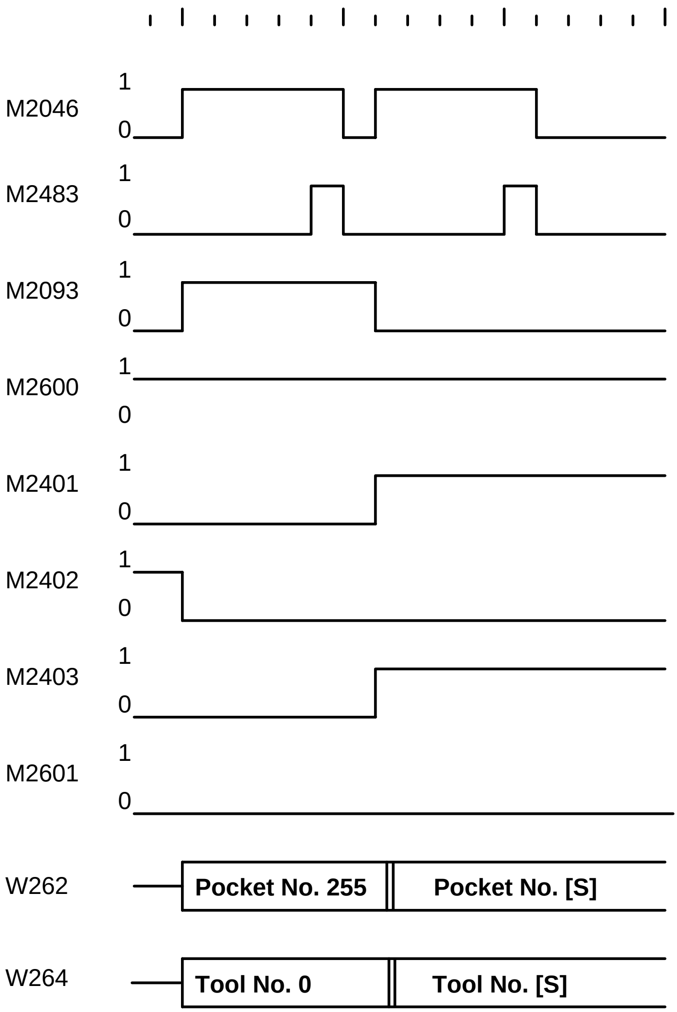
The pocket number and tool number of the new tool are transferred first. After acknowledging with M2483, the pocket number of the old tool and tool number 0 are transferred. Tool number 0 tells the PLC to clear the spindle.



M → S: Special tool follows Manual tool

With this tool-change sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

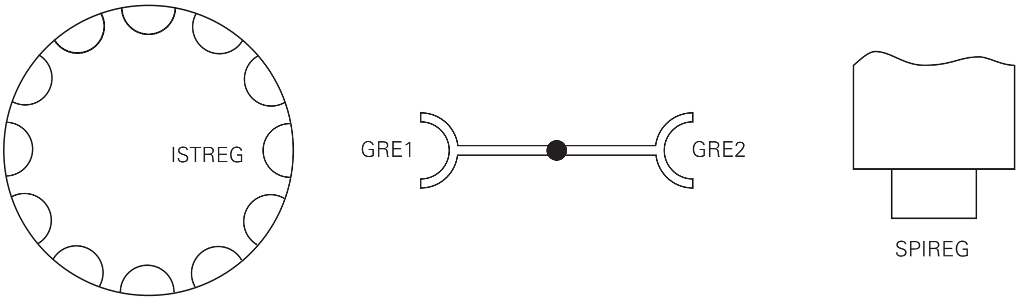
Irrespective of flag M2600 and M2601, pocket number 255 and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. Pocket number 255 means that there is no pocket in the tool magazine for the called tool. After acknowledging with M2483, the pocket number and tool number of the new tool (called tool) are transferred.



15.3 PLC program example

This section describes a tool changer and contains the basic sequence diagrams of the corresponding PLC programs. When creating a program the constraints on the PLC program sequence must be remembered (set buffer markers etc.).

The example describes a tool changer with the following features:
- Up to 254 tools
- Variable pocket coding (MP7480 = 4)
- Special tools are permitted
- Next tool standby with TOOL DEF
- Toolchange with TOOL CALL
- Tools with no pocket number defined in the tool table can be changed by hand
- Double changing arm
- Special tools variable (M2601 = 0)



The following sequence diagram uses variables for greater clarity. In the PLC program these variables are replaced by byte addresses.

ISTREG	=	B10	=	Pocket number at the tool-change position of the tool magazine
GRE1	=	B11	=	Pocket number of tool in changing arm facing tool magazine
GRE2	=	B12	=	Pocket number of tool in changing arm facing spindle
SPIREG	=	B13	=	Pocket number of tool in spindle

Other PLC operands which are used:

Marker	Function	Set	Reset
M2046	Strobe signal T code (P code) with TOOL CALL	NC	NC
M2047	Strobe signal T code (P code) with TOOL DEF	NC	NC
M2093	Another T code (P code) follows with TOOL CALL	NC	NC
M2403	Special tool called (TOOL CALL)	NC	NC
M2483	Feedback T code (P code) with TOOL CALL	PLC	PLC
M2484	Feedback T code (P code) with TOOL DEF	PLC	PLC
M2600	Sequence of tool number or pocket number transfer (M2093 = 1)	PLC	PLC
M2601	Special tool to original pocket despite variable pocket coding	PLC	PLC
W262	Pocket number		
W264	Tool number		

Machine parameters used:

MP7260 = 90	Number of tools in tool table
MP7261 = 12	Number of pockets in tool magazine
MP7264 = 1	Number of pockets vacant beside special tool
MP7480.0 = 4	Output of pocket number and tool number with every TOOL CALL block
MP7480.1 = 4	Output of pocket number and tool number with every TOOL DEF block

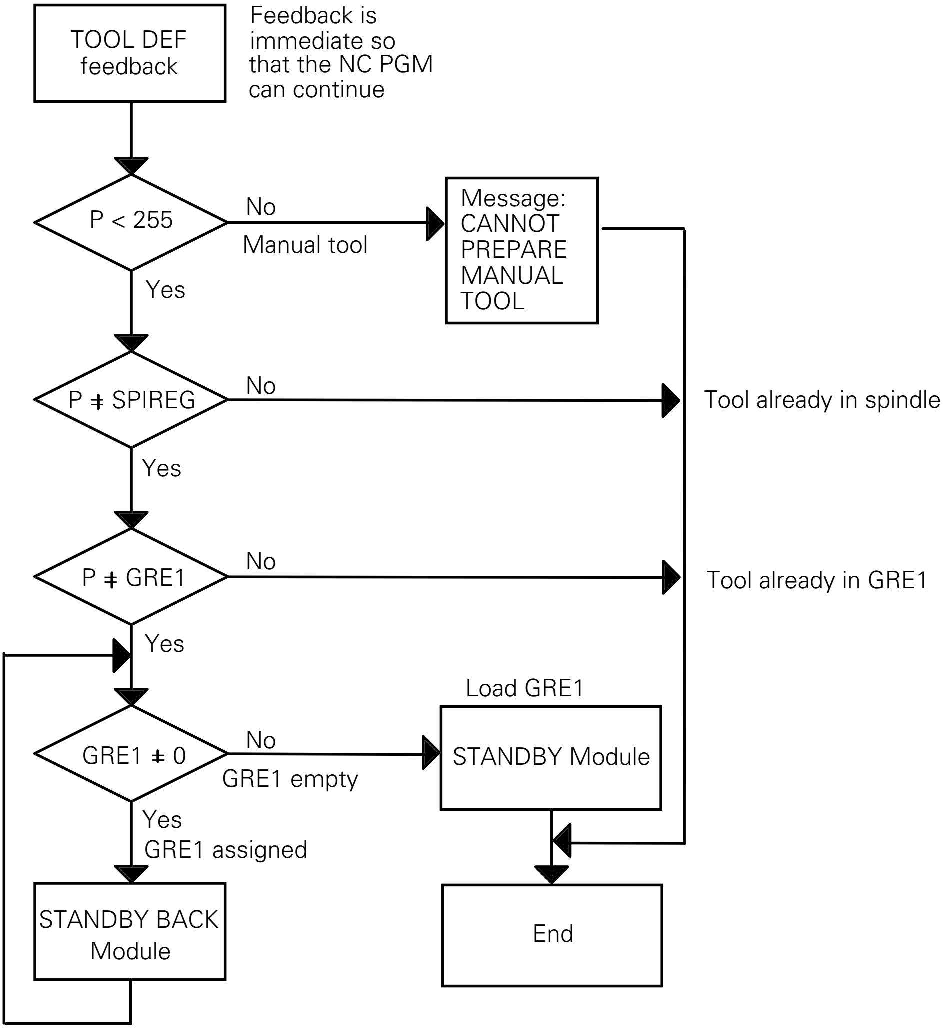
The sequence diagram for this tool changer is subdivided into modules.

List of modules (subroutines):

-	TOOL DEF	Search tool and load in GRE1
-	TOOL CALL	Automatic tool-change
-	STANDBY	Search tool and load in GRE1
-	STANDBY BACK	Return tool from GRE1 to magazine
-	MANUAL TOOL IN	Manual tool follows Normal or Special tool
-	MANUAL TOOL OUT	Normal or Special tool follows Manual tool
-	MANUAL OUT/IN	Manual tool follows Manual tool
-	CHANGE	Take old tool out and put new tool in
-	COMPUTE SHORTEST PATH	
-	COMPARE P-CODE WITH ISTREG	
-	COMPARE GRE1 WITH ISTREG	

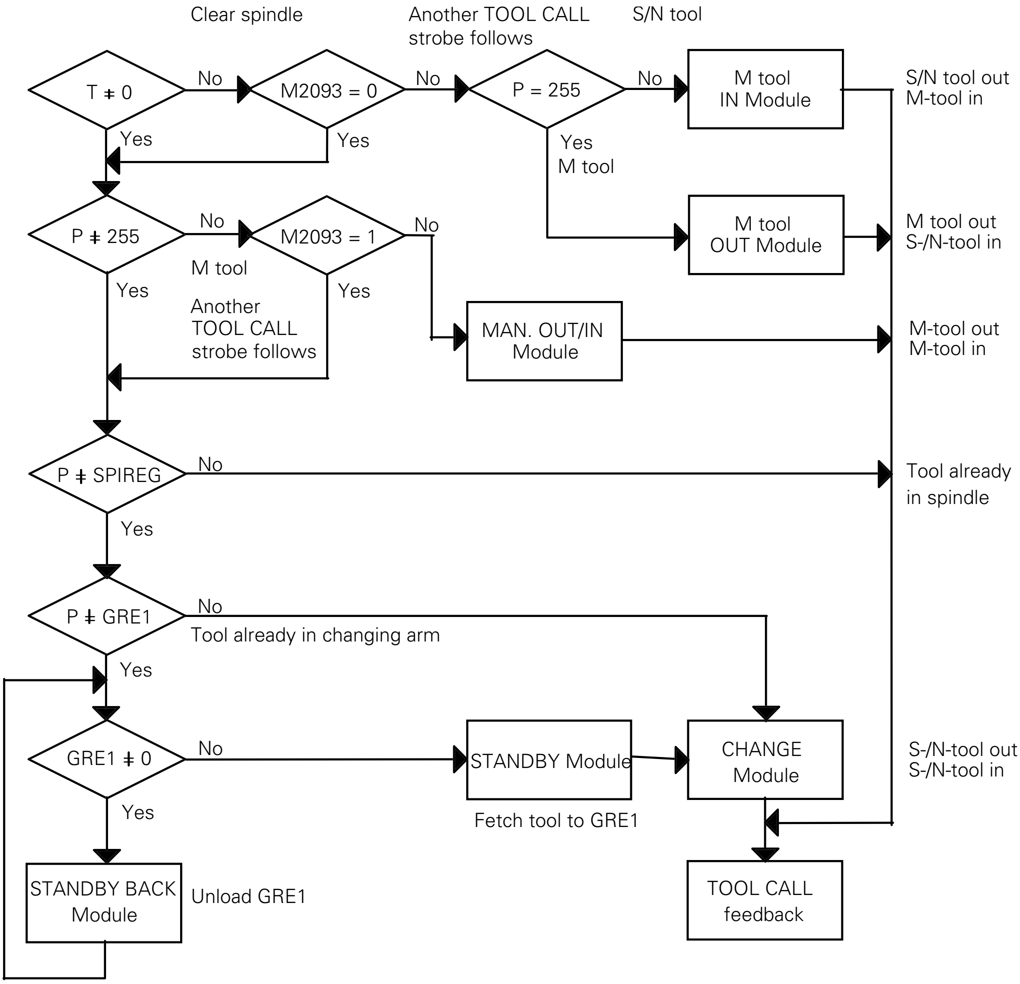
15.3.1 Program module TOOL DEF

Search tool and load in GRE1



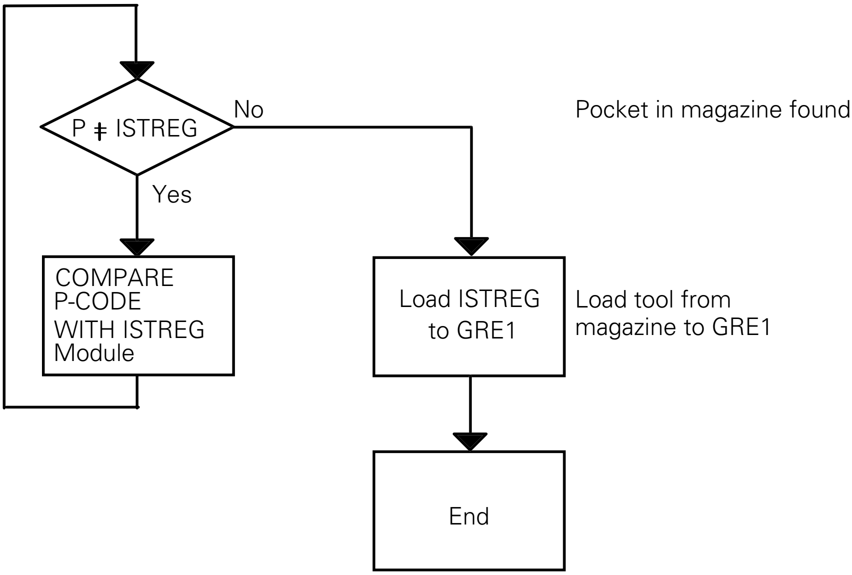
15.3.2 Program module TOOL CALL

Automatic Toolchange (Main Program)



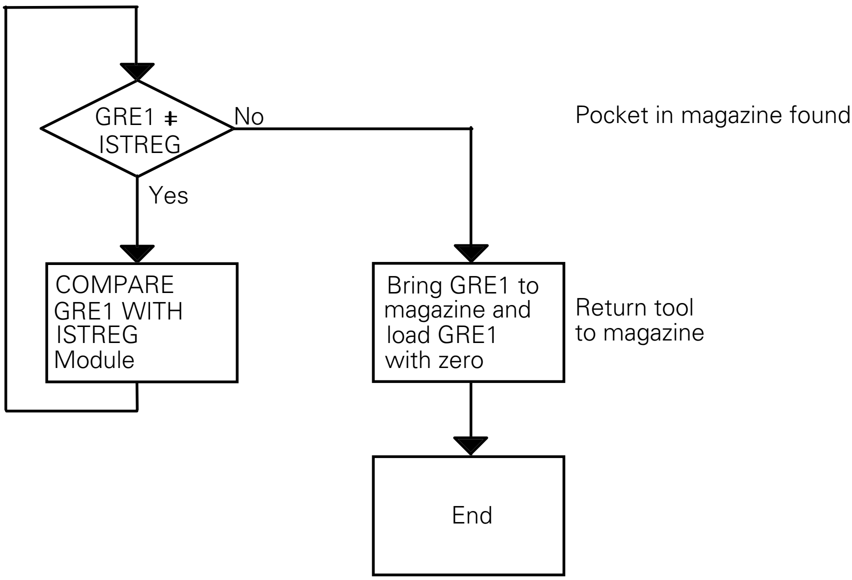
15.3.3 Program module STANDBY

Search for tool and load in GRE1



15.3.4 Program module STANDBY BACK

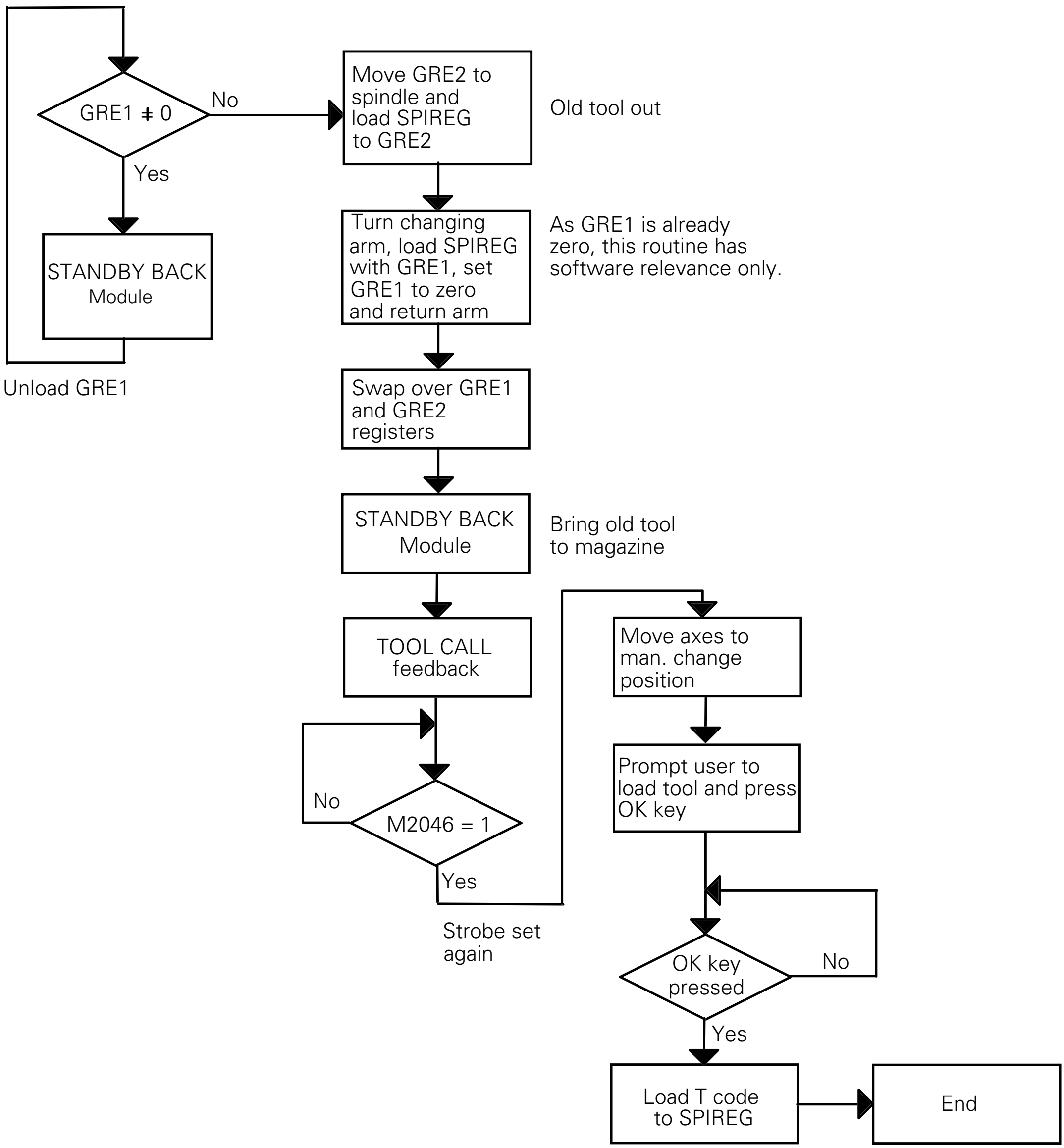
Return tool from GRE1 to magazine



15.3.5 Program module MANUAL TOOL IN

N -> M or S -> M: Manual tool follows Normal or Special tool.

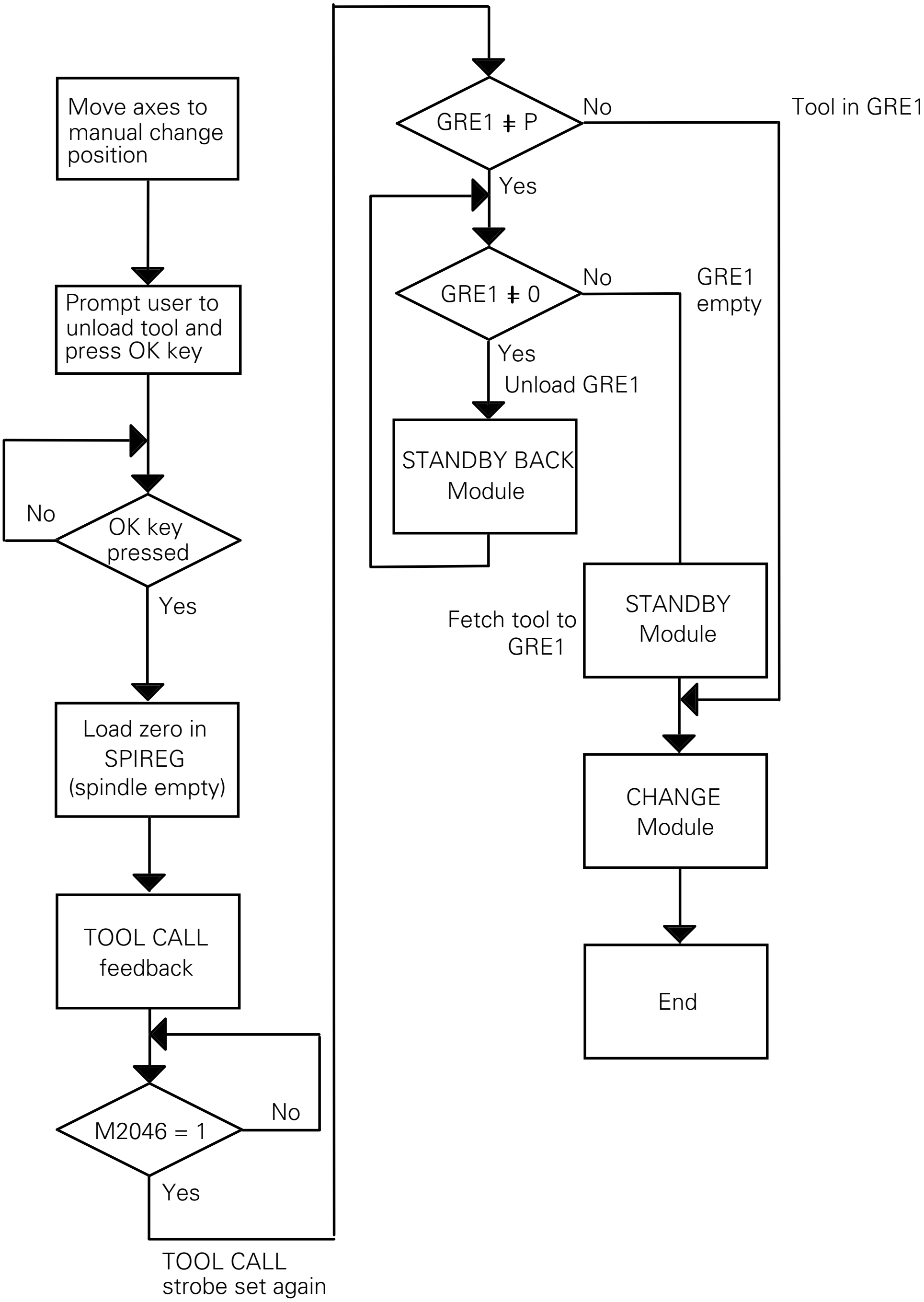
The old tool is returned to the tool magazine and the operator is prompted to load a manual tool (not in tool magazine).



15.3.6 Program module MANUAL TOOL OUT

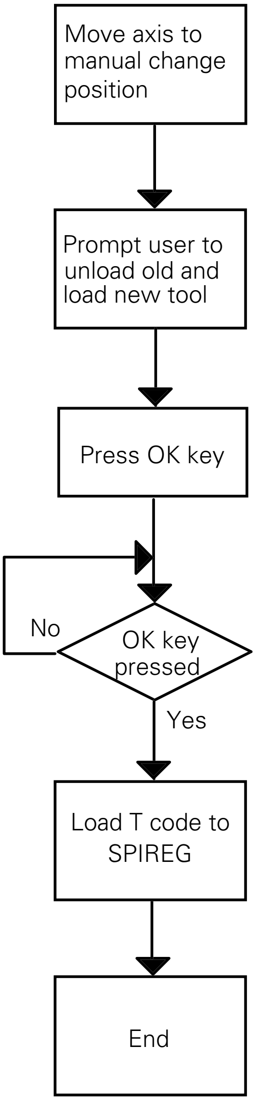
M -> N or M -> S: Normal or Special tool follows Manual tool

The operator is prompted to unload the spindle manually as there is no pocket for the current tool in the magazine. The called tool is loaded automatically.



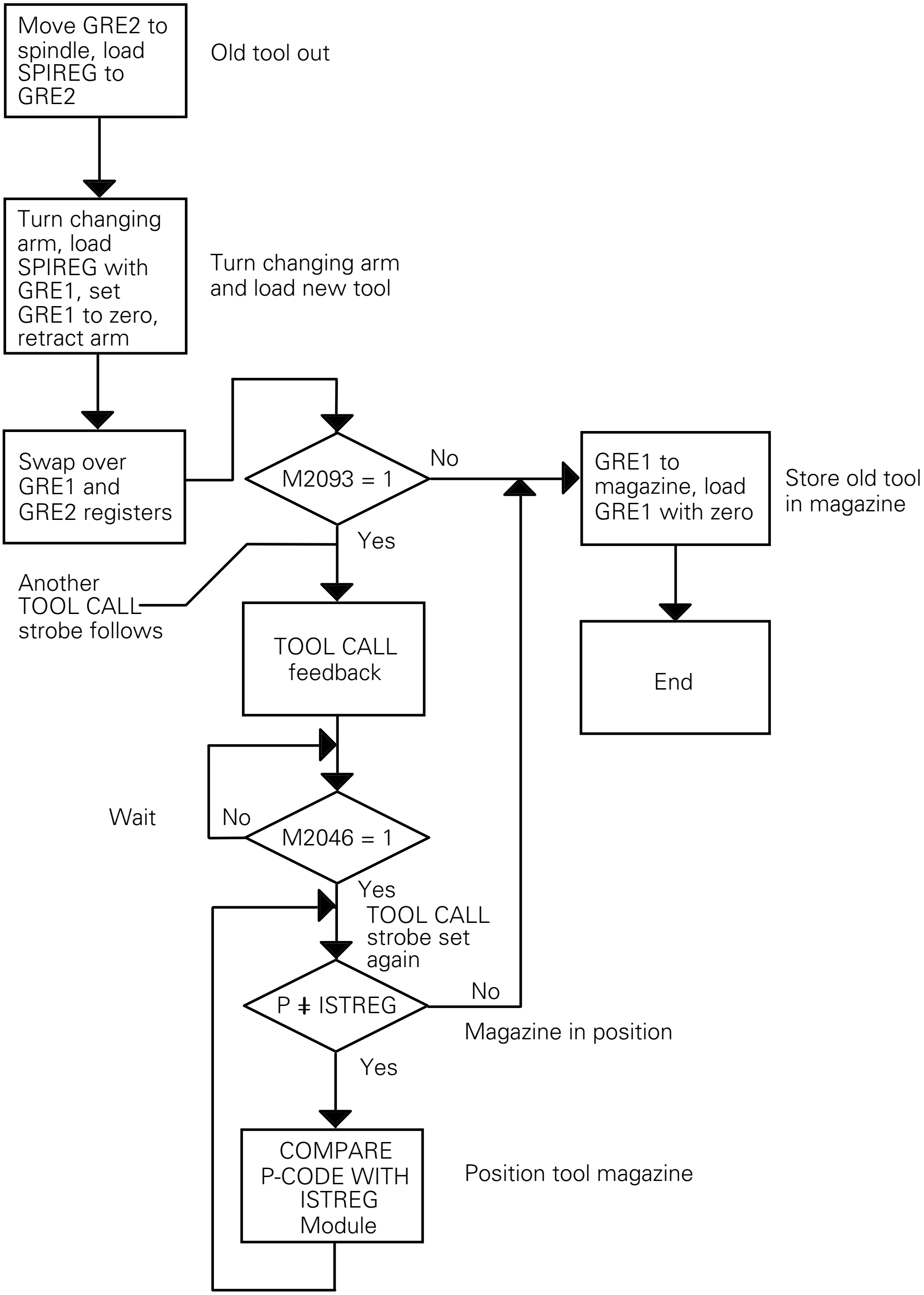
15.3.7 Program module MANUAL TOOL IN/OUT

M -> M: Manual tool follows Manual tool.
The operator is prompted to unload the spindle and load the new tool manually as there is no pocket for these tools in the magazine.



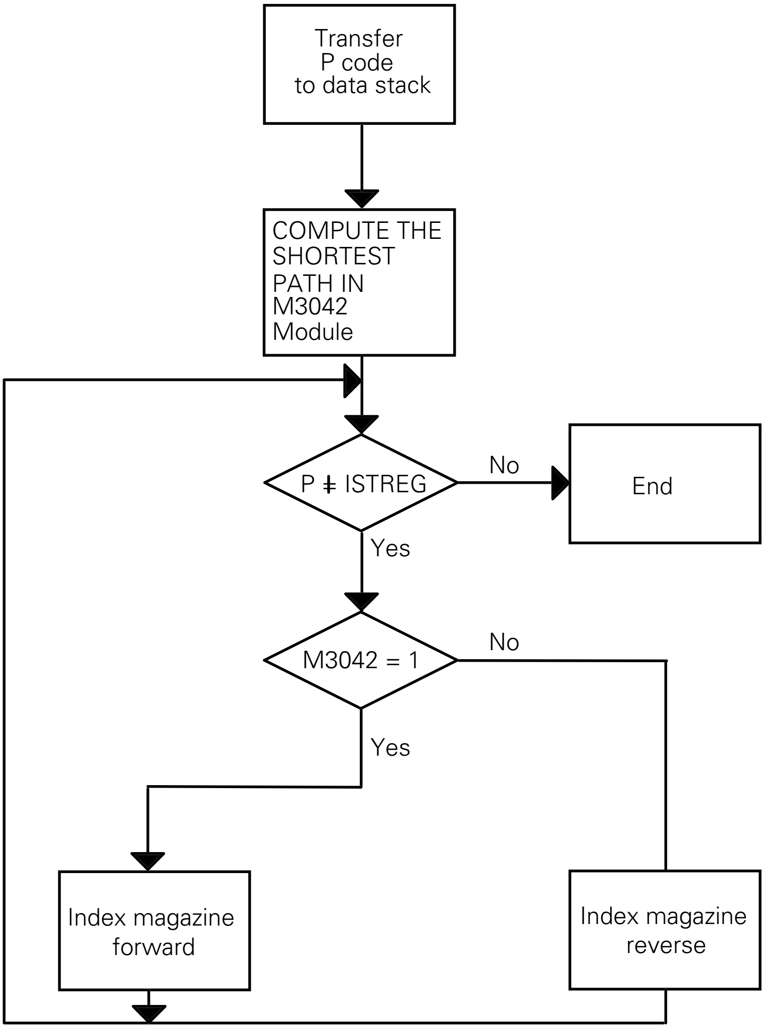
15.3.8 Program module CHANGE

The spindle is unloaded and the new tool is loaded automatically. This module defines whether the tool is to be returned to its old pocket in the tool magazine or not (e.g. Special tool).



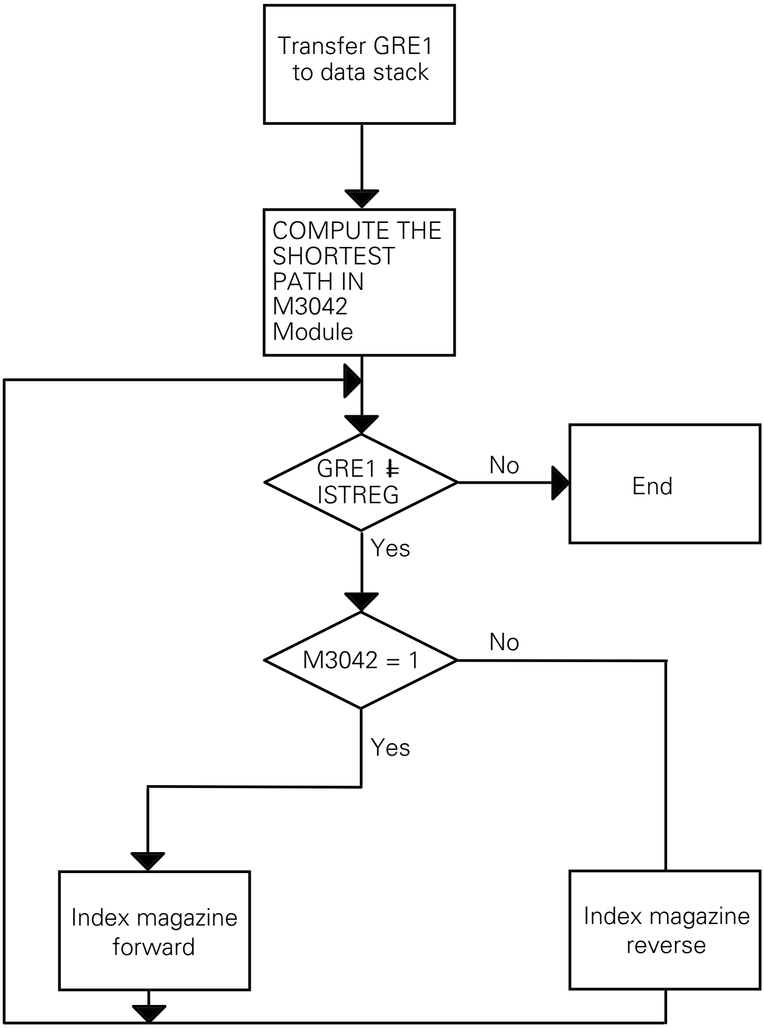
15.3.9 Program module COMPARE P-CODE WITH ISTREG

The tool magazine is positioned at the search pocket number by the shortest path.



15.3.10 Program module COMPARE GRE1 WITH ISTREG

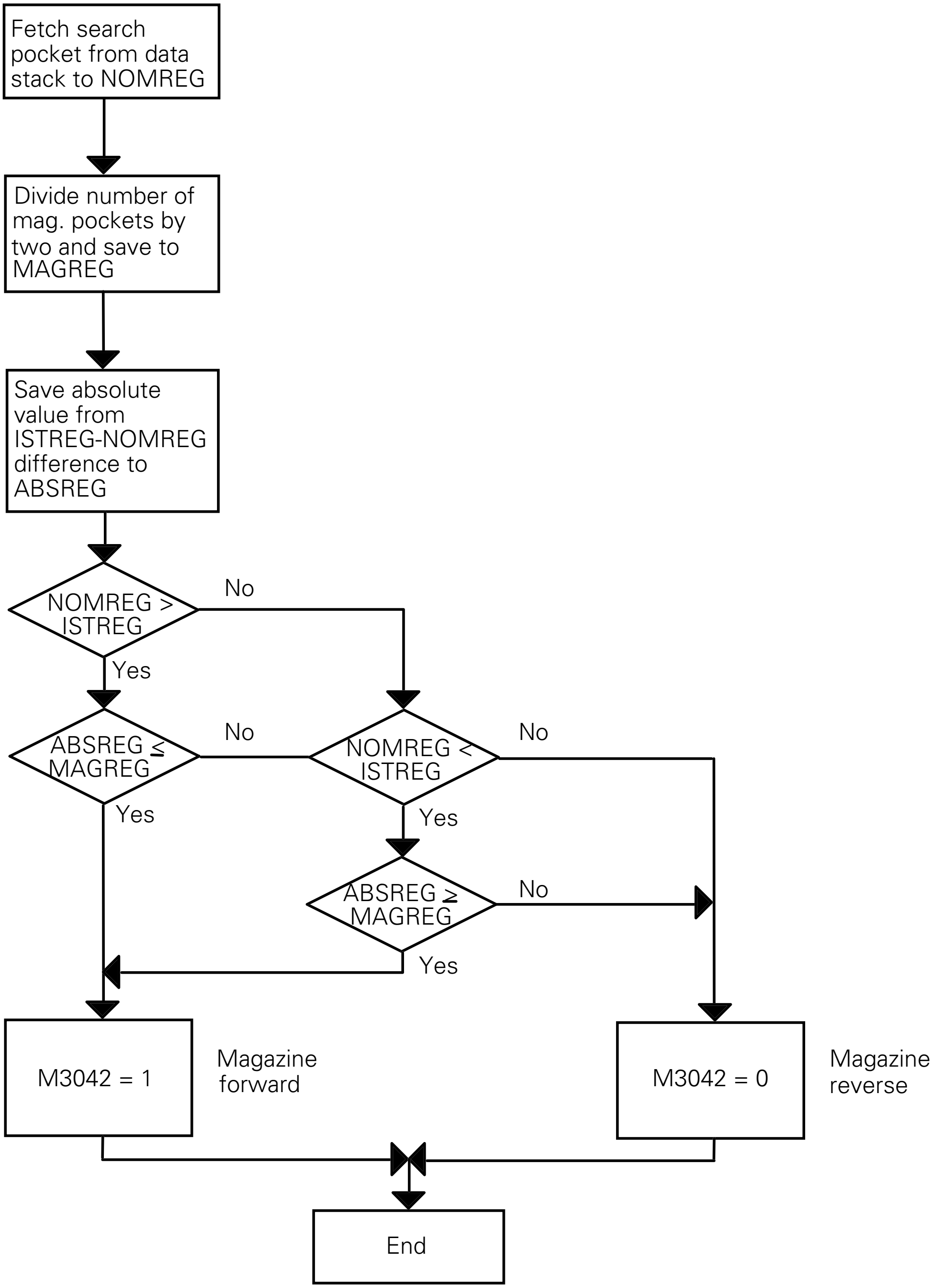
The tool magazine is positioned at the pocket number of the tool in GRE1 by the shortest path.



15.3.11 Program module COMPUTE SHORTEST PATH

The program determines the direction in which the tool magazine must move in order to reach the search pocket number by the shortest path. The path is filed in marker M3042:

M3042 = 0: reverse
M3042 = 1: forward



16 Commissioning and start-up procedure

This section describes the commissioning procedure for the controls step-by-step. The precise descriptions of the functions may be found by referring to the appropriate sections.

16.1 Code numbers for commissioning

Certain operating conditions and functions for the commissioning procedure must be selected by code numbers (see section "Display and operation").

Code number	Function
95 148	List of machine parameters (see "Machine parameters")
807 667	PLC-operation (see "PLC-programming")
531 210	Cancel markers M1000 to M2000 and B0 to B127
75 368	Automatic offset adjustment (see section "Servo positioning")
105 296	Compensation table (see section "Non-linear axis error correction")
688 379	Oscilloscope

16.2 Preparation of the machine

The machine is prepared without a control being connected.

Follow the sequence below to commission the drive amplifiers:

– Clamp the nominal value lead to 0 V:
  Disconnect and short together the nominal value inputs of the drive amplifiers.
  0 V must be applied to the input.

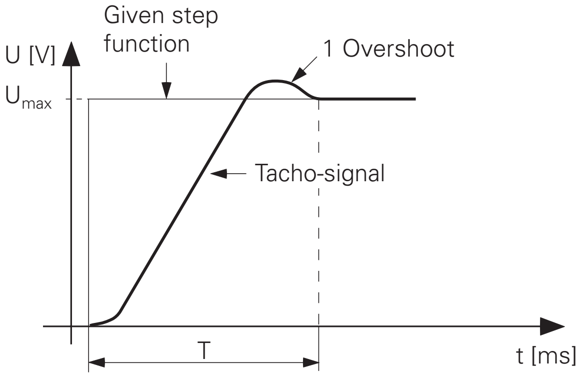
– Connect the servo-enable input:
  Connect the servo-enable input to 24 V, thus activating the servo.

– Connect the power supply to the drive amplifiers.

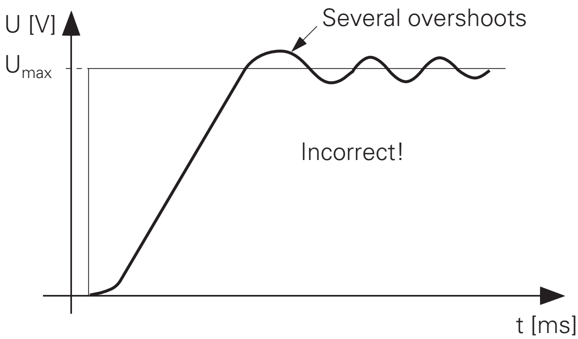
– Rough offset adjustment:
  If the axis moves even though the nominal value input is clamped to 0 V, the offset potentiometer must be adjusted such that the axis comes to a stop. After the offset adjustment, remove the short-circuit bridge on the nominal value input.

– Rough velocity adjustment:
  Connect battery supply to the nominal value input. Adjust the battery supply to 9 V and adjust the drive motor with the tacho-potentiometer to the desired speed (which equals the maximum rapid traverse rate). The rated speed can be read from the drive motor with a tachometer.

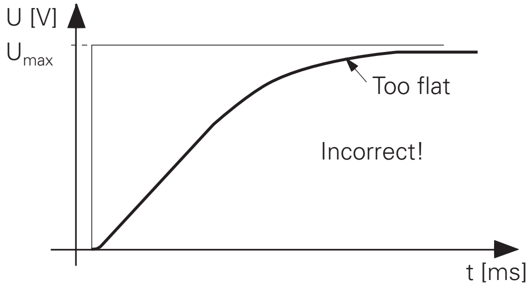
– Tuning of the drive amplifier:
  As far as the control is concerned, the actual servo-loop consists of the drive amplifier, motor and axis slide (see section "Servo positioning"). The servo-loop must be tuned before the position control loop in the control can be optimized.
  To achieve this, a battery supply is used to apply a (9 V) step function to the nominal value input of the drive amplifier. The step response can be recorded by an oscilloscope via the tachometer signal. The axis should be loaded with the permissible workpiece weight during the acquisition of the step response. The subordinate control loop (current control, spindle speed control) must be so optimized that the step response shows an overshoot. The following picture shows the ideal response of the tacho-signal.



The following pictures show **incorrectly adjusted** tacho-signals:



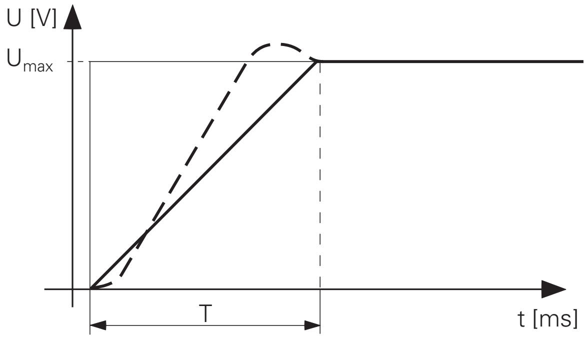
The P-component of the subordinate control loop is too high, or the I-component too low.



The P-component of the subordinate control loop is too low, or the I-component too high..

Calculating acceleration

The maximum acceleration time T can be deduced from the step response.



To calculate the acceleration increase T by 10%.

From this it follows that:

$$a = \frac{v_{max}}{T \times 1.1}$$

v_{max} = velocity at U_{max}

a = acceleration

The step response must be investigated for all axes.

The acquired acceleration values are entered in MP1060.0 to MP1060.4.

After adjustment, switch off the supply voltage to the drive amplifiers.

This concludes the preparation of the drive amplifiers.

16.3 Commissioning the control

The machine must be prepared in accordance with section 16.2 before the machine parameters for commissioning can be optimized.

Before connecting the control, the NC- and PLC- supply voltages and the ground connections should be tested once more (see "Mounting and Electrical Installation ").

16.3.1 Entry of the provisional and pre-defined machine parameters

After switching on the machine for the first time, the control will first of all display the message "Operating parameters erased", which means that the machine parameters still have to be entered (see "Machine parameters").

Most machine parameters can be pre-defined and entered according to the machine parameter list and the functional descriptions. The parameters which affect the control loop must be assigned provisional values (see section 16.3.6).

In order to avoid lengthy delays on restarting during the commissioning of the control, the memory test can be suppressed by MP7690 (see section "Display and operation").

16.3.2 Entry of the PLC-program

A complete PLC-program for all machine functions must be created for commissioning and stored either in EPROM or RAM (see "PLC-programming").

Machine parameter MP4010 (see section "Display and operation") selects whether the processing uses a PLC-program from EPROM or from RAM.

The PLC-program in RAM is used for commissioning purposes. It only makes sense to create an EPROM when all functions are operating without error.
In case of doubt, please contact HEIDENHAIN customer service.

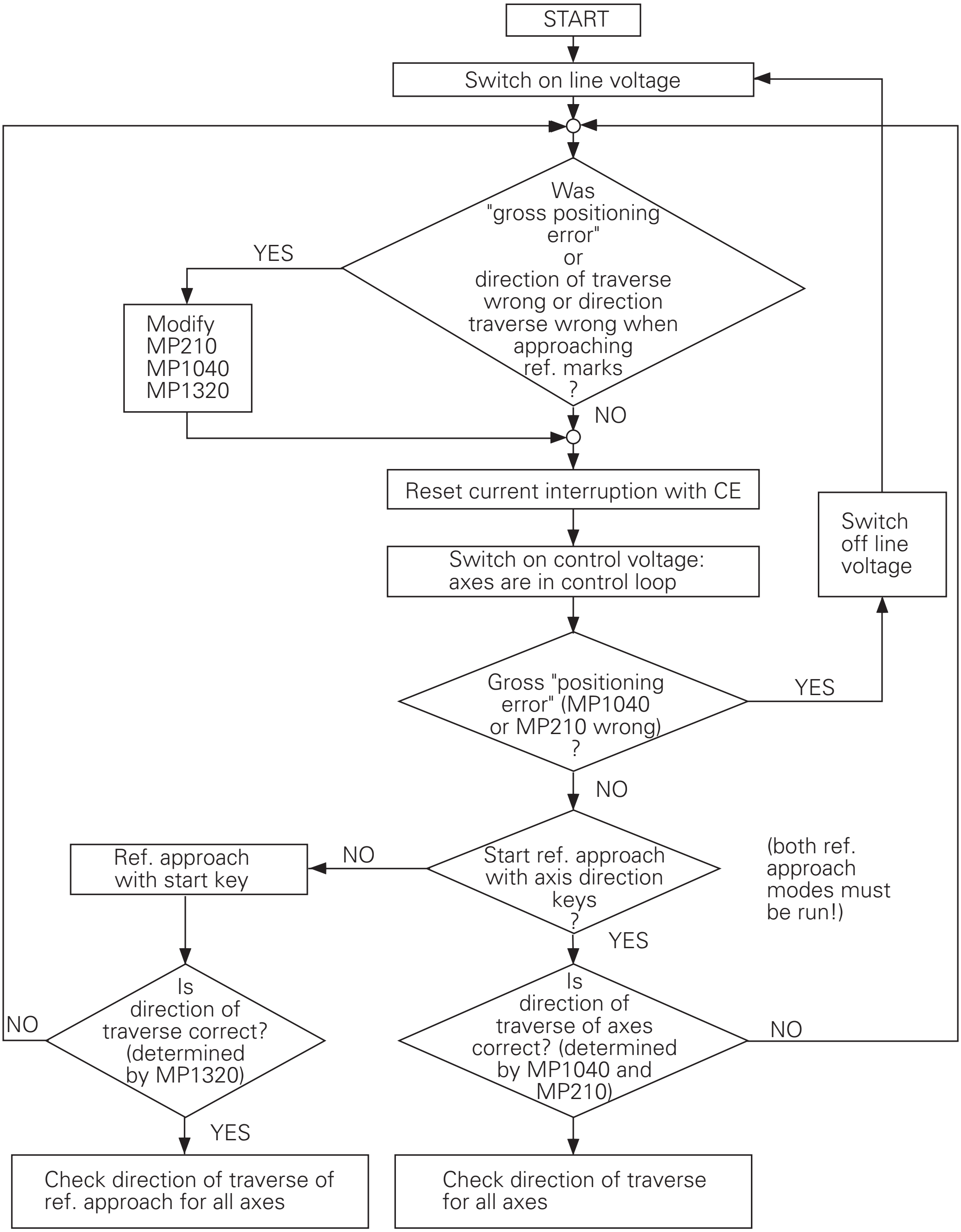
16.3.3 Testing the EMERGENCY STOP routine

Since the EMERGENCY STOP-circuit is very important for the machine it is vital that it is tested!

– Test the Function "Control operational" according to the section "EMERGENCY STOP routine".
– Test the EMERGENCY STOP-circuit by pressing the EMERGENCY STOP keys and by traversing past the EMERGENCY STOP limit switches.

16.3.4 Testing the direction of traverse

Test the controls for the direction of traverse according to the following diagram (effective machine parameters: MP210 count direction of the measuring system signals, MP1040 polarity of the nominal value voltage, MP1320 direction of traverse on passing over the reference marks).



This flow diagram must be worked through for every axis!

16.3.5 Fixing the software limit switch ranges

Determine the software limit switch ranges (see section "Machine axes") as follows:

In "Manual operation" mode select the REF-display with the MOD key.

↓

Position-displays show the distance to the machine datum .

↓

Move all axes with the aid of the direction keys or the handwheel in positive and negative direction to just before the EMERGENCY STOP limit switches. Note the values in the positional-displays and their sign.

↓

Enter the noted values in MP910 or MP920.

↓

Select actual-value display with MOD key.

16.3.6 Optimizing the control with lag (MP1390 = 1)

The following provisional values can be entered for the machine parameters which determine the control characteristics:

Machine parameter	Function	provisional entry value
MP1050	Analogue voltage for rapid traverse	9 V
MP1060	Acceleration	As measured on the machine (see "Preparation of the machine")
MP1810	k_V factor	1
MP1820	Multiplication factor	1
MP1830	Kink point	$\text{Entry value} = \frac{\text{max. mach. speed} \cdot 100\ \%}{\text{Rapid traverse}}$

These values can usually be further optimized.

k_V factor

Adjust the k_V factor (MP1810) so that the voltage characteristic is as described in the section "Servo positioning". If a different k_V-factor is required for rapid traverse to that for the machining feed-rate, it must be optimized separately.

Optimize k_V factor for the machining feed-rate (X-axis)

Connect storage oscilloscope to tachometer of the servo amplifier of the X axis

↓

Enter following program in "PROGRAMMING AND EDITING" operating mode:

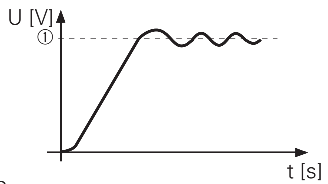
```
LBL 1
X 100¹⁾ R0 F²⁾...
X   0   R0 F²⁾...
CALL LBL 1 REP 100/100
```

↓

Press external START button in "PROGRAM RUN/FULL SEQUENCE" operating mode, machine runs.
Caution: Set feed-rate override to 100 %.

↓

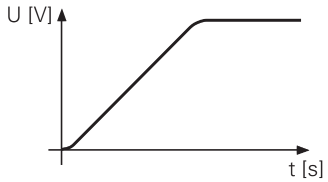
MP1810: Increase entry value until control loop oscillates or overshoot occurs after the acceleration ramp.



① Machining feed-rate

↓

MP1810 (k_V-factor): Reduce entry value until no oscillations can be detected.



↓

Repeat trimming procedure for axes Y, Z, IV and V.

[1] Program the traverse paths for the axis concerned as large as possible.
[2] Enter the max. feed-rate for machining.
For axes which are mutually interpolated the k_v factor must be the same.
In this case the worst axis determines the entry value.

Kink point

Check multiplication factor for the kv (MP1820 0 1) and kink point

$$\text{(MP1830} = \frac{\text{Max. machining feed-rate x 100\%}}{\text{Rapid traverse}}$$

↓

Connect storage oscilloscope to tachometer of the servo amplifier of the X axis.

↓

Enter following program in "PROGRAMMING AND EDITING" operating mode:

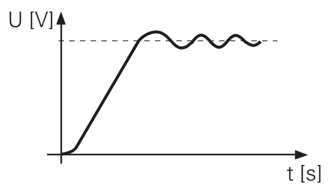
```
LBL 1
X 100¹⁾ R0 F MAX
X     0  R0 F MAX
CALL LBL 1 REP 100/100
```

↓

Press external START button in "PROGRAM RUN/FULL SEQUENCE" operating mode, machine runs.

↓

Does tacho signal overshoot?



YES → Reduce MP1820 multiplication factor progressively until overshoot disappears.

NO → Kink point is not required.

↓

Repeat trimming procedure for axes Y, Z, IV and V.

[1] Program the traverse paths for the axis concerned as large as possible.

16.3.7 Optimizing the control with feedforward control (MP1390 = 0)

The following provisional values can be entered for the machine parameters which determine the control characteristics:

Machine parameter	Function	Provisional entry value
MP1050	Analogue voltage for rapid traverse	9 V
MP1060	Acceleration	As measured on the machine (see "Preparation of the machine")
MP1510	k_V factor	1
MP1520	Position approach	0.5 [m/min]

These values can usually be further optimized.

Optimize k_V factor

Connect storage oscilloscope to tachometer of the servo-amplifier of the X axis.

↓

Enter following program in "PROGRAMMING AND EDITING" operating mode:

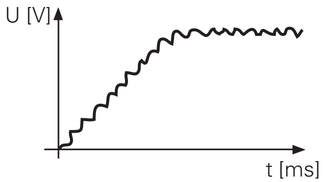
```
LBL 1
X 100 1) R0 F MAX
X     0   R0 F MAX
CALL LBL 1 REP 100/100
```

↓

Press machine START button in the "PROGRAM RUN/SINGLE BLOCK" operating mode. Machine runs.

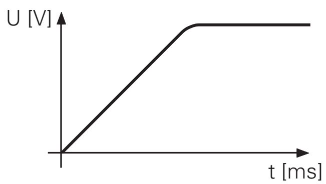
↓

k_V factor MP1510: Increase entry value until position loop oscillates.



↓

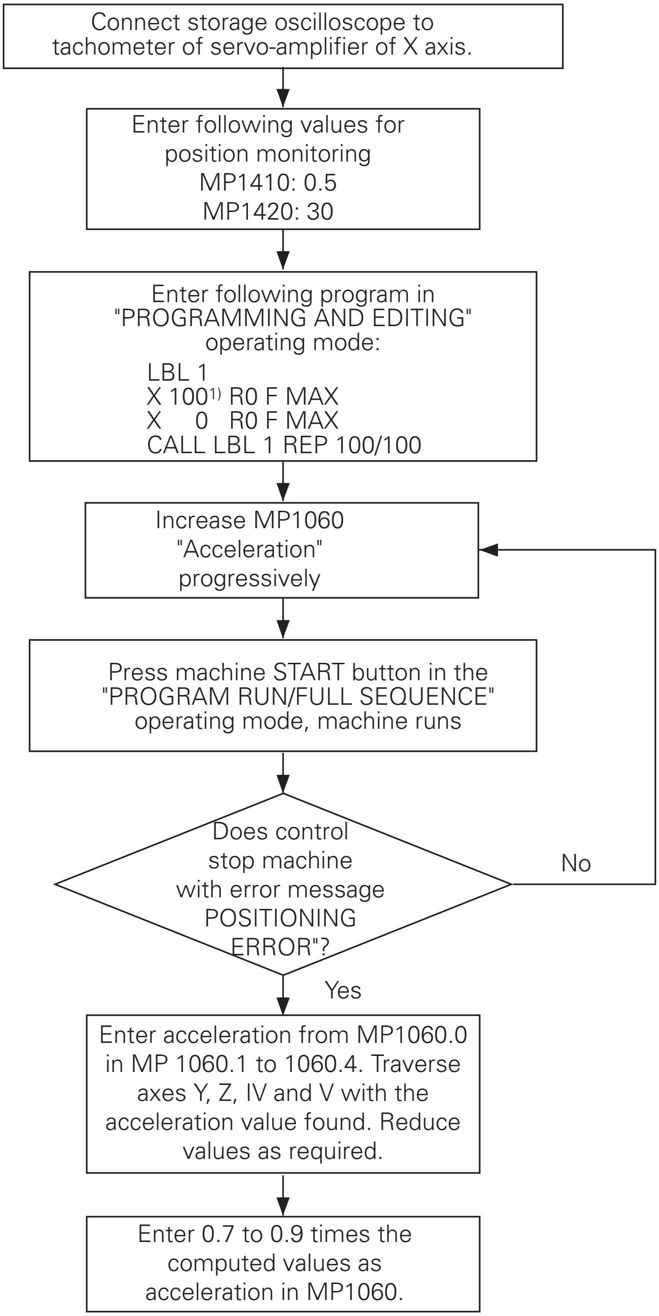
k_V factor MP1510: Reduce entry value until no oscillations can be detected.



1) Program the traverse paths for the axis concerned as large as possible.

Optimize acceleration

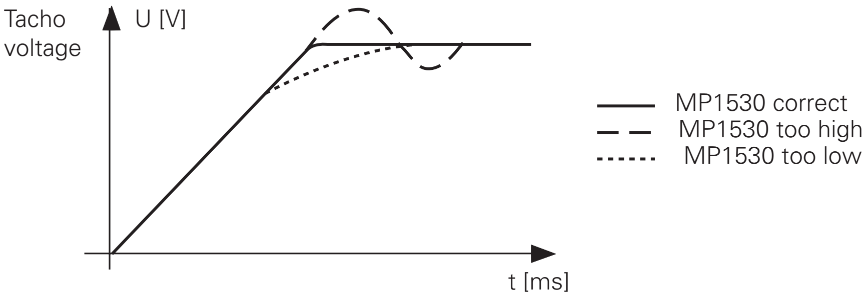
If the maximum acceleration of the servo loop cannot be determined with the battery supply, the acceleration can be optimized as follows:



[1)] Program the traverse paths for the axis concerned as large as possible.

Optimize transient response

MP1520 defines the transient response when accelerating and braking. The approach behaviour is optimized to the target position.



Since MP1520 is effective for all axes, the worst axis determines the entry value. No axis should show an overshoot.

If MP1520 is set too low then the positioning times to the target position will increase substantially. Optimize MP1520 with great care!

Connect storage oscilloscope to tachometer of servo-amplifier of X axis.

↓

Enter following program in "PROGRAMMING AND EDITING" operating mode:

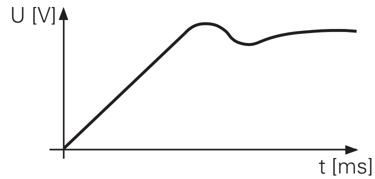
```
LBL 1
X 100 1) R0 F MAX
X     0   R0 F MAX
CALL LBL 1 REP 100/100
```

↓

Press machine START button in the "PROGRAM RUN/SINGLE BLOCK" operating mode.
Machine runs.

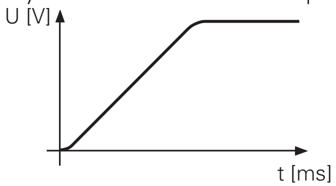
↓

MP1520: Increase entry value until overshoot occurs.



↓

MP1520: Reduce entry value until overshoot stops.



↓

Check axes Y, Z, IV and V and correct MP1520, if necessary.

[1] Program the traverse paths for the axis concerned as large as possible.

16.3.8 Optimize the integral factor

MP7290: "4" (display step = 1 μm)
Machine stands in position, under servo control.

↓

MP1080 integral factor (for X-axis):
raise entry value [1], until the
Servo positioning becomes noticeably unstable
(observe the actual value display).

↓

MP1080: reduce the entry value in small steps
until the servo positioning is steady again [2].

↓

Repeat adjustment for Y axis, Z, IV and V.

Whether or not an integral factor is possible in connection with the automatic offset adjustment with MP 1220 can only be judged when observing the complete machine.
The optimum solution must be found during the commissioning procedure.

[1] First entry value 100, raise subsequent values in steps of 100.
[2] For drives which are not quite free of backlash the entry value should be"0".

16.3.9 Adjust the offset

The rough offset-adjustment has already been carried out on the servo-amplifier. A fine offset-adjustment can be performed with the offset-adjustment possibility described in the section "Servo positioning".

16.3.10 Adjustment of the monitoring functions

The following entry values are recommended for the monitoring functions (see section "Servo positioning"):

Machine parameter	Function	Entry value
MP1710	Position monitoring	1.2 × lag in rapid traverse
MP1720	Position monitoring (EMERGENCY STOP)	1.4 × lag in rapid traverse
MP1410	Position monitoring	0.5 [mm]
MP1420	Position monitoring (EMERGENCY STOP)	10 [mm]
MP1140	Movement monitoring	0.5 [V]
MP1030	Positioning window	0.01 [mm]
MP1110	Standstill monitoring	0.2 [mm]

If the drives of the machine permit narrower limits, these may be entered.

17 Special functions for laser cutting machines

Special functions can be activated to interface the TNC to laser cutting machines and water jet machines.

17.1 Analogue voltage output

If the analogue output (X8, pin 8) is not needed for the spindle, MP3011 can define another function for this analogue output. The input values in MP3011 show an effect only if the value in MP3010 is less than 3.

MP3011	Function of analogue output S, if MP3010 < 3 Entry: 0 to 3 0 = No special function 1 = Voltage proportional to current contour feed rate (depending on MP3012) 2 = Voltage as defined by PLC module 9130 3 = Voltage is defined via M function (M200 to M204)

17.1.1 Voltage proportional to feed rate (MP3011 = 1)

A voltage proportional to the current contour feed rate is output. The feed rate achieved when 10 V is output is entered in MP3012.

MP3012	Feed rate for output of an analogue voltage of 10 V (MP3011 = 1) Entry: 0 to 300 000 [$\frac{mm}{min}$]

17.1.2 Voltage from the PLC (MP3011 = 2)

The voltage defined through the PLC module 9130 is output. See also chapter "PLC Programming".

17.1.3 Definition of the voltage via M function (MP3011 = 3)

The analogue voltage output can be defined in the positioning block with the miscellaneous functions M200 to M204. These M functions are available only if the value 3 is entered in MP3011. The M functions are executed synchronously to the positioning blocks and are effective at the beginning of the block.

Direct output of the programmed voltage: M200 V...

The TNC outputs the value programmed behind M200 V... as a voltage.
Entry: 0 to 9.999 [V]
Duration: M200 V... is effective until a new voltage is output through M200 to M204.

Voltage varies with distance: M201 V...

The TNC outputs the voltage as a function of the traversed distance. The TNC increases or decreases the voltage linearly from the active voltage to the value programmed behind M201 V...
Entry: 0 to 9.999 [V]
Duration: M201 V... is effective unit a new voltage is output through M200 to M204.

Voltage varies with the velocity: M202 FNR.

The TNC outputs the voltage as a function of the velocity. With machine parameters MP3013.x and MP3014.x up to 3 characteristic curves are defined in a table. In the table certain analogue voltages are assigned to certain feed rates. M202 FNR. selects the characteristic curve in which the TNC finds the voltage to be output.
Entry: 1 to 3
Duration: M202 FNR. is effective until a new voltage is output through M200 to M204.

You can enter in the table up to four kink points per characteristic curve. The output values are interpolated linearly between the kink points. The first kink point must have the value 0. The entry values of the following kink points must increase in sequence. The TNC recognizes the beginning of a new characteristic curve from the entry value 0.

Example:

Velocity		Voltage		
MP3013.0	0	MP3014.0	0	Characteristic 1
MP3013.1	25	MP3014.1	0	"
MP3013.2	500	MP3014.2	4.5	"
MP3013.3	1000	MP3014.3	9.999	"
MP3013.4	0	MP3014.4	0	Characteristic 2
MP3013.5	10 000	MP3014.5	9.999	"
MP3013.6	0	MP3014.6	9	Characteristic 3
MP3013.7	50	MP3014.7	0.5	"
MP3013.8	300	MP3014.8	1.5	"
MP3013.9	5000	MP3014.9	9.999	"
MP3013.10	0	MP3014.10	0	Not used
MP3013.11	0	MP3014.11	0	here

MP3013.0 to MP3013.11	Characteristic kink points for analogue voltage output with M202
	Entry: 10 to 300 000 [mm/min]

MP3014.0 to MP3014.11	Characteristic kink points for analogue voltage output with M202
	Entry: 0.000 to 9.999 [V]

Voltage varies with the time (time-voltage ramp): M203 V... TIME...

The TNC outputs the voltage as a function of the time. It increases or decreases the voltage linearly in the time programmed behind TIME from the current voltage to the voltage value programmed behind V...

Entry:	Voltage V:	0 to 9.999 [V]
	TIME:	0 to 1.999 [sec]
Duration:	M203 V... TIME... is effective until a new voltage is output through M200 to M204.	

Voltage for a specific time (time pulse): M204 V... TIME...

The TNC outputs the voltage programmed behind V... as a pulse. The duration of the pulse is entered with TIME....

Entry:	Voltage V:	0 to 9.999 [V]
	TIME:	0 to 1.999 [sec]
Duration:	M204 V... TIME... is effective until a new voltage is output through M200 to M204.	

17.2 Graphic simulation without TOOL CALL

Graphic simulation is also available on machines that operate without tool definition (e.g. water jet and laser cutters).

The tool radius for graphic simulation is defined in machine parameter MP7315. MP7316 defines the depth of penetration of the simulated tool. The program sections to be displayed are marked with M functions that are defined in machine parameters MP7317.0 and MP7317.1.

MP7315	Tool radius for graphic simulation without TOOL CALL Entry: 0.0000 to 99 999.9999 [mm]
MP7316	Tool penetration depth Entry: 0.0000 to 99 999.9999 [mm]
MP7317.0	M function at start of graphic display Entry: 0 to 88
MP7317.1	M function to interrupt graphic display Entry: 0 to 88

17.3 Program stop with M functions

When an M function is output in the "Program run, full sequence" and "Program run, single block" modes, the program run is normally interrupted until the PLC reports that the M function was executed.

This can be a disadvantage in certain applications, such as with laser cutting machines. For such applications it is more desirable not to wait for the acknowledgement of the M function, but rather to run the program continuously. This function can be selected through machine parameter MP7440, Bit 2. There must be no PLC positioning, datum shift, spindle orientation or limit switch during M function output.

This function must not be used on milling machines or boring mills.

Machine parameters – Contents

1 What is a machine parameter?

A contouring control must have access to specific data (e.g., traverse distances, acceleration) before it can execute its programmed instructions.

The machine tool builder provides these data in so-called machine parameters. In addition, machine parameters can be used to activated certain functions, which are possible with HEIDENHAIN contouring controls, but are required only on certain types of machines (e.g. automatic tool changing).

The list of machine parameters is not numbered in sequence but is split into groups according to function.

Machine parameters	Functional group
0 to 999	Measuring systems and machines
1000 to 1399	Positioning
1400 to 1699	Operation with feed precontrol
1700 to 1999	Operation with servo lag
3000 to 3999	Spindle
4000 to 4999	Integral PLC
5000 to 5999	Setting the data interface
6000 to 6199	Measuring with a 3D touch probe
6200 to 6299	Digitizing with TS 120
7100 to 7199	Tapping
7200 to 7399	Display and programming
7400 to 7599	Machining and program run
7600 to 7699	Hardware

If there is more than one input value for a single function (e.g., a separate input for each axis), the parameter number is provided with indices.

Example:

MP330 Grating period
MP330.0 Grating period for axis X
MP330.1 Grating period for axis Y
MP330.2 Grating period for axis Z
MP330.3 Grating period for axis 4
MP330.4 Grating period for axis 5

The indices are assigned to the corresponding axes according to a fixed pattern. For example, if an entry is possible only in axes 4 and 5, then only the indices 3 and 4 will appear.

1.1 User parameters

The MOD function "User Parameters" permits the control operator to easily access and change certain machine parameters. The machine tool builder can define up to 16 different machine parameters as user parameters through MP7330 (see chapter "Machine Adjustment", section "Display and operation").

2 Input/output of machine parameters

If the machine parameters have not yet been entered in a HEIDENHAIN contouring control (e.g., during commissioning), the TNC presents the list of machine parameters after the memory test. Now the input values must be entered either by hand on the keyboard or through the data interface. The data interface is activated by pressing the EXT key. It is pre-set to RS-232-C format and FE1 mode. This default setting can be changed through the MOD functions (see chapter "Data Interface").

2.1 Entry format

A number is entered for each machine parameter. This value can be, for example, the acceleration in mm/s^2 of an individual axis, or the analogue voltage in volts. In the TNC 415B and TNC 425, values are entered and stored to exactly 0.0001 mm (or °). With the TNC 407, TNC 415F and TNC 425E you can enter to 0.0001 mm (or °), but the entry value is rounded to 0.001 mm (or °).

You can add a written comment to your entry by placing a semicolon ";" behind the numerical entry, followed by your comment.

The input values can be entered in decimal, binary (%) or hexadecimal ($) format.

There are machine parameters with which individual functions are activated bit-coded. Binary entry (%) is recommended for these machine parameters. The hexadecimal format ($) may be advisable for other machine parameters.

Example:

Disabling file types with machine parameter MP7224.0.

Bit 0	HEIDENHAIN programs	0 = not disabled 1 = disabled
Bit 1	DIN/ISO programs	
Bit 2	Tool tables	
Bit 3	Datum tables	
Bit 4	Pallet tables	
Bit 5	Text files	

The datum tables, pallet tables and text files are to be disabled.

Input value for MP7224.0 =

Binary %111 000
Hexadecimal $ 38

Decimal 56 ($2^3 + 2^4 + 2^5$)

With software types 243 05, 259 96, 243 07 and 243 02 only the decimal format is possible for the machine parameters. The corresponding decimal input value must be computed for bit-coded entries. You will find a "Power of 2" table in the chapter entitled "Appendix".

2.2 Activating the machine parameter settings

After the values for the machine have been entered, exit the machine parameter list by pressing the END □- key. Missing or incorrect entries result in error messages from the control that prompt you to correct your entry.

The following errors are displayed:

Entry error	Meaning
0	No MP number found
1	Invalid MP number
2	No separator ";" found
3	Entry value incorrect
4	MP doubly defined
5	MP not defined
6	MP can not be stored

If the control does not recognise any errors, it automatically exits the machine parameter editor and is ready for operation.

If during commissioning no entries are made in the parameter list (MP NAME), the TNC will generate a standard machine parameter list when the END □ key is pressed. In this list the TNC is defined as a programming station with the HEIDENHAIN standard colours. All other machine parameters assume the minimum value.

It is also possible to keep several machine parameter lists and load the desired list into the TNC when needed. The desired list can be selected in the machine parameter editor by pressing the PGM NAME key and the SELECT soft key. The parameter list which is active when you exit the machine parameter editor goes into effect.

2.3 Changing the entry values

After a machine parameter list has been created, it can be changed either through the machine parameter editor or directly through the PLC.

2.3.1 Manual input

Call the machine parameter editor through the MOD function "code number".

Enter the **code number 95 148** to access the complete list of machine parameters.

Entering the **code number 123** opens a partial list of machine parameters. These are the machine parameters that may be changed by the control user (see User's Manual for TNC 407/TNC 415). The machine parameters which can be changed with the code number 123 are marked in the following list with CN 123.

Exit the machine parameter editor by pressing the END □ key.

2.3.2 Changing the entry values via PLC

The entry values of the active machine parameter list can be changed with the module 9031 (see chapter "PLC Programming", section "PLC-modules"). These changes are then erased when the TNC is switched off.

The machine parameters which can by changed through the PLC are marked in the following list with PLC EDIT or PLC RUN. PLC EDIT means that the machine parameters can only be changed if no program is being executed on the machine. PLC RUN means that the machine parameter can be changed even while a program is being executed.

3 List of machine parameters

3.1 Measuring systems and machines

Machine parameter	Function and input	Change via	Reaction	Page
MP10	Active axes Entry: %xxxxx Bit 0 X axis 0 = not active Bit 1 Y axis 1 = active Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis		RESET	4-6
MP30	Checking the absolute position of the distance-coded reference marks Entry: %xxxxx Bit 0 X axis 0 = not active Bit 1 Y axis 1 = active Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis	PLC RUN		4-8
MP31	Checking the amplitude of the measuring system signals Entry: %xxxxx Bit 0 X axis 0 = not active Bit 1 Y axis 1 = active Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis Bit 5 S-axis	PLC RUN		4-10
MP32	Checking the edge separation of the measuring system signals Entry: %xxxxx Bit 0 X axis 0 = not active Bit 1 Y axis 1 = active Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis Bit 5 S-axis	PLC RUN		4-10

Machine parameter	Function and input	Change via	Reaction	Page
MP40	VDU display Entry: %xxxxx Bit 0 X axis 0 = not active Bit 1 Y axis 1 = active Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis	PLC RUN		4-14
MP50	Controlled axes Entry: %xxxxx Bit 0 Axis X 0 = not controlled Bit 1 Axis Y 1 = controlled Bit 2 Axis Z Bit 3 4th Axis Bit 4 5th Axis	PLC RUN		4-89
MP60	PLC auxiliary axes Entry: %xxxxx Bit 0 Axis X 0 = NC axis Bit 1 Axis Y 1 = PLC axis Bit 2 Axis Z Bit 3 4th Axis Bit 4 5th Axis	PLC RUN		4-38
MP110.0-4	Assignment of the measuring system inputs to the axes Entry: 0 to 5 0 = measuring system input X1 1 = measuring system input X2 2 = measuring system input X3 3 = measuring system input X4 4 = measuring system input X5 5 = measuring system input X6 MP110.0 X axis MP110.1 Y axis MP110.2 Z axis MP110.3 4th axis MP110.4 5th axis		RESET	4-13

Machine parameter	Function and input	Change via	Reaction	Page
MP120.0-4	Assignment of the nominal value outputs Entry: 0 to 5 0 = output 1 1 = output 2 2 = output 3 3 = output 4 4 = output 5 5 = output S MP120.0 X axis MP120.1 Y axis MP120.2 Z axis MP120.3 4th axis MP120.4 5th axis		RESET	4-14
MP210	Count direction of the measuring system signals Entry: %xxxxx Bit 0 X axis 0 = positive Bit 1 Y axis 1 = negative Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis		RESET	4-8
MP330.0-4	Signal period Entry: 0.100 to 1000.000 [µm] or $\left[\frac{1°}{1000}\right]$ MP330.0 X axis MP330.1 Y axis MP330.2 Z axis MP330.3 4th axis MP330.4 5th axis	PLC EDIT	REF	4-7
MP340.0-4	Interpolation factor of the EXE at X5, X6 (only TNC407) Entry: 0,1, 5 0 = no measuring system on X5, X6 or EXE without interpolation 1 = EXE without interpolation 5 = EXE with 5-fold interpolation MP340.0 X axis MP340.1 Y axis MP340.2 Z axis MP340.3 4th axis MP340.4 5th axis			4-8

Machine parameter	Function and input	Change via	Reaction	Page
MP410	Axis designation Entry: 0 to 5 0 = A 1 = B 2 = C 3 = U 4 = V 5 = W MP410.3 4th axis MP410.4 5th axis	PLC EDIT	RESET	4-13
MP420	Hirth coupling Entry: 0 or 1 0 = No Hirth coupling 1 = Hirth coupling active MP420.3 4th axis MP420.4 5th axis	PLC RUN		4-212
MP430	Prescribed step for Hirth coupling Entry: 0.0000 to 30.0000 [°] MP430.3 4th axis MP430.4 5th axis	PLC RUN		4-212
MP710	Backlash compensation Entry: –1.0000 to +1.0000 [mm] or [°] MP710.0 X axis MP710.1 Y axis MP710.2 Z axis MP710.3 4th axis MP710.4 5th axis	PLC EDIT		4-23
MP711	Compensation for reversal spikes in circular interpolation Entry: –1.0000 to1.0000 [mm] MP711.0 X axis MP711.1 Y axis MP711.2 Z axis MP711.3 4th axis MP711.4 5th axis	PLC RUN		4-24

Machine parameter	Function and input	Change via	Reaction	Page
MP712	Compensation per control loop cycle time Entry: 0.000000 to 99.999999 [mm] MP712.0 X axis MP712.1 Y axis MP712.2 Z axis MP712.3 4th axis MP712.4 5th axis	PLC RUN		4-25
MP715	Compensation for reversal spikes in circular interpolation (M105) Entry: –1.0000 to +1.0000 [mm] MP715.0 X axis MP715.1 Y axis MP715.2 Z axis MP715.3 4th axis MP715.4 5th axis	PLC RUN		4-25
MP716	Compensation per control loop cycle time (M105) Entry: 0.000000 to 99.999999 [mm] MP716.0 X axis MP716.1 Y axis MP716.2 Z axis MP716.3 4th axis MP716.4 5th axis	PLC RUN		4-25
MP720	Linear axis-error compensation Entry: –1.0000 to +1.0000 [mm/m] MP720.0 X axis MP720.1 Y axis MP720.2 Z axis MP720.3 4th axis MP720.4 5th axis	PLC EDIT		4-26
MP730	Selection of linear or non-linear axis error compensation Entry: %xxxxx 0 = Linear axis error compensation 1 = Non-linear axis error compensation Bit 0 X axis 0 = not active Bit 1 Y axis 1 = active Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis	PLC EDIT		4-26 4-31

Machine parameter	Function and input	Change via	Reaction	Page
MP810	Display mode for rotary axes and PLC auxiliary axes Entry: 0.0000 to 9 999.9999[°] 0 = Display +/-99 999.9999; software limit switch active ≠ 0 = Modulo value display; software limit switch not active MP810.0 X axis MP810.1 Y axis MP810.2 Z axis MP810.3 4th axis MP810.4 5th axis	PLC EDIT		4-135
MP850	Synchronized axes Entry: 0 to 5 0 = Main axis 1 = Axis tracked to X axis 2 = Axis tracked to Y axis 3 = Axis tracked to Z axis 4 = Axis tracked to 4th axis 5 = Axis tracked to 5th axis MP850.0 X axis MP850.1 Y axis MP850.2 Z axis MP850.3 4th axis MP850.4 5th axis	PLC EDIT		4-49
MP855	Synchronization monitoring Entry: 0 to 100.0000 [mm] 0 = Monitoring inactive MP855.0 X axis MP855.1 Y axis MP855.2 Z axis MP855.3 4th axis MP855.4 5th axis	PLC EDIT		4-50
MP860	Datum for synchronization control Entry: 0 or 1 0 = Datum at position upon switch-on 1 = Datum at reference marks (machine datum) MP860.0 X axis MP860.1 Y axis MP860.2 Z axis MP860.3 4th axis MP860.4 5th axis	PLC EDIT		4-50

Machine parameter	Function and input	Change via	Reaction	Page
MP910	Traverse range 1 Default setting after switch-on; activated by PLC M2817 = 0, M2816 = 0 MP910.0 Software limit switch X+ MP910.1 Software limit switch Y+ MP910.2 Software limit switch Z+ MP910.3 Software limit switch 4+ MP910.4 Software limit switch 5+	PLC EDIT		4-16
MP911	Traverse range 2 activated by PLC M2817 = 0, M2816 = 1 MP911.0 Software limit switch X+ MP911.1 Software limit switch Y+ MP911.2 Software limit switch Z+ MP911.3 Software limit switch 4+ MP911.4 Software limit switch 5+	PLC EDIT		4-16
MP912	Traverse range 3 activated by PLC M2817 = 1, M2816 = 0 MP912.0 Software limit switch X+ MP912.1 Software limit switch Y+ MP912.2 Software limit switch Z+ MP912.3 Software limit switch 4+ MP912.4 Software limit switch 5+	PLC EDIT		4-17
MP920	Traverse range 1 Default setting after switch-on; activated by PLC M2817 = 0, M2816 = 0 MP920.0 Software limit switch X– MP920.1 Software limit switch Y– MP920.2 Software limit switch Z– MP920.3 Software limit switch 4– MP920.4 Software limit switch 5–	PLC EDIT		4-16
MP921	Traverse range 2 activated by PLC M2817 = 0, M2816 = 1 MP921.0 Software limit switch X– MP921.1 Software limit switch Y– MP921.2 Software limit switch Z– MP921.3 Software limit switch 4– MP921.4 Software limit switch 5–	PLC EDIT		4-17
MP922	Traverse range 3 activated by PLC M2817 = 1, M2816 = 0 MP922.0 Software limit switch X– MP922.1 Software limit switch Y– MP922.2 Software limit switch Z– MP922.3 Software limit switch 4– MP922.4 Software limit switch 5–	PLC EDIT		4-17

Machine parameter	Function and input	Change via	Reaction	Page
MP950	Datum point for positioning blocks with M92 Entry: –99 999.9999 to +99 999.9999 [mm] or [°] Values referenced to machine datum MP950.0 X axis MP950.1 Y axis MP950.2 Z axis MP950.3 4th axis MP950.4 5th axis	PLC RUN		4-127
MP951	Simulated toolchange position for TOOL CALL with block scan: –99 999.9999 to +99 999.9999 [mm] or [°] Values referenced to machine datum MP951.0 X axis MP951.1 Y axis MP951.2 Z axis MP951.3 4th axis MP951.4 5th axis	PLC RUN		4-149
MP960	Machine datum Entry: –99 999.9999 to +99 999.9999 [mm] or [°] MP960.0 X axis MP960.1 Y axis MP960.2 Z axis MP960.3 4th axis MP960.4 5th axis	PLC EDIT	REF	4-127

3.2 Positioning

Machine parameter	Function and input	Change via	Reaction	Page
MP1010	Rapid traverse Entry: 10 to 300 000 [mm/min] MP1010.0 X axis MP1010.1 Y axis MP1010.2 Z axis MP1010.3 4th axis MP1010.4 5th axis	PLC EDIT		4-70

Machine parameter	Function and input	Change via	Reaction	Page
MP1020	Manual feed Entry: 10 to 300 000 [mm/min] MP1020.0 X axis MP1020.1 Y axis MP1020.2 Z axis MP1020.3 4th axis MP1020.4 5th axis	PLC EDIT		4-70
MP1030	Positioning window Entry: 0.0001 to 2.0000 [mm] or [°] MP1030.0 X axis MP1030.1 Y axis MP1030.2 Z axis MP1030.3 4th axis MP1030.4 5th axis	PLC EDIT		4-87
MP1040	Polarity of the nominal value voltage for the positive direction of traverse Entry: %xxxxx Bit 0 Axis X 0 = positive Bit 1 Axis Y 1 = negative Bit 2 Axis Z Bit 3 4th axis Bit 4 5th axis			4-8
MP1050	Analogue voltage for rapid traverse Entry: 1.000 to 9.000 [V] MP1050.0 X axis MP1050.1 Y axis MP1050.2 Z axis MP1050.3 4th axis MP1050.4 5th axis	PLC EDIT		4-70
MP1060	Acceleration Entry: 0.001 to 5.000 [m/s^2] MP1060.0 X axis MP1060.1 Y axis MP1060.2 Z axis MP1060.3 4th axis MP1060.4 5th axis	PLC EDIT		4-68
MP1070	Radial acceleration Entry: 0.001 to 5.000 [m/s^2]	PLC RUN		4-80

Machine parameter	Function and input	Change via	Reaction	Page
MP1080	Integral factor Entry: 0 to 65 535 MP1080.0 integral factor X axis MP1080.1 integral factor Y axis MP1080.2 integral factor Z axis MP1080.3 integral factor 4th axis MP1080.4 integral factor 5th axis	PLC RUN		4-78
MP1110	Standstill monitoring Entry: 0.0010 to 30.0000 [mm] MP1110.0 X axis MP1110.1 Y axis MP1110.2 Z axis MP1110.3 4th axis MP1110.4 5th axis	PLC EDIT		4-87
MP1140	Movement monitoring Entry: 0.030 to 10.000 [V] MP1140.0 X axis MP1140.1 Y axis MP1140.2 Z axis MP1140.3 4th axis MP1140.4 5th axis	PLC RUN		4-86
MP1150	Delay time before switching off the residual voltage on error message "Position error" Entry: 0 to 65 535 [s]	PLC RUN		4-85
MP1220	Automatic cyclical offset adjustment Entry: 0 to 65 536 [s] 0 = no automatic adjustment	PLC RUN		4-77
MP1320	Direction for traversing the reference marks Entry: %xxxxx Bit 0 Axis X 0 = positive Bit 1 Axis Y 1 = negative Bit 2 Axis Z Bit 3 4th axis Bit 4 5th axis	PLC EDIT		4-62
MP1330	Feed rate for traversing the reference marks Entry: 80 to 300 000 [mm/min] MP1330.0 X axis MP1330.1 Y axis MP1330.2 Z axis MP1330.3 4th axis MP1330.4 5th axis	PLC RUN		4-62

Machine parameter	Function and input	Change via	Reaction	Page
MP1331	Feed rate for leaving the reference end-position (only for rotary encoders MP1350 = 2) Entry: 80 to 500 [mm/min] MP1331.0 X axis MP1331.1 Y axis MP1331.2 Z axis MP1331.3 4th axis MP1331.4 5th axis	PLC RUN		4-62
MP1340	Sequence for traversing reference marks Entry: 0 = no evaluation of reference mark 1 = X axis 2 = Y axis 3 = Z axis 4 = 4th axis 5 = 5th axis MP1340.0 1st axis MP1340.1 2nd axis MP1340.2 3rd axis MP1340.3 4th axis MP1340.4 5th axis	PLC RUN	REF	4-62
MP1350	Type of reference mark approach Entry: 0 = measuring system with distance-coded reference marks 1 = measuring system with one reference mark 2 = special sequence (linear measurement by rotary encoder) 3 = measuring system with distance-coded reference marks MP1350.0 X axis MP1350.1 Y axis MP1350.2 Z axis MP1350.3 4th axis MP1350.4 5th axis	PLC EDIT	REF	4-53
MP1390	Feed precontrol in the "Positioning with manual data input", "Program run, single block" and "Program run, full sequence" modes Entry: 0 or 1 0 = operation with feed precontrol 1 = operation with lag	PLC EDIT		4-75
MP1391	Feed precontrol in the "Manual" and "Handwheel" modes Entry: %xxxxx 0 = operation with lag 1 = operation with feed precontrol Bit 0 X axis Bit 3 4th axis Bit 1 Y axis Bit 4 5th axis Bit 2 Z axis	PLC EDIT		4-33 4-75

3.3 Operation with feed precontrol

Machine parameter	Function and input	Change via	Reaction	Page
MP1410	Position monitoring in operation with feed precontrol (cancellable) Entry: 0.0010 to 30.0000 [mm] MP1410.0 X axis MP1410.1 Y axis MP1410.2 Z axis MP1410.3 4th axis MP1410.4 5th axis	PLC EDIT		4-85
MP1420	Position monitoring in operation with feed precontrol (EMERGENCY STOP) Entry: 0.0010 to 30.0000 [mm] MP1420.0 X axis MP1420.1 Y axis MP1420.2 Z axis MP1420.3 4th axis MP1420.4 5th axis	PLC EDIT		4-85
MP1510	kv factor for feed precontrol Entry: 0.100 to 20.000 [$\frac{m/min}{mm}$] MP1510.0 X axis MP1510.1 Y axis MP1510.2 Z axis MP1510.3 4th axis MP1510.4 5th axis	PLC RUN		4-76
MP1511	Factor for stiction compensation Entry: 0 to 16 777 215 MP1511.0 X axis MP1511.1 Y axis MP1511.2 Z axis MP1511.3 4th axis MP1511.4 5th axis	PLC RUN		4-33
MP1512	Limit of extent of stiction compensation Entry: 0 to 16 777 215 [counting steps] MP1512.0 X axis MP1512.1 Y axis MP1512.2 Z axis MP1512.3 4th axis MP1512.4 5th axis	PLC RUN		4-33

Machine parameter	Function and input	Change via	Reaction	Page
MP1513	Feed-rate limit for stiction compensation Entry: 0 to 300 000 [mm/min] MP1513.0 X axis MP1513.1 Y axis MP1513.2 Z axis MP1513.3 4th axis MP1513.4 5th axis	PLC RUN		4-33
MP1515	k_v factor for feed precontrol activated (M105) Entry: 0.100 to 20.000 [$\frac{m/min}{mm}$] MP1510.0 X axis MP1510.1 Y axis MP1510.2 Z axis MP1510.3 4th axis MP1510.4 5th axis	PLC RUN		4-76
MP1520	Transient response Entry: 0.100 to 10.000	PLC EDIT		4-76
MP1525	Maximum velocity for checking the positioning window Entry: 0.100 to 10.000 [mm/min] Recommended value: 0.5 [mm/min]	PLC EDIT		4-87
MP1530	*Damping factor for transient response* *Only with software types 243 05, 259 96, 243 07, 243 02.* Entry: 0.010 to 0.999	PLC EDIT		4-76

3.4 Operation with servo lag

Machine parameter	Function and input	Change via	Reaction	Page
MP1710	Position monitoring for operation with lag (cancellable) Entry: 0.0000 to 300.0000 [mm] MP1710.0 X axis MP1710.1 Y axis MP1710.2 Z axis MP1710.3 4th axis MP1710.4 5th axis	PLC EDIT		4-84

Machine parameter	Function and input	Change via	Reaction	Page
MP1720	Position monitoring for operation with lag (EMERGENCY STOP) Entry: 0.0000 to 300.0000 [mm] MP1720.0 X axis MP1720.1 Y axis MP1720.2 Z axis MP1720.3 4th axis MP1720.4 5th axis	PLC EDIT		4-84
MP1810	k_v factor for operation with lag Entry: 0.100 to 20.000 [$\frac{m/min}{mm}$] MP1810.0 X axis MP1810.1 Y axis MP1810.2 Z axis MP1810.3 4th axis MP1810.4 5th axis	PLC EDIT		4-70
MP1815	k_v factor for operation with lag activated by M105 Entry: 0.100 to 20.000 [$\frac{m/min}{mm}$] MP1815.0 X axis MP1815.1 Y axis MP1815.2 Z axis MP1815.3 4th axis MP1815.4 5th axis	PLC EDIT		4-70
MP1820	Multiplication factor for kv Entry: 0.001 to 1.000 MP1820.0 X axis MP1820.1 Y axis MP1820.2 Z axis MP1820.3 4th axis MP1820.4 5th axis	PLC EDIT		4-72
MP1830	Kink point Entry: 0.000 to 100.000 [%] MP1830.0 X axis MP1830.1 Y axis MP1830.2 Z axis MP1830.3 4th axis MP1830.4 5th axis	PLC EDIT		4-72

3.5 Digital speed control (only TNC 425)

Machine parameter	Function and input	Change via	Reaction	Page
MP1900	Select axes with digital speed controller Entry: %xxxxx			11-8
MP1910	Speed controller monitoring Entry: 1 to 16 777 215 MP1910.0 X axis MP1910.1 Y axis MP1910.2 Z axis MP1910.3 4th axis MP1910.4 5th axis			11-9
MP1920	Integral component for speed controller Entry: 0 to 65 535 MP1920.0 X axis MP1920.1 Y axis MP1920.2 Z axis MP1920.3 4th axis MP1920.4 5th axis			11-9
MP1925	Limiting the integral component for speed controller Entry: 0.000 to 65.535 [s] Suggested value: 0.1 to 2 s MP1925.0 X axis MP1925.1 Y axis MP1925.2 Z axis MP1925.3 4th axis MP1925.4 5th axis			11-9
MP1940	Proportional component for speed controller Entry: 0 to 65 535 MP1940.0 X axis MP1940.1 Y axis MP1940.2 Z axis MP1940.3 4th axis MP1940.4 5th axis			11-9
MP1945	Factor for acceleration precontrol of the rotational speed controller Entry: 0.000 to 9.999 [V/(m/s^2)] MP1945.0 X axis MP1945.1 Y axis MP1945.2 Z axis MP1945.3 4th axis MP1945.4 5th axis			11-9

Machine parameter	Function and input	Change via	Reaction	Page
MP1950	Polarity for torque signal Entry: %xxxxx 0 = positive 1 = negative			11-10
MP1951	Select measuring system for position control Entry: %xxxxx 0 = linear measuring system for position control 1 = motor rotary encoder for position control			11-10
MP1955	Ratio of grating period LS to ROD Entry: 0.1 to 100 MP1955.0 X axis MP1955.1 Y axis MP1955.2 Z axis MP1955.3 4th axis MP1955.4 5th axis			11-10
MP1960	Compensation for reversal spikes Entry: –1.0000 to +1.0000 MP1960.0 X axis MP1960.1 Y axis MP1960.2 Z axis MP1960.3 4th axis MP1960.4 5th axis			11-11
MP1970	Motion monitor for position and speed Entry: 0 to 300.0000 [mm] 0 = no monitor MP1970.0 X axis MP1970.1 Y axis MP1970.2 Z axis MP1970.3 4th axis MP1970.4 5th axis			11-11
MP1980	Delayed shutdown of speed controller in EMERGENCY STOP Entry: 0 to 1.9999 [sec]			11-11

3.6 Spindle

Machine parameter	Function and input	Change via	Reaction	Page
MP3010	Spindle-speed output, Spindle orientation Entry: 0 to 8 0 = No spindle-speed output 1 = BCD-coded output of the spindle speed, only if the spindle speed changes 2 = BCD-coded output of the spindle-speed at every TOOL CALL 3 = Analogue output of the spindle speed, but gear-change signal only if the gear range changes 4 = Analogue output of the spindle speed and gear-change signal at every TOOL CALL 5 = Analogue output of the spindle speed and no gear-change signal 6 = As for entry value 3, but with controlled spindle for orientation 7 = As for entry value 4, but with controlled spindle for orientation 8 = As for entry value 5, but with controlled spindle for orientation	PLC EDIT	RESET	4-94
MP3011	Function of analogue output S if MP3010 < 3 Entry: 0 to 2 0 = No special function 1 = Voltage proportional to current contour feed rate (dependent on MP3012) 2 = Voltage as defined by PLC module 9130 3 = Voltage as defined via M function (M200 to M204)			4-275
MP3012	Feed-rate for output of an analogue voltage of 10 V (MP3011 = 1) Entry: 0 to 300 000 [mm/min]			4-207
MP3013.0 to MP3013.11	Characteristic curve kink points for analogue voltage output with M202 Entry: 10 to 300 000 [mm/min]			4-277
MP3014.0 to MP3014.11	Characteristic curve kink points for analogue voltage output with M202 Entry: 0.000 to 9.999 [V]			4-277
MP3020	Definition of the spindle speed range Entry: 0 to 99 999	PLC EDIT		4-104

Machine parameter	Function and input	Change via	Reaction	Page
MP3030	Axis-halt on TOOL CALL with only a spindle speed output Entry: 0 or 1 1 = No axis-halt on TOOL CALL 0 = Axis-halt on TOOL CALL	PLC EDIT		4-102
MP3120	Zero spindle speed permitted Entry: 0 or 1 0 = 0 rpm permitted 1 = 0 rpm not permitted	PLC EDIT		4-97
MP3130	Polarity of the S-analogue voltage Entry: 0 to 3 0 = M03 positive voltage M04 negative voltage 1 = M03 negative voltage M04 positive voltage 2 = M03 and M04 positive voltage 3 = M03 and M04 negative voltage	PLC RUN		4-95
MP3140	Count direction of the measuring system signals for the spindle Entry: 0 or 1 0 = positive count direction with M03 1 = negative count direction with M03	PLC RUN		4-96
MP3210.0-7	S-analogue voltage for gear range Entry: 0 to 9.999 [V]	PLC RUN		4-97
MP3240.1	Minimum S-analogue voltage output Entry: 0 to 9.999 [V]	PLC RUN		4-97
MP3240.2	Jog-voltage for gear change Entry: 0 to 9.999 [V]	PLC RUN		4-101
MP3310.0	Limit with S-override (maximum) Entry: 0 to 150 [%]	PLC RUN		4-99
MP3310.1	Limit with S-override (minimum) Entry: 0 to 150 [%]	PLC RUN		4-99
MP3410.0	Spindle ramp gradient for M03, M04, M05 Entry: 0 to 1.9999 [V/ms]	PLC RUN		4-97
MP3410.1	Spindle ramp gradient for spindle orientation Entry: 0 to 1.9999 [V/ms]	PLC RUN		4-106
MP3410.2	Spindle ramp gradient for tapping Entry: 0 to 1.9999 [V/ms]	PLC RUN		4-114
MP3410.3	Spindle ramp gradient for rigid tapping Entry: 0 to 1.9999 [V/ms]	PLC RUN		4-118

Machine parameter	Function and input	Change via	Reaction	Page
MP3420	Positioning window for spindle Entry: 0 to 65 535 [increments] 1 increment is equivalent to about 0.088 degree $= \frac{360 \text{ degrees}}{1024 \text{ lines} \times \text{ quadruple evaluation}}$	PLC RUN		4-108
MP3430	Reference mark deviation from the desired position (Spindle-pre-set) Entry: 0 to 360 [°]	PLC RUN		4-108
MP3440.0-7	kv factor for spindle-orientation Entry: 0.1 to 10 [$\frac{1000°/\text{min}}{°}$]	PLC RUN		4-108
MP3510.0-7	Spindle speed for each gear range Entry: 0 to 99 999.999 [rpm]	PLC EDIT		4-97
MP3515.0-7	Maximum spindle speed in each gear range Entry: 0 to 99 999.999 [rpm]	PLC EDIT		4-97
MP3520.0	Spindle-speed activated by marker M2501 Entry: 0 to 99 999.999 [rpm]	PLC RUN		4-108
MP3520.1	Spindle-speed for spindle-orientation Entry: 0 to 99 999.999 [rpm]	PLC RUN		4-106

3.7 Integral PLC

Machine parameter	Function and input	Change via	Reaction	Page
MP4010	PLC-program from RAM or from EPROM Entry: 0 or 1 0 = EPROM-operation 1 = RAM-operation		RESET	7-28
MP4060.0-4	Path dependent lubrication Entry: 0 to 65 535 [units of 65 536 µm]	PLC RUN		4-19
MP 4070	Compensation value per PLC cycle for axis error compensation with lag tracking Entry: 0.0001 to 0.005 [mm]			4-32
MP4110.0 to MP4110.47	Value for Timer T0 to T47 Entry: 0 to 65 535 [PLC-cycles] (TNC 415: 20 ms; TNC 407: 24 ms)	PLC RUN		7-25
MP4120.0 to MP4120. 31	Pre-set value for counters C0 to C31 Entry: 0 to 65 535 [PLC cycles]	PLC RUN		7-26
MP4130	Fast PLC input to suppress the monitoring functions Entry: 0 to 255 [No. of the PLC input]	PLC EDIT		4-83
MP4131	Activation condition for fast PLC input from MP4130 Entry: 0 or 1	PLC EDIT		4-83
MP4210.0 to MP4210.47	Set a number in the PLC (D768 to D956) Entry: –99 999.9999 to +99 999.9999 [mm] or [°]	PLC EDIT		7-21
MP4220.0-4	Machine parameter with multiple function Entry: 10 to 30 000 – Feed rate for returning to the contour – To set a number in the PLC in the word range W960 to W968	PLC EDIT		 4-148 7-23
MP4230.0 to MP4230.31	Set a number in the PLC (Module 9032) Entry: –99 999.9999 to +99 999.9999			7-23
MP4231.0 to MP4231.31	Set a number in the PLC (Module 9032) Entry: –99 999.9999 to +99 999.9999			7-23
MP4310.0-6	Set a number in the PLC (W976 to W988) Entry: 0 to 65 535	PLC EDIT		7-22 7-178
MP4410	Activation of analogue inputs Entry: %xx Bit 0 = 0 no analogue inputs on 1st extension 1 analogue input on 1st extension Bit 1 = 0 no analogue inputs on 2nd extension 1 analogue input on 2nd extension	PLC RUN		4-206

3.8 Setting the data interface

Machine parameter	Function and input	Change via	Reaction	Page
MP5000	Inhibit data interface Entry: 0 to 2 0 = Not inhibited 1 = RS-232-C/V.24 inhibited 2 = RS-422/V.11 inhibited	PLC RUN		8-21
MP5020.0 MP5020.1 MP5020.2	Operating mode EXT 1 Operating mode EXT 2 Operating mode EXT 3 (PLC) Entry: %xxxxxxx Bit 0 7 or 8 Data bits 0 = 7 Data bits 1 = 8 Data bits Bit 1 Block-Check-Character 0 = BCC-character optional 1 = BCC-character not Control character Bit 2 Transmission stop through RTS 0 = not active 1 = active Bit 3 Transmission stop through DC3 0 = not active 1 = active Bit 4 Character parity 0 = even 1 = odd Bit 5 Character parity 0 = not required 1 = required Bit 6/7 Stop bits Bit 6 Bit 7 1 1/2 Stop-Bits 0 0 2 Stop-Bits 1 0 1 Stop-Bit 0 1 1 Stop-Bit 1 1	PLC RUN CN 123		8-24
MP5030.0 MP5030.1 MP5030.2	Operating mode EXT 1 Operating mode EXT 2 Operating mode EXT 3 (PLC) Entry: 0 or 1 0 = "Standard data transmission" 1 = "Blockwise transfer"	PLC RUN CN 123		8-24

Machine parameter	Function and input	Change via	Reaction	Page
MP5040	Data transfer rate in operating mode EXT3 (data transfer via PLC) Entry: 0 to 9 0 = 110 baud; 5 = 2 400 baud 1 = 150 baud; 6 = 4 800 baud 2 = 300 baud; 7 = 9 600 baud 3 = 600 baud; 8 = 19 200 baud 4 = 1200 baud; 9 = 38 400 baud	PLC RUN		8-47
MP5200 MP5200.0 MP5200.1 MP5200.2	Control character for beginning of text (STX) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-25
MP5201 MP5201.0 MP5201.1 MP5201.2	Control character for end of text (ETX) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-25
MP5202 MP5202.0 MP5202.1 MP5202.2	ASCII character for file type for data entry in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-26
MP5203 MP5203.0 MP5203.1 MP5203.2	ASCII character for input identification (E) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-26
MP5204 MP5204.0 MP5204.1 MP5204.2	ASCII character for file type for data output in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-26
MP5205 MP5205.0 MP5205.1 MP5205.2	ASCII character for output identification (A) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-26
MP5206 MP5206.0 MP5206.1 MP5206.2	ASCII characters for start of heading (SOH) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-25

Machine parameter	Function and input	Change via	Reaction	Page
MP5207 MP5207.0 MP5207.1 MP5207.2	ASCII characters for end of transmission block (ETB) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-25
MP5208 MP5208.0 MP5208.1 MP5208.2	ASCII characters for acknowledgement (ACK) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-25
MP5209 MP5209.0 MP5209.1 MP5209.2	ASCII negative acknowledgement characters(NAK) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-25
MP5210 MP5210.0 MP5210.1 MP5210.2	Control characters for end of transmission (EOT) in operating mode EXT 1 in operating mode EXT 2 in operating mode EXT 3 (PLC) Entry: 0 to 127	PLC RUN CN 123		8-25

3.9 Measuring with a 3D touch probe

Machine parameter	Function and input	Change via	Reaction	Page
MP6010	Selection of the touch probe system Entry: 0 or 1 0 = TS 120 1 = TS 511	PLC EDIT CN 123		4-177

3.9.1 Digitizing with TS 120 (only with digitizing option)

Machine parameter	Function and input	Change via	Reaction	Page
MP6120	Probing feed rate (triggering touch probe) Entry: 10 to 3000 [mm/min]	PLC RUN CN 123		4-177
MP6130	Maximum measuring range Entry: 0.001 to 99 999.9999 [mm]	PLC RUN CN 123		4-177
MP6140	Safety-clearance over measurement point Entry: 0.001 to 99 999.9999 [mm]	PLC RUN CN 123		4-177
MP6150	Rapid traverse for probe cycle (triggering touch probe) Entry: 10 to 10000 [mm/min]	PLC RUN CN 123		4-177
MP6160	Spindle orientation for 180° rotation Entry: –1 to 88 –1 = Spindle orientation directly through NC 0 = Function not active 1 to 88 = Number of the M function for spindle orientation by PLC	PLC RUN CN 123		4-178
MP6200	Selection of triggering or measuring touch probe system (only with "Digitizing with TM 110" option) Entry: 0 or 1 0 = Triggering touch probe (e.g. TS 120) 1 = Measuring touch probe(e.g.. TM 110)			4-177
MP6210	Number of oscillations per second in normal direction Entry: 0 to 65 535 [Hz]	PLC RUN CN 123		4-181
MP6220	Traverse for lubricating the probe axis at end of line Entry: 0.000 to 99 999.999 [mm]			4-182
MP6221	Time after which the probe axis must be lubricated Entry: 0 to 65 535 [min]			4-182
MP6230	Feed rate in normal direction Entry: 0 to 1000 [mm/min]	PLC RUN CN 123		4-181
MP6240	Maximum deflection of stylus Entry: 0 to 10.000 [mm]	PLC RUN CN 123		4-181
MP6260	Output of M90 on NC blocks of digitized data Entry: 0 or 1 0 = no output of M90 1 = output of M90 in every NC block	PLC RUN CN 123		4-181

Machine parameter	Function and input	Change via	Reaction	Page
MP6270	Rounding of decimal places Entry: 0 to 2 0 = Output in 0.001 mm increments [1 µm] 1 = Output in 0.01 mm increments [10 µm] 2 = Output in 0.0001 mm increments [0.1 µm]	PLC RUN CN 123		4-181

3.9.2 Digitizing with TM 110 (only with digitizing option)

Machine parameter	Function and input	Change via	Reaction	Page
MP6310	Stylus deflection depth (measuring touch probe) Entry: 0.1000 to 2.0000 [mm]			4-190
MP6320	Counting direction of the measuring system signals (measuring touch probe) Entry: %xxx Bit 0 = X axis 0 = Positive Bit 1 = Y axis 1 = Negative Bit 2 = Z axis			4-190
MP6321	Measuring the centre offset while calibrating the TM110 Entry: 0 or 1 0 = Calibrate and measure centre offset 1 = Calibrate without measuring centre offset			4-178
MP6322	Assignment of touch probe axes to the machine axes Entry: 0 to 2 0 = Touch probe axis X 1 = Touch probe axis Y 2 = Touch probe axis Z MP6322.0 Machine axis X MP6322.1 Machine axis Y MP6322.2 Machine axis Z			4-191
MP6330	Maximum stylus deflection (measuring touch probe) Entry: 0.1 to 4.000 [mm]			4-191
MP6350	Feed rate for positioning to the MIN point and contour approach (Measuring touch probe) Entry: 10 to 3000 [mm/min]			4-191
MP6360	Probing feed rate (measuring touch probe) Entry: 10 to 3000 [mm/min]			4-177 4-191
MP6361	Rapid traverse in scanning cycle (measuring touch probe) Entry: 10 to 10 000 [mm/min]			4-178 4-191

Machine parameter	Function and input	Change via	Reaction	Page
MP6362	Feed rate reduction if stylus of the TM 110 is deflected away from its path Entry: 0 or 1 0 = Feed rate reduction not active 1 = Feed rate reduction active			4-191
MP6390	Target window for contour-line end point Entry: 0.1000 to 4.0000 [mm]			4-191

3.9.3 Tool calibration with TT 110

Machine parameter	Function and input	Change via	Reaction	Page
MP6500	Tool Calibration with TT 110 Entry: 0 or 1 0 = Cycles for tool calibration inhibited 1 = Cycles for tool calibration not inhibited			4-194
MP6505	Probing direction for tool measurement Entry: 0 to 3 0 = Positive probing direction in the angle reference axis (0°axis) 1 = Positive probing direction in the +90°-axis 2 = Negative probing direction in the angle reference axis (0° axis) 3 = Negative probing direction in the +90°-axis			4-194
MP6507	Calculation of the probing feed rate Entry: 0 to 2 0 = Calculation of the probing feed rate with constant tolerance 1 = Calculation of the probing feed rate with variable tolerance 2 = Constant probing feed rate			4-194
MP6510	Max. permissible measuring error for tool measurement with rotating tool Entry: 0.002 to 0.999 [mm]			4-194
MP6520	Probing feed rate for tool measurement with non-rotating tool Entry: 10 to 3000 [mm/min]			4-194
MP6530	Distance from tool lower edge to probe contract upper edge for tool radius measurement Entry: 0.001 to 99.9999 [mm]			4-194
MP6531	Diameter or edge length of TT 110 probe contact Entry: 0.001 to 99 999.9999 [mm]			4-194
MP6540	Safety zone around the probe contact TT 110 for pre-positioning Entry: 0.001 to 99 999.9999 [mm]			4-194

Machine parameter	Function and input	Change via	Reaction	Page
MP6550	Rapid traverse in the probing cycle for TT 110 Entry: 10 to 10 000 [mm/min]			4-194
MP6560	M function for spindle orientation for measuring individual teeth Entry: –0 to 88 –1 = Spindle orientation directly via NC 0 = Function inactive (error message) 1 to 88 = Number of the M function for spindle orientation via PLC			4-194
MP6570	Max. permissible surface cutting speed at the tool edge Entry: 1.0000 to 120.0000 [m/min]			4-194
MP6580	Coordinates of the TT 110 stylus centre referenced to the machine datum Entry: –99 999.9999 to 99 999.9999 [mm] MP6580.0 Axis X MP6580.1 Axis Y MP6580.2 Axis Z			4-194

3.10 Tapping

Machine parameter	Function and input	Change via	Reaction	Page
MP7110.0	Minimum feed-override when tapping Entry: 0 to 150 [%]	PLC RUN		4-114
MP7110.1	Maximum feed-override when tapping Entry: 0 to 150 [%]	PLC RUN		4-114
MP7120.0	Dwell-time for change of direction of spindle rotation in "tapping" cycle Entry: 0 to 65.535 [s]	PLC RUN		4-114
MP7120.1	Advanced switch time of spindle for "tapping" cycle with coded output Entry: 0 to 65.535 [s]	PLC RUN		4-116
MP7120.2	Spindle run-on time after reaching total hole depth Entry: 0 to 65.535 [s]	PLC RUN		4-114
MP7130	Spindle run-in characteristic Entry: 0.001 to 10 [°/min] (matching MP1520)	PLC EDIT		4-118
MP7140	Transient response of spindle on acceleration Entry: 0.001 to 1 (matching MP1530)	PLC EDIT		4-118

Machine parameter	Function and input	Change via	Reaction	Page
MP7150	Positioning window for tool axis Entry: 0.0001 to 2 [mm]	PLC EDIT		4-118
MP7160	Spindle orientation with Cycle 17 Entry: 0 or 1 0 = spindle orientation before execution of Cycle 17 1 = no spindle orientation before execution of Cycle 17			4-118

3.11 Display and operation

Machine parameter	Function and input	Change via	Reaction	Page
MP7210	Programming station Entry: 0, 1, 2 0 = Control and programming 1 = Programming station, "PLC active" 2 = Programming station "PLC inactive"	CN 123	RESET	4-153
MP7212	"POWER INTERRUPTED" Entry: 0 or 1 0 = "POWER INTERRUPTED" message must be acknowledged with CE key 1 = no "POWER INTERRUPTED" message	PLC RUN CN 123		4-155
MP7220	Block-number increment size for ISO programs Entry: 0 to 255	PLC RUN CN 123		4-151
MP7222	Filename length Entry: 0 to 2 0 = 8 characters 1 = 12 characters 2 = 16 characters	PLC RUN CN 123		4-150
MP7224.0	Disable file type Entry: %xxx xxxxx 0 = Not disabled 1 = Disabled Bit 0 HEIDENHAIN programs Bit 1 DIN/ISO programs Bit 2 Tool tables Bit 3 Datum tables Bit 4 Pallet tables Bit 5 Text files Bit 6 Help files Bit 7 Point tables	PLC RUN CN 123	RESET	4-150

Machine parameter	Function and input	Change via	Reaction	Page
MP7224.1	Protect file type Entry: %xxxxxxxx 0 = Not protected 1 = Protected Bit 0 HEIDENHAIN programs Bit 1 DIN/ISO programs Bit 2 Tool tables Bit 3 Datum tables Bit 4 Pallet tables Bit 5 Text files Bit 6 Help files Bit 7 Point tables	PLC RUN CN 123	RESET	4-150
MP7226.0	Size of pallet tables Entry: 0 to 255 [lines]	PLC RUN CN 123		4-151
MP7226.1	Size of datum tables Entry: 0 to 255 [lines]	PLC RUN CN 123		4-151
MP7228	Storage requirement in drip feed mode Entry: 0 to 1024 bytes MP7228.0 Minimum storage MP7228.1 Maximum storage			8-20
MP7230	Change dialogue language Entry: 0 or 1 0 = First dialogue language 1 = Basic language – English	PLC RUN CN 123		4-153
MP7240	PGM entry inhibit at (PGM NAME) = (OEM cycle number in EPROM) Entry: 0 or 1 0 = Inhibited 1 = Not inhibited	PLC RUN CN 123		9-5

Machine parameter	Function and input	Change via	Reaction	Page
MP7245.0	Inhibit the HEIDENHAIN cycles 1 to 15 Entry: 0 to 65 535 $0 to $FFFF 0 = not disabled 1 = disabled Bit 1 Cycle 1 Bit 2 Cycle 2 Bit 3 Cycle 3 Bit 4 Cycle 4 Bit 5 Cycle 5 Bit 6 Cycle 6 Bit 7 Cycle 7 Bit 8 Cycle 8 Bit 9 Cycle 9 Bit 10 Cycle 10 Bit 11 Cycle 11 Bit 12 Cycle 12 Bit 13 Cycle 13 Bit 14 Cycle 14 Bit 15 Cycle 15	PLC RUN		4-144
MP7245.1	Inhibit the HEIDENHAIN standard cycles 16 to 30 Entry: 0 to 65 535 $0 to $FFFF 0 = not disabled 1 = disabled Bit 0 Cycle 16 Bit 1 Cycle 17 Bit 2 Cycle 18 Bit 3 Cycle 19 Bit 4 Cycle 20 Bit 5 Cycle 21 Bit 6 Cycle 22 Bit 7 Cycle 23 Bit 8 Cycle 24 Bit 9 Cycle 25 Bit 10 Cycle 26 Bit 11 Cycle 27 Bit 12 Cycle 28 Bit 13 Cycle 29 Bit 14 Cycle 30 Bit 15 Cycle 31	PLC RUN		4-145
MP7246	Disable paraxial positioning blocks Entry: 0 or 1 0 = paraxial positioning blocks enabled 1 = paraxial positioning blocks disabled	PLC RUN		4-155

Machine parameter	Function and input	Change via	Reaction	Page
MP7250	Difference between Q-parameter numbers for DLG-CALL-block and DLG-DEF-block in OEM cycle Entry: 0 to 50	PLC RUN		9-5
MP7251	Number of global Q parameters transferred from OEM cycle to calling program Entry: 0 to 100	PLC RUN		9-5
MP7260	Number of tools in the tool table Entry: 0 to 254	CN 123	RESET	4-225
MP7261	Number of pockets in tool carousel Entry: 0 to 254	CN 123	RESET	4-225
MP7264	*Number of reserved pockets for special tools* *Entry: 0 to 3*		*RESET*	*4-226*
MP7266	Items in tool table Entry: 0 to 99 0 = No display 1 to 99 = Position in tool table MP7266.0 Tool name (NAME) MP7266.1 Tool length (L) MP7266.2 Tool radius (R) MP7266.3 Tool radius 2 (R2) MP7266.4 Oversize tool length (DL) MP7266.5 Oversize tool radius (DR) MP7266.6 Oversize tool radius 2 (DR2) MP7266.7 Tool locked? (TL) MP7266.8 Replacement tool (RT) MP7266.9 TIME 1 MP7266.10 TIME 2 MP7266.11 CURRENT TIME MP7266.12 Commentary on the tool (DOC) MP7266.13 No. of tool cutting edges (CUT) MP7266.14 Wear tolerance for tool length (LTOL) MP7266.15 Wear tolerance for tool radius (RTOL) MP7266.16 Cutting direction of the tool (DIRECT) MP7266.17 PLC status (PLC) MP7266.18 Tool length offset (TT:L-OFFS) MP7266.19 Tool radius offset (TT:R-OFFS) MP7266.20 Breakage tolerance for tool length (LBREAK) MP7266.21 Breakage tolerance for tool radius (RBREAK)	CN 123		4-225

Machine parameter	Function and input	Change via	Reaction	Page
MP7267	Items in pocket table Entry: 0 to 99 0 = No display 1 to 99 = Position in pocket table MP7267.0 Tool number (T) MP7267.1 Special tool (ST) MP7267.2 Fixed pocket (F) MP7267.3 Locked pocket (L) MP7267.4 PLC status (PLC)	CN 123		4-226
MP7270	Display of the feed rates in manual operating modes (Manual operation, Electronic handwheel) Entry: 0 or 1 0 = Display of the axis feed-rate only when an axis-direction key is pressed (axis-specific feed from MP1020.X) 1 = Display of the axis feed-rate before operating an axis-direction key (smallest value from MP1020.X for all axes)	PLC RUN CN 123		4-138
MP7280	Decimal sign Entry: 0 or 1 0 = Decimal comma 1 = Decimal point	PLC RUN CN 123		4-154
MP7285	Calculation of the tool length in the position display of the tool axis Entry: 0 or 1 0 = tool length ignored 1 = tool length taken into account	PLC RUN CN 123		4-134
MP7290	Position display step Entry: 0 to 6 0 = 0.1 mm or 0.1° 1 = 0.05 mm or 0.05° 2 = 0.01 mm or 0.01° 3 = 0.005 mm or 0.005° 4 = 0.001 mm or 0.001° 5 = 0.0005 mm or 0.0005° TNC 415 B only 6 = 0.0001 mm or 0.0001° TNC 415 B only MP7290.0 = X axis MP 7290.1 = Y axis MP 7290.2 = Z axis MP 7290.3 = 4th axis MP 7290.4 = 5th axis	PLC RUN CN 123		4-134

Machine parameter	Function and input	Change via	Reaction	Page
MP7295	Inhibit datum setting Entry: %xxxxx Bit 0 X axis0 = not disabled Bit 1 Y axis1 = disabled Bit 2 Z axis Bit 3 4th axis Bit 4 5th axis	PLC RUN CN 123		4-127
MP7296	Set datum with axis keys Entry: 0 or 1 0 = datum can be set with axis keys and soft key 1 = datum can be set with soft key only	PLC RUN CN 123		4-128
MP7300	Cancel Status-display and Q-parameters Entry: 0 to 7 0 = Cancel status display and Q parameters and tool data when program is selected 1 = Cancel status display, Q parameters and tool data with M02, M30, END PGM and when program is selected 2 = Cancel status display and tool data when program is selected 3 = Cancel status display and tool data when program is selected and with M02, M30, END PGM 4 = Cancel status display and Q parameters when program is selected 5 = Cancel status display and Q parameters when program is selected and with M02, M30, END PGM 6 = Cancel status display when program is selected 7 = Cancel status display when program is selected and with M02, M30, END PGM	PLC RUN CN 123		4-140

Machine parameter	Function and input	Change via	Reaction	Page
MP7310	Graphics-display Entry: %xxxx Bit 0 Changeover of 3-plane display 0 = European preferred 1 = American preferred Bit 1 Rotation of the coordinate system in the machining plane + 90° 0 = no rotation 1 = coordinate system rotated by +90° Bit 2 BLK form after datum shift 0 = BLK form will not shift 1 = BLK form will shift Bit 3 Show cursor position in 3-plane display 0 = not shown 1 = cursor position shown	PLC RUN CN 123		4-133
MP7315	Tool radius for graphic simulation without TOOL CALL Entry: 0.0000 to 99 999.9999 [mm]	PLC RUN CN 123		4-272
MP7316	Tool penetration depth Entry: 0.0000 to 99 999.9999 [mm]	PLC RUN CN 123		4-272
MP7317.0	M function to start graphic display Entry: 0 to 88	PLC RUN CN 123		4-272
MP7317.1	M function to interrupt graphic display Entry: 0 to 88	PLC RUN CN 123		4-272
MP7330.0 to MP7330.15	Determination of the User-parameters Entry: 0 to 9999.00 (Number of desired machine parameter)	PLC RUN		4-151
MP7340.0 to MP7340.15	Dialogues for User-parameters Entry: 0 to 4095 (EPROM address line number of the dialogue)	PLC RUN		4-151
  MP7350 MP7351 MP 7352 MP 7352.0 MP 7352.1 MP 7352.2 MP7353 MP7353.0 MP7353.1 MP7353.2	Colour for — Standard colour adjustment Window frame $030200C Error messages $03F3F0F Operating-mode display "Machine" Background $0000000 Text for operating mode $0342008 Dialogue $03F3828 Operating-mode display "Programming" Background $0000000 Text for operating mode $0342008 Dialogue $03F3828	PLC RUN		4-130

Machine parameter	Function and input	Change via	Reaction	Page
	Colour for Standard colour adjustment	PLC RUN		4-130
MP 7354	Program-text display ”Machine”			
MP 7354.0	Background $0080400			
MP 7354.1	General program-text $038240C			
MP 7354.2	Current block $038341C			
MP 7354.3	Background of active window $00C0800			
MP7355	Program-text-display ”Programming”			
MP7355.0	Background $0080400			
MP7355.1	General program-text $038240C			
MP7355.2	Current block $038341C			
MP7355.3	Background of active window $00C0800			
MP7356	Status- and PLC-window			
MP7356.0	Background $00C0800			
MP7356.1	Axis positions in the status-display $03F2C18			
MP7356.2	Status-display, except axis positions $03F280C			
MP7357	Soft key-display ”Machine”			
MP7357.0	Background $0000000			
MP7357.1	Symbols $03F3828			
MP7358	Soft- key display ”Programming”			
MP7358.0	Background $0000000			
MP7358.1	Symbols $03F3828			
MP7360	Graphics: 3D-depiction			
MP7360.0	Background $0000000			
MP7360.1	Surface $0203038			
MP7360.2	Front face $00C1820			
MP7360.3	Text-display in graphics window $03F3F3F			
MP7360.4	Side face $0102028			
MP7361	Graphics: view in three planes (and oscilloscope)			
MP7361.0	Background $0000000			
MP7361.1	Plan (Grating) $0203038			
MP7361.2	Front and side view (not selected channel) $0203038			
MP7361.3	Axis cross and text in graphics-display (cursor, data, screen window) $03F3F3F			
MP7361.4	Cursor (selected channel) $03F0000			
MP7362	Additional status-display in graphics window			
MP7362.0	Background graphics window $0080400			
MP7362.1	Background status display $00C0800			
MP7362.2	Status symbols $038240C			
MP7362.3	Status values $03F2C18			
MP7363	FK-graphics			
MP7363.0	Background $0000000			
MP7363.1	Resolved contour $03F3F3F			
MP7363.2	Subprograms and frame for zoom $0003F00			
MP7363.3	Alternative solutions $0003F00			
MP7363.4	Unresolved contour $03F0000			

3.12 Machining and program run

Machine parameter	Function and input	Change via	Reaction	Page
MP7410	"Scaling factor" cycle in two or three axes Entry: 0 or 1 0 = Cycle "Scaling factor" operates in all three principal axes 1 = Cycle "Scaling factor" only operates in the machining plane	PLC RUN CN 123		4-147
MP7411	Tool data in touch probe block Entry: 0 or 1 0 = With the touch probe block the current tool data are overwritten with the calibrated data of the probe system. 1 = The current tool data are retained even with a touch probe block.	PLC RUN CN 123		4-178
MP7420	Cycles for milling pockets with free-programmed contours Entry: %xxxx Bit 0 Slot-Milling direction 0 = Anti-clockwise slot-milling of the pocket contours, clockwise for islands 1 = Clockwise slot-milling of the pocket contours, anti clockwise for islands Bit 1 Sequence for clearing out and slot milling 0 = First slot-milling, then clear out pocket 1 = First clear out pocket, then slot-milling Bit 2 Merge programmed contours 0 = Contours merged only if the tool centre paths intersect 1 = Contours merged if the programmed contours overlap Bit 3 Clear out and slot-milling to pocket depth, or for each peck 0 = Clearing out and slot-milling performed in one operation for all pecks 1 = For each peck, first perform slot-milling and then feed clearing out (depending on Bit 1) before next peck	PLC RUN CN 123		4-147

Machine parameter	Function and input	Change via	Reaction	Page
	Bit 4 Position after machining 0 = Return to position that was approached before cycle was called 1 = TNC moves tool to clearance height			
MP7430	Overlap-factor for pocket milling Entry: 0.1 to 1.414	PLC RUN CN 123		4-145
MP7431	Arc end-point tolerance Entry: 0.0001 to 0.016 [mm]	PLC RUN CN 123		4-155
MP7440	Output of M-functions Entry: %xxxxx Bit 0 Program-halt on M06 0 = Program halt on M06 1 = No program-halt on M06 Bit 1 **modal** cycle-call M89 0 = normal code-transfer of M89 at beginning of block 1 = **modal** cycle-call M89 at end of block Bit 2 Program-halt on M-functions 0 = Program-halt until acknowledgement of M-function 1 = No program-halt (does not wait for acknowledgement) Bit 3 Select Kv factors with M105/M106 0 = function not active 1 = function active Bit 4 Reduced feed-rate in tool axis with M103 0 = function not active 1 = function active	PLC RUN CN 123		4-162

Machine parameter	Function and input	Change via	Reaction	Page
MP7450	Calculate the tool change position from MP951 in block scan Entry: %xxxxx Bit 0 Axis X 0 = Do not calculate Bit 1 Axis Y 1 = Calculate Bit 2 Axis Z Bit 3 4th axis Bit 4 5th axis	PLC RUN		4-149
MP7460	Constant feed rate in corners Entry: 0.0001 to 179.9999 [°]	PLC RUN		4-81
MP7475	Datum in datum table Entry: 0 or 1 0 = Datum point is workpiece datum 1 = Datum point is machine datum	PLC RUN CN 123		4-151
MP7480.0	Output of tool number or pocket number with TOOL CALL block Entry: 0 to 6 0 = No output 1 = Output of tool number only when tool number changes (W262) 2 = Output of tool number with every TOOL CALL block (W262) 3 = Output of pocket number (W262) and tool number (W264) only when tool number changes 4 = Output of pocket number (W262) and tool number (W264) with every TOOL CALL block 5 = Output of pocket number (W262) and tool number (W264) only when tool number changes. Pocket table does not change. 6 = Output of pocket number (W262) and tool number (W264) with every TOOL CALL block. Pocket table does not change.	PLC RUN		4-229

Machine parameter	Function and input	Change via	Reaction	Page
MP7480.1	Output of tool number or pocket number with TOOL DEF block Entry: 0 to 4 0 = No output 1 = Output of tool number only when tool number changes (W262) 2 = Output of tool number with every TOOL DEF block (W262) 3 = Output of pocket number (W262) and tool number (W264) only when tool number changes 4 = Output of pocket number (W262) and tool number (W264) with every TOOL DEF block	PLC RUN		4-229
MP7490	Number of traverse ranges Entry: 0 to 3 0 = 1 traverse range, 3 datum points 1 = 3 traverse ranges, 3 datum points 2 = 1 traverse range, 1 datum point 3 = 3 traverse ranges, 1 datum point	PLC RUN		4-17
MP7500	MP7500 "Tilt working plane" function Entry: 0 or 1 0 = inactive 1 = active	PLC EDIT		4-42
MP75x0	MP75x0 Transformed axis Entry: %xxxxxx Entry 0 means "End of transformation sequence" Bit 0 X axis Bit 1 Y axis Bit 2 Z axis Bit 3 A axis Bit 4 B axis Bit 5 C axis	PLC EDIT		4-42

Machine parameter	Function and input	Change via	Reaction	Page
MP75x1	Supplementary identifier for transformation Entry: %xx Bit 0 Swivel axis 0 = swivel head 1 = tilting table Bit 1 Dimension in MP75x2 0 = incremental step (for swivel head) 1 = absolute related to machine datum (for tilting table)	PLC EDIT		4-42
MP75x2	Dimension for transformation Entry: –99,999.9999 to +99,999.9999 Entry 0 means "free rotating axis"	PLC EDIT		4-42

3.13 Hardware

Machine parameter	Function and input	Change via	Reaction	Page
MP7620	Feed rate and spindle override Entry: %xxxx Bit 0 Feed rate override if rapid traverse button is pressed in "program run" 0 = Override not active 1 = Override active Bit 2 Feed rate override if rapid traverse buttons and machine direction buttons are pressed in "manual" operating mode 0 = Override not active 1 = Override active Bit 3 Spindle override in 1% increments or according to a non-linear characteristic curve 0 = 1% increments 1 = non-linear characteristic curve	PLC RUN		4-99

Machine parameter	Function and input	Change via	Reaction	Page
MP7640	Machine with handwheel Entry: 0 to 5 0 = no handwheel 1 = HR 330 (all keys evaluated by NC) 2 = HR 130 or HR 330 3 = HR 330 (+, – and "rapid traverse" keys evaluated by PLC) 4 = HR 332 (all keys evaluated by PLC) 5 = up to three HR 150 via HRA 110 6 = HR 410	PLC RUN CN 123		4-197
MP7641	Entry of handwheel interpolation factor Entry: 0 or 1 0 = entry at TNC control panel 1 = entry via PLC module 9036	PLC RUN CN 123		4-197
MP7645.0-7	Initializing parameters for handwheel Entry: 0 to 255 MP7645.0 MP7645.1 MP7645.2 MP7645.3-7 at present without function	PLC RUN CN 123		4-197 4-200 4-202 4-202 4-202
MP7650	Count direction for handwheel Entry: 0 or 1 0 = Negative count direction 1 = Positive count direction	PLC RUN		4-197
MP7660	Threshold sensitivity for electronic handwheel Entry: 0 to 65 535 [increments]	PLC RUN		4-197
MP7670	Minimum interpolation factor for handwheel Entry: 0 to 10	PLC RUN		4-197
MP7670.0	Interpolation factor for low speed			
MP7670.1	Interpolation factor for medium speed (HR 410 only)			
MP7670.2	Interpolation factor for high speed (HR 410 only)			
MP7671	Manual feed rate in "Handwheel" operating mode with HR 410 Entry: 0 to 1000 [% to MP 1020]			4-201
MP7671.0	Low speed			
MP7671.1	Medium speed			
MP7671.2	High speed			

Machine parameter	Function and input	Change via	Reaction	Page
MP7680	Machine parameter with multiple function Entry: %x xxx xxx Bit 0 Memory function for axis-direction keys 0 = not stored 1 = stored Bit 1 Re-approaching the contour 0 = not active 1 = active Bit 2 block scan (mid-program startup) 0 = not active 1 = active Bit 3 Interruption of block scan with STOP or M06 0 = interruption 1 = no interruption Bit 4 Include programmed dwell time during block scan 0 = include dwell time 1 = do not include dwell time Bit 5 Start calculation with block scan 0 = start from cursor position 1 = start from beginning of program Bit 6 Tool length for blocks with surface normal vector 0 = Without DR2 from the tool table 1 = With DR2 from the tool table	PLC RUN		4-155 4-227
MP7690	Memory test at switch-on Entry: %xx Bit 0 RAM-test 0 = Memory test at switch-on 1 = No memory test at switch-on Bit 1 EPROM-test 0 = Memory test at switch-on 1 = No memory test at switch-on			4-154

1 List of markers

The markers listed in italics have been retained to ensure compatibility with the TNC 355. It is recommended, however, that these functions be activated through the new word functions.

Marker	Function	Set	Reset	Page
M2000	Axis enable X	NC	NC	4-89
M2001	Axis enable Y	NC	NC	4-89
M2002	Axis enable Z	NC	NC	4-89
M2003	Axis enable 4	NC	NC	4-89
M2004	S-analogue voltage not on a ramp	NC	NC	4-97
M2005	S-analogue voltage = 0 V	NC	NC	4-97
M2007	Spindle in position	NC	NC	4-108
M2008	X axis in position	NC	NC	4-90
M2009	Y axis in position	NC	NC	4-90
M2010	Z axis in position	NC	NC	4-90
M2011	4th axis in position	NC	NC	4-90
M2012	Lubrication pulse X axis, since value of MP4060.0 was exceeded	NC	NC	4-20
M2013	Lubrication pulse Y axis, since value of MP4060.1 was exceeded	NC	NC	4-20
M2014	Lubrication pulse Z axis, since value of MP4060.2 was exceeded	NC	NC	4-20
M2015	Lubrication pulse 4th axis, since value of MP4060.3 was exceeded	NC	NC	4-20
M2016	Axis enable 5	NC	PLC	4-89
M2017	Axis 5 in position	NC	NC	4-90
M2018	Soft key "Manual operation" pressed	NC	NC	4-148
M2019	"Restore position" is active	NC	NC	4-148
M2022	Touch probe system not ready (ready-signal missing on connector X12 or signals from TM are faulty)	NC	NC	4-178
M2023	Stylus deflected before start of probing cycle	NC	NC	4-178
M2025	Stylus deflected (probe operation has been performed)	NC	PLC	4-178
M2026	Probe operation ended or interrupted	NC	NC	4-178
M2027	Battery voltage too low (battery-warning on connector X12); is evaluated only during the probe operation	NC	NC	4-178
M2029	Lubrication pulse 5 axis, since value of MP4060.4 was exceeded	NC	NC	4-20
M2032	*T-Code 1st bit (lsb)*	*NC*	*NC*	*7-177*
M2033	*T-Code 2nd bit*	*NC*	*NC*	*7-177*
M2034	*T-Code 3rd bit*	*NC*	*NC*	*7-155*
M2035	*T-Code 4th bit*	*NC*	*NC*	*7-155*
M2036	*T-Code 5th bit*	*NC*	*NC*	*7-155*
M2037	*T-Code 6th bit*	*NC*	*NC*	*7-155*
M2038	*T-Code 7th bit*	*NC*	*NC*	*7-155*
M2039	*T-Code 8th bit (msb)*	*NC*	*NC*	*7-155*
M2041	Basic language – English is selected	NC	NC	4-153
M2042	Analogue output of the spindle speed	NC	NC	4-95
M2043	Gear code change signal	NC	NC	4-101
M2044	S-code change signal	NC	NC	4-104

Marker	Function	Set	Reset	Page
M2045	Strobe signal for M-function	NC	NC	4-159
M2046	Strobe signal T code (P code) with TOOL CALL	NC	NC	4-229
M2047	Strobe signal 2nd T code (2nd P code) with TOOL DEF	NC	NC	4-229
M2048	Tapping cycle called	NC	NC	4-114
M2051	*Operating mode: Manual*	*NC*	*NC*	*7-177*
M2052	*Operating mode: Electronic handwheel*	*NC*	*NC*	*7-177*
M2053	*Operating mode: Positioning with manual data input*	*NC*	*NC*	*7-177*
M2054	*Operating mode: Program run, single block*	*NC*	*NC*	*7-177*
M2055	*Operating mode: Program run, full sequence*	*NC*	*NC*	*7-177*
M2057	*Operating mode: Traversing the reference marks*	*NC*	*NC*	*7-177*
M2059	Block scan (mid-program startup) is active	NC; PLC	NC; PLC	4-148
M2061	END-PGM, or M02 or M30 has been executed	NC	NC	4-154
M2064	*S-Code 1st bit (lsb)*			*7-177*
M2065	*S-Code 2nd bit*			*7-177*
M2066	*S-Code 3rd bit*			*7-177*
M2067	*S-Code 4th bit*			*7-177*
M2068	*S-Code 5th bit*			*7-177*
M2069	*S-Code 6th bit*			*7-177*
M2070	*S-Code 7th bit*			*7-177*
M2071	*S-Code 8th bit (msb)*			*7-177*
M2072	*M-Code 1st bit (lsb)*			*7-177*
M2073	*M-Code 2nd bit*			*7-177*
M2074	*M-Code 3rd bit*			*7-177*
M2075	*M-Code 4th bit*			*7-177*
M2076	*M-Code 5th bit*			*7-177*
M2077	*M-Code 6th bit*			*7-177*
M2078	*M-Code 7th bit*			*7-177*
M2079	*M-Code 8th bit (msb)*			*7-177*
M2080	*Minimum rpm from MP3020 1st bit (lsb)*			*7-177*
M2081	*Minimum rpm from MP3020 2nd bit*			*7-177*
M2082	*Minimum rpm from MP3020 3rd bit*			*7-177*
M2083	*Minimum rpm from MP3020 4th bit*			*7-177*
M2084	*Minimum rpm from MP3020 5th bit*			*7-177*
M2085	*Minimum rpm from MP3020 6th bit*			*7-177*
M2086	*Minimum rpm from MP3020 7th bit*			*7-177*
M2087	*Minimum rpm from MP3020 8th bit (msb)*			*7-177*
M2088	*Increment from MP3020 1st bit (lsb)*			*7-177*
M2089	*Increment from MP3020 2nd bit*			*7-177*
M2090	*Increment from MP3020 3rd bit*			*7-177*
M2091	*Increment from MP3020 4th bit (msb)*			*7-177*
M2092	Illegal spindle speed	NC	NC	4-102
M2093	Another T code (P code) follows with TOOL CALL	NC	NC	4-231
M2094	Maximum tool life elapsed (Time 1 in the tool table)	PLC	NC;PLC	4-227
M2095	Tapping – rigid tapping active	NC	NC	4-118
M2096	X Key last pressed	NC	NC	4-135
M2097	Y Key last pressed	NC	NC	4-135
M2098	Z key last pressed	NC	NC	4-135
M2099	Key IV last pressed	NC	NC	4-135
M2100	X-axis is tool axis	NC	NC	4-14

Marker	Function	Set	Reset	Page
M2101	Y-axis is tool axis	NC	NC	4-14
M2102	Z-axis is tool axis	NC	NC	4-14
M2103	4-axis is tool axis	NC	NC	4-14
M2104	*G-Code S-analogue 1st bit (lsb)*			*7-178*
M2105	*G-Code S-analogue 2nd bit*			*7-178*
M2106	*G-Code S-analogue 3rd bit (msb)*			*7-178*
M2112	*T-Number (P-Number) 1st decade (lsb)*			*7-178*
M2113	*T-Number (P-Number) 1st decade*			*7-178*
M2114	*T-Number (P-Number) 1st decade*			*7-178*
M2115	*T-Number (P-Number) 1st decade (msb)*			*7-178*
M2116	*T-Number (P-Number) 2nd decade (lsb)*			*7-178*
M2117	*T-Number (P-Number) 2nd decade*			*7-178*
M2118	*T-Number (P-Number) 2nd decade*			*7-178*
M2119	*T-Number (P-Number) 2nd decade (msb)*			*7-178*
M2127	Spindle in motion	NC	NC	4-109 4-177 4-198
M2128	X axis in motion	NC	NC	4-91
M2129	Y axis in motion	NC	NC	4-91
M2130	Z axis in motion	NC	NC	4-91
M2131	axis 4 in motion	NC	NC	4-91
M2132	axis 5 in motion	NC	NC	4-91
M2136	reference marks X axis not yet passed	NC	NC	4-63
M2137	reference marks Y axis not yet passed	NC	NC	4-63
M2138	reference marks Z axis not yet passed	NC	NC	4-63
M2139	reference marks axis 4 not yet passed	NC	NC	4-63
M2140	reference marks axis 5 not yet passed	NC	NC	4-63
M2148	Key V last pressed	NC	NC	4-135
M2149	Transfer with FN19 active	NC	NC	7-19
M2150	Dimensional unit for transfer with FN19 0 = mm; 1 = inch	NC	NC	7-19
M2151	Rapid traverse is output	NC	NC	4-138
M2160	X axis direction of traverse 0 = positive 1 = negative	NC	NC	4-9
M2161	Y axis direction of traverse 0 = positive 1 = negative	NC	NC	4-9
M2162	Z axis direction of traverse 0 = positive 1 = negative	NC	NC	4-9
M2163	Axis 4 direction of traverse 0 = positive 1 = negative	NC	NC	4-9
M2164	Axis 5 direction of traverse 0 = positive 1 = negative	NC	NC	4-9
M2176	*Code-Operating mode (lsb)*			*7-178*
M2177	*Code-Operating mode*			*7-178*
M2178	*Code-Operating mode*			*7-178*
M2179	*Code-Operating mode (msb)*			*7-178*
M2180	*1st PLC scan after switch-on*	*NC*	*NC*	–
M2182	Inhibited key was operated	NC	PLC	4-165
M2183	Program interruption ("Control operational" display flashes)	NC	NC	4-140
M2184	Control operational ("Control operational" display goes on or flashes)	NC	NC	4-140
M2185	1st PLC scan after interruption of the PLC program	NC	NC	–
M2186	*Code number 84159 entered*			*7-178*
M2187	Soft key function not executed	NC	NC	4-165

Marker	Function	Set	Reset	Page
M2190	Non-flashing error message is displayed	NC	NC	4-142
M2191	"External EMERGENCY OFF" message is displayed	NC	NC	4-120
M2192 to M2239	*Markers changeable via MP4310.0, MP 4310.1 and MP 4310.2*	*NC*	*NC*	*7-178*
M2240	OEM cycle 68 inhibit	NC	PLC	9-5
M2241	OEM cycle 69 inhibit	NC	PLC	9-5
M2242	OEM cycle 70 inhibit	NC	PLC	9-5
M2243	OEM cycle 71 inhibit	NC	PLC	9-5
M2244	OEM cycle 72 inhibit	NC	PLC	9-5
M2245	OEM cycle 73 inhibit	NC	PLC	9-5
M2246	OEM cycle 74 inhibit	NC	PLC	9-5
M2247	OEM cycle 75 inhibit	NC	PLC	9-5
M2248	OEM cycle 76 inhibit	NC	PLC	9-5
M2249	OEM cycle 77 inhibit	NC	PLC	9-5
M2250	OEM cycle 78 inhibit	NC	PLC	9-5
M2251	OEM cycle 79 inhibit	NC	PLC	9-5
M2252	OEM cycle 80 inhibit	NC	PLC	9-5
M2253	OEM cycle 81 inhibit	NC	PLC	9-5
M2254	OEM cycle 82 inhibit	NC	PLC	9-5
M2255	OEM cycle 83 inhibit	NC	PLC	9-5
M2256	OEM cycle 84 inhibit	NC	PLC	9-5
M2257	OEM cycle 85 inhibit	NC	PLC	9-5
M2258	OEM cycle 86 inhibit	NC	PLC	9-5
M2259	OEM cycle 87 inhibit	PLC	PLC	9-5
M2260	OEM cycle 88 inhibit	PLC	PLC	9-5
M2261	OEM cycle 89 inhibit	PLC	PLC	9-5
M2262	OEM cycle 90 inhibit	PLC	PLC	9-5
M2263	OEM cycle 91 inhibit	PLC	PLC	9-5
M2264	OEM cycle 92 inhibit	PLC	PLC	9-5
M2265	OEM cycle 93 inhibit	PLC	PLC	9-5
M2266	OEM cycle 94 inhibit	PLC	PLC	9-5
M2267	OEM cycle 95 inhibit	PLC	PLC	9-5
M2268	OEM cycle 96 inhibit	PLC	PLC	9-5
M2269	OEM cycle 97 inhibit	PLC	PLC	9-5
M2270	OEM cycle 98 inhibit	PLC	PLC	9-5
M2271	OEM cycle 99 inhibit	PLC	PLC	9-5
M2390	Cycle for tool calibration started	NC	NC	4-199
M2391	0 = Tool measurement 1 = Tool inspection	NC	NC	4-199
M2392	Wear tolerance exceeded	NC	NC	4-199
M2400	Tool number 0 programmed	NC	NC	4-229
M2401	Active tool with pocket number (MP7260/MP7261)	NC	NC	4-232
M2402	Active tool without pocket number (MP7260/MP7261)	NC	NC	4-232
M2403	Active tool = special tool	NC	NC	4–232
M2404	TOOL CALL programmed or by end of tool life	NC	NC	4-232

Marker	Function	Set	Reset	Page
M2408	Cycle 13 is executed	NC	PLC	4-108
M2448	NC start (edge evaluation)	PLC	PLC	4-174
M2449	Rapid traverse	PLC	PLC	4-174
M2450	Memory function for axis-direction keys	PLC	PLC	4-174
M2451	Feed-release	PLC	PLC	4-90
M2452	*Activate PLC-positioning axis X*	*PLC*	*NC*	*7-178*
M2453	*Activate PLC-positioning axis Y*	*PLC*	*NC*	*7-178*
M2454	*Activate PLC-positioning axis Z*	*PLC*	*NC*	*7-178*
M2455	*Activate PLC-positioning axis 4*	*PLC*	*NC*	*7-178*
M2456	Manual traverse X+	PLC	PLC	7-174
M2457	Manual traverse X+	PLC	PLC	7-174
M2458	Manual traverse Y+	PLC	PLC	4-174
M2459	Manual traverse Y-	PLC	PLC	4-174
M2460	Manual traverse Z+	PLC	PLC	4-175
M2461	Manual traverse Z-	PLC	PLC	4-175
M2462	Manual traverse 4+	PLC	PLC	4-175
M2463	Manual traverse 4-	PLC	PLC	4-175
M2464	Complement NC-start	PLC	PLC	4-174
M2465	Complement – rapid traverse	PLC	PLC	4-174
M2466	Complement – memory function for axis-direction keys	PLC	PLC	4-174
M2467	Complement – feed-release	PLC	PLC	4-90
M2468	*Complement PLC positioning axis X*	*NC*	*PLC*	*7-178*
M2469	*Complement PLC positioning axis Y*	*NC*	*PLC*	*7-178*
M2470	*Complement PLC positioning axis Z*	*NC*	*PLC*	*7-178*
M2471	*Complement PLC positioning axis 4*	*NC*	*PLC*	*7-178*
M2472	Complement – manual traverse X+	PLC	PLC	4-174
M2473	Complement – manual traverse X-	PLC	PLC	4-174
M2474	Complement – manual traverse Y+	PLC	PLC	4-174
M2475	Complement – manual traverse Y-	PLC	PLC	4-174
M2476	Complement – manual traverse Z+	PLC	PLC	4-175
M2477	Complement – manual traverse Z-	PLC	PLC	4-175
M2478	Complement – manual traverse 4+	PLC	PLC	4-175
M2479	Complement – manual traverse 4-	PLC	PLC	4-175
M2480	Acknowledgement "Gear change completed"	PLC	PLC	4-101
M2481	S-code acknowledgement	PLC	PLC	4-104
M2482	Acknowledgement of M-function	PLC	PLC	4-159
M2483	Acknowledgement T code (P code) with TOOL CALL	PLC	PLC	4-229
M2484	Acknowledgement T code (P code) with TOOL DEF	PLC	PLC	4-229
M2485	Status-display and sign of S-analogue for M03	PLC	PLC	4-96
M2486	Status-display and sign of S-analogue for M04	PLC	PLC	4-96
M2487	Status-display for M05 and spindle stop	PLC	PLC	4-96
M2488	NC-stop ("0" signifies stop)	PLC	PLC	4-174
M2489	Change direction of spindle rotation	PLC	PLC	4-96
M2490	Spindle rotation left (for gear change)	PLC	PLC	4-101
M2491	Spindle rotation right (for gear change)	PLC	PLC	4-101
M2492	Await open control loop X axis	PLC	PLC	4-92
M2493	Await open control loop Y axis	PLC	PLC	4-92
M2494	Await open control loop Z axis	PLC	PLC	4-92
M2495	Await open control loop axis 4	PLC	PLC	4-92

Marker	Function	Set	Reset	Page
M2496	Enable marker for the decoded M-code transfer in Markers M1900 to M1999	PLC	PLC	4-159
M2497	Activate the edge evaluation for PLC-inputs Rising-edge marker M1500 to M1659 Falling-edge marker M1700 to M1859	PLC	PLC	7-27
M2498	Release jog positioning	PLC	PLC	4-209
M2499	Open spindle control loop	PLC	PLC	4-106 4-118
M2500	Await open control loop axis 5	PLC	PLC	4-92
M2501	Activates spindle-speed MP3520.0 and direction of rotation from marker M2656	PLC	PLC	4-109
M2502	NC-STOP in all operating modes if stylus is deflected	PLC	PLC	4-178
M2503	Enable-marker for probe functions	NC	PLC	4-178
M2505	Actual – nominal value transfer axis 5	PLC	PLC	4-92
M2506	Reference end-position for axis 5	PLC	PLC	4-63
M2507	Open control loop axis 5	PLC	PLC	4-92
M2508	Status display M07, M08, M09 and MK	PLC	PLC	4-139
M2512	Jog positioning axis X+	PLC	PLC	4-203
M2513	Jog positioning axis X–	PLC	PLC	4-209
M2514	Jog positioning axis Y+	PLC	PLC	4-209
M2515	Jog positioning axis Y–	PLC	PLC	4-209
M2516	Jog positioning axis Z+	PLC	PLC	4-209
M2517	Jog positioning axis Z–	PLC	PLC	4-209
M2518	Jog positioning axis 4+	PLC	PLC	4-209
M2519	Jog positioning axis 4–	PLC	PLC	4-209
M2520	Jog positioning axis 5+	PLC	PLC	4-209
M2521	Jog positioning axis 5–	PLC	PLC	4-209
M2522	*Activate PLC-positioning axis 5*	*PLC*	*NC*	*7-178*
M2524	Manual traverse 5+	PLC	PLC	4-175
M2525	Manual traverse 5–	PLC	PLC	4-175
M2527	*Activate spindle orientation*	*PLC*	*NC*	*7-178*
M2528	Complement Jog positioning axis X+	PLC	PLC	4-209
M2529	Complement Jog positioning axis X–	PLC	PLC	4-209
M2530	Complement Jog positioning Axis Y+	PLC	PLC	4-209
M2531	Complement Jog positioning Axis Y–	PLC	PLC	4-209
M2532	Complement Jog positioning Axis Z+	PLC	PLC	4-209
M2533	Complement Jog positioning Axis Z–	PLC	PLC	4-209
M2534	Complement Jog positioning Axis 4+	PLC	PLC	4-209
M2535	Complement Jog positioning Axis 4–	PLC	PLC	4-209
M2536	Complement Jog positioning Axis 5+	PLC	PLC	4-209
M2537	Complement Jog positioning Axis 5–	PLC	PLC	4-209
M2538	*Complement PLC-Positioning Axis 5*	*NC*	*PLC*	*7-178*
M2540	Complement – manual traverse 5+	PLC	PLC	4-175
M2541	Complement – manual traverse 5–	PLC	PLC	4-175
M2543	*Complement Spindle-Orientation*	*NC*	*PLC*	*7-178*
M2544	Open control loop X axis	PLC	PLC	4-92
M2545	Open control loop Y axis	PLC	PLC	4-92
M2546	Open control loop Z axis	PLC	PLC	4-92
M2547	Open control loop axis 4	PLC	PLC	4-92
M2548	Reset of accumulated distance for lubrication X axis	PLC	PLC	4-20

Marker	Function	Set	Reset	Page
M2549	Reset of accumulated distance for lubrication Y axis	PLC	PLC	4-20
M2550	Reset of accumulated distance for lubrication Z axis	PLC	PLC	4-20
M2551	Reset of accumulated distance for lubrication of axis 4	PLC	PLC	4-20
M2552	Actual – nominal value transfer X axis	PLC	PLC	4-92
M2553	Actual – nominal value transfer Y axis	PLC	PLC	4-92
M2554	Actual – nominal value transfer Z axis	PLC	PLC	4-92
M2555	Actual – nominal value transfer 4 axis	PLC	PLC	4-92
M2556	Reference end-position for X axis	PLC	PLC	4-63
M2557	Reference end-position for Y axis	PLC	PLC	4-63
M2558	Reference end-position for Z axis	PLC	PLC	4-63
M2559	Reference end-position for axis 4	PLC	PLC	4-63
M2560 to M2589	*BCD numerical values for PLC positioning, tool number, spindle orientation and Q parameters*	*PLC*	*PLC*	*7-178*
M2597	*Tool number: output mode* *0 = binary 1 = BCD*	*PLC*	*PLC*	*7-178*
M2600	Tool change sequence for changing from special tools to normal tool	*PLC*	*PLC*	4-232
M2601	Return special tool to its original pocket in spite of flexible pocket coding	*PLC*	*PLC*	4-226 4-232
M2608	Status-display M03, M04, M05 inverse and S-analogue output = 0V	PLC	PLC	4-139
M2609	Status-display M08, M09 inverse	PLC	PLC	4-139
M2610	T-character for tool status is displayed inverted	PLC	PLC	—
M2611	Confirm the transfer with FN19	PLC	PLC	7-19
M2612	Do not update pocket number in pocket table	PLC	PLC	4-148 4-232
M2613	Reset of accumulated distance for lubrication axis 5	PLC	NC	4-20
M2614	Read-in inhibited (after acknowledgement of the M/S/T/Q strobes the subsequent NC blocks are not executed)	PLC	PLC	—
M2615	Repeat reference mark evaluation for spindle-orientation	PLC	NC	4-108
M2624	Limit switch X+	NC	NC	4-18
M2625	Limit switch X–	NC	NC	4-18
M2626	Limit switch Y+	NC	NC	4-18
M2627	Limit switch Y–	NC	NC	4-18
M2628	Limit switch Z+	NC	NC	4-18
M2629	Limit switch Z–	NC	NC	4-18
M2630	Limit switch 4+	NC	NC	4-18
M2631	Limit switch 4–	NC	NC	4-18
M2632	Limit switch 5+	NC	NC	4-18
M2633	Limit switch 5–	NC	NC	4-18
M2656	Spindle-orientation from stop 0 = Orientation with M03 1 = Orientation with M04	PLC	PLC	4-106
M2657	Status display M07, M08, M09 and MK	PLC	PLC	4-139
M2688	No monitoring X axis	PLC	PLC	4-83
M2689	No monitoring Y axis	PLC	PLC	4-83
M2690	No monitoring Z axis	PLC	PLC	4-83
M2691	No monitoring axis 4	PLC	PLC	4-83
M2692	No monitoring axis 5	PLC	PLC	4-83

Marker	Function	Set	Reset	Page
M2704	Activate PLC-positioning X axis	PLC	NC;PLC	4-35
M2705	Activate PLC-positioning Y axis	PLC	NC;PLC	4-35
M2706	Activate PLC-positioning Z axis	PLC	NC;PLC	4-35
M2707	Activate PLC-positioning axis 4	PLC	NC;PLC	4-35
M2708	Activate PLC-positioning axis 5	PLC	NC;PLC	4-35
M2712	Activate PLC-positioning for spindle-orientation	PLC	NC	4-106
M2713	Activate the transfer of the value from D528 to the Q-Parameter defined in W516	PLC	NC	4-155
M2716	Strobe marker for datum correction	PLC	NC	4-221
M2717	Geometry of the tool from W264	PLC	NC	4-228
M2719	*Deactivate the TNC 355 mode*	*PLC*	*PLC*	7-175
M2720	Activate/deactivate function "Free rotation" of rotary axis	PLC	NC	4-136
M2800 to M2808	*Key code*	*PLC*	*PLC*	*7-178*
M2809	*Activate the numerical value transfer from PLC to NC*	*PLC*	*PLC*	*7-178*
M2813	Activate the key from W516	PLC	NC	4-165
M2814	Activation of gear range and rpm via PLC	PLC	NC	4-101
M2815	Flashing PLC error message	PLC	PLC	4-142
M2816	Select range	PLC	PLC	4-17
M2817	Select range	PLC	PLC	4-17
M2818	*Q-Number (msb)*	*PLC*	*PLC*	*7-178*
M2819	*Activate datum correction*	*PLC*	*NC*	*7-178*
M2824	Activation of the selected range (M2816/M2817)	PLC	NC	4-18
M2826	Suppress handwheel pulses	*PLC*	*PLC*	*4-197*
M2827	Suppress EMERGENCY STOP, open the control loop, stop NC	PLC	PLC	4-92 4-120
M2830	Stop NC and open the control loop	PLC	PLC	4-92
M2832 to M2839	*Key code of the activated inhibited key*	*NC*	*NC*	*7-178*
M2854 to M2875	Inhibit keys	PLC	PLC	4-166 4-167
M2876	Inhibit alpha keyboard	PLC	PLC	4-165
M2877	Inhibit soft keys below the screen	PLC	PLC	4-165
M2878	Inhibit switch-over keys next to screen	PLC	PLC	4-165
M2880 to M2923	Inhibit keys	PLC	PLC	4-146
M2924 to M3023	Error messages	PLC	NC PLC	4-142
M3168	Overflow during multiplication	NC	PLC	7-68 7-85 7-123
M3169	Division by 0	NC	PLC	7-69 7-85 7-123

Marker	Function	Set	Reset	Page
M3170	MODULO wrongly executed	NC	PLC	7-70 7-86 7-123
M3171	Error status for PLC modules	NC	PLC NC	7-123 7-136ff.
M3172	Reserved for errors, which the PLC programmer would like to trap	PLC	PLC	7-123
M3200 to M3263	*Values from MP4310.3 to MP4310.6*			*7-179*

2 List of words

Words	Function	Page
W256	G-Code for S analogue	4-101
W258	S-Code	4-104
W260	Code for M-Function	4-159
W262	Pocket number with MP7480 = 3 or 4 Tool number with MP7480 = 1 or 2	4-247
W264	Tool number with MP7480 = 3 or 4	4-247
W270	Help file line number –1 = No help file selected –2 = No valid numerical value 0 to 9999 = Line number	4-156
W272	Operating mode 1 = Manual operation 2 = Electronic handwheel 3 = Positioning with manual entry 4 = Program run/single block 5 = Program run/full sequence 7 = Pass over reference point	4-63 4-164
W274	Key-code for the operated, inhibited key. Signal via M2182	4-164
D276	Value of the code-number most recently entered by MOD	4-153
D280	1st number value from FN19	7-19
D284	2nd number value from FN19	7-19
W320	Nominal spindle speed	4-95
W322	Actual spindle speed	4-95
D356	Programmed spindle speed	4-95
D360	Programmed feed rate (NC → PLC)	–
D388	Current feed rate	–
W464	Analogue input 0 2nd I/O extension	4-206
W466	Analogue input 1 2nd I/O extension	4-206
W468	Analogue input 2 2nd I/O extension	4-206
W470	Analogue input 3 2nd I/O extension	4-206
W472	Thermistor input 0 2nd I/O extension	4-206
W474	Thermistor input 1 2nd I/O extension	4-206
W476	Thermistor input 2 2nd I/O extension	4-206
W478	Thermistor input 3 2nd I/O extension	4-206
W492	%-factor – spindle override (NC → I/O)	4-99
W494	%-factor – feed rate override (NC → I/O)	4-138
W496	Analogue input 0 1st I/O extension	4-206
W498	Analogue input 1 1st I/O extension	4-206
W500	Analogue input 2 1st I/O extension	4-206
W502	Analogue input 3 1st I/O extension	4-206
W504	Thermistor input 0 1st I/O extension	4-206
W506	Thermistor input 1 1st I/O extension	4-206
W508	Thermistor input 2 1st I/O extension	4-206
W510	Thermistor input 3 1st I/O extension	4-206

Words	Function	Page
W516	Word with multiple function Key code for simulating TNC keys activated by M2813 Number of the Q-parameter to be overwritten (Q100 to Q107 = 0 to 7)	4-154 4-164
B518	0 = "Free-rotation" function cancelled 8 = "Free-rotation" function for axis 4 16 = "Free-rotation" function for axis 5	4-136
B519	Definition of the direction of traverse	4-136
B520	Axis specific feed rate release Bit 0 +1 Feed rate enable in the X axis +0 No feed rate enable X in the axis Bit 1 +2 Feed rate enable in the Y axis +0 No feed rate enable in the Y axis Bit 2 +4 Feed rate enable in the Z axis +0 No feed rate enable in the Z axis Bit 3 +8 Feed rate enable in the 4th axis +0 No feed rate enable in the 4th axis Bit 4 +16 Feed rate enable in the 5th axis +0 No feed rate enable in the 5th axis	4-90
W522	Monitoring functions suppressed if PLC input from MP4130 is activated	4-83
D528	Value to be transferred to the Q parameter Axis X datum correction Position of X axis [1/10 000 mm]	4-154 4-222 4-35
D532	Axis Y datum correction Position Y axis	
D536	Axis Z datum correction Position Z axis	
D540	Axis IV datum correction Position 4 axis	
D544	Axis V datum correction Position 5 axis	
W560	Feed X axis [mm/min]	4-35
W562	Feed Y axis	4-35
W564	Feed Z axis	4-35
W566	Feed axis 4	4-35
W568	Feed axis 5	4-35
W576 to W584 W576 W578 W580 W582 W584	Lag tracking error compensation (compensation speed from MP4070) Input range: +32 767 to –32 768 [$\frac{1}{10}$ µm] Axis X Axis Y Axis Z Axis 4 Axis 5	4-32
D592	Nominal position for spindle orientation (Strobe M2712)	4-106
D596	Rapid traverse from PLC	4-70
W754	%-factor of feed-override "feed rotation" (0 to 300 %)	4-136
D756	Set spindle-speed from the PLC; programmed spindle speed	4-101

Words	Function	Page
W764	%-factor – spindle override (PLC → NC)	4-99
W766	%-factor - feed rate override (PLC → NC)	4-138
D768 to D956	Value from MP4210.0 to MP4210.47	7-21 7-22
W960 to W968	Value from MP4220.0 to MP4220.4	7-22
W976 to W988	Value from MP 4310.0 to MP4310.6	7-22
W1008	S-code for minimum rpm	4-104

PLC Programming — Contents

1 PLC functions

The integrated PLC in the TNC contains its own Text Editor for creating the list of instructions for the PLC program. Commands and comments are entered via the ASCII keyboard on the control panel (see section "Programming and editing files").

The functions TRACE and TABLE, as well as a Syntax check on entering the PLC commands and a logical test with the Function COMPILE were introduced to make it easier to find faults in the PLC program (see section "Test functions").

10 ms are available for a PLC run. Up to 8 000 logic commands, equivalent to 64 KB, can be processed within this period (executable memory). A new PLC run commences every 20 ms in the TNC 415 and every 24 ms in the TNC 407 (PLC cycle time), i.e. every 20 ms or 24 ms the inputs are read and outputs are set.

1.1 Select PLC operation

PLC operation covers all functions for creating and testing the PLC programs, for creating the PLC error messages and the dialogue texts for OEM cycles, the Help files and compensation lists for non-linear axis error correction.

It can be selected as follows, using the code number **807 667**:

Select NC operating mode
"Programming and editing"

↓

Press MOD
key

↓

Enter code number 807 667

↓

ENT key

↓

PLC operation is selected
(Main menu)

PLC operation can be stopped by pressing the soft key END or the END key.

Provided the TNC has not been switched off and the PLC operation has already been selected once with the code number, it is possible to return to PLC operation quite simply, without using the code number, by means of the MOD key and the soft key PLC EDIT.

1.2 PLC – Main menu

After entering the code number (or soft key PLC EDIT) the following screen display will appear (main menu):

PROCESSING TIME:	MAXIMUM	XXX%
	CURRENT	XXX%
CODE LENGTH		XX KBYTE
RUNTIME PGM		XXXX XXXX.XXX
EDITOR PGM		XXXX XXXX.XXX

EDIT	TABLE	TRACE	COMPILE	OUTPUT BINARY CODE 0001	SELECT .DI1/.ER1 FILES	MP EDIT	END
Select edit mode	Select table M/I/O/T/C	Select TRACE function	Compile PLC program	Binary output of PLC program	Activate dialogue or error message file	Call machine parameter list	Quit PLC Mode

Processing time
The PLC processing time (time for a PLC run) is given in % of the maximum time of 10 ms, whereby 10 ms is equivalent to 200 %

Processing time: Maximum
The longest occurring run time for the current executable program is displayed in % . If this time exceeds the absolute maximum of 10 ms, then the flashing error message "PLC-program error 53" will be displayed.

Current (processing time)
The time for the latest PLC run, displayed in % .

Code length
This is the length of the compiled executable program in KB.

Executable program
The last compiled PLC program is displayed here. If you are working from the RAM (MP4010) then the desired program can be selected using the PGM NAME and SELECT soft key and must be compiled with the COMPILE function (soft key).
A program which was already selected as an executable program before switching on will be compiled automatically. The PLC program is only active after compilation!

Editor program
The name of a file which was selected with the soft key SELECT can be seen in the line PGM IN EDIT MEM.

If no executable program has been activated and no Editor program has been selected, then the relevant name fields will be empty!

The various PLC functions can be accessed from inside the Main menu.

1.3 File management

All the EPROM and RAM files can be listed by operating the PGM NAME key on the keyboard unit and the soft key SHOW ALL. The following functions appear at the same time in the soft-key row:

PAGE ⇧	PAGE ⇩	SELECT	COPY ABC ⇒ XYZ	MODIFY WINDOW	SELECT TYPE	WINDOW	END
Scroll		Select file	Copy	Modify display for selected window	Select file type	Switch to display with two windows	Return to main menu

PAGE ⇧	PAGE ⇩	PROTECT	UNPROTECT	RENAME ABC = XYZ	CONVERT ABC ⇒ XYZ		END
Scroll		Protect program	Cancel program steps	Rename	Change file type		Return to main menu

The following soft-key row is displayed when the **MODIFY WINDOW** soft key is operated.

SELECT TYPE	SELECT TNC	SELECT TNC/EPROM	SELECT EXT				END
Select file type	List from RAM	List from EPROM	List across data interface				

You can choose whether the file list from the RAM, the EPROM or across the data interface will be displayed in the selected window. You can also define the type of files to be listed using the SELECT TYPE soft key.

Use the **WINDOW** soft key to toggle between the display with one and the display with two windows. The display with two windows is useful for the input/output of data and for copying data from the EPROM. The following soft-key rows are offered with the two-window display.

PAGE ⇧	PAGE ⇩	TRANSFER	TRANSFER	TRANSFER ?	SELECT TYPE	WINDOW	END
Scroll		Copy file from window to window			Select file type	Switch to display with one window	Return to main menu

PAGE ⇧	PAGE ⇩	CONVERT ABC ⇨ XYZ	MODIFY WINDOW				END
Scroll		Change file type	Modify display for selected window				

The soft key functions are only active for the selected window. Use the arrow keys to toggle between the two windows.

Operate the **SELECT TYPE** soft key to select a different file type. The following soft-key rows are shown:

SHOW ALL	.PLC FILES	.ER1 FILES	.ERE FILES	.DI1 FILES	.DIE FILES	SHOW .A	END
Display all file types	PLC programs	PLC error messages, first language	PLC error messages, English	Dialogues for OEM cycles, first language	Dialogues for OEM cycles, English	ASCII files	

.HLP FILES	.BIN FILES	.COM FILES	.CMA FILES				END
Help files	Binary files	Compensation tables	Compensation assignment				

Operate the **CONVERT** soft key to change the type of the selected file. The following soft-key row is shown:

CONV. PLC LISTING	CONV. PLC CROSSREF	CONV. PLC LISTING+ CROSSREF	CONV. PLC UNFORM.	CONVERT → .A			END
PLC program with block number	Cross-reference list	PLC program with block number and cross-reference list	Unformatted				

1.4 Text Editor

A file which is loaded into the editor (EDITOR PGM) can be displayed on the screen with the soft key EDIT from the main menu. If it is a new file then the screen will be empty, apart from the status display for the editor.

The status display contains the following information:

FILE:	Name of the file
LINE:	Current line number of the cursor (bright field).
COLUMN:	Current column number of the cursor (bright field).
INSERT or OVERWRITE:	Use the soft key to choose between - insert mode (INSERT) and - overwrite mode (OVERWRITE).

See User's Manual for a detailed description of the Text Editor.

1.5 File output via data interface

The EXT key is used to initiate the output of data across the data interface (see chapter entitled "Data Interface").

PLC programs, PLC error messages and dialogues for OEM cycles can be transferred as ASCII text via the data interface (see chapter "Data interface").

PLC programs, PLC error messages, dialogues for OEM cycles and the OEM cycles themselves can also be given out as binary code for the creation of EPROMs.

After operating the soft key **"OUTPUT BINARY CODE 0001"** it is possible to collect the files that are to be transferred in a list. The required ID number for the EPROM which is to be created must be entered in the first line of this table. The preset names in the list can be overwritten.

To start data transfer, the user must press the EXT key.

The HEIDENHAIN service department is at your disposal for further information about creating EPROMs.

A table could look like this:

```
IDENT   123 456 99
LOAD    PLC0CODE.PLC
LOAD    PLC0CODE.ER1
LOAD    PLC0CODE.ERE
LOAD    PLC0CODE.DI1
LOAD    PLC0CODE.THE
LOAD    999 999 69.H (OEM-cycle)
```

If the PLC program is to be transferred to a 2MB EPROM, the command "SIZE 2MB" must be entered in the table (as of software 280 58, 280 54, 280 56). The PLC main program is designated with /M. Help files (.HLP) and machine parameters lists (.MP) can also be transferred into the EPROM.

1.6 Test functions for the PLC program

1.6.1 TRACE functions

The TRACE function makes it possible to check the logical states of the markers, inputs, outputs, timers and counters as well as to test the contents of Byte, Word and Doubleword.

These functions are available from the Main menu by using the soft key TRACE.

The list of instructions for the compiled program will then be displayed. In addition, the contents of the Operand and Accumulator for each line of the program are shown in HEX or decimal code. Every active command in the LIST is identified with a "*". The cursor keys or the GOTO function can be used to display the required portion of the program.
The soft keys now have the following meaning:

SELECT M/I/O/T/C	LOGIC DIAGRAM	FIND	HEX ⇕ DECIMAL	START STOP DISPLAY	START TRACE	STOP TRACE	END

HEX ↔ DECIMAL

The contents of the Operands and the Accumulators can be displayed either in decimal or HEX code. The soft key can be used to toggle the display.

START/STOP DISPLAY
The dynamic display of the Operands and the Accumulators can be frozen with STOP DISPLAY.
On operating START DISPLAY they will again be continuously updated.

Further soft key functions are described in the next section.

1.6.2 Logic diagram

The logical states of up to 16 Operands (M, I, O, T, C) can be shown graphically and simultaneously on the VDU screen. 1 024 PLC runs can be recorded.
The selection is made by soft key SELECT M/I/O/T/C, which makes it possible to create a table with the required Operands. The individual positions in the table are determined by dialogue. Wrong entries can be erased with the DEL key. For each Operand a trigger condition can be entered. 512 states are recorded before and 512 after a trigger event. The following trigger conditions are possible:

"1" => Record when Operand is logical "1"
(Triggering on positive edge)

"0" => Record when Operand is logical "0"
(Triggering on negative edge)

If no trigger condition is wanted then confirm with NO ENT.
If no trigger condition is entered for any of the Operands then the Operand states will be continuously recorded and the last 1024 remain in memory.

A recording commences with START TRACE and is ended either with STOP TRACE or when the trigger event takes place. During recording of the logical states, the message PCTR will blink in the status window. The blinking will stop when the recording is finished.

The cursor keys can be used to select the desired range in the TRACE buffer.

After operating the soft keys SELECT M/I/O/T/C the soft keys will have the following meaning:

TRACE IN-CODE	LOGIC DIAGRAM				START TRACE	STOP TRACE	END
Return to instruction list display with operands and accumulator content	Display logic diagrams of the selected operands				Start TRACE recording	Stop TRACE recording	Return to main menu

1.6.3 TABLE function

The table of markers, inputs, outputs, timers and counters can be dynamically displayed on the VDU screen by using the soft key TABLE from within the Main menu. The selection, as well as the set and reset of the markers, inputs, outputs, timers and counters is performed by soft keys.
The cursor keys or the GOTO key are used to select the positions within the table. Return to the Main menu by pressing the END key.

1.6.4 COMPILE function

A PLC program is only transferred to the executable memory when it is compiled (see PLC Main menu). After pressing the soft key COMPILE a directory of the available PLC programs is displayed. The program which is to be compiled must be chosen with SELECT.
During compilation, error messages may be displayed which result from programming errors (see section: "Program creation").

2 Program creation

The PLC program can be created directly on the HEIDENHAIN contouring controls. For this, the PLC Editor must be called with the code number **807 667** (see section "PLC functions").

HEIDENHAIN can supply the PLC.EXE PLC development software for creating PLC programs on your personal computer. Further particulars can be obtained from your HEIDENHAIN customer service.

2.1 Program structure

2.1.1 Command

A command is the smallest unit in a PLC program. It consists of the operation portion and the Operand portion.

A	I	28	;Comment
↓	↓	↓	
Operation (A, AN...)	Operand designation (I, O, M, T, C, B, W, D, K)	Operand address; constant value	
	Operand		

The operation describes the function which is to be performed. It explains what is done with the Operands.

The Operand shows what is to be operated on. It consists of the Operand abbreviation and a parameter (Address). Register and memory contents can be gated, erased and loaded by using PLC commands.

Both Bit and Word processing are possible. In Word processing it is possible to address memory contents with a length of 8 Bits (Byte), 16 Bits (Word) or 32 Bits (Doubleword) (see section "Commands").

2.1.2 Module technique

It is good practice to make the maintenance of the PLC programs easier by creating the program with the most transparent structure possible. This can be best achieved by dividing the PLC program into individual modules (structured programming).

Only the most important PLC functions should be programmed in the main routine.

Individual PLC functions such as spindle orientation and key simulation are programmed in their own modules.

```
0       L M2719
1       SN M2719     ;Activate the strobes for Word processing
2       L M2497
3       SN M2497     ;Activate the edge evaluation
4       L M2496
5       SN M2496     ;Activate transfer of decoded M codes
                     ;(M1900 to M1999)
.       .            .
.       .            .
.       .            .
20      L M1919      ;M Function M19
21      A M2045      ;Change signal for M function
22      AN M12       ;Spindle orientation already  active?
23      CMT 180      ;Spindle orientation
24      LM 2182      ;Disabled key operated?
25      CMT 31       ;Yes, then call key simulation
.       .            .
.       .            .
.       .            .
150     EM           ;End main program
151     LBL 31       ;Key simulation
.       .            .
.       .            .
.       .            .
200     EM
201     LBL 180      ;Spindle orientation
.       .            .
.       .            .
.       .            .
259     EM
.       .            .
.       .            .
.       .            .
```

Error conditions in the machine should be interrogated in the PLC program and a plain language error message should be displayed on the VDU screen. See chapter "Machine integration", section "Display and operation" and chapter "PLC programming", section "Modules".

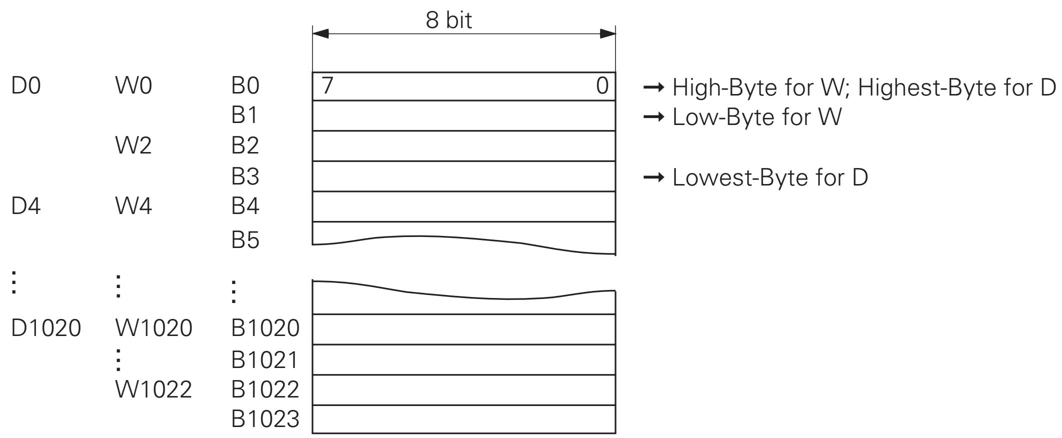
2.2 Address allocation

2.2.1 Operand directory

Operand	Abbreviation	Address range
Marker	M (Marker)	0 to 3023
Input	I (Input)	I0 to I31; I128 to I152; I64 to I126 (first extension board); I192 to I254 (second extension board)
Output	O (Output)	O0 to O30; O32 to O62(first extension board); O64 to O94 (second extension board);
Counter	C (Counter)	Set counter : C0 to C31 Counter contents: C48 to C79 Release count pulse: C96 to C127
Timer	T (Timer)	Timer start: T0 to T47 Timer running: T48 to T95
Byte	B (Byte)	0 to 1023 (8 Bit)
Word	W (Word)	0 to 1022 (16 Bit)
Doubleword	D (Doubleword)	0 to 1020 (32 Bit)
Constant	K	- 2 147 483 647 to + 2 147 483 647

2.2.2 Addressing the memory

The memory for the Operands B (8 Bit), W (16 Bit), D (32 Bit) is only 8 Bit wide. Since the Operands can be 8,16 or 32 Bit wide, an overlap of the memory areas will occur, which must be taken into account in addressing the memory.



In Byte addressing every address from 0 to 1023 is accessible. In Word addressing, every second address from 0 to 1022 is accessible and in Doubleword addressing every fourth from 0 to 1020.

The address parameter gives the High Byte for a Word Address (W) , or the Highest Byte for a Doubleword address (D).

Markers M1000 to M1999 and Bytes B0 to B127 are non-volatile, i.e. the contents of this memory are maintained when the supply voltage is switched off.
After entry of the code number **531 210** the markers M1000 to M1999 and Bytes B0 to B127 are erased.

B0 to B127	Freely available, not deleted with RESET
B128 to B255	Freely available, deleted with RESET
B256 to B511	Data transfer NC → PLC
B512 to B767	Data transfer PLC → NC
B768 to B1023	Machine parameters

2.3 Data transfer NC/PLC

The information exchange between PLC and NC is conducted by markers, Bytes, Words and Doublewords. The function of the individual markers, Bytes, Words and Doublewords is fixed.

Either numerical data or general data (controlled by strobes) can be transferred.

2.3.1 Number transfer

The Parameter function FN19 can be used to transfer two numbers to the PLC. The transferred values are deposited in the Doublewords D280 and D284.

During the transfer the marker M2149 is set by the NC. The transfer must be confirmed by the PLC by setting marker M2611.

Marker M2150 determines whether dimensions are in mm or inches.

The transferred value is deposited as an integer number in units of 1/10 000.

Marker	Function	Set	Reset
M2149	Transfer with FN19 active	NC	NC
M2611	Confirm the transfer with FN19	PLC	PLC
M2150	Dimensional unit for transfer with FN19 0 = mm; 1 = inch	NC	NC

Address	Function
D 280	1st Number value from FN 19
D 284	2nd Number value from FN 19

Example:

NC program

```
0 BEGIN PGM 5 MM
.
.
.
11 FN19:PLC =+20.25/+100
.
.
.
33 END PGM 5 MM
```

PLC program

```
.
.
.
LN M2149                        ;Strobe signal/Transfer
R M2611                         ;Acknowledgement
.
.
.
L M2149                         ;Strobe signal/Transfer
CMT 199
.
.
.
LBL 199
L D280                          ;1st value from FN19
= D120                          ;Internal buffer 1st value
.
.
.
L D284                          ;2nd value from FN19
= D124                          ;Internal buffer 2nd value
L M2149                         ;Change signal for transfer
S M2611                         ;Acknowledgement
```

After the program has been performed, the following data will be in the Doublewords D120 and D124 :

D120 =   202 500
D124 = 1 000 000

M2150 = 0, since program no. 5 was programmed in mm

The number transfer from the PLC to the machining program is carried out by the Q parameters Q100 to Q107, i.e. the Q Parameters Q100 to Q107 can be overwritten by the PLC. The required numerical value is entered in doubleword D528 and the Q parameter number (0-7) is defined in Word W516. The transfer is activated by the strobe marker M2713. The Q parameter values are transferred with the next M/S/T strobe.

Marker	Function	Set	Reset
M2713	Activate the Q Parameter transfer to the NC Data from D528 Q No. from W516	PLC	NC

Address	Function
D 528	Doubleword with multiple function, in this instance data for transfer from the PLC to the NC (0–7, Q number from W516, strobe marker M2713)
W516	Q number for numerical value transfer from PLC to NC (transfer of the value from D528, strobe marker M2713)

122 different machine parameters are reserved for data transfer in the PLC. MP4210.x, MP4220.x and MP4310.x are stored in PLC words. The contents of MP4230.x and 4231.x must be called by module 9032. For example, PLC positioning, datum shifts, feed rates for PLC positioning or coding for the release of certain PLC functions can be filed in these machine parameters. These numerical values are evaluated in the PLC program.

Although the TNC 407 / TNC 415F / TNC 425E accept entry values to 0.0001 mm (or °), the controls round these values to 0.001 mm (or °) for their calculations.

Address	Function
D768	Value from MP4210.0
D772	Value from MP4210.1
D776	Value from MP4210.2
D780	Value from MP4210.3
D784	Value from MP4210.4
D788	Value from MP4210.5
D792	Value from MP4210.6
D796	Value from MP4210.7
D800	Value from MP4210.8
D804	Value from MP4210.9
D808	Value from MP4210.10
D812	Value from MP4210.11
D816	Value from MP4210.12
D820	Value from MP4210.13
D824	Value from MP4210.14
D828	Value from MP4210.15
D832	Value from MP4210.16
D836	Value from MP4210.17
D840	Value from MP4210.18
D844	Value from MP4210.19
D848	Value from MP4210.20

Address	Function
D852	Value from MP4210.21
D856	Value from MP4210.22
D860	Value from MP4210.23
D864	Value from MP4210.24
D868	Value from MP4210.25
D872	Value from MP4210.26
D876	Value from MP4210.27
D880	Value from MP4210.28
D884	Value from MP4210.29
D888	Value from MP4210.30
D892	Value from MP4210.31
D896	Value from MP4210.32
D900	Value from MP4210.33
D904	Value from MP4210.34
D908	Value from MP4210.35
D912	Value from MP4210.36
D916	Value from MP4210.37
D920	Value from MP4210.38
D924	Value from MP4210.39
D928	Value from MP4210.40
D932	Value from MP4210.41
D936	Value from MP4210.42
D940	Value from MP4210.43
D944	Value from MP4210.44
D948	Value from MP4210.45
D952	Value from MP4210.46
D956	Value from MP4210.47
W960	Value from MP 4220.0
W962	Value from MP 4220.1
W964	Value from MP 4220.2
W966	Value from MP 4220.3
W968	Value from MP 4220.4
W976	Value from MP 4310.0
W978	Value from MP 4310.1
W980	Value from MP 4310.2
W982	Value from MP 4310.3
W984	Value from MP 4310.4
W986	Value from MP 4310.5
W988	Value from MP 4310.6

MP4210.0 to MP4210.47	Set a number in the PLC Entry : –99 999.9999 to +99 999.9999
MP 4220	Machine parameter with multiple function Entry : 80 to 30 000 - Set a number in the PLC. In Word range W960 to W968. - Feed rate for re-approaching the contour

MP4220.0 X axis
MP4220.1 Y axis
MP4220.2 Z axis
MP4220.3 axis 4
MP4220.4 axis 5

MP4310.0 to MP4310.6	Set a number in the PLC, in the Word range W976 to W988 Entry : 0 to 65 535
MP4230.0 to MP4230.31	Set a number in the PLC (Module 9032) Entry : - 99 999.0000 to +99 999.9999
MP4231.0 to MP 4231.31	Set a number in the PLC (Module 9032) Entry : - 99 999.0000 to +99 999.9999

2.3.2 General data transfer (Strobes)

The NC transfers general data to the PLC, e.g. M functions, S word, T word, G code.

The transfer is controlled by strobes.

Example:

When an M function is output, the NC sets the strobe signal M2045. After evaluating the M function, the PLC sets the acknowledge marker M2482. The PLC must reset M2482 otherwise no further strobes can be transferred by the NC.

2.4 Timers and counters

2.4.1 Timers

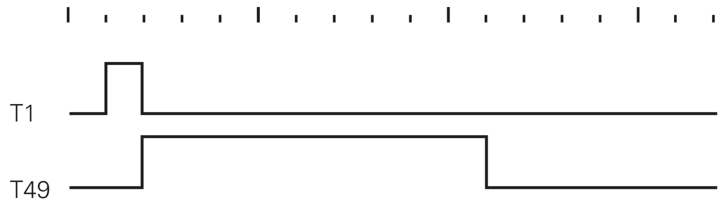
48 timers are available in the PLC. These 48 timers are controlled by special markers with the abbreviation symbol T. The time period for the timer is defined in the machine parameter MP4110.X. The time unit corresponds to the PLC cycle time (TNC 415: 20 ms; TNC 407: 24 ms).

The timers are started by setting the markers T0 to T47 which also sets the timer to the value from MP4110.X. This activation may only be performed for a single PLC run, as otherwise the timers will be restarted by every succeeding run.

The markers T48 to T95 (timer running) will remain set for the period defined in the machine parameters.

Example:

Start of Timer 1
Period in MP4110.1 = 9 (PLC cycles)



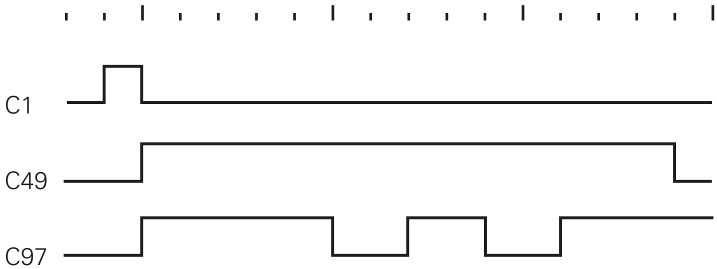
Timer start	Timer running	Machine parameter
T0	T48	MP4110.0
T1	T49	MP4110.1
T2	T50	MP4110.2
T3	T51	MP4110.3
T4	T52	MP4110.4
T5	T53	MP4110.5
T6	T54	MP4110.6
T7	T55	MP4110.7
T8	T56	MP4110.8
T9	T57	MP4110.9
T10	T58	MP4110.10
T11	T59	MP4110.11
T12	T60	MP4110.12
T13	T61	MP4110.13
T14	T62	MP4110.14
T15	T63	MP4110.15
T16	T64	MP4110.16
T17	T65	MP4110.17
T18	T66	MP4110.18
T19	T67	MP4110.19
T20	T68	MP4110.20
T21	T69	MP4110.21
T22	T70	MP4110.22
T23	T71	MP4110.23
T24	T72	MP4110.24
T25	T73	MP4110.25
T26	T74	MP4110.26
T27	T75	MP4110.27
T28	T76	MP4110.28
T29	T77	MP4110.29
T30	T78	MP4110.30
T31	T79	MP4110.31
T32	T80	MP4110.32
T33	T81	MP4110.33
T34	T82	MP4110.34
T35	T83	MP4110.35
T36	T84	MP4110.36
T37	T85	MP4110.37
T38	T86	MP4110.38
T39	T87	MP4110.39
T40	T88	MP4110.40
T41	T89	MP4110.41
T42	T90	MP4110.42
T43	T91	MP4110.43
T44	T92	MP4110.44
T45	T93	MP4110.45
T46	T94	MP4110.46
T47	T95	MP4110.47

Entry range for MP4110.X: 0 to 65 535 [PLC cycles] (TNC 415: 20 ms, TNC 407: 24 ms)

2.4.2 Counters

32 counters are available in the PLC. Each of these 32 counters is controlled by special markers with the abbreviation symbol C. After setting a marker from the range C0 to C31 the counter is loaded with the value from machine parameter MP4120.X. The marker range C48 to C79 indicates whether the count has been completed or not. The marker range C96 to C127 is used to start the counter (counter release pulse).

Example: Logic diagram for counter C1
Contents of machine parameter MP4120.1 = 10 (PLC cycles)



C0	C48	C96	MP4120.0
C1	C49	C97	MP4120.1
C2	C50	C98	MP4120.2
C3	C51	C99	MP4120.3
C4	C52	C100	MP4120.4
C5	C53	C101	MP4120.5
C6	C54	C102	MP4120.6
C7	C55	C103	MP4120.7
C8	C56	C104	MP4120.8
C9	C57	C105	MP4120.9
C10	C58	C106	MP4120.10
C11	C59	C107	MP4120.11
C12	C60	C108	MP4120.12
C13	C61	C109	MP4120.13
C14	C62	C110	MP4120.14
C15	C63	C111	MP4120.15
C16	C64	C112	MP4120.16
C17	C65	C113	MP4120.17
C18	C66	C114	MP4120.18
C19	C67	C115	MP4120.19
C20	C68	C116	MP4120.20
C21	C69	C117	MP4120.21
C22	C70	C118	MP4120.22
C23	C71	C119	MP4120.23
C24	C72	C120	MP4120.24
C25	C73	C121	MP4120.25
C26	C74	C122	MP4120.26
C27	C75	C123	MP4120.27
C28	C76	C124	MP4120.28
C29	C77	C125	MP4120.29
C30	C78	C126	MP4120.30
C31	C79	C127	MP4120.31

MP4120.0 to MP4120.31	Preset value for counters C0 to C31 Entry range: 0 to 65 535 [PLC cycles]

2.5 Edge evaluation of the PLC inputs

The edge evaluation for the PLC inputs can be activated by marker M2497. An automatic edge evaluation cannot be activated for the PLC inputs I192 to I254 on the second PLC I/O expansion board.
Edge evaluation means that if the signal at the PLC input changes, a certain marker will be set for the duration of a PLC run. If marker M2497 is set, the following markers will be set if the signals change at the PLC inputs.

Marker for rising edges at the PLC inputs:

Marker	PLC inputs
M1500 to M1531	I0 to I31
M1564 to M1626	I64 to I126 (first extension board)
M1628 to M1652	I128 to I152

Marker for falling edges at the PLC inputs:

Marker	PLC inputs
M1700 to M1731	I0 to I31
M1764 to M1826	I64 to I126 (first extension board)
M1828 to M1852	I128 to I152

Marker	Function	Set	Reset
M2497	Activate the edge evaluation for PLC inputs Rising-edge marker M1500 to M1652 Falling-edge marker M1700 to M1852	PLC	PLC

2.6 EPROM creation

Machine parameter MP4010 selects whether the PLC program is run from the RAM area of the control or the EPROM area.

During the creation and test of the PLC program the control should operate from the RAM area. HEIDENHAIN recommends that an EPROM is created for the PLC program before delivering the machine to the customer (see also chapter "Introduction").

MP4010	PLC program from RAM or from EPROM Entry: 0 = EPROM operation 1 = RAM operation

A PC and the MEGA-PROMMER-Software (Version 2.12.or later) are necessary for programming the PLC EPROMS.

The PLC files from the RAM area in the control can be transferred to the PC via the data interface (see chapter "Data interface").The output is initiated from the control with the soft key "BINARY OUTPUT" (see section "PLC functions").

The PLC files are then transferred to the PC in the Motorola EXORMAX S3 format. With the MEGA-PROMMER these data can be read in the MOX format and the EPROM programmed.

Please contact the HEIDENHAIN customer service if you have further questions.

2.7 Error messages

Creating the instruction list and testing the program is supported by the display of error messages.

In the PLC Editor or when compiling the program in the PLC programming mode, errors are displayed by the message "INPUT ERROR n". The flashing message "ERROR IN PLC Program n" is displayed when compiling the program after switching on the control system or when a timing error occurs. After resetting, the error can be located by calling the PLC Editor. Basically, when an error is displayed in the PLC Editor, the relevant file is opened and the cursor is on the error line. With timing error 53 (maximum PLC cycle time has timed out) the cursor is on the last jump instruction to be executed.

Error classification:

The system indicates when every error was detected. The following abbreviations are used:

E - Detected while editing, the line is not formatted.
S - Detected during syntax check in PLC Editor (Compile soft key).
(S) - May have been detected during syntax check, otherwise during Compiler run.
C - Detected during Compiler run either after control is switched on or in PLC programming mode.
R - Detected at PLC program run time.

List of error codes:

Error Code	Explanation
0 E S C	Not a valid command - The read line cannot be interpreted as a PLC command
1 -	spare (earlier versions: Operand for jump is no label)
2 E S C	Invalid operand type - An invalid operand type has been defined. - The command cannot be used with the defined operand type.
3 E S C	Operand not found - A type has been specified for the operand, but not a value.
4 E S C	Operand not in acceptable range - An operand number has been defined that is outside the range available for the operand.
5 E S C	No limiter after command - The PLC command is followed by other characters that cannot be interpreted.
6 E S C	No end of line found - The line is more than 128 characters long.
7 S C	Label not defined - There is a reference to a label that is not defined elsewhere with LBL, KFIELD or EXTERN.

8 S C No end of block found

- At the end of the program file there are PLC commands not terminated by an EM or JP command. There is an attendant risk of an undefined program range being run at run time.

9 S C Program too long (RAM Overflow)

- The total length of the program code to be generated exceeds the storage space available in the control..

10 S C Assignment to a parenthesis

- An attempt has been made to assign a gating result to an operand even though not all open parentheses have been closed again.

11 S C Parenthesis nesting too deep

- An attempt has been made to nest more than 16 parentheses..

12 S C Jump in a gating chain

- An unconditional jump has been programmed even though the assignment chain previously begun has not yet been assigned.

13 S C Close parenthesis with no open parenthesis

- A "close parenthesis" command has been programmed before writing the relevant "open parenthesis" command.

14 S C Label within a parenthesis

- A label has been set inside a parenthesis gate. This is not permitted because "close parenthesis" commands cannot be executed before the corresponding "open parenthesis" commands.

15 S C Label within a gating chain

- A label has been programmed in a gate that has already begun. This is not permitted because the first command after the label would have to be interpreted either as a gate or as a load command depending on the program flow.

16 S C Jump within a parenthesis

- A jump command has been programmed in a parenthesis. This is not possible because opened parentheses must always be closed again due to internal implementation, and this would not happen with the jump.

17 S C Parenthesis opened at end of block

- An EM instruction has been programmed with parenthesis open. Parentheses must always be closed again.

18 S C Label defined twice

- A label name imported from another module with EXTERN has been used again with an LBL or KFIELD instruction.
- A name reserved for internal modules (9000 .. 9255) has been used with an LBL, KFIELD or EXTERN instruction.

19 S C No word assignment

- A word gate has been executed but the result has not been assigned to any operand and a new gate has begun instead.

20 S C No logic assignment

- A logic gate has been executed but the result has not been assigned to any operand and a new gate has begun instead.

21	S C	Word accumulator not loaded - A command has been programmed that gates, assigns or manipulates the already loaded word accumulator even though the accumulator was not loaded before.
22	S C	Logic accumulator not loaded - A command has been programmed that gates, assigns or manipulates the already loaded logic accumulator even though the accumulator was not previously loaded.
23	S C	Accumulators not loaded with open parenthesis - An "open parenthesis" command has been programmed even though neither a logic nor a word string was previously begun.
24	S C	Wrong type of parenthesis result - Depending on the gate formed before a parenthesis and the parenthesis command used, the system expects the string in the parenthesis to provide a result of the same type (word/logic). With different types the gate required by the "open parenthesis" command cannot be formed.
25	S C	Conditional jump with invalid logic accumulator - A conditional jump (CMT/CMF/JPT/JPF/EMT/EMF) has been programmed without first beginning a gating chain in the logic accumulator.
26	S C	ENDC/ENDK outside a CASE/KFIELD instruction - An ENDC command has been programmed without a prior CASE instruction. - And ENDK command has been programmed without a prior KFIELD label.
27	S C	Wrong command within CASE table/KFIELD - A command other than CM has been programmed after a CASE instruction and before the corresponding ENDC instruction. - A command other than K has been programmed after a KFIELD label and before the corresponding ENDK label.
28	S C	Too many table entries in CASE - A CASE table with more than 128 entries has been programmed.
29	S C	Blank CASE instruction/KFIELD - A CASE instruction has been programmed followed immediately by an ENDC label. - A KFIELD instruction has been programmed followed immediately by an ENDK label.
30	S C	String accumulator not loaded - A command has been programmed which gates, assigns or manipulates the already loaded string accumulator even though the accumulator was not previously loaded.
31	S C	String instruction within parentheses - A string instruction has been programmed within parentheses even though string gates cannot be nested with parentheses.
32	S C	No string assignment - A new gating chain has started without assigning the gating result previously formed in the string accumulator.

33	S C	GLOBAL/EXTERN not at start of file - The commands GLOBAL or EXTERN have been written after another program code in the file. These commands must always come before the program code.
34	(S)C	Too many modules - An attempt has been made to link more than 64 files into one program with the USES instruction.
35	(S)C	File not found - A file linked with USES cannot be found. - An attempt has been made to link a file of the .PLC type with MP4010=0 (EPROM).
36	S C	File too long - The program code for an individual file is bigger than 64 K so cannot be compiled. The file must be split up into several files and linked with USES
37	S C	Too many local labels - More than 1000 labels have been issued in a file. All LBL, KFIELD and EXTERN instructions are added together, also the (invisible) labels generated by structured commands. The file must be split up into several files and linked with USES.
38	C	Too many global labels - Over 1000 global labels have been defined from all participating files.
39	C	External label not defined - A label declared as EXTERN has not been defined as GLOBAL in any of the participating modules.
40	S C	External label in CASE instruction - A label declared as EXTERN has been entered in the CM List of a CASE instruction. A local module must be defined that only calls the global module with CM in the simplest case.
41	S C	External label in JP instruction - An attempt has been made to jump to a label defined as EXTERN with a JP/JPF/JPT instruction.
42	(S)C	Global label defined twice - The same label has been defined as GLOBAL several times in the same or in different files.
43	S C	Wrong instruction structure - An ELSE/ENDI/ENDW/UNTIL instruction has been programmed without an IF/ELSE/WHILE/REPEAT instruction first. - Instructions with different structures have been interleaved instead of nested. The structures must always be closed in the reverse order in which they were opened!
44	S C	Structure open at end of file - A structured instruction has been opened but not closed again by the end of file.
45	S C	GLOBAL instruction in main file - A module from the main file has been defined as GLOBAL. Only modules from files linked with USES can be made accessible to other files using the GLOBAL instruction.

46..49 spare

50 R Nesting too deep
- An attempt has been made to nest more than 32 module calls.
- A recursive module call has been programmed which exceeds the nesting depth limit of 32.

51 R Stack underflow
- An attempt has been made to fetch data from the stack even though they were not previously stored there.

52 R Stack overflow
- An attempt has been made to store more than 128 bytes of data to the stack. Word operands (B/W/D/K) occupy 4 bytes each, logic operands (M/I/O/T/C) occupy 2 bytes each.

53 R Time out
- It took longer than 10 ms to process the program section that runs cyclically. Check the subprogram structure, if necessary very processor-intensive sections will have to be started as SUBMIT jobs.
- The displayed computer time may be increased by V.24 transfers and handwheel mode. In case of doubt select handwheel mode and start data transfer with V.24 (baud rate 38400 if possible), then check "MAXIMUM PROCESSING TIME" in PLC programming environment. 100% corresponds to 5 ms, the block processing speed is still achieved with this load. Values above 150% should not occur (safety margin for adverse operating conditions!).

54 R CASE out of range
- The operand for the CASE instruction contains a value that cannot be interpreted as an offset in the CM table (<0 or > table length -1).

55 R Subprogram not defined
- This error cannot occur at present..

56 R
- The address for a write access to data types B/W/D/M/I/O/T/C is in a range invalid for this operand type owing to inclusion of the index register.
- While accessing a constants field the index register contains a value which is not possible for that field (<0 or > field length -1).
- The address of a string leads to a prohibited value owing to inclusion of the index register.
- The number of a dialogue (S#Dn[X]) or an error message (S#En[X]) leads to a prohibited value owing to inclusion of the index register (<0 or >999).
- While addressing a substring (Sn^X) the value range for the index register (0..127) was overshot.

3 Commands

3.1 Load and Assign Commands

3.1.1 LOAD (L)

Abbreviation for the PLC-Editor: L (LOAD)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.2 to 0.7	0.5 to 0.8	0.2 to 0.7	0.0 to 0.3
Number of bytes	4	6	4	6

Logic execution with LOAD command

Operands: M, I, O, T, C

Operation:
The addressed operand is copied into the Accumulator. A load command is always used at the start of a logic chain, in order to enable subsequent gating commands. The same function is achieved when the gating commands A, O, XO are used at the start of a logic chain, however this should only be used when compatibility with the TNC 355 is required.

Example:
Input I4 and Input I5 is to be gated with AND and the result assigned to Output O2. Thus the logic state of Input I4 is loaded into the Accumulator to enable subsequent gating commands.

Initial state:	Input	I4	= 1
	Input	I5	= 0
	Output	O2	= ?

Line	Instruction	Accumulator Contents	Operand-Contents
	Bit	31 . . . 7 0	
		... x x x x x x | X | x x x x x x x	
1	L I4	... x x x x x x | 1 | x x x x x x x	1
2	A I5	... x x x x x x | 0 | x x x x x x x	0
3	= O2	... x x x x x x | 0 | x x x x x x x	0

Line 1: The operand contents are loaded into the Logic Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with AND.
Line 3: The gating result is assigned to output O2.

Word execution with the LOAD command

Operands: B, W, D, K

Operation:
The addressed Operand (B, W, D) or a Constant (K) is copied into the Word Accumulator. In addition, the Accumulator is filled, if necessary, according to the sign bit. In contrast to logic execution the start of a word gating chain must always be with the L command. It is not possible to use a gating command.

Example:
A Constant and Byte B5 is to be gated with AND and the result assigned to Byte B8.

Initial state:	Byte	B5 = 2A (hex)
	Constant:	54 = 36 (hex)
	Byte	B8 = ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 15 7 0	7 0
		... x x x x x x x x x x | x | x x x x x x x	
1	L K+54	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 1 0 1 1 0	
2	A B5	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 0 0 0 1 0	00101010
3	= B8	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 0 0 0 1 0	00100010

Line 1: The Constant is loaded into the Word Accumulator.
Line 2: The contents of the Word Accumulator and Byte B5 are gated with AND.
Line 3: The gating result is assigned to Byte B8.

3.1.2 LOAD NOT (LN)

Abbreviation for the PLC-Editor: LN (LOAD NOT)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.2 to 0.7	0.5 to 1.6	0.5 to 0.7	0.2 to 0.3
Number of bytes	6	8	6	8

Logic execution with the LOAD NOT command

Operands: M, I, O, T, C

Operation:
The complement of the addressed operand is loaded into the Logic Accumulator. A load command is always used at the start of a logic chain in order to enable subsequent gating commands. The same function is achieved when the gating commands AN, ON, XON are used at the start of a logic chain, however this should only be used when compatibility with TNC 355 is required.

Example:
The inverted logic state of Input I4 and Input I5 is to be gated with AND and the result assigned to Output O2. Thus the inverted logic state of Input I4 is loaded into the Accumulator to enable subsequent gating commands.

Initial state:	Input	I4 = 0
	Input	I5 = 1
	Output	O2 = ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0 ... x x x x x x | X | x x x x x x x	
1	LN I4	... x x x x x x | 1 | x x x x x x x	0
2	A I5	... x x x x x x | 1 | x x x x x x x	1
3	= O2	... x x x x x x | 1 | x x x x x x x	1

Line 1: The inverted operand contents are loaded into the Logic Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with AND.
Line 3: The gating result is assigned to Output O2.

Word execution with the LOAD NOT command

Operands: B, W, D, K

Operation:
The complement of the contents of the addressed Operand (B, W, D) or Constant (K) is loaded into the Word Accumulator. In addition, the Accumulator is filled, if necessary, according to the sign bit. In contrast to logic execution a word gating chain must always start with a load command. It is not possible to use a gating command.

Example:
The complement of Byte B6 and Byte B5 is to be gated with AND and the result assigned to Byte B8.

Initial state: Byte B5 = 2A (hex)
Byte B6 = B6 (hex)
Byte B8 = ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 15 7 0	7 0
		... x x x x x x x x x x x x x x x x x x	
1	LN B6	... 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1	10110110
2	A B5	... 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0	00101010
3	= B8	... 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0	00001000

Line 1: The inverted contents of Byte B6 are loaded into the Word Accumulator.
Line 2: The contents of the Word Accumulator and Byte B5 are gated with AND.
Line 3: The gating result is assigned to Byte B8.

3.1.3 LOAD TWO'S-COMPLEMENT (L–)

Abbreviation for the PLC-Editor: L– (LOAD MINUS)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	----	0.5 to 1.6	0.5 to 0.7	0.2 to 0.3
Number of bytes	----	8	6	8

Operands: B, W, D, K

Operation:
The contents of the addressed Operand (B, W, D) or a Constant (K) are loaded into the Word Accumulator as a two's complement. In addition, the Accumulator is filled, if necessary, according to the sign bit. The two's complement allows negative numbers to be stored. i.e. a number loaded with the L- command appears in the Accumulator with an inverted sign.
This command may only be used with Word execution.

Example:
The contents of Byte B5 is to be negated, added to Byte B6 and the result assigned to Byte B8.

Initial state:
Byte B5 = 15 (dec)
Byte B6 = 20 (dec)
Byte B8 = ?

Line	Instruction	Accumulator Contents (Bit 31 . . . 15 … 7 … 0)		Operand Contents (7 … 0)	
		... x x x x x x x x x x x x x x x x x x			
1	L– B5	... 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 1	(–15)	(+15)	00001111
2	+ B6	... 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1	(+ 5)	(+20)	00010100
3	= B8	... 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1	(+ 5)	(+5)	00000101

To aid understanding of this example, the contents of the Accumulator and operands are shown as decimal values in parentheses.

Line 1: The contents of Byte B5 are loaded into the Accumulator and the sign of the value is inverted.
Line 2: The contents of the Word Accumulator and Byte B6 are added.
Line 3: The result is assigned to Byte B8.

3.1.4 LOAD BYTE (LB)

Abbreviation for the PLC-Editor: LB (LOAD BYTE)

Execution time [µs] 10.8 to 12.5
Number of bytes 18

Operands: M, I, O, T, C

Operation:
With the command LB, 8 Markers, Inputs, Outputs, Timers or Counters with ascending numbering are loaded into the Word Accumulator. Each operand occupies 1 bit in the Accumulator. The designated operand address occupies the LSB in the Accumulator, the designated address + 1 the LSB + 1 and so on. In this way, the last affected operand occupies the MSB!
If necessary, the Accumulator is filled according to the sign bit.

3.1.5 LOAD WORD (LW)

Abbreviation for the PLC-Editor: LW (LOAD WORD)

Execution time [µs] 21.0 to 23.8
Number of bytes 18

Operands: M, I, O, T, C

Operation:
With the command LW, 16 Markers, Inputs, Outputs, Timers or Counters with ascending numbering are loaded into the Word Accumulator. Each operand occupies 1 bit in the Accumulator. The designated operand address occupies the LSB in the Accumulator, the designated address + 1 the LSB + 1 and so on. In this way, the last affected operand occupies the MSB!
If necessary, the Accumulator is filled according to the sign bit.

3.1.6 LOAD DOUBLEWORD (LD)

Abbreviation for the PLC-Editor: LD (LOAD DOUBLE WORD)

Execution time [µs] 40.7 to 46.5
Number of bytes 16

Operands: M, I, O, T, C

Operation:
With the command LD, 32 Markers, Inputs, Outputs, Timers or Counters with ascending numbering are loaded into the Word Accumulator. Each operand occupies 1 bit in the Accumulator. The designated operand address occupies the LSB in the Accumulator, the designated address + 1 the LSB + 1 and so on. In this way, the last affected operand occupies the MSB!
If necessary, the Accumulator is filled according to the sign bit.

Example for the Commands LB, LW and LD:
Via the Inputs I3 to I10, a binary coded value is to be read in and assigned to Byte B8 for further use.

Initial state:	Input I3 = 1	Input I7 = 0
	Input I4 = 1	Input I8 = 1
	Input I5 = 1	Input I9 = 1
	Input I6 = 0	Input I10 = 0

Line	Instruction		Accumulator Contents	Operand Contents
		Bit	31 . . . 15 7 0	I10 I3
			... x x x x x x x x x x x x x x x x x x	
1	LB I3		... 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 1	01100111
				7 0
2	= B8		... 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 1 1	01100111

Line 1: Inputs I3 to I10 are loaded into the Word Accumulator (Bit 0 to Bit 7).
Line 2: The Accumulator Contents are assigned to Byte 8.

The Commands LW and LD are processed in the same way except that 16 or 32 operands are used accordingly.

3.1.7 ASSIGN (=)

Abbreviation for the PLC-Editor: = (ASSIGN)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.2 to 0.5	0.2 to 0.5	0.2 to 0.5	----
Number of bytes	4	4	4	

Logic execution with the ASSIGN command

Operands: M, I, O, T, C

Operation:
ASSIGN in conjunction with the Logic-Operands (M, I, O, T, C) copies the contents of the Logic Accumulator to the addressed operand. The = command is only used at the end of a logic chain in order that a gating result is available. The command may be used several times in succession (see example).

Example:
Input I4 and Input I5 should be gated with AND and the result assigned to Outputs O2 and O5.

Initial state:		
	Input	I4 = 1
	Input	I5 = 0
	Output	O2 = ?
	Output	O5 = ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0	
		... x x x x x x X x x x x x x x	
1	L I4	... x x x x x x 1 x x x x x x x	1
2	A I5	... x x x x x x 0 x x x x x x x	0
3	= O2	... x x x x x x 0 x x x x x x x	0
4	= O5	... x x x x x x 0 x x x x x x x	0

Line 1: The operand contents are loaded into the Logic Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with AND.
Line 3: The gating result is assigned to Output O2.
Line 4: The gating result is assigned to Output O5.

Word execution with the ASSIGN command

Operands: B, W, D

Operation:
ASSIGN in conjunction with the Word-Operands (B, W, D) copies the contents of the Word Accumulator to the addressed operand. In contrast to bit processing, the = command can also be used within a word gating chain. The command can be used several times in succession (see example).

Example:
A Constant (K) and the contents of Byte B5 should be gated with AND and the result assigned to Byte B8 and Byte B10.

Initial state:	Byte	B5 = 2A	(hex)
	Constant	54 = 36	(hex)
	Byte	B8 = ?	
	Byte	B10 = ?	

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 15 7 0	7 0
		... x x x x x x x x x x x x x x x x x x	
1	L K+54	... 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0	
2	A B5	... 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0	00101010
3	= B8	... 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0	00100010
4	= B10	... 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0	00100010
5	= B10	... 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0	01010000

Line 1: The Constant is loaded into the Word Accumulator.
Line 2: The contents of the Word Accumulator is stored in B8
Line 2: The contents of the Word Accumulator and Byte B5 are gated with AND.
Line 3: The gating result is assigned to Byte B8.
Line 4: The gating result is assigned to Byte B10.

3.1.8 ASSIGN BYTE (B=)

Abbreviation for the PLC-Editor: B= (ASSIGN BYTE)

Execution time [µs] 8.7 to 11.7
Number of bytes 14

Operands: M, I, O, T, C

Operation:
With the command B=, 8 bits are copied from the Word Accumulator to Markers, Inputs, Outputs, Timers or Counters with ascending numbering. Each bit corresponds to 1 operand. The LSB in the Accumulator is copied to the designated operand address, the LSB + 1 to the designated address + 1 and so on. The last affected operand is occupied by the MSB.

3.1.9 ASSIGN WORD (W=)

Abbreviation for the PLC-Editor: W= (ASSIGN WORD)

Execution time [µs] 16.7 to 22.7
Number of bytes 14

Operands: M, I, O, T, C

Operation:
With the command W=, 16 bits are copied from the Word Accumulator to Markers, Inputs, Outputs, Timers or Counters with ascending numbering. Each bit corresponds to 1 operand. The LSB in the Accumulator is copied to the designated operand address, the LSB + 1 to the designated address + 1 and so on. The last affected operand is occupied by the MSB.

3.1.10 ASSIGN DOUBLEWORD (D=)

Abbreviation for the PLC-Editor: D= (ASSIGN DOUBLE)

Execution time [µs] 32.7 to 44.7
Number of bytes 14

Operands: M, I, O, T, C

Operation:
With the command D=, 32 bits are copied from the Word Accumulator to Markers, Inputs, Outputs, Timers or Counters with ascending numbering. Each bit corresponds to 1 operand. The LSB in the Accumulator is copied to the designated operand address, the LSB + 1 to the designated address + 1 and so on. The last affected operand is occupied by the MSB.

Example:
A bit pattern, as defined in Word W8 , is to be assigned to Outputs O5 to O20.

Initial state: Word W8: 36 FF (hex)

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 15 7 0	15 8 7 0
		... x x x x x x x x x x | x | x x x x x x x	
1	L W8	... 0 0 0 0 1 1 0 1 1 0 | 1 | 1 1 1 1 1 1 1	00110110 | 11111111
			O20 ... ... O5
2	W= O5	... 0 0 0 0 1 1 0 1 1 0 | 1 | 1 1 1 1 1 1 1	00110110 | 11111111

Line 1: The contents of Word W8 are loaded into the Accumulator.
Line 2: The contents of the Accumulator are assigned to the Outputs O5 to O20.

The Commands B= and D= are processed in the same way except that 8 or 32 bits are used accordingly.

3.1.11 ASSIGN NOT (=N)

Abbreviation for the PLC Editor: =N (STORE NOT)

Logic processing
Operands: M, I, O, T, C

Operation:
An ASSIGN NOT in conjunction with a logic operand (M,I,O,T,C) copies the one's complement of the contents of the logic accumulator to the addressed operand.
For example see ASSIGN command (=).

Word processing
Operands: B, W, D

Operation:
An ASSIGN NOT in conjunction with a word operand (B,W,D) copies the one's complement of the contents of the word accumulator to the addressed operand.

For example see ASSIGN command (=).

3.1.12 ASSIGN TWO'S COMPLEMENT (= -)

Abbreviation for the PLC Editor: = - (STORE MINUS)
Operands: B, W, D

Operation:
An ASSIGN TWO'S COMPLEMENT copies the two's complement of the contents of the word accumulator to the addressed operand.
For example see ASSIGN command (=).

3.2 Set-Commands

3.2.1 SET (S)

Abbreviation for the PLC Editor: S (SET)

	Operand changed	Operand unchanged
Execution time [µs]	0.1 to 0.8	0.2 to 0.5
Number of bytes	8 (6)	

Byte value in parentheses:
With certain preceding program sequences, the command may be shortened.

Operands: M, I, O, T, C

Operation:
The function of the command depends on the contents of the Logic Accumulator. If the Logic Accumulator = 1, the addressed operand is set to 1, otherwise the operand remains unchanged. An S-command is used at the end of a logic chain so that the gating result may influence the operand. The command may be used several times in succession (see example).

Example:
Input I4 and input I5 should be gated with OR.
If the gating result is 1, output O2 and marker M500 should be set.

Initial state:			
	Input	I4	= 1
	Input	I5	= 0
	Output	O2	= ?
	Marker	M500	= ?

Line	Instruction	Accumulator Contents (Bit 31 . . . 7 0)	Operand Contents
		... x x x x x x | X | x x x x x x x	
1	L I4	... x x x x x x | 1 | x x x x x x x	1
2	O I5	... x x x x x x | 1 | x x x x x x x	0
3	S O2	... x x x x x x | 1 | x x x x x x x	1
4	S M500	... x x x x x x | 1 | x x x x x x x	1

Line 1: The contents of the operand are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with OR.
Line 3: The gating result = 1: output O2 is set.
Line 4: The gating result = 1: marker M500 is set.

3.2.2 RESET (R)

Abbreviation for the PLC-Editor: R (RESET)

	Operand changed	Operand unchanged
Execution time [µs]	0.1 to 0.8	0.2 to 0.5
Number of bytes	8 (6)	

Byte value in parentheses:
With certain preceding program sequences the command may be shortened.

Operands: M, I, O, T, C

Operation:
The function of the command is dependent on the contents of the Logic Accumulator. If the Logic Accumulator = 1, the addressed operand is set to 0, otherwise the operand remains unchanged. An R command is used at the end of a logic chain, in order that a gating result may influence the operand. The command may be used several times in succession (see example).

Example:
Input I4 and Input I5 should be gated with OR.
If the gating result = 1, Output O2 and Marker M500 should be reset.

Initial state:			
	Input	I4	= 1
	Input	I5	= 0
	Output	O2	= ?
	Marker	M500	= ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0	
		... x x x x x x | X | x x x x x x x	
1	L I4	... x x x x x x | 1 | x x x x x x x	$\boxed{1}$
2	O I5	... x x x x x x | 1 | x x x x x x x	$\boxed{0}$
3	R O2	... x x x x x x | 1 | x x x x x x x	$\boxed{0}$
4	R M500	... x x x x x x | 1 | x x x x x x x	$\boxed{0}$

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and input I5 are gated with OR.
Line 3: The gating result = 1: Output O2 is reset.
Line 4: The gating result = 1: Marker 500 is reset.

3.2.3 SET NOT (SN)

Abbreviation for the PLC-Editor: SN (SET NOT)

	Operand changed	Operand unchanged
Execution time [µs]	0.1 to 0.8	0.2 to 0.5
Number of bytes	8 (6)	

Byte value in parentheses:
With certain preceding program sequences the command may be shortened.

Operands: M, I, O, T, C

Operation:
The function of the command is dependent upon the contents of the Logic Accumulator. If the Logic Accumulator = 0 , then the addressed operand is set to 1, otherwise the operand remains unchanged. An SN command is used at the end of a logic chain, in order that a gating result may influence the operand. The command may be used several times in succession (see example).

Example:
Input I4 and Input I5 are to be gated with OR.
If the gating result = 0, Output O2 and Marker M500 are set.

Initial state:			
	Input	I4	= 0
	Input	I5	= 0
	Output	O2	= ?
	Marker	M500	= ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0	
		... x x x x x x | X | x x x x x x x	
1	L I4	... x x x x x x | 0 | x x x x x x x	0
2	O I5	... x x x x x x | 0 | x x x x x x x	0
3	SN O2	... x x x x x x | 0 | x x x x x x x	1
4	SN M500	... x x x x x x | 0 | x x x x x x x	1

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with OR.
Line 3: The gating result = 0: Output O2 is set.
Line 4: The gating result = 0: Marker 500 is set.

3.2.4 RESET NOT (RN)

Abbreviation for the PLC-Editor: RN (RESET NOT)

	Operand changed	Operand unchanged
Execution time [µs]	0.1 to 0.8	0.2 to 0.5
Number of bytes	8 (6)	

Byte value in parentheses:
With certain preceding program sequences the command may be shortened.

Operands: M, I, O, T, C

Operation:
The function of the command is dependent upon the contents of the Logic Accumulator. If the Logic Accumulator = 0, then the addressed operand is set to 0, otherwise the operand remains unchanged. An RN command is used at the end of a logic chain, in order that a gating result may influence the operand. The command may be used several times in succession (see example).

Example:
Input I4 and Input I5 are to be gated with OR.
If the gating result = 0, Output O2 and Marker M500 are reset.

Initial state:			
	Input	I4	= 0
	Input	I5	= 0
	Output	O2	= ?
	Marker	M500	= ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit 31 . . . 7 0	... x x x x x x | X | x x x x x x x	
1	L I4	... x x x x x x | 0 | x x x x x x x	$\boxed{0}$
2	O I5	... x x x x x x | 0 | x x x x x x x	$\boxed{0}$
3	RN O2	... x x x x x x | 0 | x x x x x x x	$\boxed{0}$
4	RN M500	... x x x x x x | 0 | x x x x x x x	$\boxed{0}$

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with OR.
Line 3: The gating result = 0: Output O2 is reset.
Line 4: The gating result = 0: Marker M500 is reset.

3.3 Logic Gates

3.3.1 AND (A)

Abbreviation for the PLC-Editor: A (AND)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.2 to 0.7	0.2 to 0.7	0.2 to 0.7	0.1 to 0.5
Number of bytes	4	4	4	6

Logic execution with the AND command

Operands: M, I, O, T, C

Operation:
This command functions in different ways according to its position in the program:
a) At the start of a logic chain the command functions as an L command, i.e. the logic state of the operand is loaded into the Logic Accumulator. This is to ensure compatibility with the TNC 355 control which did not have the special L command.
In PLC programs for the TNC 407/TNC 415, a logic chain should always be started with a load command (see L, LN, L–).
b) Within a logic chain the contents of the Logic Accumulator and the logic state of the operand (M, I, O, T, C) are gated with AND. The gating result is stored in the Logic Accumulator.

Example:
Input I4 and Input I5 are to be gated with AND and the result assigned to Output O2.

Initial state:			
	Input	I4	= 1
	Input	I5	= 0
	Output	O2	= ?

Line	Instruction	Accumulator Contents			Operand Contents
	Bit	31 . . .	7	0	
		... x x x x x x	x	x x x x x x x	
1	L I4	... x x x x x x	1	x x x x x x x	1
2	A I5	... x x x x x x	0	x x x x x x x	0
3	= O2	... x x x x x x	0	x x x x x x x	0

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with AND.
Line 3: The gating result is assigned to Output O2.

Word execution with the AND Command

Operands: B, W, D, K

Operation:
The contents of the Word Accumulator and the contents of the operand (B, W, D, K) are gated with AND. In accordance with the different sizes of operand (B = 8 bit; W = 16 bit; D = K = 32 bit), 8, 16 or 32 bits will be influenced in the Accumulator.
Thus: Bit 0 of the Accumulator is gated with bit 0 of the operand
Bit 1 of the Accumulator is gated with bit 1 of the operand and so on.
The result of the operation is stored in the Word Accumulator.

Example:
The contents of Byte B5 and Byte B6 should be gated with AND and the result assigned to Byte B8.

Initial state: Byte B5 = 2A (hex)
Byte B6 = 36 (hex)
Byte B8 = ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 15 7 0	15 8 7 0
		... x x x x x x x x x x | x | x x x x x x x	
1	L B6	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 1 0 1 1 0	00110110
2	A B5	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 0 0 0 1 0	00101010
3	= B8	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 0 0 0 1 0	00100010

Line 1: The contents of Byte B6 are loaded into the Accumulator.
Line 2: The contents of the Word Accumulator and Byte B5 are gated with AND.
Line 3: The gating result is assigned to Byte B8.

3.3.2 AND NOT (AN)

Abbreviation for the PLC-Editor: AN (AND NOT)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.5 to 0.9	0.5 to 0.9	0.5 to 0.9	0.2 to 0.5
Number of bytes	8	8	8	10

Logic execution with the AND NOT command

Operands: M, I, O, T, C

Operation:
This command functions in different ways according to its position in the program:

a) At the start of a logic chain the command functions as an LN command, i.e. the complement of the operand is loaded into the Logic Accumulator. This is to ensure compatibility with the TNC 355 control which did not have the special LN command.
In PLC programs for the TNC 407/TNC 415 a logic chain should always be started with a load command (see L, LN, L–).

b) Within a logic chain, the contents of the Logic Accumulator and the logic state of the operand (M, I, O, T, C) are gated with AND NOT.
The gating result is stored in the Logic Accumulator.

Example:
Input I4 and Input I5 should be gated with AND NOT and the result assigned to Output O2.

Initial state:	Input	I4	= 1
	Input	I5	= 1
	Output	O2	= ?

Line	Instruction	Accumulator Contents			Operand Contents
	Bit	31 . . .	7	0	
		... x x x x x x	x	x x x x x x x	
1	L I4	... x x x x x x	1	x x x x x x x	1
2	AN I5	... x x x x x x	0	x x x x x x x	1
3	= O2	... x x x x x x	0	x x x x x x x	0

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with AND NOT.
Line 3: The gating result is assigned to Output O2.

Word execution with the AND NOT command

Operands: B, W, D, K

Operation:
The contents of the Word Accumulator and the contents of the operand (B, W, D, K) are gated with AND NOT. In accordance with the different sizes of operand (B = 8 bit; W = 16 bit; D = K = 32 bit), 8, 16 or 32 bits will be influenced in the Accumulator.
Thus: Bit 0 in the Accumulator is gated with bit 0 in the operand.
Bit 1 in the Accumulator is gated with bit 1 in the operand and so on.
The result of the operation is stored in the Word Accumulator.

Example:
The contents of Word W4 and Word W6 should be gated with AND NOT and the result assigned to Word W8.

Initial state: Word W4 = 36 AA (hex)
Word W6 = 3C 36 (hex)
Word W8 = ?

Line	Instruction	Accumulator Contents			Operand Contents	
	Bit	31 . . . 15	7	0	15 8	7 0
		... x x x x x x x x x	x	x x x x x x x		
1	L W6	... 0 0 0 0 1 1 1 1 0 0	0	0 1 1 0 1 1 0	00111100	00110110
2	AN W4	... 0 0 0 0 0 0 1 0 0 0	0	0 0 1 0 1 0 0	00110110	10101010
3	= W8	... 0 0 0 0 0 0 1 0 0 0	0	0 0 1 0 1 0 0	00001000	00010100

Line 1: The contents of Word W6 are loaded into the Accumulator.
Line 2: The contents of the Word Accumulator and Word W4 are gated with AND NOT.
Line 3: The gating result is assigned to Word W8.

3.3.3 OR (O)

Abbreviation for the PLC-Editor: O (OR)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.2 to 0.7	0.2 to 0.7	0.2 to 0.7	0.1 to 0.5
Number of bytes	4	4	4	6

Logic execution with the OR command

Operands: M, I, O, T, C

Operation:
This command functions in different ways according to its position in the program:
a) At the start of a logic chain the command functions as an L command, i.e. the logic state of the operand is loaded into the Logic Accumulator. This is to ensure compatibility with the TNC 355 control which did not have the special L command.
In PLC programs for the TNC 407/TNC 415 a logic chain should always be started with a load command (see L, LN, L–).
b) Within a logic chain, the contents of the Logic Accumulator and the logic state of the operand (M, I, O, T, C) are gated with OR.
The result of the operation is stored in the Logic Accumulator.

Example:
Input I4 and Input I5 are to be gated with OR and the result assigned to Output O2.

Initial state:	Input	I4 = 0
	Input	I5 = 1
	Output	O2 = ?

Line	Instruction	Accumulator Contents			Operand Contents
	Bit	31 . . .	7	0	
		... x x x x x x	x	x x x x x x x	
1	L I4	... x x x x x x	0	x x x x x x x	0
2	O I5	... x x x x x x	1	x x x x x x x	1
3	= O2	... x x x x x x	1	x x x x x x x	1

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with OR.
Line 3: The gating result is assigned to Output O2.

Word execution with the OR command

Operands: B, W, D, K

Operation:
The contents of the Word Accumulator and the contents of the operand (B, W, D, K) are gated with OR. In accordance with the different sizes of operand (B = 8 bit; W = 16 bit; D = K = 32 bit), 8, 16 or 32 bits will be influenced in the Accumulator.

Thus: Bit 0 in the Accumulator is gated with bit 0 in the operand
Bit 1 in the Accumulator is gated with bit 1 in the operand and so on.

The result of the operation is stored in the Word Accumulator.

Example:
The contents of Byte B5 and Byte B6 are to be gated with OR and the result assigned to Word W8.

Initial state:	Byte	B5	= 2A	(hex)
	Byte	B6	= 36	(hex)
	Word	W8	= ?	

Line	Instruction	Accumulator Contents (Bit 31 . . . 15 / 7 / 0)	Operand Contents (15 / 8 7 / 0)
		... x x x x x x x x x x | x | x x x x x x x	
1	L B6	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 1 0 1 1 0	00110110
2	O B5	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 1 1 1 1 0	00101010
3	= W8	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 1 1 1 1 0	00000000 | 00111110

Line 1: The contents of Byte B6 are loaded into the Accumulator.
Line 2: The contents of the Word Accumulator and Byte B5 are gated with OR.
Line 3: The gating result is assigned to Word W8.

3.3.4 OR NOT (ON)

Abbreviation for the PLC-Editor: ON (OR NOT)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.5 to 0.9	0.5 to 0.9	0.5 to 0.9	0.2 to 0.5
Number of bytes	8	8	8	10

Logic execution with the OR NOT command

Operands: M, I, O, T, C

Operation:
This command functions in different ways according to its position in the program:
a) At the start of a logic chain this command functions as an LN command, i.e. the complement of the operand is loaded into the Logic Accumulator. This is to ensure compatibility with the TNC 355 control which did not have the special LN command. In PLC programs for the TNC 407/TNC 415 a logic chain should always be started with a load command (see L, LN, L–).
b) Within a logic chain, the contents of the Logic Accumulator and the logic state of the operand (M, I, O, T, C) are gated with OR NOT. The result of the operation is stored in the Logic Accumulator.

Example:
Input I4 and Input I5 are to be gated with OR NOT and the result assigned to Output O2.

Initial state:	Input	I4	= 0
	Input	I5	= 0
	Output	O2	= ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0	
		... x x x x x x | x | x x x x x x x	
1	L I4	... x x x x x x | 0 | x x x x x x x	0
2	ON I5	... x x x x x x | 1 | x x x x x x x	0
3	= O2	... x x x x x x | 1 | x x x x x x x	1

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with OR NOT.
Line 3: The gating result is assigned to Output O2.

Word execution with the OR NOT command

Operands: B, W, D, K

Operation:
The contents of the Word Accumulator and the contents of the operand (B, W, D, K) are gated with OR NOT. In accordance with the different sizes of operand (B = 8 bit; W = 16 bit; D = K = 32 bit), 8, 16 or 32 bits will be influenced in the Accumulator.
Thus: Bit 0 in the Accumulator is gated with bit 0 in the operand
Bit 1 in the Accumulator is gated with bit 1 in the operand and so on.
The result of the operation is stored in the Word Accumulator.

Example:
The contents of Word W4 and Word W6 are to be gated with OR NOT and the result assigned to Word W8.

Initial state: Word W4 = 36 AA (hex)
Word W6 = 3C 36 (hex)
Word W8 = ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 15 7 0	15 8 7 0
		... x x x x x x x x x x | x | x x x x x x x	
1	L W6	... 0 0 0 0 1 1 1 1 0 0 | 0 | 0 1 1 0 1 1 0	00111100 | 00110110
2	ON W4	... 1 1 1 1 1 1 1 1 0 1 | 0 | 1 1 1 0 1 1 1	00110110 | 10101010
3	= W8	... 1 1 1 1 1 1 1 1 0 1 | 0 | 1 1 1 0 1 1 1	11111101 | 01110111

Line 1: The contents of Word W6 are loaded into the Accumulator.
Line 2: The contents of the Word Accumulator and Word W4 are gated with OR NOT.
Line 3: The gating result is assigned to Word W8.

3.3.5 EXCLUSIVE OR (XO)

Abbreviation for the PLC-Editor: XO (EXCLUSIVE OR)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.5 to 0.7	0.5 to 0.7	0.5 to 0.7	0.2 to 0.5
Number of bytes	6	6	6	8

Logic execution with the EXCLUSIVE OR command

Operands: M, I, O, T, C

Operation:
This command functions in different ways according to its position in the program:

a) At the start of a logic chain the command functions as an L command, i.e. the logic state of the operand is loaded into the Logic Accumulator. This is to ensure compatibility with the TNC 355 control which did not have the special L command.
In PLC programs for the TNC 407/TNC 415 a logic chain should always be started with a load command (see L, LN, L–).

b) Within a logic chain the contents of the Logic Accumulator and the logic state of the operand (M, I, O, T, C) are gated with EXCLUSIVE OR.
The result of the operation is stored in the Logic Accumulator.

Example:
Input I4 and Input I5 are to be gated with EXCLUSIVE OR and the result assigned to Output O2.

Initial state:	Input	I4 = 1
	Input	I5 = 1
	Output	O2 = ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0	
		... x x x x x x X x x x x x x x	
1	L I4	... x x x x x x 1 x x x x x x x	1
2	XO I5	... x x x x x x 0 x x x x x x x	1
3	= O2	... x x x x x x 0 x x x x x x x	0

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and Input I5 are gated with EXCLUSIVE OR.
Line 3: The gating result is assigned to Output O2.

Word execution with the EXCLUSIVE OR command

Operands: B, W, D, K

Operation:
The contents of the Word Accumulator and the contents of the operand (B, W, D, K) are gated with EXCLUSIVE OR. In accordance with the different sizes of operand (B = 8 bit; W = 16 bit; D = K = 32 bit), 8, 16 or 32 bits will be influenced in the Accumulator.
Thus: Bit 0 in the Accumulator is gated with bit 0 in the operand
Bit 1 in the Accumulator is gated with bit 1 in the operand and so on.
The result of the operation is stored in the Word Accumulator.

Example:
The contents of Byte B5 and Byte B6 are to be gated with EXCLUSIVE OR and the result assigned to Word W8.

Initial state	Byte	B5	= 2A	(hex)
	Byte	B6	= 36	(hex)
	Word	W8	= ?	

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 15 7 0	15 8 7 0
		... x x x x x x x x x x | x | x x x x x x x	
1	L B6	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 1 1 0 1 1 0	00110110
2	XO B5	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 0 1 1 1 0 0	00101010
3	= W8	... 0 0 0 0 0 0 0 0 0 0 | 0 | 0 0 1 1 1 0 0	00000000 00011100

Line 1: The contents of Byte B6 are loaded into the Accumulator.
Line 2: The contents of the Word Accumulator and Byte B5 are gated with EXCLUSIVE OR.
Line 3: The gating result is assigned to Word W8.

3.3.6 EXCLUSIVE OR NOT (XON)

Abbreviation for the PLC-Editor: XON (EXCLUSIVE OR NOT)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	0.5 to 0.9	0.5 to 0.9	0.5 to 0.9	0.2 to 0.5
Number of bytes	8	8	8	10

Logic execution with the EXCLUSIVE OR NOT command

Operands: M, I, O, T, C

Operation:
This command functions in different ways according to its position in the program:
a) At the start of a logic chain this command functions as a LN command, i.e. the complement of the operand is loaded into the Logic Accumulator. This is to ensure compatibility with the TNC 355 control which did not have the special LN command.
In PLC programs for the TNC 407/TNC 415 a logic chain should always be started with a load command (see L, LN, L–).
b) Within a logic chain the contents of the Logic Accumulator and the logic state of the operand (M, I, O, T, C) are gated with EXCLUSIVE OR NOT.
The result of the operation is stored in the Logic Accumulator.

Example:
Input I4 and Marker M500 are to be gated with EXCLUSIVE OR NOT and the result assigned to Output O2.

Initial state:			
	Input	I4	= 0
	Marker	M500	= 0
	Output	O2	= ?

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0	
		... x x x x x x | X | x x x x x x x	
1	L M500	... x x x x x x | 0 | x x x x x x x	0
2	XON I4	... x x x x x x | 1 | x x x x x x x	0
3	= O2	... x x x x x x | 1 | x x x x x x x	1

Line 1: The operand contents are loaded into the Accumulator.
Line 2: The contents of the Logic Accumulator and the Input I4 are gated with EXCLUSIVE OR NOT.
Line 3: The gating result is assigned to Output O2.

Word execution with the EXCLUSIVE OR NOT command

Operands: B, W, D, K

Operation:
The contents of the Word Accumulator and the contents of the operand (B, W, D, K) are gated with EXCLUSIVE OR NOT. In accordance with the different sizes of operand (B = 8 bit; W = 16 bit; D = K = 32 bit), 8, 16 or 32 bits will be influenced in the Accumulator.
Thus: Bit 0 in the Accumulator is gated with bit 0 in the operand
Bit 1 in the Accumulator is gated with bit 1 in the operand and so on.
The result of the operation is stored in the Logic Accumulator.

Example:
The contents of Word W4 and Word W6 are to be gated with EXCLUSIVE OR NOT and the result assigned to Word W8.

Initial state:
Word W4 = 36 AA (hex)
Word W6 = 3C 36 (hex)
Word W8 = ?

Line	Instruction	Accumulator Contents			Operand Contents	
	Bit	31 . . . 15	7	0	15 8	7 0
		... x x x x x x x x x x	x	x x x x x x x		
1	L W6	... 0 0 0 0 1 1 1 1 0 0	0	0 1 1 0 1 1 0	00111100	00110110
2	XON W4	... 1 1 1 1 1 1 0 1 0 1	0	1 1 0 0 0 1 1	00110110	10101010
3	= W8	... 1 1 1 1 1 1 0 1 0 1	0	1 1 0 0 0 1 1	11110101	01100011

Line 1: The contents of Word W6 are loaded into the Accumulator.
Line 2: The contents of the Word Accumulator and Word W4 are gated with EXCLUSIVE OR NOT.
Line 3: The gating result is assigned to Word W8.

3.4 Arithmetic Commands

3.4.1 ADDITION (+)

Abbreviation for the PLC-Editor: + (PLUS)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	----	0.5 to 0.7	0.2 to 0.7	0.1 to 0.5
Number of bytes	----	8	4	6

Operands: B, W, D, K

Operation:
With arithmetic functions the operand is firstly expanded to the size of the Accumulator (32 bits). Then the contents of the operand are added to the Word Accumulator. The result of the operation is stored in the Word Accumulator and may be processed further.

Example:
A constant and a stored value in Word W6 are to be added. The result is then stored in Doubleword D8.

Initial state:	Constant		= 100 000	(dec)
	Word	W6	= 200	(dec)
	Doubleword D8		= ?	

In the interests of clarity the contents of the Accumulator and operand are shown in decimal notation.
The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L K100000	100 000	
2	+ W6	100 200	200
3	= D8	100 200	100 200

Line 1: The Constant is loaded into the Accumulator.
Line 2: The contents of the Accumulator and Word W6 are added.
Line 3: The result is assigned to Doubleword D8.

3.4.2 SUBTRACTION (–)

Abbreviation for the PLC-Editor: – (MINUS)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	----	0.5 to 0.7	0.2 to 0.7	0.1 to 0.5
Number of bytes	----	8	4	6

Operands: B, W, D, K

Operation:
With arithmetic functions the operand is firstly expanded to the size of the Accumulator (32 bits). Then the contents of the operand are subtracted from the contents of the Word Accumulator. The result of the operation is stored in the Word Accumulator and may be processed further.

Example:
A stored value in Word W6 is to be subtracted from a Constant. The result is then stored in Doubleword D8.

Initial state:	Constant	= 100 000	(dec)
	Word W6	= 200	(dec)
	Doubleword D8	= ?	

In the interests of clarity the contents of the Accumulator and the operand are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L K100000	100 000	
2	– W6	99 800	200
3	= D8	99 800	99 800

Line 1: The Constant is loaded into the Accumulator.
Line 2: The contents of Word W6 are subtracted from the Accumulator.
Line 3: The result is assigned to Doubleword D8.

3.4.3 MULTIPLICATION (x)

Abbreviation for the PLC-Editor: x (MULTIPLY)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	----	3.5 to 4.3	3.2 to 3.8	3.0 to 3.8
Number of bytes	----	14	10	14

Operands: B, W, D, K

Operation:
With arithmetic functions the operand is firstly expanded to the size of the Accumulator (32 bits). Then the contents of the operand are multiplied with the contents of the Word Accumulator. The result of the operation is stored in the Word Accumulator and may be processed further. If the multiplication is not correctly executed the Marker M3168 is set, otherwise it is reset.

Example:
A Constant and a value stored in Word W6 are to be multiplied. The result is then stored in Doubleword D8.

Initial state	Constant		= 100	(dec)
	Word	W6	= 20	(dec)
	Doubleword	D8	= ?	

In the interests of clarity the contents of the Accumulator and the operand are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L K100	100	
2	x W6	2 000	20
3	= D8	2 000	2 000

Line 1: The Constant is loaded into the Accumulator.
Line 2: The contents of the Accumulator are multiplied by the contents of Word W6.
Line 3: The result is assigned to Doubleword D8.

3.4.4 DIVISION (/)

Abbreviation for the PLC-Editor: / (DIVIDE)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	----	6.6 to 7.7	6.6 to 7.0	6.3 to 6.7
Number of bytes	----	16	14	16

An error condition (Divisor = 0) results in an execution time of 1.0 to 1.8 µs.

Operands: B, W, D, K

Operation:
With arithmetic functions the operand is firstly expanded to the size of the Accumulator (32 bits) . Then the contents of the Word Accumulator are divided by the contents of the operand. The result of the operation is stored in the Word Accumulator and may be processed further. If the division is not correctly executed the Marker M3169 is set, otherwise it is reset.

Example:
A Constant is to be divided by the value stored in Word W6 . The result is then assigned to Doubleword D8.

Initial state:	Constant		= 100	(dec)
	Word	W6	= 20	(dec)
	Doubleword	D8	= ?	

In the interests of clarity the contents of the Accumulator and the operand are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L K100	100	
2	/ W6	5	20
3	= D8	5	5

Line 1: The Constant is loaded into the Accumulator.
Line 2: The contents of the Accumulator are divided by the contents of Word W6.
Line 3: The result is assigned to Doubleword D8.

3.4.5 REMAINDER (MOD)

Abbreviation for the PLC-Editor: MOD (MODULO)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	----	6.6 to 8.0	6.6 to 7.3	6.3 to 7.8
Number of bytes	----	18	16	12

An error condition (Divisor = 0) results in an execution time of 1.0 to 1.8 µs.

Operands: B, W, D, K

Operation:
With arithmetic functions the operand is firstly expanded to the size of the Accumulator (32 bits). Then the REMAINDER is determined from a division of the contents of the Word Accumulator by the contents of the operand. The REMAINDER is stored in the Word Accumulator and may be processed further. If the MOD command is not correctly executed then the Marker M3170 is set, otherwise it is reset.

Example:
The REMAINDER of a division of the value stored in Word W6 by a constant is to be determined. The REMAINDER is then stored in Doubleword D8.

Initial state:	Word	W6	= 50	(dec)
	Constant	K	= 15	(dec)
	Doubleword	D8	= ?	

In the interests of clarity the contents of the Accumulator and the operand are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x x x x x x x x x x	
1	L W6	5 0	
2	MOD K15	5	5 0
3	= D8	5	5

Line 1: The contents of Word W6 are loaded into the Accumulator.
Line 2: The contents of the Accumulator are divided by the constant and the integer REMAINDER is left in the Accumulator.
Line 3: The REMAINDER is assigned to Doubleword D8.

3.4.6 INCREMENT (INC)

INCREMENT Operand
Abbreviation for the PLC Editor: INC
Operands: B, W, D

Operation:
The contents of the addressed operand increases by one.

INCREMENT Word Accumulator
Abbreviation for the PLC Editor: INCW

Operation:
The contents of the word accumulator increases by one.

INCREMENT Index Register
Abbreviation for the PLC Editor: INCX

Operation:
The contents of the index register increases by one.

3.4.7 DECREMENT (DEC)

DECREMENT Operand
Abbreviation for the PLC Editor: DEC
Operands: B, W, D

Operation:
The contents of the addressed operand decreases by one.

INCREMENT Word Accumulator
Abbreviation for the PLC Editor: DECW

Operation:
The contents of the word accumulator decreases by one.

INCREMENT Index Register
Abbreviation for the PLC Editor: DECX

Operation:
The contents of the index register decreases by one.

3.5 Comparisons

3.5.1 EQUAL TO (==)

Abbreviation for the PLC-Editor: == (EQUAL)

	Byte/Word/Double	Constant
Execution time [µs]	1.0 to 1.2	0.3 to 0.5
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, a direct transfer from Word to Logic processing occurs. The contents of the Word Accumulator and the contents of the addressed operand are compared. If the Word Accumulator and the operand are equal, the condition is true and the Logic Accumulator is set to 1. If they are not equal the Logic Accumulator is set to 0. The comparison takes place over the number of bits corresponding to the operand, i.e. B = 8 bit, W = 16 bit and D = K = 32 bit.

Example:
A constant is to be compared with the contents of Doubleword D8. The result is then assigned to Marker M500.

Initial state: Constant = 16 000
Doubleword D8 = 15 000

The Accumulator and operand contents are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x x x x x x x x x x	
1	L K16000	1 6 0 0 0	
	Bit	31 . . . 7 0	
2	== D8	... x x x x x x | 0 | x x x x x x x	1 5 0 0 0
3	= M500	... x x x x x x | 0 | x x x x x x x	0

Line 1: The constant is loaded into the Accumulator.
Line 2: The contents of the Accumulator and the Doubleword D8 are compared (Accumulator = Operand ?).
As the condition is not fulfilled the Logic Accumulator is set to 0.
Line 3: The contents of the Logic Accumulator (The result of the comparison) are assigned to Marker M500.

3.5.2 LESS THAN (<)

Abbreviation for the PLC-Editor: < (LESS THAN)

	Byte/Word/Double	Constant
Execution time [µs]	1.0 to 1.2	0.3 to 0.5
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, a direct transfer from Word to Logic processing occurs. The contents of the Word Accumulator are compared with the contents of the addressed operand. If the Word Accumulator is smaller than the operand, the condition is true and the Logic Accumulator is set to 1. If the Word Accumulator is smaller or equal to the operand, the Logic Accumulator is set to 0. The comparison takes place over the number of bits in the operand, i.e. B = 8 bit, W = 16 bit and D = K = 32 bit.

Example:
A constant is to be compared with the contents of Doubleword D8. The result is then assigned to Marker M500.

Initial state:	Constant	= 16 000
	Doubleword D8	= 15 000

The Accumulator and operand contents are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L K16000	16 000	
	Bit	31 . . . 7 0	
1	< D8	... xxxxxx | 0 | xxxxxxx	15 000
2	= M500	... xxxxxx | 0 | xxxxxxx	0

Line 1: The constant is loaded into the Accumulator
Line 2: The contents of the Accumulator and the Operand are compared (Accumulator < Operand ?).
As the condition is not fulfilled the Logic Accumulator is set to 0.
Line 3: The contents of the Logic Accumulator (The result of the comparison) are assigned to Marker M500.

3.5.3 GREATER THAN (>)

Abbreviation for the PLC-Editor: > (GREATER THAN)

	Byte/Word/Double	Constant
Execution time [µs]	1.0 to 1.2	0.3 to 0.5
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, a direct transfer from Word to Logic processing occurs. The contents of the Word Accumulator are compared with the contents of the addressed operand. If the Word Accumulator is greater than the operand, the condition is true and the Logic Accumulator is set to 1. If the Word Accumulator is less than or equal to the operand, the Logic Accumulator is set to 0. The comparison takes place over the number of bits in the operand, i.e. B = 8 bit, W = 16 bit and D = K = 32 bit.

Example:
A constant is to be compared with the contents of Doubleword D8. The result is then assigned to Marker M500.

Initial state:	Constant	= 16 000
	Doubleword D8	= 15 000

The Accumulator and operand contents are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L K16000	16 000	
	Bit	31 . . . 7 0	
1	> D8	... x x x x x x 1 x x x x x x x	15 000
2	= M500	... x x x x x x 1 x x x x x x x	1

Line 1: The constant is loaded into the Accumulator
Line 2: The contents of the Accumulator and the Operand are compared (Accumulator > Operand ?). As this condition is fulfilled the Logic Accumulator is set to 1.
Line 3: The contents of the Logic Accumulator (The result of the comparison) are assigned to Marker M500.

3.5.4 LESS THAN OR EQUAL TO (<=)

Abbreviation for the PLC-Editor: <= (LESS EQUAL)

	Byte/Word/Double	Constant
Execution time [µs]	1.0 to 1.2	0.3 to 0.5
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, a direct transfer from Word to Logic processing occurs. The contents of the Word Accumulator are compared with the contents of the addressed operand. If the Word Accumulator is less than or equal to the operand, the condition is true and the Logic Accumulator is set to 1. If the Word Accumulator is greater than the operand, the Logic Accumulator is set to 0. The comparison takes place over the number of bits in the operand, i.e. B = 8 bit, W = 16 bit and D = K = 32 bit.

Example:
A constant is to be compared with the contents of Doubleword D8. The result is then assigned to Marker M500.

Initial state: Constant = 16 000
Doubleword D8 = 15 000

The Accumulator and operand contents are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L K16000	16 000	
	Bit	31 . . . 7 0	
1	<= D8	... x x x x x x | 0 | x x x x x x x	15 000
2	= M500	... x x x x x x | 0 | x x x x x x x	0

Line 1: The constant is loaded into the Accumulator.
Line 2: The contents of the Accumulator and the Operand are compared (Accumulator <= Operand). As this condition is not fulfilled the Logic Accumulator is set to 0.
Line 3: The contents of the Logic Accumulator (The result of the comparison) are assigned to Marker M500.

3.5.5 GREATER THAN OR EQUAL TO (>=)

Abbreviation for PLC Editor: >= (GREATER EQUAL)

	Byte/Word/Double	Constant
Execution time [µs]	1.0 to 1.2	0.3 to 0.5
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, a direct transfer from Word to Logic execution occurs. The content of the Word Accumulator is compared with the content of the addressed operand. If the Word Accumulator is greater than or equal to the operand, the condition is true and the Logic Accumulator is set to 1. If the Word Accumulator is smaller than the operand, the Logic Accumulator is set to 0. The comparison takes place over the number of bits corresponding to the operand i.e. B=8 bit, W=16 bit and D=K=32 bit.

Example:
A constant is to be compared with the content of Doubleword D8. The result is then assigned to marker M500.

Initial state:	Constant	= 16 000
	Doubleword D8	= 15 000

Accumulator and operand contents are entered here in decimal notation. The ten-position Accumulator thus permits the maximum possible Accumulator content (2 147 483 647).

Line	Instruction	Accumulator Content	Operand Content
		x xxx xxx xxx	
1	L K16000	16 000	
	Bit	31 . . . 7 0	
1	>= D8	... x x x x x x | 1 | x x x x x x x	15 000
2	= M500	... x x x x x x | 1 | x x x x x x x	1

Line 1: The constant is loaded into the Word Accumulator.
Line 2: The contents of the Word Accumulator and operand are compared according to the following criteria: Word Accumulator >= Operand. As this condition is fulfilled, the Logic Accumulator is set to 1.
Line 3: The content of the Logic Accumulator (result of the comparison) is assigned to marker M500.

3.5.6 UNEQUAL (<>)

Abbreviation for PLC Editor: <> (NOT EQUAL)

	Byte/Word/Double	Constant
Execution time [µs]	1.0 to 1.2	0.3 to 0.5
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, a direct transfer from Word to Logic execution occurs. The content of the Word Accumulator is compared with the content of the addressed operand. If the Word Accumulator and the operand are not equal, the condition is true and the Logic Accumulator is set to 1. If the Word Accumulator is equal to the operand, the Logic Accumulator is set to 0. The comparison takes place over the number of bits corresponding to the operand i.e. B=8 bit,W=16 bit and D=K=32 bit.

Example:
A constant is to be compared with the contents of Doubleword D8. The result is then assigned to marker M500.

Output state	Constant	= 16 000
	Doubleword D8	= 15 000

Accumulator and operand contents are entered here in decimal notation. The ten position Accumulator thus permits the maximum possible Accumulator content (2 147 483 647).

Line	Instruction	Accumulator Content	Operand Content
		x x x x x x x x x x	
1	L K16000	1 6 0 0 0	
	Bit	31 . . . 7 0	
1	<> D8	... x x x x x x 1 x x x x x x x	1 5 0 0 0
2	= M500	... x x x x x x 1 x x x x x x x	1

Line 1: The constant is loaded into the Word Accumulator.
Line 2: Contents of the Word Accumulator and operand are compared according to the following criteria: Word Accumulator <> Operand. If this condition is fulfilled, the Logic Accumulator is set to 1.
Line 3: The contents of the Logic Accumulator [result of the comparison] is assigned to marker M500.

3.6 Parentheses with logical gating

Execution time and code length are summarised respectively for the "open-parentheses" and corresponding "close-parentheses" commands.

3.6.1 AND [ ] (A[ ])

Abbreviation for PLC Editor: A[ ] (AND [ ])

	Logic	Byte/Word/Double
Execution time [µs]	0.6 to 1.2	0.7 to 1.3
Number of bytes	6	6

Operands: none

3.6.2 AND NOT [ ] (AN[ ])

Abbreviation for PLC Editor: AN[ ] (AND NOT [ ])

	Logic	Byte/Word/Double
Execution time [µs]	0.5 to 1.2	0.7 to 1.5
Number of bytes	6	6

Operands: none

3.6.3 OR [ ] (O[ ])

Abbreviation for PLC Editor: O[ ] (OR [ ])

	Logic	Byte/Word/Double
Execution time [µs]	0.6 to 1.2	0.7 to 1.3
Number of bytes	6	6

Operands: none

3.6.4 OR NOT [ ] (ON[ ])

Abbreviation for PLC Editor: ON[ ] (OR NOT [ ])

	Logic	Byte/Word/Double
Execution time [µs]	0.5 to 1.2	0.7 to 1.5
Number of bytes	6	6

Operands: none

3.6.5 EXCLUSIVE OR [ ] (XO[ ])

Abbreviation for PLC Editor: XO[ ] (EXCL: OR [ ])

	Logic	Byte/Word/Double
Execution time [µs]	0.5 to 1.2	0.7 to 1.5
Number of bytes	6	6

Operands: none

3.6.6 EXCLUSIVE OR NOT [ ] (XON[ ])

Abbreviation for PLC Editor: XON[ ] (EXCL: OR NOT [ ])

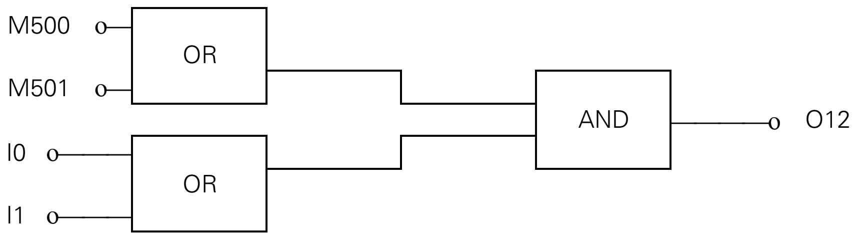
	Logic	Byte/Word/Double
Execution time [µs]	0.7 to 1.5	0.7 to 1.5
Number of bytes	8	8

Operands: none

Function of Parentheses with Logic Commands:
The execution sequence in a ladder may be altered by the use of parentheses. The "open-parentheses" command loads the contents of the Accumulator onto the Program Stack. If the Logic Accumulator is addressed in the previous command, prior to a "parentheses-open" instruction, the content of the Logic Accumulator is loaded into the Program Stack. By addressing the Word Accumulator, the content of the Word Accumulator will be distributed.

The "close-parentheses" instruction initiates the gating of the buffered value from the Program Stack with the Logic Accumulator and/or the Word Accumulator, depending on which Accumulator was addressed prior to the "parentheses-open" instruction. The result is then available in the corresponding Accumulator. The maximum nesting level is 16 parentheses.

Examples for the commands AND [ ], AND NOT [ ], OR [ ], OR NOT [ ], EXCLUSIVE OR [ ], EXCLUSIVE OR NOT [ ]. With the use of parentheses, an instruction listing may be developed according to the following logic block-diagram.



Initial state: Marker M500 = 0 Input I0 = 0 Output O12 = ?
Marker M501 = 1 Input I1 = 1

Line	Instruction	Accumulator Contents	Operand Contents
		bit 31 ... 7 ... 0: ... x x x x x x x | x | x x x x x x x	
1	L M500	... x x x x x x x | 0 | x x x x x x x	0
2	O M501	... x x x x x x x | 1 | x x x x x x x	1
3	A [	... x x x x x x x | 1 | x x x x x x x	
4	L I0	... x x x x x x x | 0 | x x x x x x x	0
5	O I1	... x x x x x x x | 1 | x x x x x x x	1
6	]	... x x x x x x x | 1 | x x x x x x x	
7	= O12	... x x x x x x x | 1 | x x x x x x x	1

Program-stack:

xxxxxxxx | 1 | xxxxxxx

bit 15 7 0

Line 1: Marker state M500 is loaded into the Logic Accumulator.
Line 2: The Logic Accumulator is gated with Marker M501.
Line 3: Open parentheses: the Accumulator contents are buffered on the Program Stack.
Line 4: Input state I0 is loaded into the Logic Accumulator.
Line 5: The Logic Accumulator is gated with Input I1.
Line 6: Close parentheses: Accumulator content is gated with the content of the Program Stack, according to the command (A[, O[, NO[...).
Line 7: The result of the complete logical process is assigned to Output O12.

Note:
The functional sequence is in principle the same for word execution, with the exception that the whole Accumulator is written onto the Stack.

3.7 Parentheses with arithmetic Commands

Execution time and code length are summarised respectively for the "open-parentheses" and corresponding "close-parentheses" commands.

3.7.1 ADD [ ] (+[ ])

Abbreviation for PLC Editor: + [ ] (PLUS [ ])

	Logic	Byte/Word/Double
Execution time [µs]	-----	0.9 to 2.2
Number of bytes	-----	6

Operands: none

3.7.2 SUBTRACT [ ] (–[ ])

Abbreviation for PLC Editor: – [ ] (MINUS [ ])

	Logic	Byte/Word/Double
Execution time [µs]	-----	0.9 to 2.1
Number of bytes	-----	6

Operands: none

3.7.3 MULTIPLICATION [ ] (x[ ])

Abbreviation for PLC Editor: x [ ] (MULTIPLY [ ])

	Logic	Byte/Word/Double
Execution time [µs]	-----	3.7 to 4.7
Number of bytes	-----	12

Operands: none

Marker M3168 is set if an error occurs.

3.7.4 DIVISION [ ] (/[ ])

Abbreviation for PLC Editor: / [ ] (DIVIDE [ ])

	Logic	Byte/Word/Double
Execution time [µs]	-----	6.5 to 7.5
Number of bytes	-----	16

In the event of an error (Divisor = 0) in the Division and MODULO functions, the execution time will be in the range 0.9 to 1.3 µs.

Operands: none

Marker M3169 is set if an error occurs.

3.7.5 REMAINDER [ ] (MOD[ ])

Abbreviation for PLC Editor: MOD [ ] (MODULO [ ])

	Logic	Byte/Word/Double
Execution time [µs]	-----	6.5 to 7.5
Number of bytes	-----	14

In the event of an error (Divisor = 0) in the Division and MODULO functions, the execution time will be in the range 0.9 to 1.3 µs.

Operands: none

Function of Parentheses with Arithmetic Commands:
With arithmetic commands, only word execution comes into question. The execution sequence in a ladder may be altered by the use of parentheses. The "open-parentheses" command loads the content of the Word Accumulator onto the Program Stack. Then the Accumulator is available for the calculation of intermediate results. The "close-parentheses" instruction initiates the gating of the buffered value from the Program Stack with the content of the Word Accumulator. The result is again loaded into the Accumulator. The maximum nesting level is 16 parentheses.

Marker M3170 is set if an error occurs.

Example for the commands ADD [ ], SUBTRACT [ ], MULTIPLY [ ], DIVIDE [ ], DIVISION REMAINDER [ ]
The following example demonstrates how parentheses influence the result of the operation.

Initial state:	Constant	= 1000		(decimal)
	Doubleword D12	= 15000	(decimal)	
	Doubleword D36	= 100	(decimal)	
	Doubleword D100	= ?		

The specification of Accumulator and operand contents is given in decimal notation. The ten-place Accumulator thus permits the maximum possible Accumulator content of (2 147 483 647).

Command sequence without parentheses:

Line	Instruction	Accumulator Content	Operand Content
		x xxx xxx xxx	
1	L D12	15 000	15 000
2	+ K1000	16 000	
3	/ D36	160	100
4	= D100	160	100

Command sequence with parentheses:

Line	Instruction	Accumulator Contents	Operand Contents
		x xxx xxx xxx	
1	L D12	15 000	15 000
2	+ [	15 000	
3	L K1000	1 000	
4	/ D36	10	100
5	]	15 010	
6	= D100	15 010	15 010

Program-stack:

15 000

Line 1: The content of Doubleword D12 is loaded into the Word Accumulator.
Line 2: Open parentheses: buffer the Accumulator content in the Program Stack.
Line 3: A constant is loaded into the Word Accumulator.
Line 4: The content of the Word Accumulator is divided by the content of Doubleword D12.
Line 5: Close parentheses: Accumulator content is gated, corresponding to the command (+[, -[, x[...) with the content of the Program Stack.
Line 6: The result of the complete logical process is assigned to Doubleword D100.

3.8 Parentheses with comparison Commands

Execution time and code length are summarized respectively for the "open-parenthesis" and the corresponding "close-parenthesis" commands.

3.8.1 EQUAL TO [ ] (==[ ])

Abbreviation for PLC Editor: == [ ] (EQUAL [ ])

	Logic	Byte/Word/Double
Execution time [µs]	----	0.5 to 0.8
Number of bytes	----	6

Operands: none

3.8.2 LESS THAN [ ] (<[ ])

Abbreviation for PLC Editor: < [ ] (LESS THAN [ ])

	Logic	Byte/Word/Double
Execution time [µs]	----	0.5 to 0.8
Number of bytes	----	6

Operands: none

3.8.3 GREATER THAN [ ] (>[ ])

Abbreviation for PLC Editor: > [ ] (GREATER THAN [ ])

	Logic	Byte/Word/Double
Execution time [µs]	----	0.5 to 0.8
Number of bytes	----	6

Operands: none

3.8.4 LESS THAN OR EQUAL TO [ ] (<=[ ])

Abbreviation for PLC Editor: <= [ ] (LESS EQUAL [ ])

	Logic	Byte/Word/Double
Execution time [µs]	----	0.5 to 0.8
Number of bytes	----	6

Operands: none

3.8.5 GREATER THAN OR EQUAL TOL[ ] (>=[ ])

Abbreviation for PLC Editor: >= [ ] (GREATER EQUAL [ ])

	Logic	Byte/Word/Double
Execution time [µs]	----	0.5 to 0.8
Number of bytes	----	6

Operands: none

3.8.6 NOT EQUAL TO [ ] (<>[ ])

Abbreviation for PLC Editor: <> [ ] (NOT EQUAL [ ])

	Logic	Byte/Word/Double
Execution time [µs]	----	0.5 to 0.8
Number of bytes	----	6

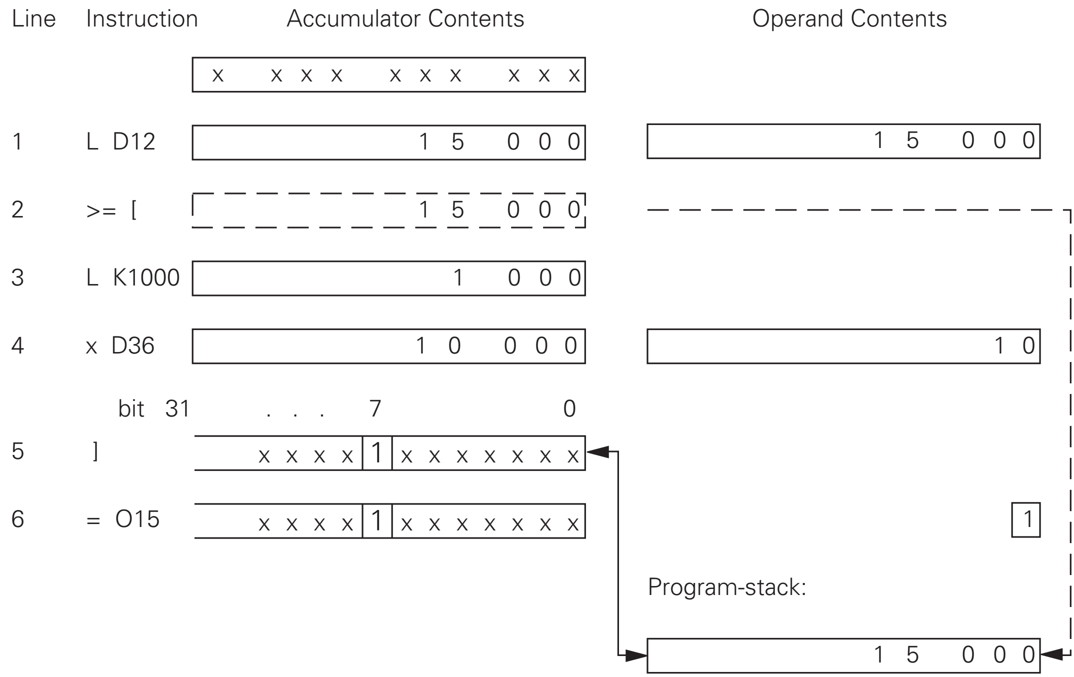
Operands: none

Function of parentheses with comparison commands:
The execution sequence in a ladder may be altered by the use of parentheses. The "open-parentheses" command loads the contents of the Word Accumulator onto the Program Stack. The Accumulator is now available for the calculation of intermediate results.
The "close-parentheses" instruction initiates the gating of the buffered value from the Program Stack with the content of the complete Word Accumulator. The result is loaded again into the Accumulator. The maximum nesting depth is 16 parentheses.
A direct transition from Word to Logic execution takes place with comparison commands. If the comparison condition is "true", the Logic Accumulator is set to "1". If the condition is not fulfilled, the Logic Accumulator is set to "0".

Example:

Initial state:	Constant	= 1000	(decimal)
	Doubleword D12	= 15000	(decimal)
	Doubleword D36	= 10	(decimal)
	Output O15	= ?	

The Accumulator contents and operand contents are shown in decimal notation. The ten-position Accumulator thus permits the maximum possible Accumulator content of 2 147 483 647.
The Accumulator is again represented in binary notation after program line 5, as the transition to logic execution occurs here.



Line 1: The content of Doubleword D12 is loaded into the Word Accumulator.
Line 2: Open parentheses: buffering of the Accumulator content in the Program Stack.
Line 3: Loading of a Constant into the Word Accumulator.
Line 4: The content of the Word Accumulator is multiplied by the content of Doubleword D12.
Line 5: Close parentheses: Word Accumulator content is gated, corresponding to the command(==[, >=[, <=[...) with the content of the Program Stack . The transition from Word to Logic processing occurs in this program line. The Logic Accumulator is set or reset, depending on the result of the comparison.
Line 6: The result of the complete logical process is assigned to output O15.

3.9 Shift Commands

3.9.1 SHIFT LEFT (<<)

Abbreviation for PLC Editor: << (SHIFT LEFT)

	Byte/Word/Double	Constant
Execution time [µs]	0.7 to 1.0	0.5 to 0.6
Number of bytes	6	8

Operands: B, W, D, K

Operation:
Since the sign bit (MSB) is included with this command, it is grouped in with arithmetic commands. For this reason and out of time considerations, this command should not be used for the isolation of bits. A SHIFT LEFT instruction causes the contents of the Word Accumulator to be multiplied by two. For this purpose, the bits in the Accumulator are simply shifted by one place to the left. The result must lie in the range of –2 147 483 648 to +2 147 483 647, otherwise the Accumulator contains an undefined value. The number of shift events is defined by the operand. The Accumulator is filled on the right side with nulls.

Initial state: Doubleword D8 = 3E 80 (hex)
Doubleword D12 = ?

The Accumulator content is shown here in binary notation, and the operand content in hexadecimal notation.

Line	Instruction	Accumulator Content	Operand Content
		xxxxxxxx xxxxxxxx xxxxxxxxx xxxxxxxxx	
1	L D8	00000000 00000000 00111110 10000000	00 00 3E 80
2	<< K+1	00000000 00000000 01111101 00000000	
3	<< K+1	00000000 00000000 11111010 00000000	
4	<< K+1	00000000 00000001 11110100 00000000	
5	<< K+1	00000000 00000011 11101000 00000000	
6	= D12	00000000 00000011 11101000 00000000	00 03 E8 00

Line 1: Load Doubleword D8 into the Accumulator.
Line 2 to 5: The content of the Word Accumulator is shifted to the left by the number of bits specified in the operand. The complete operation can also be undertaken with the command << K+4.
Line 6: The result is stored in the Doubleword D12.

3.9.2 SHIFT RIGHT (>>)

Abbreviation for PLC Editor: >> (SHIFT RIGHT)

	Byte/Word/Double	Constant
Execution time [µs]	0.6 to 0.9	0.3 to 0.5
Number of bytes	6	8

Operands: B, W, D, K

Operation:
Since the sign bit (MSB) is included with this command, it is grouped in with arithmetic commands. For this reason and out of time considerations, this command should not be used for the isolation of bits. A SHIFT RIGHT instruction causes the contents of the Word Accumulator to be divided by two. For this purpose, the bits in the Accumulator are simply shifted by one place to the right. The number of the shift operations is determined via the operand. Thus the set bits, which are shifted beyond the Accumulator to the right, are lost; the Accumulator is filled according to the sign, from the left-hand side. With operand contents greater than 32, the operand value Modulo 32 is used, i.e. the integer remainder from the division (operand value)/32.

Example:
The content of the Doubleword D8 is to be shifted four times to the right and then stored in D12.

Initial state: Doubleword D8 = 3E 80 (hex)
Doubleword D12 = ?

The Accumulator content is shown here in binary notation and the operand content in hexadecimal notation.

Line	Instruction	Accumulator Content	Operand Content
		xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx	
1	L D8	00000000 00000000 00111110 10000000	00 00 3E 80
2	>> K1	00000000 00000000 00011111 01000000	
3	>> K1	00000000 00000000 00001111 10100000	
4	>> K1	00000000 00000000 00000111 11010000	
5	>> K1	00000000 00000000 00000011 11101000	
6	= D12	00000000 00000000 00000011 11101000	00 00 03 E8

Line 1: Load Doubleword D8 into the Accumulator.
Line 2 to 5: The content of the Word Accumulator is shifted to the right by the number of bits specified in the operand. The complete operation can also be undertaken with the command >> K+4.
Line 6: The result is stored in Doubleword D12.

3.10 Bit Commands

3.10.1 BIT SET (BS)

Abbreviation for PLC Editor: BS (BIT SET)

	Byte/Word/Double	Constant
Execution time [µs]	0.5 to 0.8	0.3 to 0.4
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, each bit in the Accumulator can be acted on. The addressed bit is set to "1" through the use of the BS command. The selection (addressing) of the corresponding bit is derived from the content of the specified Operand or a Constant. In the bit-numbering, bit 0 corresponds to the LSB and bit 31 the MSB. For operand contents larger than 32, the operand value Modulo 32 is used, i.e. the integer remainder from the division (operand value)/32.

Example:
Load Doubleword D8 in the Accumulator, set the bit 0 of the Accumulator to "1" and store the result in Doubleword D12.

Initial state: Doubleword D8 = 3E 80 (hex)
Doubleword D12 = ?

Accumulator and operand contents are shown here in hexadecimal notation.

Line	Instruction	Accumulator Content	Operand Content
		xx xx xx xx	
1	L D8	00 00 3E 80	00 00 3E 80
2	BS K+0	00 00 3E 81	
3	= D12	00 00 3E 81	00 00 3E 81

Line 1: Load Doubleword D8 into the Accumulator.
Line 2: The bit specified in the operand is set to 1.
Line 3: The result is stored in Doubleword D12.

3.10.2 BIT RESET (BC)

Abbreviation for PLC Editor: BC (BIT CLEAR)

	Byte/Word/Double	Constant
Execution time [µs]	0.5 to 0.8	0.3 to 0.4
Number of bytes	6	8

Operands: B, W, D, K

Operation:
With this command, each bit in the Accumulator can be acted on. The addressed bit is set to "0" through the use of the BC command. The selection (addressing) of the corresponding bit is derived from the content of the specified Operand or a Constant. In the bit-numbering, bit 0 corresponds to the LSB and bit 31 the MSB. For operand contents larger than 32, the operand value Modulo 32 is used, i.e. the integer remainder from the division (operand value)/32.

Example:
Load Doubleword D8 in the Accumulator, set bit 0 of the Accumulator to "0" and store the result in Doubleword D12.

Initial state: Doubleword D8 = 3E 81 (hex)
Doubleword D12 = ?

Accumulator and operand contents are shown here in hexadecimal notation.

Line	Instruction	Accumulator Content	Operand Content
		xx xx xx xx	
1	L D8	00 00 3E 81	00 00 3E 81
2	BC K+0	00 00 3E 80	
3	= D12	00 00 3E 80	00 00 3E 80

Line 1: Load Doubleword D8 into the Accumulator.
Line 2: The bit specified in the operand is set to "0".
Line 3: The result is stored in Doubleword D12.

3.10.3 BIT TEST (BT)

Abbreviation for PLC Editor: BT (BIT TEST)

	Byte/Word/Double	Constant
Execution time [µs]	0.6 to 1.1	0.3 to 0.5
Number of bytes	8	0

Operands: B, W, D, K

Operation:
With this command, the status of each individual bit in the Accumulator may be interrogated. With BT commands, a direct transition from Word to Logic execution takes place. The BIT TEST tests the status of a bit from the Word Accumulator and then acts correspondingly on the Logic Accumulator. If the tested bit is "1", then the Logic Accumulator is also set to "1"; if it is "0" ,it is set to "0". The program continues in logic execution. The selection (addressing) of the corresponding bit is derived from the content of the specified Operand or a Constant. In the bit-numbering, bit 0 corresponds to the LSB and bit 31 the MSB. For operand contents larger than 32, the operand value Modulo 32 is used, i.e. the integer remainder from the division (operand value)/32.

Example:
Load Doubleword D8 in the Accumulator, and assign the logic state of bit 0 to an Output.

Initial state: Doubleword D8 = 3E 81 (hex)
Output O12 = ?

Word Accumulator and operand contents are shown here in hexadecimal notation, the Logic Accumulator in binary representation.

Line	Instruction	Accumulator Content	Operand Content
		xx xx xx xx	
1	L D8	00 00 3E 81	00 00 3E 81
2	BT K+0	00 00 3E 81	
3	= O12	x x x x x x 1 x x x x x x x	1

Line 1: Load Doubleword D8 into the Accumulator.
Line 2: The bit specified in the operand is tested as to its status.
Line 3: The Logic Accumulator is assigned to Output O12.

3.11 Stack Operations

It should be noted that with Stack operations all read/write operations on the Data Stack take place according to the LIFO principle (Last In – First Out).

3.11.1 Load Data onto the Data Stack (PS)

Abbreviation for PLC Editor: PS (PUSH)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	1.4 to 2.0	1.5 to 2.0	1.3 to 1.8	1.0 to 1.1
Number of bytes	24	26	22	24

Logic Execution with the PS Command

Operands: M, I, O, T, C

Operation:
With the PS command, data can be buffered. Thus the addressed operand is loaded onto the Data Stack. Since the Data Stack is organised as 16 bit, a minimum width of one Word must be used in writing to it. During this the operand value is copied into bit 7 of the current address in the Data Stack. The free bits of the reserved memory are undefined or unused. In the event of a Stack overflow, an error message will be issued.

Memory allocation in the Data Stack:

Bit 15	7	0
x x x x x x x x	L	x x x x x x x

Word Execution with the PS Command

Operands: B, W, D, K

Operation:
With the PS command, data can be buffered. Thus the addressed memory area (B, W, D, K) is copied into the current address of the Data Stack. With Word execution, two Words are reserved as standard on the Data Stack per PS command. The operand is extended in the Stack with sign justification corresponding to the MSB. In the event of a Stack overflow, an error message will be issued.

Memory allocation in the Data Stack upon saving of:

	Bit 31	15 7 0
Byte	X X X X X X X X X X X X X X X X	X X X X X X X X B B B B B B B B
Word	X X X X X X X X X X X X X X X X	WWWWWWWWWWWWWWWW
Doubleword	D D D D D D D D D D D D D D D D	D D D D D D D D D D D D D D D D
Constant	K K K K K K K K K K K K K K K K	K K K K K K K K K K K K K K K K

3.11.2 Acquire Data from the Data Stack (PL)

Abbreviation for PLC Editor: PL (PULL)

	Logic	Byte/Word	Double	Constant
Execution time [µs]	1.1 to 1.5	0.9 to 1.3	0.9 to 1.3	----
Number of bytes	20	20	18	----

Logic Execution with the PL Command

Operands: M, I, O, T, C

Operation:
The PL command complements the PS command. Data which are saved with PUSH can be taken from the Data Stack again with PULL. With logic execution, bit 7 is copied from the current address of the Data Stack into the addressed operand with a PL command. If the Stack is empty, an error message will be issued.

Logic Execution with the PL Command

Operands: B, W, D

Operation:
The PL command complements the PS command. Data which are saved with PUSH can be taken from the Data Stack again with PULL. With Word execution, two Words are copied from the current address of the Data Stack into the addressed memory area with a PL command. If the Stack is empty, an error message will be issued.

3.11.3 Load Logic Accumulator onto the Data Stack (PSL)

Abbreviation for PLC Editor: PSL (PUSH LOGICACCU)

Execution time [µs]	0.6 to 1.0
Number of bytes	20

Operands: none

Operation:
The Logic Accumulator can be buffered with the PSL command. For this purpose, the Logic Accumulator is loaded onto the Data Stack. Since the Data Stack is organised as 16 bits, it must be written to with a minimum width of one Word. During this the content of the Logic Accumulator is copied into the current address of the Data Stack. The free bits of the reserved memory are undefined or unused. In the event of a Stack overflow, an error message will be issued.

Memory allocation in the Data Stack:

Bit 15	7	0
x x x x x x x x	L	x x x x x x x

3.11.4 Load Word Accumulator onto the Data Stack (PSW)

Abbreviation for PLC Editor: PSW (PUSH WORDACCU)

Execution time [µs] 1.0 to 1.1
Number of bytes 20

Operands: none

Operation:
The content of the Word Accumulator can be buffered with the PSW command. For this purpose, the Word Accumulator is copied into the Data Stack. The content of the Word Accumulator (32 bit) reserves two Words on the Data Stack. In the event of a stack overflow, an error message will be issued.

3.11.5 Acquire Logic Accumulator from the Data Stack (PLL)

Abbreviation for PLC Editor: PLL (PULL LOGICACCU)

Execution time [µs] 0.6 to 1.0
Number of bytes 16

Operands: none

Operation:
The PLL command complements the PSL command. With a PLL instruction, bit 7 from the current address of the Data Stack is copied into the Logic Accumulator. If the stack is empty, an error message will be issued.

3.11.6 Acquire Word Accumulator from the Data Stack (PLW)

Abbreviation for PLC Editor: PLW (PULL WORDACCU)

Execution time [µs] 1.0 to 1.1
Number of bytes 16

Operands: none

Operation:
The PLW command complements the PSW command. With a PLW instruction, two Words are copied from the Data Stack into the Word Accumulator. If the stack is empty, an error message will be issued.

Examples for the commands PS, PL, PSL, PSW, PLL, PLW
The Module 15 is to be called at a specific point in the program. After the return into the main program, the original Accumulator content is again required for further program run.

Accumulator contents prior to the Call Module: 1A 44 3E 18

Line	Instruction	Accumulator Contents
	.	
	.	
49	.	1 A 4 4 3 E 1 8
50	PSW	1 A 4 4 3 E 1 8
52	CM 15	
54	PLW	1 A 4 4 3 E 1 8
55	.	
	.	
	.	

Data stack:

1 A 4 4
3 E 1 8

Line 50: Save the Word Accumulator onto the Data Stack.
Line 52: Subprogram 15 is called up.
Line 54: The original Accumulator contents are transferred back from the Data Stack and is available for further program run.

Note:
The sequence for stack operations is the same for all commands. Only the data width varies.

3.12 Jump Commands

3.12.1 Unconditional Jump (JP)

Abbreviation for PLC Editor: JP (JUMP)

	Jump processed	Jump not processed
Execution time [µs]	0.6 to 1.0	----
Number of bytes	8	

Operands: jump address (LBL)

Operation:
A JP command instructs the processor to continue the program at the specified jump address (Label). This command interrupts a logic sequence.

3.12.2 Jump if Logic Accumulator = 1 (JPT)

Abbreviation for PLC Editor: JPT (JUMP IF TRUE)

	Jump processed	Jump not processed
Execution time [µs]	0.4 to 1.3	0.1 to 0.5
Number of bytes	12 (10)	

Byte information in brackets:
A shorter command is employed in certain high-priority program sequences.

Operands: jump address (LBL)

Operation:
A JPT command is a conditional jump command. If the Logic Accumulator is "1", the program is continued from the specified jump address (Label). If the Logic Accumulator is "0" the jump is not processed. This command interrupts a logic sequence.

3.12.3 Jump if Logic Accumulator = 0 (JPF)

Abbreviation for PLC Editor: JPF (JUMP IF FALSE)

	Jump processed	Jump not processed
Execution time [µs]	0.4 to 1.3	0.1 to 0.5
Number of bytes	12 (10)	

Byte information in brackets:
A shorter command is employed in certain high-priority program sequences.

Operands: jump address (LBL)

Operation:
A JPF command is a conditional jump command. If the Logic Accumulator is "0", the program is continued from the specified jump address (Label). If the Logic Accumulator is "1", the jump is not processed. This command interrupts a logic sequence.

Example for the commands JP, JPT, JPF

A certain program section is to be skipped, depending on Input 15.

Initial state: Input I5 = 1

Line	Instruction	Accumulator Content	Operand Content
	Bit	31 . . . 7 0	
		... x x x x x x X x x x x x x x	
1	L I5	... x x x x x x 1 x x x x x x x	1
2	JPT 10	... x x x x x x 1 x x x x x x x	
3	L I3		
4	O M500		
5	= O20		
6	LBL 10		
7	L M100	... x x x x x x 0 x x x x x x x	0

•
•
•

Line 1: Load the operand contents in the Accumulator.
Line 2: Dependent on Input I5, a program jump is processed.
Line 3: Skipped in this example.
Line 4: Skipped in this example.
Line 5: Skipped in this example.
Line 6: Jump address: The program run is continued from here.

3.12.4 Call Module (CM)

Abbreviation for PLC Editor: CM (CALL MODULE)

	Jump processed	Jump not processed
Execution time [µs]	1.4 to 2.2	----
Number of bytes	22	
Special Library Call:		
Execution time [µs]	1.6 to 2.1	----
Number of bytes	24	

Operands: jump address (LBL)

Operation:
A Call Module instructs the processor to leave the main program and process the Module designated by the jump address (LBL). Modules are independent subprograms and are terminated by the command EM. They can also be called at multiple points in the main program. This command interrupts a logic sequence.

3.12.5 Call Module if Logic Accumulator = 1 (CMT)

Abbreviation for PLC Editor: CMT (CALL MODULE IF TRUE)

	Jump processed	Jump not processed
Execution time [µs]	1.5 to 2.5	0.1 to 0.5
Number of bytes	26 (24)	
Special Library Call:		
Execution time [µs]	2.0 to 2.4	0.4 to 0.5
Number of bytes	28	

Byte information in brackets:
A shorter command is employed in certain high-priority program sequences.

Operands: jump address (LBL)

Operation:
A CMT command is a conditional Call Module. If the Logic Accumulator is "1", the Module with the specified jump address (Label) is processed. If the Logic Accumulator is "0", the main program continues without a Call Module. This command interrupts a logic sequence.

3.12.6 Call Module if Logic Accumulator = 0 (CMF)

Abbreviation for PLC Editor: CMF (CALL MODULE IF FALSE)

	Jump processed	Jump not processed
Execution time [µs]	1.5 to 2.5	0.1 to 0.5
Number of bytes	26 (24)	
Special Library Call:		
Execution time [µs]	2.0 to 2.4	0.4 to 0.5
Number of bytes	28	

Byte information in brackets:
A shorter command is employed in certain high-priority program sequences.

Operands: jump address (LBL)

Operation:
A CMF command is a conditional Call Module. If the Logic Accumulator is "0", the Module with the specified jump address (Label) is processed. If the Logic Accumulator is "1", the main program continues without a Call Module. This command interrupts a logic sequence.

Example for the commands CM, CMT, CMF

A certain Module is to be called, depending on Input I5.

Initial state: Input I5 = 0

Line	Instruction	Accumulator Contents	Operand Contents
	Bit	31 . . . 7 0	
		... x x x x x x X x x x x x x x	
1	L I5	... x x x x x x 0 x x x x x x x	0
2	CMF 10	... x x x x x x 0 x x x x x x x	
3	L M100	... x x x x x x 1 x x x x x x x	1
	•		
	•		
	•		
499	EM		
500	LBL 10		
501	L I3	... x x x x x x 0 x x x x x x x	0
502	OM 500	... x x x x x x 1 x x x x x x x	1
503	= O20	... x x x x x x 1 x x x x x x x	1
504	EM		

Line 1: Load the operand contents in the Accumulator.
Line 2: Dependent on Input I5, the Call Module is processed.
Line 499: End Module of the main program.
Line 500: Start of the Module, identified by LBL.
Line 501: Instruction in the subprogram.
Line 502: Instruction in the subprogram.
Line 503: Instruction in the subprogram.
Line 504: End Module: Effects the return to the main program.
Line 3: The main program continues at this point once the Module is processed.

3.12.7 End of Module, Program End (EM)

Abbreviation for PLC Editor: EM (END OF MODULE)

Execution time [µs]	3.3 to 3.8
Number of bytes	4

Operands: none

Operation:
Every program and/or every subprogram (Module) is terminated with an EM command. An EM command in a Module initiates the return jump to the Call Module (CM, CMT, CMF). The program is continued with the instruction following the Call Module. The command EM is handled as program end criterion; thus subsequent program instructions can be reached using a jump address.

3.12.8 End of Module if Logic Accumulator = 1 (EMT)

Abbreviation for the PLC Editor: EMT (END OF MODULE IF TRUE)

Operation:
An EMT command only initiates a return jump to the Call Module (CM, CMT, CMF) when the Logic Accumulator is "1".

3.12.9 End of Module if Logic Accumulator = 0 (EMF)

Abbreviation for the PLC Editor: EMF (END OF MODULE IF FALSE)

Operation:
An EMF command only initiates a return jump to the Call Module (CM, CMT, CMF) when the Logic Accumulator is "0".

3.12.10 Jump Label (LBL)

Abbreviation for PLC Editor: LBL (LABEL)

Execution time [µs]	0
Number of bytes	0

Operands: ASCII name (up to 32 characters long)

Operation:
The jump label defines a program position as an entry point for the CM and JP commands. Jump labels may be allocated addresses in the range 0 to 511. Up to 1000 jump labels per file can be defined.

The ASCII name of the jump label may be up to 32 characters long, but only the first 16 characters are used to distinguish jump labels.
For importing global jump labels, see EXTERN instruction.

3.13 INDEX-Register

Under the control of the PLC programmer this register can be used for data transfer, intermediate storage of results and for index addressing of operands. The register is 32 bits wide but only the lower 16 bits are used for index addressing. The X register can be used anywhere in the program – there is no contents validity check – however there is a check for address space overflow with indexed write accesses.

Example: = B100[X]

X can assume values from -100 to +923 which corresponds to addressing the complete available address space for the Byte operand type. If the address space is overshot the error message "ERROR IN PLC PROGRAM 56" flashes in the display. Reset with END to display the error line in the PLC Editor.

Before using a command with the index-register it must be assigned a defined value. At the beginning of each PLC cycle the index register is set to 0.

The following operands can be addressed.

Mn[X]	
In[X]	
On[X]	
Cn[X]	
Tn[X]	Operand number = n+X
Bn[X]	Operand number = n+X
Wn[X]	Operand number = n+2*X
Dn[X]	Operand number = n+4*X
Sn[X]	String number = n+X
S#Dn[X]	Dialogue text number = n+X
S#En[X]	Error text number = n+X
Sn^X	Substring from X-th characters of n-th string

The types S"", K and K$ cannot be indexed.
Caution: When addressing S#Dn[X] and S#En[X] the sequence <SUB>Dnnn or <SUB>Ennn is loaded in the string accumulator, with nnn as the modified string number!

Commands for operating the Index Register:

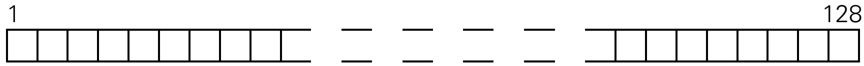
The following commands have been introduced to permit data interchange between the Word Accumulator and the Index Register or between the Stack and Index Register:

LX	(Load Index to Accu)	Index Register --> Word Accumulator
=X	(Store Accu to Index)	Word Accumulator --> Index Register
PSX	(Push Index Register)	Index Register --> Stack
PLX	(Pull Index Register)	Stack --> Index Register
INCX	(Increment Index Register)	
DECX	(Decrement Index Register)	

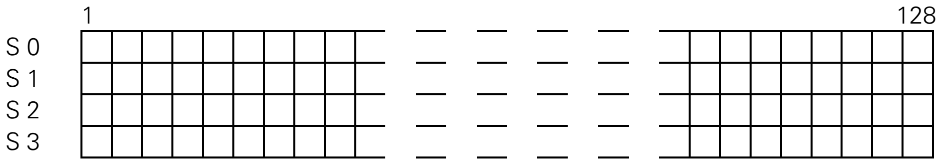
3.14 Commands for STRING Execution

STRING execution allows the creation and manipulation of any texts via the PLC program. These texts may be displayed in the PLC window of the screen by the use of Module 9082, and/or deleted again with Module 9080 (refer to PLC modules). A STRING Accumulator and four STRING memories are provided in the control for STRING execution. A maximum of 128 characters may be loaded into this.

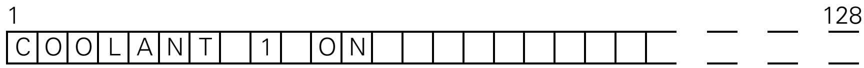
STRING accumulator: 128 Characters



STRING memory: 128 Characters



Example:



STRING Accumulator and STRING memory are volatile, and so are again deleted upon powering off. For STRING execution, the new operand "S" has been introduced. The operand "S" may be used with various arguments.

Explanation of the Operand:

The operand "S" is only used in STRING execution. The following locations may be addressed with the various arguments:

– STRING memory: Should a STRING memory be addressed, the number of the required memory (S0-S3) must be specified after the Operand-Designation.

– Part of a STRING: If only part of a STRING is to be addressed, then this can be done by addressing Sn^X (see INDEX REGISTER).
  The substring will be addressed from the X-th character in the specified STRING.

– Immediate STRING: A STRING can also be entered directly into the PLC program. The Text STRING, which may contain 0 – 37 characters, must be identified by quotation marks.
  Example: S "COOLANT 1 ON"

– Text from the PLC-Error Message files and/or from the PLC-Dialogue files: Text from the active error message or dialogue files may be read by the input of the line number.

PLC-ERROR:	S#Exx	xx: Line Number from the PLC-Error Message file (0 to 999)
PLC-DIALOG:	S#Dxx	xx: Line Number from the PLC-Dialogue file (0 to 4095)

The character sequence #Exx or #Dxx is entered in the Argument <arg> for the STRING-Command. A 5 Byte long character train <SUB> E0xx or <SUB> D0xx is loaded into the Accumulator (<SUB> = ASCII <SUB>). Instead of this character train, the line xx of the active error message or dialogue file is read for display on the screen.

Note:
The execution times depend on the length of the STRINGS. The specified times indicate maximum values. For the Immediate STRINGS, the length "n" of the STRINGS must respectively be added to the command length; should this be odd, the next larger even length must be added.

3.14.1 LOAD (L)

Abbreviation for PLC Editor: L (LOAD)

Execution time [µs]		< 80
Number of bytes	STRING memory:	10
	Immediate STRING:	18 + n
	STRING from error message or dialogue files	24

Operands: S <arg>

Operation:
The STRING Accumulator is loaded with this L command. The selection of the STRINGS to be loaded, proceeds using the Argument <arg> after the operand designation.
Refer also to operand explanation.

3.14.2 ADD (+)

Abbreviation for PLC Editor: +

Execution time [µs]		< 80
Number of bytes	STRING memory:	10
	Immediate STRING:	18 + n
	STRING from error message or dialogue files	24

Operands: S <arg>

Operation:
With this command another STRING is added to the STRING in the STRING Accumulator. The selection of the STRINGS, which should be added, proceeds using the Argument <arg> after the operand designation. Refer also to operand explanation. The resultant STRING must not be longer than 128 characters.

3.14.3 Storing a STRING (=)

Abbreviation for PLC Editor: =

Execution time [µs]		< 80
Number of bytes	STRING memory	10
	Immediate STRING	—
	STRING from error message or dialogue files	—

Operands: S <arg>

Operation:
With the = command a STRING from the STRING Accumulator is stored in a STRING memory. The selection of the memory, into which the STRING should be copied, proceeds using the Argument <arg> after the operand designation. Whereby only the Arguments 0 – 3, which address a STRING memory (S0 – S3) are valid here.
Refer also to operand explanation.

3.14.4 Overwriting of a STRING (OVWR)

Abbreviation for PLC Editor: OVWR (OVERWRITE)

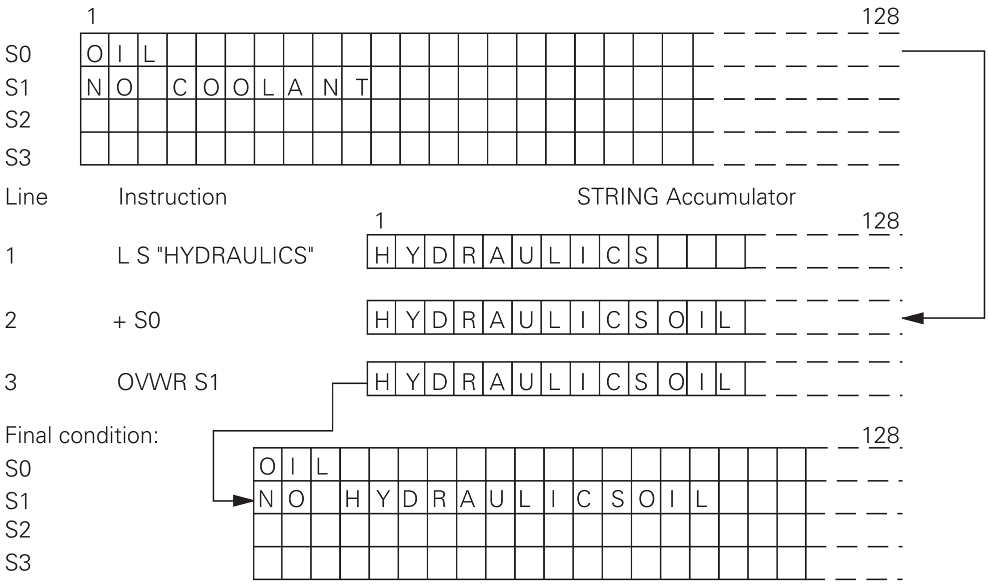
Execution time [µs]		< 80
Number of bytes	STRING memory	10
	Immediate STRING	—
	STRING from error message or dialogue files	—

Operands: S <arg>

Operation:
With the OVWR command a STRING from the STRING Accumulator is stored in a STRING memory. This command functions in a similar manner to the = command, with the difference that the character "STRING-End" is not transferred alongside. By this means, the beginning of a STRING which is already in the STRING memory, can be overwritten.
The selection of the memory, into which the STRING should be copied, proceeds using the Argument <arg> after the operand designation. Whereby only the Arguments 0 – 3, which address a STRING memory (S0 – S3) are valid here.
Refer also to operand explanation.

Example of STRING execution:
A STRING from the STRING memory S0 is to be added to an Immediate STRING. This STRING must overwrite the contents STRING memory S1.

Initial condition:	Immediate STRING:	HYDRAULICS
	STRING memory S0:	OIL
	STRING memory S1:	NO COOLANT



Line 1: Load the immediate STRING into the STRING accumulator.
Line 2: The contents of the STRING memory S0 is added to the contents of the STRING accumulator.
Line 3: The STRING accumulator overwrites the contents of the STRING memory S1.

3.15 Logical Comparisons in STRING Execution

Two STRINGS are compared according to the argument as follows:
If STRING memory or Immediate STRING are entered in the command, both STRINGS are compared character for character. The Logic Accumulator is reset after the first character for which the comparison conditions are not fulfilled. The remaining characters are checked no further. For the purposes of comparison, the number of the character in the ASCII table is always used. This results in, for example:
A < B
AA > A

If PLC-Error messages or PLC-Dialogue texts are entered, the position in the file (0 to 4095) is compared, not the actual text as with Immediate STRING.

The execution times depend on the length of the STRINGS. The quoted times represent maximum values. With the Immediate STRINGS, the length "n" of the STRINGS respectively must be added to the command length. In the event that this is odd, the next larger even-numbered length must be added.

3.15.1 EQUAL TO (==)

Abbreviation for PLC Editor: == (EQUAL)

Execution time [µs]		< 100
Number of bytes	STRING memory	12
	Immediate STRING	20 + n
	STRING from error message or dialogue files	26

Operands: S <arg>

Operation:
With this command a direct transition from STRING- to logic execution takes place. The content of the STRING Accumulator is compared with the STRING in the Argument.
If the STRING Accumulator and the operand are equal, the condition is true and the Logic Accumulator is set to 1. If they are not equal the Logic Accumulator is set to 0.

3.15.2 LESS THAN (<)

Abbreviation for PLC Editor: < (LESS THAN)

Execution time [µs]		< 100
Number of bytes	STRING memory	12
	Immediate STRING	20 + n
	STRING from error message or dialogue files	26

Operands: S <arg>

Operation:
With this command a direct transition from STRING to Logic execution takes place. The content of the STRING Accumulator is compared with the STRING in the Argument. If the STRING Accumulator is smaller than the operand, the condition is true and the Logic Accumulator is set to 1. If the STRING Accumulator is greater than or equal to the operand the Logic Accumulator is set to 0.

3.15.3 GREATER THAN (>)

Abbreviation for PLC Editor: > (GREATER THAN)

Execution time [µs]		< 100
Number of bytes	STRING memory	12
	Immediate STRING	20 + n
	STRING from error message or dialogue files	26

Operands: S <arg>

Operation:
With this command a direct transition from STRING- to logic execution takes place. The content of the STRING Accumulator is compared with the STRING in the Argument.
If the STRING Accumulator is greater than the operand, the condition is true and the Logic Accumulator is set to 1. If the STRING Accumulator is less than or equal to the operand the Logic Accumulator is set to 0.

3.15.4 LESS THAN OR EQUAL TO (<=)

Abbreviation for PLC Editor: <= (LESS EQUAL)

Execution time [µs]		< 100
Number of bytes	STRING memory	12
	Immediate STRING	20 + n
	STRING from error message or dialogue files	26

Operands: S <arg>

Operation:
With this command a direct transition from STRING- to logic execution takes place. The content of the STRING Accumulator is compared with the STRING in the Argument.
If the STRING Accumulator is less than or equal to the operand, the condition is true and the Logic Accumulator is set to 1. If the STRING Accumulator is greater than the operand the Logic Accumulator is set to 0.

3.15.5 GREATER THAN OR EQUAL TO (>=)

Abbreviation for PLC Editor: >= (GREATER EQUAL)

Execution time [µs]		< 100
Number of bytes	STRING memory	12
	Immediate STRING	20 + n
	STRING from error message or dialogue files	26

Operands: S <arg>

Operation:
With this command a direct transition from STRING- to logic execution takes place. The content of the STRING Accumulator is compared with the STRING in the Argument. If the STRING Accumulator is greater than or equal to the operand, the condition is true and the Logic Accumulator is set to 1. If the STRING Accumulator is less than the operand the Logic Accumulator is set to 0.

3.15.6 UNEQUAL (<>)

Abbreviation for PLC Editor: <> (NOT EQUAL)

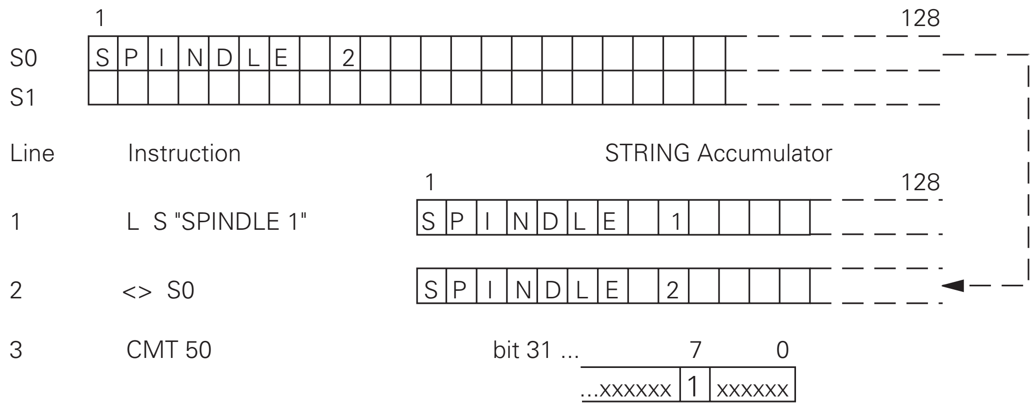
Execution time [µs]		< 100
Number of bytes	STRING memory	12
	Immediate STRING	20 + n
	STRING from error message or dialogue files	26

Operands: S <arg>

Operation:
With this command a direct transition from STRING- to logic execution takes place. The content of the STRING Accumulator is compared with the STRING in the Argument.
If the STRING Accumulator is not equal to the operand, the condition is true and the Logic Accumulator is set to 1. If the STRING Accumulator is equal to the operand the Logic Accumulator is set to 0.

Example of STRING execution
An Immediate STRING is to be compared with the content of the STRING-Memory S0. Depending on the comparison result, Module 50 is called.

Initial state: STRING memory S0: SPINDLE 2
Immediate STRING: SPINDLE 1



Line 1: Load the Immediate STRING into the STRING Accumulator.
Line 2: The content of the STRING Accumulator is compared with the content of the STRING memory S0 according to the command.
Line 3: Since the result of the comparison is "true", the Logic Accumulator is set and the Call Module is processed.

3.16 Submit Programs

Submit programs are subprograms which the PLC submits to the NC for processing. This allows tasks to be performed which are very processor-intensive, require program loops or must wait for external results. It is assumed, however, that these programs are not bound by a particular time frame. Depending on processor loading, each Submit program is allocated a certain computing power, but always at least 5% of the total power. Submit programs are started from the PLC program and can access all the same data memories (M/B/W/D) as can the main program. This can lead to problems in certain circumstances. Such problems can be avoided if the data processed by the PLC program are clearly separated from the data processed by the Submit program.
Up to eight Submit programs can be entered in a queue (Submit Queue). Each receives an "Identifier", a number between 1 and 255 allocated by the NC, which is transferred into the Word Accumulator. With this "Identifier" and the REPLY function, it can be interrogated whether or not the program is in the queue, is being processed or is already complete. The Submit programs are executed in the order of their placement in the queue. Should an error occur during the execution of the Submit programs, the following Markers are set:

M3168: Overflow during Multiplication
M3169: Division by 0
M3170: MODULO incorrectly executed
M3171: Error status for PLC module
M3172: Reserved for errors, which the PLC programmer would like to intercept

These markers are listed separately in the submit job. This means that the same markers can be edited as those in the PLC run program without changing the original markers.

Exact times cannot be given for the commands for the management of the Submit queue. The execution times denote maximum values.

3.16.1 Call up of the Submit Program (SUBM)

Abbreviation for PLC Editor: SUBM (SUBMIT)

Execution time [µs]	< 30
Number of bytes	10

Operands: jump address (LBL)

Operation:
The SUBM command allots an "Identifier" (1 to 255) to the subprogram, designated by the jump address (LBL). Simultaneously, the allocated number is written to the Word Accumulator. If there are already programs transferred into the Submit queue, the addressed program will not be processed until the program immediately prior to it is finished. A submission to the queue may only take place from a PLC program, a SUBM command in a Submit program is not possible.

If no location is free in the queue, or if the SUBM command is programmed in a Submit program (nesting), a "0" will be returned to the Word Accumulator.

3.16.2 Status Interrogation of a Submit Program (RPLY)

Abbreviation for PLC Editor: RPLY (REPLY)

Execution time [µs] < 5
Number of bytes 10

Operands: B, W, D

Operation:
With the RPLY command the Status of the Submit program is interrogated with the specified Identifier. This Identifier must already be stored in a Byte, Word or Doubleword prior to the calling up of the Submit program. With the RPLY command and the memory address specified above, which contains the Identifier, one of the following messages about the status is transferred to the Word Accumulator:

Word Accumulator 0: Program complete/not in the queue
Word Accumulator 1: Program running
Word Accumulator 2: Program in the queue

3.16.3 Cancellation of a Submit Program (CAN)

Abbreviation for PLC Editor: CAN (CANCEL)

Execution time [µs] < 40
Number of bytes 10

Operands: B, W, D

Operation:
With the CAN command the Submit-Program with the specified Identifier is cancelled during execution or removed from the queue. This Identifier must already be stored in a Byte, Word or Doubleword prior to the calling up of the Submit-Program. After the cancellation of the Program, the next Submit program in the queue will immediately be processed.
The following PLC modules cannot be cancelled with CANCEL at any desired point:
– PLC module for access to the screen (908X).
– PLC module for reading NC files (909X).
In these cases, the RPLY command must be used to check whether or not the CAN command may be used.

Example of the use of the SUBM command:
Dependent on Input I10 the subprogram with the Label LBL 300 is handed over to the NC for processing. In addition, the execution of the subprogram is checked in the main program with the RPLY command and cancelled with the CAN command in conjunction with Input I11.

Line	Instruction		Program Comments:
1	L	I10	;Interrogate state of Input I10
2	JPF	100	;Dependent on Input I10 skip
			;Call Module
3	RPLY	B 128	;Interrogate status of the Submit program
4	<>	K+0	;Submit program already transferred to
			;the NC for processing ?
5	JPT	100	;If program already transferred to the NC,
			;renewed program call skipped
6	SUBM	300	;Call up Submit program
7	=	B 128	;Store Identifier in Byte 128
8	LBL 100		;Jump address
9	L	I 11	;Interrogate state of Input I11
10	JPF	110	;Dependent on Input I11, skip the deletion
			;of the Submit program
11	CAN	B 128	;Interrupt execution of the Submit program
			;or remove program from the queue
12	LBL 110		;Jump address
	•		;Continuation
	•		;Main program
	•		;
XX	EM		;End main program
XX	LBL	300	;Begin Submit program (is added as with Modules
XX	•		;at the end of the main program)
XX	•		;
XX	•		;
XX	EM		;End Submit program

In this case, the contents of the Submit program could, for example, be a display in the PLC window, which can be done via a fixed PLC Module.

3.17 Constants Field (KF)

The Constants Field data type can be used to access one of several constants defined in tabular form, depending on the value of the Index Register X.
Addressing is with KF Name[X], where 'Name' is a jump label that identifies the beginning of the Constants Field.
Constants Fields start with the label KFIELD Name followed by a random (not zero) number of constants followed by the end label ENDK. Constants Fields may only be created when the program has been previously terminated with an EM or JP instruction.
The name of the Constants Fields conforms to the rules for jump labels.

Example:

```
L       KF VALUESFIELD     ;ACCESS TO VALUESFIELD WITH
                           VALUESFIELD[X]
                           ;X=[0 TO 3]
        = W0               ONE OF THE CONSTANTS IS
                           STORED
EM
KFIELD  VALUESFIELD
        K+10               CONSTANT TO BE LOADED IF X=0
        K+1
        K$ABC
        K-100000           CONSTANT TO BE LOADED IF X=3
        ENDK
```

Access to Constants Fields is checked in the same way as write access to indexed operands. This is why X may only assume positive values from 0 to "Length of Constants Field -1".

3.18 Program Structures

A program is split up into program sequences so as to make it clearer. To do this the programmer uses jump labels (LBL) and conditional and unconditional jumps.
When structured instructions are used, the jump labels and jump commands are created by the Compiler. Remember that internal jump labels are generated to implement these structured commands, so the total number of available jump labels will be reduced accordingly. Structured instructions can be nested to up to 16 levels but there must be no "interleaving".

```
Right:  IFT                     Wrong:  IFT
        ...                             ...
        WHILEF                          WHILEF
        ...                             ...
        ENDW                            END
        ...                             ...
        ENDI                            ENDW
```

Instructions IFT, IFF, WHILET, WHILEF, ENDW, UNTILT and UNTILF require a valid gating result in the Logic Accumulator. They terminate the gating chain.
Instructions ELSE, ENDI and REPEAT require all gating chains to be terminated first.

3.18.1 IF ... ELSE ... ENDI Structure

The IF ... ELSE ... ENDI structure permits the alternative processing of two program branches depending on the value in the Logic Accumulator. The ELSE branch can be omitted. The following commands are available:

• IFT	(If Logic Accu True)	Following code only if Logic Accumulator=1
• IFF	(If Logic Accu False)	Following code only if Logic Accumulator=0
• ELSE	(else)	Following code only if IF not fulfilled
• ENDI	(End of IF-Structure)	End of IF Structure

Example:

```
    L     I0
    IFT            ;If Logic Accu=1
    ....           ;Program code for I0 = 1
    ELSE           ;                         can be omitted

    ....           ;Program code for I0 = 0  can be omitted
    ENDI           ;end of conditional processing
```

Internal jump labels are generated for the IF and ELSE instructions.

3.18.2 REPEAT ... UNTIL Structure

The REPEAT ... UNTIL structure repeats a program sequence until a condition is fulfilled.
Under no circumstances must this structure wait for an external event in the cyclical PLC program to happen!

The following commands are available:

• REPEAT	(Repeat)	Repeat program sequence from here
• UNTILT	(Until True)	Repeat sequence until Logic Accumulator=1
• UNTILF	(Until False)	Repeat sequence until Logic Accumulator=0

A REPEAT ... UNTIL loop is always run at least once!

Example:

```
    =     M100      ;end of previous chain
REPEAT              ;repeat following code
    .....           ;code to be executed
    LX              ;load Index Register
    >=    K100      ;check Index Register
UNTILT              ;repeat until X>=100
```

An internal jump label is generated for the REPEAT ... UNTIL structure.

3.18.3 WHILE ... ENDW Structure

The WHILE ... ENDW structure repeats a program sequence if a condition is fulfilled.
Under no circumstances must this structure wait for an external event in the cyclical PLC program to happen!

The following commands are available:

- WHILET (While True) Execute sequence if Logic Accumulator=1
- WHILEF (While False) Execute sequence if Logic Accumulator=0
- ENDW (End While) End of program sequence, go back to beginning

A WHILE ... ENDW loop is only run when the WHILE condition is fulfilled at the beginning. The execution condition must be repeated before the ENDW instruction. The condition can also be repeated differently than before the WHILE instruction!

Example:

```
        .....
        L       M100        ;create condition for 1st WHILE scan
WHILET                      ;execute following code if Logic Accumulator = 1
        .....               ;code to be executed
        L       M101        ;create condition for repeat processing
        A       M102        ;next condition
ENDW                        ;return to WHILE scan
```

Two internal jump labels are generated for the WHILE ... ENDW structure.

3.18.4 CASE Branch

Indexed Module Call (CASE)

Abbreviation for PLC Editor: CASE (CASE OF)

	Byte	Word
Execution time [µs]	3.3 to 3.8	3.3 to 3.8
Number of bytes	46	44

4 bytes must be added to the length for each entry in the jump table (CM).

Operands: B, W

Operation:
The CASE command is used to select a defined subprogram from a list of module calls (CM). These CM commands come directly after the CASE command and are numbered internally in ascending order from 0 to 127 maximum. The content of the operand (B, W) addresses the desired module.

End of Indexed Module Call (ENDC)

Abbreviation for PLC Editor: ENDC (ENDCASE)

	Byte	Word
Execution time [µs]	0	0
Number of bytes	0	0

Operands: none

Operation:
The ENDC command is used together with the CASE command. It must come directly after the list of CM commands.

Structure of a CASE instruction:

```
        Internal addressing (0 to 127 max.)
1       CASE B 150
2       CM 100    <-------    (0)
3       CM 200    <-------    (1)
4       CM 201    <-------    (2)
5       CM 202    <-------    (3)
6       CM 203    <-------    (4)
7       CM 204    <-------    (5)
8       CM 300    <-------    (6)
9       ENDC
```

Line 1: Command + Operand; the internal address of the required module must be filed in the operand
Line 2: Call Module if operand content 0
Line 3: Call Module if operand content 1
Line 4: Call Module if operand content 2
Line 5: Call Module if operand content 3
Line 6: Call Module if operand content 4
Line 7: Call Module if operand content 5
Line 8: Call Module if operand content 6
Line 9: End of CASE instruction

3.19 Linking Files

The source code of the PLC program can be stored in several different files that are managed with the USES, GLOBAL and EXTERN commands. These must be written at the beginning of the file, i.e. before any PLC instruction.

The USES command links another file to the program.
The GLOBAL command supplies a jump label from its own file as an entry that can be used by all other files.
The EXTERN command supplies a jump label defined in another file where it is declared as GLOBAL for use in its own file.

Splitting the source code up into a number of files helps improve clarity and overview by swapping out individual function groups.

The number of possible jump labels increases dramatically because the maximum of 1000 jump labels can be used for each individual file. This means a theoretical number of 60,000 jump labels for 60 files.

In all, up to 64 files can be linked to form a program. Each file can generate up to 64 K of code. The total potential length of the code is limited to 128 K (TNC 407: 64 K).

Where several files exist, the main program must have the status flag "M" in the directory. In the RAM this is done by selecting the PLC program function "COMPILE" once, in the EPROM it is done by specifying the /M option after the main program in the Linker file for binary output.

3.19.1 USES Instruction

The USES instruction links other files to the main program. Files linked with USES can also link other files with the USES instruction. It is also acceptable for one file to be linked by several other files with USES, and code for that file is only generated once.

The USES instruction requires a file name as its argument.

Linking files from the RAM is not allowed when executing from the EPROM (MP4010 = 0).
When executing from the RAM (MP4010 = 1) the file is searched first in the RAM and then in the EPROM.

The USES instruction only links the file, the program code for the file is not executed, i.e. USES cannot be compared to a CM instruction. The linked files must therefore contain individual modules which can then be called with CM instructions.

Example:

```
        USES    PLCMOD1              ;module depends on MP4010 from RAM or EPROM
        USES    EPRUPG               ;
        USES    RAMPLC               ;
```

Example of file linking:

```
PLCMAIN.PLC
;main program
USES    SPINDEL.PLC
USES    TCHANGE.PLC
;code

                        TCHANGE.PLC                  SPINDEL.PLC
                        ;tool change                 ;spindle control
                        USES    PLCUPG.PLC           USES    PLCUPG.PLC
                        ;code                        ;code

                                      PLCUPG.PLC
                                      ;general subprograms
                                      ;Code
```

3.19.2 GLOBAL Instruction

Up to 1000 local jump labels can be defined in each of the files linked with USES.
Modules defined in one file must be defined globally before they can be called from another file. This is done with the GLOBAL instruction at the beginning of the file.
Jump labels can only be defined globally when they are defined as LBL later on in the program (and not as KFIELD!).

Syntax:

```
GLOBAL          jump label          ;declaration of "jump label" beyond the file boundary
```

GLOBAL definitions must not be written in the main program. A single jump label cannot be declared globally by more than one module, however a name that has been declared globally by file A can be used again locally in file B.
In all, 1000 jump labels can be defined globally by all modules.

3.19.3 EXTERN Instruction

For a jump label to be able to access in one file modules which other files have declared as GLOBAL, it must be declared as EXTERN. The EXTERN instruction must be written at the beginning of the file.
The commands CM, CMT and CMF can then jump to this label in the program code.
The instructions JP, JPT, JPF, access to a Constants Field and linking with CM to a CASE Branch are not possible with external jump labels.
The name of the jump label cannot be assigned again in this file for a local jump label. Every external jump label reduces the number of available local jump labels.

Syntax:

EXTERN	Jump label	;the "Jump Label" Module can now be called from another file with the CM instruction.

4 PLC Modules

A number of PLC modules are available for PLC functions that cannot be executed or which are very complicated to execute with PLC commands. The error status is displayed after execution of the module in Marker 3171.

4.1 Marker, Byte, Word, Doubleword

4.1.1 Copy in Marker or Word Range (Module 9000/9001)

Modules 9000 (Marker) and 9001 (Byte/Word/Double) copy a block with a certain number of markers or bytes beginning from the start address to the specified target address.
For module 9001 the length should always be defined in bytes.

Constraints:
- Copying is sequential, starting with the first memory cell. This means that the function is not guaranteed when the source and destination blocks overlap and the source block begins at a lower address than the destination block. In this case the overlapping part of the source block is overwritten before copying takes place.

Possible errors:
- A block of the defined length cannot be read from the defined address in the marker or word RAM (address is too high or block is too long).
- A block of the defined length cannot be written to the defined address in the marker or word RAM (address is too high or block is too long).

Call:

```
PS      B/W/D/K    <Number 1st marker source block>
PS      B/W/D/K    <Number 1st marker destination block>
PS      B/W/D/K    <Length of block in markers>
CM      9000       Transfer in marker range
```

or

```
PS      B/W/D/K    <Number 1st byte source block>
PS      B/W/D/K    <Number 1st byte destination block>
PS      B/W/D/K    <Length of block in bytes>
CM      9001       Transfer in word range
```

Error status after call: M3171 = 0: Block was transferred
1: Error conditions see above

4.1.2 Read in Word Range (Module 9010/9011/9012)

A byte, word or doubleword is read from the defined position in the word memory and returned to the stack as an output variable. Indexed reading in the memory is possible by specifying a variable as the name of the memory cell.

Possible errors:
- The defined address is outside the valid range (0..1023).
- Module 9011: The defined address is not a word address (not divisible by 2).
- Module 9012: The defined address is not a doubleword address (not divisible by 4).

Call:

PS	B/W/D/K	<Number of byte to be read>	(Address)
CM	9010	read byte	
PL	B	<byte read>	(Value)

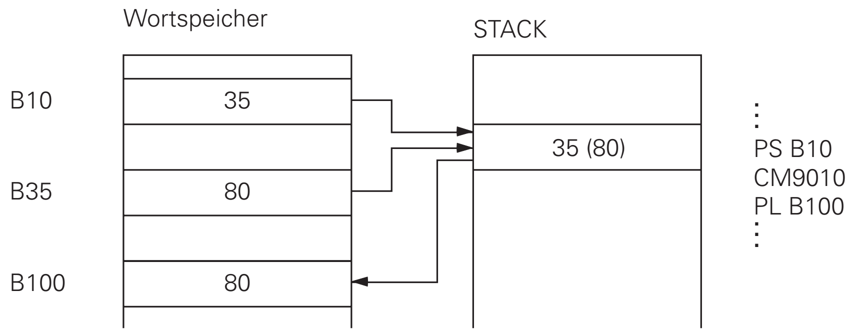
or

PS	B/W/D/K	<Number of word to be read>	(Address)
CM	9011	read word	
PL	W	<word read>	(Value)

or

PS	B/W/D/K	<Number of doubleword to be read>	(Address)
CM	9012	read doubleword	
PL	D	<doubleword read>	(Value)

Example of Module 9010



Error status after call: M3171 = 0: Byte/word/doubleword was read
1: Error condition see above

4.1.3 Write in Word Range (Module 9020/9021/9022)

The defined byte, word or doubleword is written to the defined position in the word memory. Indexed reading in the memory is possible by specifying a variable as the name of the memory cell.

Possible errors:
- The defined address is outside the valid range (0..1023).
- Module 9021: The defined address is not a word address (not divisible by 2).
- Module 9022: The defined address is not a doubleword address (not divisible by 4).

Call:

PS	B/W/D/K	<Number of byte to be written>	(Address)
PS	B/W/D/K	<byte to be written>	
CM	9020	write byte	(Value)

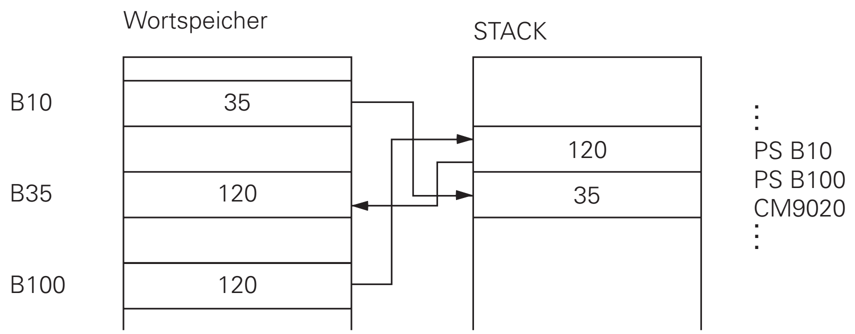
or

PS	B/W/D/K	<Number of word to be written>	(Address)
PS	B/W/D/K	<byte to be written>	
CM	9021	write word	(Value)

or

PS	B/W/D/K	<Number of doubleword to be written>	(Address)
PS	B/W/D/K	<byte to be written>	
CM	9022	write doubleword	(Value)

Example of Module 9020



Error status after call: M3171 = 0: Byte/word/doubleword was written
1: Error condition see above

4.2 Machine Parameters

4.2.1 Overwrite Machine Parameter (Module 9031)

Overwrites the value of a machine parameter that is defined by its number and index.

Constraints:
- The value of the machine parameter must be specified as an integer, with the decimal point shifted by the number of possible places after the decimal. Example: to set MP910.0 to 100.12 mm write 1001200 (four places after the decimal lead to a multiplication by 10000).
- Only the value in the run-time memory is modified, the value in the editable machine parameter list does not change. This means that the old value is valid again after editing and escaping from the machine parameter list.
- Zero must be given as the index for non-indexed machine parameters.
- Once the NC program has started the module only operates during the output of M/G/S/T/T2/Q strobes.
- Not every MP can be modified by the PLC. The machine parameters that can be modified by the PLC are marked "PLC" in the chapter "Machine Parameters".

Possible errors:
- The machine parameter specified by the MP number and index does not exist.
- The specified MP cannot be modified by the PLC or not once the NC program has started.
- The module was not called from a Submit Job.
- The module was called after the NC program started without a strobe marker being active.

Call:

PS	B/W/D/K	<MP Number>
PS	B/W/D/K	<MP Index>
PS	B/W/D	<MP Value>
CM	9031	
PL	B/W/D	<Error Code>
		0: No error
		1: MP does not exist/not modifiable/not modifiable once PC PGM has started
		2: MP value out of range
		3: Error when saving (Fatal Error)
		4: Call was not from SUBMIT Job
		5: Call once PGM started without strobe

Error status after call: M3171 = 0: MP was written
1: Error condition see above

4.2.2 Read Machine Parameter (Module 9032)

Reads the value of a machine parameter that is defined by its number and index from the editable machine parameter list.

Constraints:
- The value of the machine parameter is returned as an integer, with the decimal point being shifted by the number of possible places after the decimal. Example MP910.0 = 100.12 mm is read as 1001200 (four places after the decimal lead to a multiplication by 10000).
- Only the value from the editable machine parameter list is read, not any value in the run-time memory modified by PLC Module 9031.
- Zero must be given as the index for non-indexed machine parameters.

Possible errors:
- The machine parameter specified by the MP number and index does not exist.
- The module was not called from a Submit Job.

Call:

PS	B/W/D/K	<MP Number>
PS	B/W/D/K	<MP Index>
CM	9032	
PL	B/W/D	<MP value> / <Error Code>
		1: No such MP number
		2: No separator
		3: MP value out of range
		4: MP not in file
		5: No MP file found
		6: Call was not from SUBMIT Job

Error status after call: M3171 = 0: MP was read
1: Error condition see above

4.2.3 Select Machine Parameter File (Module 9033)

The machine parameter file with the specified name is selected (M status assigned) and a control reset is executed if another file was previously selected.

Constraints:
- The module can only be called from a SUBMIT Job.
- The module disregards any safety problems when initiating the control reset (e.g. free run-out of axes and spindle).
- The module can only be used on files in the RAM.
- The new MP file to be selected is checked, a faulty file is not selected.
- There is no return to the calling PLC program if file selection is successful.
- The file name is specified in a string that must contain the file name with extension. Additional characters (including blanks) are not allowed.
- If the PLC program is created externally ensure that lower-case letters are not used for the file name.
- Once the NC program has started the module only operates during the output of M/G/S/T/T2/Q strobes.

Possible errors:
- The module was not called from a SUBMIT Job.
- The specified string does not conform to the above conventions.
- There is no file with the specified name.
- The file to be selected is faulty.
- The module was called after the NC program started without a strobe marker being active.
- An incorrect string number was specified (out of range 0..3)

Call:

PS	B/W/D/K	<String Number>
CM	9033	Warning: Program execution ends here if a new file is selected
PL	B/W/D	<Error Code> 0: No error, file was already selected 1: String contains no valid file name 2: File not found 3: File is faulty 4: Wrong string number specified (0..3) 5: Call was not from a SUBMIT Job 6: Call once PGM started without strobe

Error status after call: M3171 = 0: File was already selected
1: Error condition see above

4.3 Status and Coordinates

4.3.1 Read Status Information (Module 9035)

Status information can be read with module 9035. A number is specified that identifies the desired information.

The following status information is available:

Specified number:		Values read:
0	Main mode Editor	0 - Edit 1 - Test run
1	Main mode Machine	0 - Approach reference points 1 - Manual mode 2 - Electronic handwheel 3 - Positioning with manual input 4 - Program run/single block 5 - Program run/full sequence

2	Overlaid Editor mode	0 - None (main mode active) 1 - Mode active 2 - Directory/Ext screen active 3 - MP Editor active 4 - PLC Editor active
3	Overlaid Machine mode	0 - None (main mode active) 1 - Mode active 2 - Directory/Ext screen active 3 - Tool Editor active
4	Displayed screen window	Bit-coded Bit #0..#7: Editing screen: #0 =1: Editing screen displayed #1 =1: Window mode active #2 =1: Block display/program select/ setup window active #3 =1: Position display active #4 =1: PLC status window active #5 =1: Status/Graphics window active #6/#7: spare Bit #8..#15: Machine screen #8 =1: Machine screen displayed #9 =1: Mode window active #10=1: Block display/program select/setup window active #11=1: Position display active #12=1: PLC status window active #13=1: Status/Graphics window active #14/#15: spare
5	Selected file in edit/test run	0 - No file 1 -.H (plain language NC PGM) 2 -.I (ISO NC PGM) 3 -.T (TOOL table) 4 -.D (Datum table) 5 -.P (Pallet table) 6 -.A (ASCII file) 7 -.S (Compensation table)
6	Selected file in single block / full sequence	0 - No file 1 - .H (plain language NC PGM) 2 - .I (ISO NC PGM)
7	Selected axis Editor (for actual value transfer)	0 - X axis 1 - Y axis 2 - Z axis 3 - 4th axis 4 - 5th axis
8	Selected axis Machine (for actual value transfer)	See above

9	Handwheel axis	-1 - None or several 0 - X axis 1 - Y axis 2 - Z axis 3 - 4th axis 4 - 5th axis
10	Handwheel axes bit coded	Bit #0 =1 - X axis controlled by handwheel Bit #1 =1 - Y axis controlled by handwheel Bit #2 =1 - Z axis controlled by handwheel Bit #3 =1 - 4th axis controlled by handwheel Bit #4 =1 - 5th axis controlled by handwheel
11	Handwheel subdivision factor	0..10 X axis
12	Handwheel subdivision factor	0..10 Y axis
13	Handwheel subdivision factor	0..10 Z axis
14	Handwheel subdivision factor	0..10 4th axis
15	Handwheel subdivision factor	0..10 5th axis
16	Input format of $MDI file	0 = .H file 1 = .I file
17	Display units	0 = MM 1 = INCH
18	Working plane	Bit0 = 1 Tilting is active Bit1 = 1 Tilting is selected for manual operation Bit2 = 1 Tilting selected for program run

Possible errors:
- The specified input parameter describes none of the status information available in the software version used.

Call:

PS	B/W/D/K	<Number of desired status information>
CM	9035	
PL	B/W/D	<Status Information>

Error status after call: M3171 = 0: Status information was read
1: Wrong number specified

4.3.2 Write Status Information (Module 9036)

NC status information that is not protected by a checksum can be modified. The information to be overwritten is identified by a specified number. The following status information can be modified:

Specified number	Function	Value
0	Handwheel interpolation X	0 to 10
1	Handwheel interpolation Y	0 to 10
2	Handwheel interpolation Z	0 to 10
3	Handwheel interpolation 4	0 to 10
4	Handwheel interpolation 5	0 to 10
5	Handwheel interpolation all axes	0 to 10
6	Select handwheel axis (MP7640 = 0 to 4)	0 = Axis X 1 = Axis Y 2 = Axis Z 3 = 4th axis 4 = 5th axis
7 to 9	Reserved	
10	Limit for jog increment	0 to 50 mm = limit for jog increment –1, <–2 or >50 = cancel the limit and activate the last jog increment entered –2 = cancel the limit and enter the minimum of the last jog increment and the last limit value

Constraints:
- Handwheel subdivision factors are limited to the lowest possible value according to the rapid traverse of the corresponding axis. There is no error message however.
- A handwheel subdivision can only be specified by MP7641=1.
- The value for the jog increment limit is transmitted in units of 1/10 000 mm
- After power switch-on the jog increment limit is always erased
- If the jog increment limit is entered in inches, the limit value [°] for rotary axes is calculated from the limit value in millimeters divided by 24.5.
- The input parameter "Number of status information" does not identify status information that can be overwritten in this software version.
- The specified value is outside the range valid for this status information.
- The entry of this status information is disabled, e.g. by machine parameters.

Call:

PS	B/W/D/K	<Number of status information>	
PS	B/W/D/K	<value to be written>	
CM	9036		
´PL	B/W/D	<Error identifier>	0: Status written 1: Wrong status identifier 2: Specified value out of range 3: Entry disabled (e.g. by MP)

Error status after call: M3171 = 0: Status was written
1: Error condition see above

4.3.3 Read Coordinates (Module 9040/9041/9042)

Module 9040: PLC reads axis coordinates (Format: 0.001 mm)

Module 9040 downloads axis coordinates for all NC axes from the control loop. The values are filed in 5 doublewords in the format 1/1000 mm starting with the specified target address.

Constraints:
- The values for all axes are always downloaded irrespective of whether individual axes are disabled by machine parameter MP10. The values for disabled axes are undefined.
- Before the reference point is traversed on an axis the coordinate value of that axis is undefined.

Possible errors:
- The argument for the coordinate type is outside the permitted range.
- The specified target address is not a doubleword address (i.e. not divisible by 4).
- 5 doublewords cannot be written at the specified target address (target address too big).

Call:

```
PS      K/B/W/D   <Target address Dxxx>
PS      K/B/W/D   <Coordinate type>
                  0 =  Actual value
                  1 =  Nominal value
                  2 =  Actual value in reference system
                  3 =  Trailing error
                  4 =  Distance to go
                  5 =  Deflection (measuring touch probe)
                  6 =  Actual values in the shifted datum system (datum shift)
CM      9040
```

Error status after call: M3171 = 0: Data was read
1: Faulty call data

Module 9041: PLC reads axis coordinates (Format: 0.0001 mm)

Module 9041 downloads axis coordinates for all NC axes from the control loop. The values are filed in 5 doublewords in the format 1/10000 mm starting with the specified target address.

Constraints:
- The values for all axes are always downloaded irrespective of whether individual axes are disabled by machine parameter MP10. The values for disabled axes are undefined.
- Before the reference point is traversed on an axis the coordinate value of that axis is undefined.

Possible errors:
- The argument for the coordinate type is outside the permitted range.
- The specified target address is not a doubleword address (i.e. not divisible by 4).
- 5 doublewords cannot be written at the specified target address (target address too big).

Call:

PS	K/B/W/D	<Target address Dxxx>
PS	K/B/W/D	0 = Actual value
		1 = Nominal value
		2 = Actual value in reference system
		3 = Trailing error
		4 = Distance to go
		5 = Deflection (measuring touch probe)
		6 = Actual values in the shifted datum system (datum shift)
CM	9041	

Error status after call: M3171 = 0: Data were read
1: Faulty call data

Module 9042: PLC reads spindle coordinates

The coordinates for actual value, nominal value, actual value in reference system, trailing error and distance-to-go of the spindle are filed in 5 consecutive doublewords starting from the specified target address. The data for actual, nominal and reference values are standardized to 0..+360.000 degrees, the data for trailing error and distance to go are displayed between -2879.912 degrees and +2879.912 degrees. The display is in 1/1000th degree format.

Constraints:
- All coordinates are read as 0 when MP3010 < 6 (not a controlled spindle).
- During operation as an analogue spindle (M3/M4 active or M5 and spindle control not active) the nominal value equals the actual value. Trailing error and distance to go are 0.

Possible errors:
- The specified target address is not a doubleword address (i.e. not divisible by 4).
- 5 doublewords cannot be written at the specified target address (target address too big).

Call:

PS	B/W/D/K	<Target address Dxxx>
CM	9042	

Error status after call: M3171 = 0: Spindle coordinates were read
1: Faulty call data

4.4 Number Conversion

4.4.1 Number Conversion Binary to ASCII (Module 9050)

Converts a binary numerical value consisting of mantissa and exponent to the base 10 to an ASCII-coded decimal number.

The number specified as mantissa and exponent is converted to a decimal number and stored at the specified address as a string. The exponent relates to the lowest-value place in the number. A negative number is detected when the mantissa corresponds to a negative number in the notation as a two's complement. A sign is only set in front of negative numbers. Trailing zeroes after the decimal point or leading zeroes before the decimal point are not converted, the string is written left-justified starting from the specified target address in the string buffer.

Constraints:
- The decimal sign is defined by machine parameter MP7280 as a decimal comma (MP7280 = 0) or a decimal point (MP7280 = 1).

Possible errors:
- The number of the target string is outside the permitted range (0..3).
- The conversion would result in more than 10 places after the decimal point.
- The conversion would result in more than 10 places before the decimal point.

Call:

```
PS      K/B/W/D   <Mantissa of numerical value to be converted>
PS      K/B/W/D   <Exponent to base 10 of the value>
PS      K/B/W/D   <Number of target string>
CM      9050
```

Error status after call: M3171 = 0: Number was converted
1: Error condition see above

Examples:

Mantissa	Exponent	Decimal number
123	0	"123"
-123	0	"-123"
123	2	"12300"
123	-3	"0.123"
123456	-3	"123.456"
123	-5	"0.00123"
100	-3	"0.1"
1234567890	-11	"0.0123456789"
123	-11	Error: more than 10 places after decimal point
123456789	1	"1234567890"
123	8	Error: more than 10 places before decimal point

4.4.2 Number Conversion Binary to ASCII (Module 9051)

Converts a binary numerical value to an ASCII-coded decimal number in the format specified.

The specified number is converted to a decimal number and stored as a string in the specified address.
The number is notated as a two's complement. When notated without a sign the absolute amount of the number is converted without a sign being put before the string. With the signed notation a sign ("+" or "-") is placed before the string in any event.
With the inch notation the numerical value is divided by 25.4 before being converted. If the number has more decimal places than the total of specified places before and after the decimal point, then the highest-value decimal places are omitted. With right-justified notation leading zeroes before the decimal point are replaced by blanks, with left-justified notation they are suppressed. Trailing zeroes after the decimal point are always converted.

Constraints:
- The decimal sign is defined by machine parameter MP7280 as a decimal comma (MP7280 = 0) or a decimal point (MP7280 = 1).

Possible errors:
- The number of the target string is outside the permitted range (0..3).
- There are more than 16 decimal places in all (before and after decimal point).
- No places before the decimal point are specified.

Call:

```
PS    K/B/W/D   <numerical value to be converted>
PS    K/B/W/D   <display mode (bit coded)>
                Bit #3: display with sign
                Bit #2: display converted to INCH
                Bit #1/#0: Format
                00: Sign and number left-justified
                1: Sign left-justified, number right-justified
                10: Sign and number right-justified
                11: Not permitted

PS    K/B/W/D   <Number of places after the decimal point>
PS    K/B/W/D   <Number of places before the decimal point>
PS    K/B/W/D   <Target address in string buffer>
CM    9051
```

Error status after call: M3171 = 0: Number was converted
1: Error condition see above

Examples:

Binary value	Mode	Before Dec.	After Dec.	Dec. Number
123	0	3	0	"123"
123	0	3	2	"1.23"
123	8	3	2	"+1.23"
123	9	3	2	"+ 1.23"
123	10	3	2	" +1.23"
254	0	3	1	"254.0"
254	4	3	1	"10.0"
1000	9	3	3	"+ 1.000"
-1000	9	3	3	"- 1.000"
123456	0	3	0	"456"

4.4.2 Number Conversion ASCII to Binary (Module 9052)

Converts an ASCII coded decimal number (possibly with places after the decimal point) into a mantissa and an exponent to the base 10.

The string identified by the source string number is read and converted to a signed number and an exponent to the base 10. If the sign is missing the number is detected as positive. Both the comma and the point are accepted as the decimal character. If the full extent of the mantissa cannot be notated in a doubleword then the last places are omitted and the exponent corrected accordingly. If possible the exponent is adjusted to correspond with the ASCII notation of the number.

Possible errors:
- The number of the source string is outside the valid range (0..3).
- The source string does not contain a string that can be interpreted as a number.
- The string overflows the end of the string buffer, 128 characters were read without an end of string being found.

Call:

```
PS      K/B/W/D    <Number of source string>
CM      9052
PL      B/W/D      <Numerical value>
PL      B/W/D      <Exponent 10Exx>
```

Error status after call: M3171 = 0: String was converted
1: Error condition see above

4.4.4 Conversion Binary to ASCII/Hexadecimal (Module 9053)

Converts a block of binary values from the word marker range into a string of ASCII coded hexadecimal numbers.

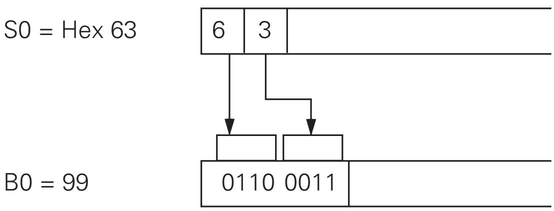
The specified number of bytes is read from the place specified by the source address and converted to a hexadecimal-coded ASCII string. Each byte in the source block makes 2 characters in the destination string. the destination string is identified by the destination string number.

Possible errors:
- The address for the source block is outside the range 0 to 1023.
- The number of the destination string is outside the valid range (0..3).

Call:

PS	K/B/W/D	<Source address in Word-RAM>
PS	K/B/W/D	<Number of destination string>
PS	K/B/W/D	<Number of data bytes>
CM	9053	

Example:



Error status after call: M3171 = 0: String was converted
1: Error condition see above

4.4.5 Conversion ASCII/Hexadecimal to Binary (Module 9054)

Converts a string of ASCII coded hexadecimal values to a block of binary values in the word marker range.

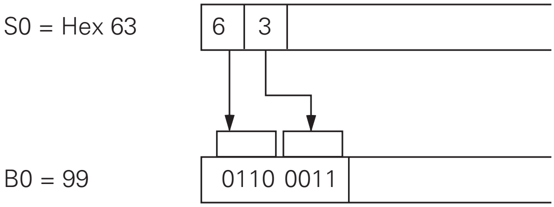
The string in the string buffer with the specified number is interpreted as a chain of ASCII coded hexadecimal numbers and converted into a block of corresponding binary bytes. Two ASCII characters make one binary byte. The binary block is stored in the word marker range starting from the specified destination address.

Possible errors:
- The number of the source string is outside the permitted range (0..3).
- The address for the destination block is outside the range 0 to 1023.
- The source string contains characters that cannot be interpreted as hexadecimal values (different characters 0..9, A..F).
- The source string contains an uneven number of characters (the last byte is not fully defined).
- The destination block has no room at the specified address.

Call:

PS	K/B/W/D	<Number of source string>
PS	K/B/W/D	<Destination address in Word-RAM>
CM	9054	

Example:



Error status after call: M3171 = 0: Binary block was converted
1: Error condition see above

4.5 String Processing

4.5.1 Copying a Number from a String (Module 9070)

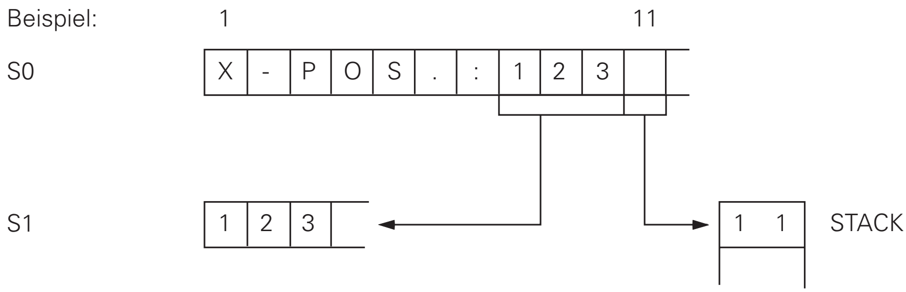
Searches for a numerical value in the source string in the string buffer with the specified source string number. The first numerical value found is copied as a string to the string identified by the destination string number.
Any conflict between source and destination strings is not checked, the source string is overwritten if necessary (but the module will still function in this case). The module recognizes signed and unsigned numbers with and without places after the decimal point. Both the comma and the point are accepted as the decimal character. The distance (in characters) of the first character after the number found in the source string is also recorded.

Possible errors:
- The numbers of source or destination string are outside the valid range (0..3).
- There is no number in the specified source string.
- The source string has been searched without an end of string being found.
- The found number string is longer than 79 characters, i.e. internal overflow.

Call:

PS	K/B/W/D	<Number of source string>
PS	K/B/W/D	<Number of destination string>
CM	9070	
PL	B/W/D	<Offset end of number string in source string>

Example:



Error status after call: M3171 = 0: Number was transferred
1: Error condition see above

4.5.2 Compute String Length (Module 9071)

Computes the length of the string with the specified number in the string buffer.

Possible errors:
- The number of the source string is outside the valid range (0..3).
- The source string has been searched without an end of string (<NUL>) being found.

Call:

PS	K/B/W/D	<Number of source string>
CM	9071	
PL	B/W/D	<Length of string>

Error status after call: M3171 = 0: String length was computed
1: Error conditions see above

4.6 PLC Window

4.6.1 Delete PLC Window (Module 9080)

Deletes the screen window for the PLC status display. The background color of the window is defined in machine parameter MP7320.2 or MP7356.0.

Constraints:
- This job cannot be aborted by a CAN command during processing of the module in a SUBMIT Job.

The module is also active when the currently selected screen shows no PLC status window (e.g. large graphic displays) or when the screen with PLC status window is in the background.

Possible errors:
- The module has not been called from a SUBMIT Job.

Call:

CM 9080

Error status after call: M3171 = 0: Screen window was deleted
1: Error condition see above

4.6.2 Interrogate PLC Window (Module 9081)

Interrogates the status of the screen window for the PLC status display.

The status is transferred bit-coded to the stack. Bit #0 is set when a window for PLC status display is on the selected screen. This is not the case with a full-page graphic display, when a program is selected or in the MOD operating mode. Bit #1 is set when the screen with the PLC status window is in the foreground. All other bits are cancelled.

Call:

```
CM    9081
PL    B/W/D    <Status of screen window>
```

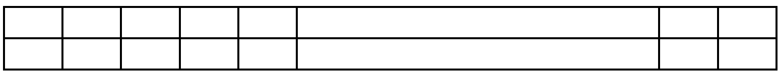
Marker M3171 is not affected.

4.6.3 Display String (Module 9082)

Displays a string in the screen window for the PLC status display at the specified position and in the specified color.

The string that is identified by the string number and which ends on the ASCII character <NUL> is displayed in the screen window for the PLC status display on line 0 (top line) or 1 (bottom line) and from column 0 (left margin) to 37 (right margin) in the specified color (1 to 15).

Line 0
Line 1



Column 0 37

References to PLC dialogues or PLC error messages are deleted. If the specified dialogue or error number is greater than the length of the corresponding file, then ASCII character `@' is displayed instead . If the text contains a non-displayable character except the string end, then ASCII character `^' is displayed instead.

Constraints:
- The character set that is used is determined by the size of the screen window, i.e. the current operating mode, and cannot be modified. The color refers to one of the machine parameters MP735x or MP732x and can be seen from the following table:

Color 1: MP7354.0 or MP7320.1
Color 2: MP7356.0 or MP7320.2
Color 3: MP7352.0 or MP732x.3 depending on the displayed graphics
Color 4: MP7353.0 or MP732x.0 depending on the displayed graphics
Color 5: MP7357.0 or MP732x.1 depending on the displayed graphics
Color 6: MP7352.1 or MP732x.2 depending on the displayed graphics
Color 7: MP7353.1 or MP732x.3 depending on the displayed graphics
Color 8: MP7350 or MP7320.8
Color 9: MP7357.1 or MP7320.9
Color 10: MP7354.1 or MP7320.10
Color 11: MP7356.2 or MP7320.11
Color 12: MP7356.1 or MP7320.12
Color 13: MP7354.2 or MP7320.13
Color 14: MP7352.2 or MP7320.14
Color 15: MP7351 or MP7320.15

Colors 1 to 5 (or 1, 2 and 4 with NC software 24305 and 24307) are background colors and contrast poorly. Color 2 in particular is the background color for the PLC window and so cannot be used as a foreground color in this window. Colors 11 and 12 are the foreground colors that go with background color 2 depending on how the machine parameters are grouped, and should therefore be given preference for displaying dialogues and status information. The NC control uses color 15 to display error messages and is therefore recommended for displaying alarms and error conditions.

If color 0 is specified, then the text appears in the same color as the last displayed character. Because the complete line is always displayed again in the window when a string is displayed (even when a column greater than 0 is specified), a text (e.g. "Spindle Output") with the color 0 is always displayed in the color of the numerical value to its left (e.g. color 11 when output is under 110% and color 15 when output is over 110%) even though PLC Module 9082 only displays the numerical value again. If the color 0 is specified for the first characters on a line however, then the color in which these characters are displayed is not defined and can change between two displays of the line.

When PLC Module 9082 is used in conjunction with software 24305 or 24307, the display of colors 3 to 7 varies with the selected graphics. These colors should therefore be avoided where possible to prevent the display changing when different graphics are selected. This problem does not arise with NC software releases 25996, 24302 and 25993.

If no screen window is currently shown for the PLC status display (window is not opened or in background) the module will run through normally and the string is not displayed until the corresponding screen window is displayed again and provided the string has not been overwritten by a repeat call of Module 9082 in the meantime. Module 9081 can be used to check whether the display is currently active. This job cannot be aborted by a CAN command during processing of the module in a SUBMIT Job.

Possible errors:
- The module was not called from a SUBMIT Job.
- A line less than 0 or greater than 1 was specified.
- A column less than 0 or greater than 37 was specified.
- The number of the string is outside the permitted range (0..3).
- No end of string was found.
- The last character(s) in the string cannot be displayed in the screen window.

The string is not displayed on screen in any of these error modes.

Call:

PS	K/B/W/D	Line number (0...1)
PS	K/B/W/D	Column number (0...37)
PS	K/B/W/D	Color number (0...15)
PS	K/B/W/D	String number (0...3)
CM	9082	

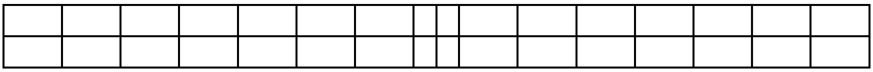
Error status after call: M3171 = 0: String displayed (when screen window for PLC status is displayed)
1: No display, error condition see above

4.6.4 Display Bar Chart (Module 9083)

Displays a bar chart in the screen window for the PLC status display on the specified line, with the specified lengths and in the specified colors.

A bar chart can be displayed in the left half of each line in the PLC status window. In this mode the ASCII text only appears in the right half of every line (19 characters max.).

Line 0
Line 1



Column 0 150 0 19

The operator must specify the line, maximum length (0...150), current length (<= maximum length) and the colors of the bars or the margin and scale graduation (0...15). If the maximum length exceeds 150 it is limited to 150. If current length exceeds maximum length then it is limited to the maximum length.

The chart comprises a rectangular grid with the maximum length and height of the ASCII characters. A scale graduation is shown at the top after every 10 units. The bar starts from the left hand edge of the grid. The unused part of the grid is filled in with the background color.

Constraints:
- The height of the bar chart varies according to the size of the screen window, i.e. the current operating mode, and cannot be modified.

The specified color refers to one of the machine parameters MP735x and can be seen from the following table:

Color 0: MP736x.0 depending on the displayed graphics
Color 1: MP7354.0
Color 2: MP7356.0
Color 3: MP7352.0
Color 4: MP7353.0
Color 5: MP7357.0
Color 6: MP7352.1
Color 7: MP7353.1
Color 8: MP7350
Color 9: MP7357.1
Color 10: MP7354.1
Color 11: MP7356.2
Color 12: MP7356.1
Color 13: MP7354.2
Color 14: MP7352.2
Color 15: MP7351

Color 2 is the background color for the screen window and can be used for margin and scale graduations if these are not to be displayed. If no screen window is currently shown for the PLC status display (window is not opened or in background) the module will cycle normally and the bar chart will not be displayed until the corresponding screen window reappears and provided the chart is not overwritten by a repeat call of Module 9083 in the meantime. Module 9081 can be used to check whether the display is currently active. This job cannot be aborted by a CAN command during processing of the module in a SUBMIT Job.

Possible errors:
- The module was not called from a SUBMIT Job.
- A line less than 0 or greater than 1 was specified.

The bar chart is not displayed on screen in any of these error modes.

Call:

PS	K/B/W/D	Line number (0...1)
PS	K/B/W/D	Color for bar (0...15)
PS	K/B/W/D	Color for frame and scale graduation (0...15)
PS	K/B/W/D	Current length of bar (0...150)
PS	K/B/W/D	Maximum length of bar (0...150)
CM	9083	

Error status after call: M3171 = 0: String displayed (when screen window for PLC status is displayed)
1: No display, error condition see above

4.7 Files

4.7.1 Executing a Pallet Program (Module 9090)

Selects an NC program and a datum table from a pallet file.

With module 9090 the PLC can select a block of NC files for execution. The file names must be defined in a pallet file (.P) that must be selected by the machine (status "M" in directory). Module 9090 can only run within a SUBMIT Job in the PLC Queue.

Constraints:
- If no datum table is specified in the pallet file then the previous file is retained.
- If the file type "Datum Table" is disabled in machine parameter MP7224 then any datum table specified in the pallet file will be ignored.
- If the file types "Plain Language NC Program" or "DIN/ISO NC Program" are disabled in machine parameter MP7224 then the system will not search for files of this type. If a file of this type is explicitly defined in the pallet file then the routine will report an error (NC program not found).
- This job cannot be aborted by a CAN command during processing of the module in a SUBMIT Job.

Possible errors:
- The module was not called from a SUBMIT Job.
- The module was called while an NC program was being executed.
- The control is not in the basic mode for executing NC programs (full sequence or single block mode).
- There is no pallet file in the NC memory that bears an "M" flag in the directory.
- A line number has been specified but is not in the selected Pallet file.
- The NC program in the pallet file has an incorrect type designation or the point between filename and file extension is missing (i.e. all extensions except ".I" or ".H" and more than 8 characters in the filename).
- The NC program specified in the pallet file is not in the NC memory.
- The NC program name specified in the pallet file is not clear. It has no extension and the NC memory contains both an ".H" program and an ".I" program with this name.
- The datum table (".D") specified in the pallet file is not in the NC memory.

Call:

PS	B/W/D/K	<Line number in pallet file>
CM	9090	
PL	B/W/D	<Error condition> 0: Files were selected 1: Call not from SUBMIT Job 2: Call after program started 3: Control not in correct mode 4: Pallet file not found 5: No such line in pallet file 6: Wrong type of NC program or point missing 7: NC program not found 8: NC program name not clear 9: Datum table not found

Error status after call: M3171 = 0: Files were selected
1: Error condition as Status word

4.7.2 Tool and Datum Table (Module 9092/9093/9094)

Module 9092: Searching for an entry in tables selected for machining (.T/.D/.TCH)

With Module 9092 values can be sought in specific columns in a table (with M status set) selected for machining. The function supplies the number of the line in which the value is found. This makes it possible, for example, to search for the vacant pocket (corresponding to T0) in the pocket table (.TCH).

Constraints:
- The module can only run within a SUBMIT Job.
- The value must be entered as an integer, shifted by the number of definable places after the decimal point.
- If the value is to be sought again, the starting line must be entered as the line in which the value was found plus one.

Possible errors:
- The module was not called from a SUBMIT Job.
- There is no file of the specified type that has M status.
- The specified line number is not in the file.
- The specified file type does not exist.
- The specified element does not exist.
- The entered numerical value was not found.

Call:

PS	B/W/D/K	<File type>
PS	B/W/D/K	<Element value>
PS	B/W/D/K	<Element number>
PS	B/W/D/K	<Line number for beginning the search>
CM	9092	
PL	B/W/D	<Line number> (–1, if M3171=1)
PL	B/W/D	<Error number>

Error status after call: M3171 = 0: Element was read
1: Error condition see above

Values for file type:
- 0: .T file (Tool table)
- 1: .D file (Datum table)
- 2: .TCH file (Pocket table)

Values for element number for .D file:
- 0: Offset X
- 1: Offset Y
- 2: Offset Z
- 3: Offset 4
- 4: Offset 5

Values for element number for .TCH file (pocket table):
- 0: Tool number (–1 if no tool entered)
- 1: Special pocket (0= no, 1= yes)
- 2: Fixed pocket (0= no, 1= yes)
- 3: Disabled pocket (0= no, 1= yes)
- 4: PLC status (PLC) (available for OEM)

Values for element number for .T file:
- 0: Tool length
- 1: Tool radius
- 2: spare
- 3: Replacement tool (-1 if not defined)
- 4: spare
- 5: Max. tool life
- 6: Max. tool life with Tool Call
- 7: Current tool life
- 8: 2nd tool radius
- 9: Allowance on tool length
- 10: Allowance on tool radius
- 11: Allowance on 2nd tool radius
- 12: Tool disabled (0= no, 1= yes)
- 13: Number of tool cutting edges (CUT)
- 14: Tolerance for tool length (LTOL)
- 15: Tolerance for tool radius (RTOL)
- 16: Cutting direction of the tool (DIRECT) (0 (0 = " " = "–")
- 17: PLC status (PLC) (available for OEM)
- 18: Tool length offset (TT:LOFFS)
- 19: Tool radius offset (TT:ROFFS)
- 20: Breakage tolerance for tool length (LBREAK)
- 21: Breakage tolerance for tool radius (RBREAK)

Error numbers:
- 0: No error, element was read
- 1: Call was not from SUBMIT Job
- 2: No such file type
- 3: No file of specified type with M status found
- 4: Line number not in file
- 5: Wrong element number
- 6: Element value not found

PLC Module 9093: Read data from tables selected for execution (.T/.D/.TCH)

With Module 9093 the contents of a line can be read in a table (with M status set) selected for execution. The module must be told the identifier of the desired table, the line number (= tool number for .T, vector number for .D or pocket number for .TCH) and the number of the element to be read.

Constraints:
- The module can only run within a SUBMIT Job.
- The values are returned as integers shifted by the number of definable places after the decimal point.

Possible errors:
- The module was not called from a SUBMIT Job.
- There is no file of the specified type that has M status.
- The specified line number is not in the file.
- The specified file type does not exist.
- The specified element does not exist.

Call:

PS	B/W/D/K	<File type>
PS	B/W/D/K	<Line number>
PS	B/W/D/K	<Element number>
CM	9093	
PL	B/W/D	<Element value>
PL	B/W/D	<Error number>

Error status after call: M3171 = 0: Element was read
1: Error condition see above

Entry value as with Module 9092.

Error numbers:
0: No error, element was read
1: Call was not from SUBMIT Job
2: No such file type
3: No file of specified type with M status found
4: Line number not in file
5: Wrong element number

Module 9094: Write data to a Tool and Datum Table

With Module 9094 the contents of a line can be overwritten in a table (with M status set) selected for execution. The module must be told the identifier of the desired table, the line number (= tool number for .T, vector number for .D or pocket number for .TCH), the number of the element to be written and the new value. Running the module 9094 results in a new initialization of the geometry.

Constraints:
- The module can only run within a SUBMIT Job.
- The values must be entered as integers shifted by the number of definable places after the decimal point.

Possible errors:
- The module was not called from a SUBMIT Job.
- There is no file of the specified type that has M status.
- The specified line number is not in the file.
- The specified file type does not exist.
- The specified element does not exist.
- The specified value is outside the permitted range.

Call:

PS	B/W/D/K	<File type>
PS	B/W/D/K	<Line number>
PS	B/W/D/K	<Element number>
PS	B/W/D/K	<Element value>
CM	9094	
PL	B/W/D	<Error status>

Error status after call: M3171 = 0: Element was written
1: Error condition see above

Entry values as for Module 9092

Error numbers:
0: No error, element was written
1: Call was not from SUBMIT Job
2: No such file type
3: No file of specified type with M status found
4: Line number not in file
5: Wrong element number
6: Element value is outside permitted range

4.7.3 Non-linear Axis Error Compensation (Module 9095)

Selects a line in the compensation table for non-linear axis error compensation.

The module selects the specified line in the active compensation assignment table (.CMA) as the active line and activates the non-linear axis error compensation according to the compensation tables entered on this line.

Constraints:
- The specified line stays selected as the active line after a control reset.
- Once an NC program starts the module only operates during the output of M/G/S/T/T2/Q strobes.
- The axis nominal values may alter slightly when the compensation table is switched over.

Possible errors:
- There is no .CMA file.
- The specified line does not exist in the selected .CMA file.
- One or more compensation tables (.COM) from the selected line do not exist.
- The module was not called from a Submit Job.
- The module was called after the start of an NC program without any strobe marker being active.

Call:

PS	B/W/D/K	<active line>
CM	9095	
PL	B/W/D	<Error code> 0: No error 1: Entered line does not exist 2: No such compensation table (.COM) 3: Compensation value table (.COM) has more than 256 entries 4: Permissible total number of compensation points exceeded 5: Too many compensation value tables (.COM) 6: No such .CMA file 7: Call was not from SUBMIT Job 5: Call after started PGM without Strobe

Error status after call: M3171 = 0: Compensation was activated
1: Error condition see above

4.8 Data Interface

The following modules enable the PLC to transfer data across the V.24/RS-232-C or V.11/RS-422 data interfaces.

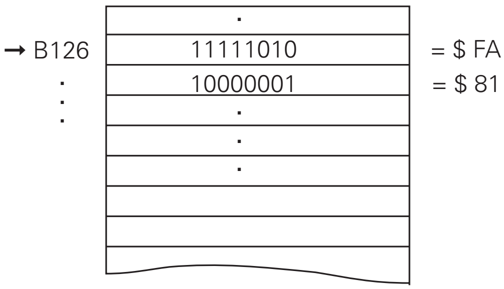
(For transfer parameters see chapter entitled "Data Interface")

The PLC assigns and enables the interfaces with Modules 9100 and 9101 respectively; the current status of the data interface can be interrogated with Module 9102.

The transmit and receive buffers for the PLC are 128 characters long. Since every STRING ends with an END character a STRING in the transmit or receive buffer can only be up to 127 characters long. As well as transmitting and receiving a STRING from the STRING memory (Modules 9103 and 9104), Modules 9105 and 9106 can be used to transfer a block of binary values (bytes) from the Word memory.

However ASCII characters are transmitted and received across the interface in both instances (STRING and binary transmission).

Example: Transferring a binary block



When transferring binary data from the Word memory from address B126 the ASCII characters "F", "A", "8", "1" etc. are transmitted consecutively across the interface.

Since each byte contains two ASCII characters when transferring binary data, the transmit and receive buffers are 63 bytes long.

With the help of Module 9107, when transferring binary data, each byte (two ASCII characters) can be read from the receive buffer without the buffer being erased.

4.8.1 Assign Data Interface (Module 9100)

Module 9100 assigns one of the serial interfaces to the PLC and configures the transfer parameters. The interface is also initialized and any errors are reset. The interface is switched to receive mode. Once assigned to the PLC, the interface is disabled for use by the Input/Output program of the user interface.

Constraints:
- The assignment of an interface to the PLC is cancelled when the PLC program is recompiled.
- Configuration for 19200 baud is not possible when the other interface is already configured for 38400 baud, and vice versa (irrespective of whether this interface is assigned to the PLC or the NC).
- Module 9100 only operates within the scope of a Submit Job.

Possible errors:
- The interface is already assigned elsewhere (by the Input/Output program of the NC user interface).
- The call parameters contain no valid values (0/1 for selecting the interface and defining the transfer parameters).
- The required baud rate is not possible owing to the configuration of the other interface (see Constraints).
- The Module was not called from a Submit Job.

Call:

PS	B/W/D/K	<Interface 0:RS232/1:RS422>
PS	B/W/D/K	<Transfer parameters 0: from MP5xxx.2 / 1: from mode>
CM	9100	

Error status after call: M3171 = 0: Interface was configured for PLC
1: Error condition see above

4.8.2 Enable Data Interface (Module 9101)

Module 9101 cancels the assignment of a serial interface to the PLC and reassigns it to the Input/Output program of the NC user interface. The receive mode of the interface is cancelled.

Constraints:
- Module 9101 only operates within the scope of a Submit Job.

Possible errors:
- The interface was not assigned to the PLC.
- The Module was not called from a Submit Job.

Call:

PS	B/W/D/K	<Interface 0:RS232/ 1:RS422>
CM	9101	

Error status after call: M3171 = 0: Interface assignment was cancelled
1: Error condition see above

4.8.3 Status of Data Interface (Module 9102)

Module 9102 reads all the relevant status information about one of the two serial interfaces in bit-coded form.

Constraints:
- The information "interface ready" is only updated when the interface is assigned to either the PLC or the NC. If the interface is not assigned, the Module reads the status which was valid before the last interface enable (whether by PLC or NC).
- Module 9102 can also be called from the cyclical PLC program section.

Possible errors:
- The call parameter for interface selection is outside the permitted range (0..1).

Call:

PS	B/W/D/K	<Interface 0:RS232/ 1:RS422>
CM	9102	
PL	B/W/D	<Interface status> Bit 0: Interface is assigned 1: Interface is assigned to PLC 2: Interface is ready (see above) 3: Transmit buffer is empty 4: Transmit error 5: Receive buffer is full 6: Receive error 7: ETX was received (not ready to receive)

Error status after call: M3171 = 0: Status was read
1: Wrong call parameter

4.8.4 Transmit String across Data Interface (Module 9103)

Module 9103 transmits a string from one of the 4 string memories across one of the two serial interfaces. References to the PLC error file and PLC dialogue file are deleted (see description of Module 9082).

Constraints:
- The interface must be assigned to the PLC and initialized by Module 9100 before Module 9103 is called.
- Module 9103 only operates within the scope of a Submit Job.

Possible errors:
- The call parameters are outside the permitted range (0..1 for the interface, 0..3 for the string number).
- The interface is not assigned to the PLC.
- The module was not called from a SUBMIT Job.
- The transmit buffer is not empty.
- Deleting the references to error and dialogue files has resulted in a string that is longer than 127 characters.
- A transfer error was detected when the transfer was initialized.

Call:

PS	B/W/D/K	<Interface 0:RS232/ 1:RS422>
PS	K/B/W/D	<Number of source string in string buffer (0..3)>
CM	9103	

Error status after call: M3171 = 0: String is transmitted
1: Error condition see above

4.8.5 Receive String across Data Interface (Module 9104)

With Module 9104 a string can be read from the receive buffer of an serial interface to one of the 4 string buffers and the receive buffer reset.

Constraints:
- The interface must be assigned to the PLC and initialized by Module 9100 before Module 9104 is called.
- Module 9104 only operates within the scope of a Submit Job.

Possible errors:
- The call parameters are outside the permitted range (0..1 for the interface, 0..3 for the string number).
- The interface is not assigned to the PLC.
- The module was not called from a SUBMIT Job.
- There is no complete string in the receive buffer.
- The string in the receive buffer is longer than 127 characters.

Call:

PS	B/W/D/K	<Interface 0:RS232/1:RS422>
PS	K/B/W/D	<Number of destination string in string buffer (0..3)>
CM	9104	

Error status after call: M3171 = 0: String was read
1: Error condition see above

4.8.6 Transmit Binary Data across Data Interface (Module 9105)

Module 9105 transmits a block of binary values from the Word RAM of the PLC to one of the two serial interfaces. The transfer is in the form of ASCII coded hexadecimal values, so every byte in the source block makes 2 ASCII characters at the serial interface.

Constraints:
- The interface must be assigned to the PLC and initialized by Module 9100 before Module 9105 is called.
- Module 9105 only operates within the scope of a Submit Job.

Possible errors::
- The call parameters are outside the permitted range (0..1 for the interface, 0..1023 for the start of the binary block, 0..63 for the length of the binary block).
- The interface is not assigned to the PLC.
- The module was not called from a Submit Job.
- The transmit buffer is not empty.
- Because of its length the binary block cannot be read from the specified address (start+length > 1024).
- A transmit error was detected when the transmission was initialized.

Call:

```
PS      B/W/D/K    <Interface 0:RS232/ 1:RS422>
PS      K/B/W/D    <Number of 1st byte in binary block (0..1023)>
PS      K/B/W/D    <Length of binary block (0..63)>
CM      9105
```

Error status after call: M3171 = 0: Hexadecimal string is transmitted
1: Error condition see above

4.8.7 Receive Binary Data across Data Interface (Module 9106)

Module 9106 reads a block of binary values from one of the two serial interfaces to the Word RAM of the PLC. The transfer is in the form of ASCII coded hexadecimal values, so every 2 ASCII characters from the serial interface make 1 byte in the binary block. The length of the read binary block is returned as the initial variable.

Constraints:
- The interface must be assigned to the PLC and initialized by Module 9100 before Module 9106 is called.
- Module 9106 only operates within the scope of a Submit Job.

Possible errors:
- The call parameters are outside the permitted range (0..1 for the interface, 0..1023 for the start of the binary block).
- The interface is not assigned to the PLC.
- The module was not called from a Submit Job.
- The receive buffer contains no data.
- The string in the receive buffer is longer than 128 characters.
- The string in the receive buffer contains an uneven number of characters.
- Because of its length the binary block cannot be written to the specified address (start+length > 1024).
- The string in the receive buffer contains characters that cannot be interpreted as ASCII coded hexadecimal values (=|= 0..9,A..F).

Call:

PS	B/W/D/K	<Interface 0:RS232/ 1:RS422>
PS	K/B/W/D	<Number of 1st byte in binary block (0..1023)>
CM	9106	
PL	B/W/D	<Length of binary block in bytes>

Error status after call: M3171 = 0: Binary block was read
1: Error condition see above

4.8.8 Read from Receive Buffer (Module 9107)

Module 9107 reads 2 ASCII characters from the receive buffer to one of the two serial interfaces and encodes them to a binary value. It is possible to specify an offset which corresponds to the position of the byte to be read in a binary block read by Module 9106 (i.e. half the offset in the ASCII string). The contents of the receive buffer are retained and can be read by Modules 9104 or 9106.

Constraints:
- The interface must be assigned to the PLC and initialized by Module 9100 before Module 9106 is called.
- Module 9106 only operates within the scope of a Submit Job.
- The system does not check whether characters of the received string are still at the place in the receive buffer indicated by the offset. If this is not the case, then an undefined value is read.

Possible errors:
- The call parameters are outside the permitted range (0..1 for the interface, 0..63 for the offset in the binary block).
- The interface is not assigned to the PLC
- The module was not called from a Submit Job.
- The receive buffer contains no data.
- The characters in the receive buffer cannot be interpreted as ASCII coded hexadecimal values (=|= 0..9,A..F).

Call:

PS	B/W/D/K	<Interface 0:RS232/1:RS422>
PS	B/W/D/K	<Offset of byte to be read in binary block>
CM	9107	
PL	B/W/D	<read binary value>

Error status after call: M3171 = 0: Binary value was read
1: Error condition see above

4.8.9 Transmit a Message by LSV/2 (Module 9110)

With Module 9110 a message (binary data or string) can be transmitted to a host computer connected by LSV/2 protocol.

Constraint:
- The message is transmitted to the host by the LSV/2 command "M PC<msg.l>".

Possible errors:
- The control has no LSV/2 connection to a host computer.
- The transmit buffer for the transfer is full.
- A wrong value was specified for the data type (doubleword or string).
- A wrong value was specified for the address. String: 0..3 Binary: 0..1020, divisible by 4.

Call:

PS	B/W/D/K	<Data type> 0: Binary data doubleword 1: String
PS	B/W/D/K	<Source address> With binary: number of doubleword (0..1020) With string: number of string (0..3)
CM	9110	
PL	B/W/D	<Error code> 0: Message transmitted 1: No connection to host 2: Transmit buffer full 3: Wrong data type (not 0 or 1) 4: Wrong source address

Error status after call: M3171 = 0: Message downloaded
1: Error condition see above

4.8.10 Read a Message by LSV/2 (Module 9111)

With Module 9111 a message (doubleword or string) that has been received from a host computer connected by LSV/2 protocol can be read.

Constraint:
- The message must be sent from the host by the LSV/2 command "M PC<msg.l>".

Possible errors:
- The control has no LSV/2 connection to a host computer.
- There is no message of the desired type in the receive buffer.
- A wrong value was specified for the data type (doubleword or string).
- A wrong value was specified for the address. String: 0..3 Binary: 0..1020, divisible by 4.

Call:

PS	B/W/D/K	<Data type> 0: Binary data doubleword 1: String
PS	B/W/D/K	<Target address> For binary: number of doubleword (0..1020) For string: number of string (0..3)
CM	9111	
PL	B/W/D	<Error code> 0: Message was read 1: No connection to host 2: No message of this type in receive buffer 3: Wrong data type (not 0 or 1) 4: Wrong target address

Error status after call: M3171 = 0: Message uploaded
1: Error condition see above

4.9 PLC Axes

4.9.1 Start PLC Axis (Module 9120)

Axis positioning is started by defining a target position (in the reference system), a feed-rate and a flag register. The axis is positioned totally independently of other control sequences, specifically there is no contour interpolation with other axes.

Constraints:
- The specified axis must be activated by MP10 and declared by MP60 as a PLC axis.
- The values for rapid traverse, analogue voltage for rapid traverse, acceleration etc. must be set correctly in the machine parameters.
- For axes with automatic reduction (modulo value in MP810.x) the axis always traverses to the target position in the direction of the shorter traverse path unless the target position has been specified as an incremental value.
- The system does not check for limit switch overshoot!
- The axis must be stationary. Any ongoing positioning must first be aborted with Module 9121.
- There is no feed-rate override.
- If the axis was previously in the "Search Reference Point" mode, this mode is cancelled. Positioning always starts from the current register contents.
- If Modules 9120 (Start PLC Axis), 9121 (Stop PLC Axis) and 9122 (Traverse reference mark) are called more than once for the same axis during a PLC cycle then only the last activated command is executed.
- Any "Positioning error" status that is set in this axis is cancelled.

Possible errors:
- A non-existent axis has been specified.
- An axis has been specified that is not declared as a PLC axis by MP10 and MP60.
- An absolute target position outside the range (0 .. modulo value) has been specified for a modulo axis.
- The axis is already positioning.

Call:

PS	B/W/D/K	<Axis> (0..4 for X/Y/Z/4/5)
PS	B/W/D/K	<Target position>
		(in reference system, format 0.0001 mm)
PS	B/W/D/K	<Feed-rate>(mm/min)
PS	B/W/D/K	<Flag register>
		Bit 0 = 1: Target position incremental
CM	9120	
PL	B/W/D	<Error code>
		0: Positioning has started
		1: Non-existent axis specified
		2: Axis not configured as PLC axis
		3: Axis is already positioning
		4: Absolute position outside modulo range

Error status after call: M3171 = 0: Positioning has started
1: Error condition see above

4.9.2 Stop PLC Axis (Module 9121)

A positioning sequence previously started with Module 9120 (Start PLC Axis) or Module 9123 (Traverse reference marks) can be aborted anywhere with Module 9121.

Constraints:
- The specified axis must be activated by MP10 and declared by MP60 as a PLC axis.
- If Modules 9120 (Start PLC Axis), 9121 (Stop PLC Axis) and 9122 (Traverse reference mark) are called more than once for the same axis during a PLC cycle then only the last activated command is executed.

Possible errors:
- A non-existent axis has been specified.
- An axis has been specified that is not declared as a PLC axis by MP10 and MP60.
- The specified axis is already stationary.

Call:

```
PS      B/W/D/K    <Axis> (0..4 for X/Y/Z/4/5)
CM      9121
PL      B/W/D      <Error code>
                   0: Position is aborted
                   1: Non-existent axis specified
                   2: Axis not configured as PLC axis
                   3: Axis was already stationary
```

Error status after call: M3171 = 0: Positioning was stopped
1: Error condition see above

4.9.3 Status PLC Axis (Module 9122)

A bit-coded status word is specified for a certain axis and provides information about the current operating mode of that axis.

Constraints:
- Status changes as a result of commands sent by the PLC to the controller of the PLC axes (Modules 9120, 9121, 9123) are not detected until the next PLC cycle.
- After power-up Bit 1 (axis over ref.) is cancelled. It is also possible to traverse the axis without first approaching the reference point.

Possible errors:
- A non-existent axis has been specified.

Call:

PS	B/W/D/K	<Axis> (0..4 for X/Y/Z/4/5)
CM	9122	
PL	B/W/D	<Status> Bit 0: 1= Axis is PLC axis Bit 1: 1= Axis already over reference point Bit 2: 1= Axis positioning Bit 3: 1= Direction of motion is negative Bit 4: 1= Positioning error

Error status after call: M3171 = 0: Status is reported
1: Error condition see above

4.9.4 Traverse Reference Marks for PLC Axis (Module 9123)

The module starts positioning in a defined direction and continues until a reference mark is found or positioning is aborted by Module 9121. Owing to the possible problems with locating a target position during the reference traverse (motion reversal etc.), stopping after locating the reference point will be unavoidable!

Constraints:
- The specified axis must be activated by MP10 and declared by MP60 as a PLC axis.
- The values for rapid traverse, analogue voltage for rapid traverse, acceleration etc. must be set correctly in the machine parameters.
- The system does not check for limit switch overshoot!
- The axis must be stationary. Any ongoing positioning must first be aborted with Module 9121.
- There is no feed-rate override.
- The "Search Reference Point" mode is set for the axis.
- An existing reference point on this axis is cancelled but not the numerical value of the axis. This is not re-initialized until the reference point is located.
- If Modules 9120 (Start PLC Axis), 9121 (Stop PLC Axis) and 9122 (Traverse reference mark) are called more than once for the same axis during a PLC cycle then only the last activated command is executed.
- Any "Positioning error" status that is set in this axis is cancelled.
- Positioning stops as soon as the reference point is reached. Depending on the braking distance of the axis, the latter will be slightly beyond the reference point in the direction of motion.

Possible errors:
- A non-existent axis has been specified.
- An axis has been specified that is not declared as a PLC axis by MP10 and MP60.
- The axis is already positioning.

Call:

PS	B/W/D/K	<Axis> (0..4 for X/Y/Z/4/5)
PS	B/W/D/K	<Feed rate> (mm/min)
PS	B/W/D/K	<Flag register>Bit 0 = 1:negative traverse direction
CM	9123	
PL	B/W/D	<Error code> 0: Positioning has started 1: Non-existent axis specified 2: Axis not configured as PLC axis 3: Axis is already positioning

Error status after call: M3171 = 0: Positioning has started
1: Non-existent axis specified

4.9.5 Override for PLC-axis (Module 9124)

The traversing speed of a PLC axis can be influenced by setting an override value.

Constraints:
- The axis must be activated through MP10 and be declared as a PLC axis in MP60.
- The override value can lie between 0% and 100.00% (resolution 0.01%) and must be transmitted as an integer (0 .. 10000).
- The last transmitted override value is accounted for at the beginning of movement.
- After a reset or interruption of the PLC program the override value of each PLC axis is set to 100.00%.
- The modules can also be called while a PLC axis is moving.
- The module can be called in addition to a module from the group (9120/9121/9123) in the same PLC scan. However, it becomes effective after this module.

Possible errors:
- A non-existent axis was transferred.
- An axis was transferred that has not been declared as a PLC axis in MP10 and MP60.
- An incorrect override value was transferred.

Call:

PS	B/W/D/K<Axis>	(0..4 for X/Y/Z/IV/V)
PS	B/W/D/K<Override>	(Format 0.01%)
CM	9124	
PL	B/W/D	<Error code> 0: Override was set 1: Invalid axis was transferred 2: Axis not defined as PLC axis 3: Incorrect override value

Error status after call: M3171 =0: Override was set
=1: See above error conditions

4.10 Analogue voltage output (Module 9130)

With Module 9130 it is possible for the PLC to have the NC control send a voltage to an analogue output.

Constraints:
- Only the analogue voltage for the spindle can be output. There is no access to the analogue outputs for the NC, nor is there any access to analogue outputs for PLC axes that are driven with Modules 9120 .. 9123.
- The voltage is output with a slight delay after the end of the PLC run. The module should be called only once per PLC run.
- The analogue voltage for the spindle can only be output when MP3010 < 3 and MP3011 = 2.
- The voltage must be transferred in 1mV format. Voltages are limited to a value between 10 V and –10 V.

Possible errors:
- The indicated analogue output does not exist.
- The indicated analogue output is not available for the PLC.

Call:

PS	B/W/D/K	<Number of the analogue output>
		0: Analogue output for spindle
PS	B/W/D/K	<Analogue voltage in millivolts>
CM	9130	

Error condition after call: M3171 = 0: Analogue voltage is output
1: See above error conditions

4.11 Inserting an NC block (Module 9150)

With Module 9150 is it possible to define an NC block during an active M/S/T/G/Q output from the NC to the PLC, or also when no program has been started. The defined NC block is executed after the acknowledgement of the M/S/T/G/Q strobe (immediately, if no PGM is started), before the control continues with the execution the NC program.

Constraints:
- If there are several commands during a strobe output, only the last one is executed.
- If there are several commands when the spindle is stationary, it is not defined which command is recognized.
- Erroneous parameters (e.g. incorrect tool number, non-existent file) result in a abortion of the NC program with the corresponding error message. If the error occurs while the spindle is stationary, there is no error message.

Possible errors:
- The call was made during a running NC program without active M/S/T/G/Q strobe.
- The call was made with an unknown command code.

Call:

PS	B/W/D/K	<Command code> 0: TOOL-CALL
PS	B/W/D/K	<Address of the Parameter> Number of the first Double of the parameter block, or number of the string
CM	9150	
PL	B/W/D	<Error code> 0: NC block was inserted 1: Running NC PGM and no strobe 2: Unknown error message

Parameter:

TOOL-CALL	B<Adr> active elements bit-coded	
	Bit 0 =1: Transfer tool number	otherwise modal
	Bit 1 =1: Transfer tool axis	otherwise modal
	Bit 2 =1: Transfer spindle speed	otherwise modal
	Bit 3 =1: Transfer length oversize	otherwise 0
	Bit 4 =1: Transfer radius oversize	otherwise 0
	B<Adr+1> Tool axis	(0..4 = X..V)
	W<Adr+2> Tool number	
	D<Adr+4> Spindle speed	(Format 0.001 1/min)
	D<Adr+8> Length oversize	(Format 0.0001 mm)
	D<Adr+12>Radius oversize	(Format 0.0001 mm)

Error status after call:	M3171	=0:	NC block was inserted
		=1:	See above error conditions

4.12 Spindle orientation (Module 9171)

Module 9171 makes it possible to define the velocity, angle and the direction of rotation for spindle orientation. Marker M2712 is set as long as the spindle is being oriented.

Constraints:
- If Marker M2712 and Module 9171 are set in the same scan, the tool is oriented with the parameters from the module call.
- If the module is called several times in the same scan, the tool is oriented with the parameters of the last call.
- If the module is called while the tool is still being oriented from a previous PLC scan, the call is ignored.
- The module works only in the cyclic PLC program.
- If the module is called while the spindle is turning, the transferred direction is ignored. The spindle is always oriented in the direction in which it is already turning.
- When the values +2 to +4 are transferred it is possible to start an orientation to the angle last defined in CYCL DEF 13. The transferred angle is added to the value from CYCL DEF 13. An additional spindle preset, therefore, can be transferred from the PLC.
- The nominal position during tool measurement with TT 110 is transferred exactly as with CYCL DEF 13
- If 0 is transferred as the spindle speed, the speed is taken from MP3520.1.
- The module can function only if M2719 = 1 (word processing).

Possible errors:
- The module was called from a SUBMIT Job.
- The parameter for the direction is erroneous.
- The spindle is already being oriented.
- A negative spindle speed was transferred.

Call:

PS	B/W/D/K	<Angle> [1/10 000 °] or additional preset with value from CYCL DEF 13	
PS	B/W/D/K	<RPM> [1/1 000 revolution per minute] 0 = value from MP3520.1	
PS	B/W/D/K	<Direction of rotation>	–1: Negative direction (M04) 0: Direction of the shortest path 1: Positive direction (M03) 2: Same as –1 but with angle from CYCL DEF 13 3: Same as 0 but with angle from CYCL DEF 13 4: Same as +1 but with angle from CYCL DEF 13
CM	9150		

Error status after call:	M3171 = 0	Spindle is being oriented (M2712 = 1)
	M3171 = 1	See above error conditions

5 Compatibility with TNC 355

The marker ranges for the TNC 355 have also been used for the TNC 407 and TNC 415 to enable PLC routines created for the TNC 355 to be run on the TNC 407 and TNC 415 as well.

However there are many functions that can be programmed much more simply with PLC words with the TNC 407 and TNC 415. PLC marker M2719 must be set to activate the strobes for word processing.

Marker	Function	Set	Reset
M2719	Deactivate the TNC 355 modules 0 = Activate strobes for marker range 1 = Activate strobes for word processing	PLC	PLC

	M2719 = 1		M2719 = 0	
Meaning	Marker	Data	Marker	Data
Start PLC-Pos. X	2704	D528	2452	M2560 to M2564
Start PLC-Pos. Y	2705	D532	2453	M2565 to M2569
Start PLC-Pos. Z	2706	D536	2454	M2570 to M2574
Start PLC-Pos. IV	2707	D540	2455	M2575 to M2579
Read out strobe for values in Q parameter (Q100 to Q107)	2713	D528	2809	M2560 to M2576
Read out strobe tool number	-- [1)]	–	2599	M2560 to M2567 M2112 to M2119
Read in strobe tool number	– [1)]	–	2598	M2560 to M2567 M2572 to M2579
Strobe datum correction	2716	D528 to D540	2819	MP4210.32 to MP4210.46
Spindle orientation	2712	D592	2527	M2585 to M2589

HEIDENHAIN recommend programming the PLC functions over the word range only. This makes the PLC program easy to read and a lot shorter.

1) Module 9093/9094

5.1 PLC Program Conversion

PLC programs created for the TNC 355 can also be used on the TNC 407 and TNC 415.

When these PLC programs are downloaded to the TNC 407 or TNC 415 the following command codes are automatically modified:

U → A
UN → AN
E → I
A → O
Z → C

An EM block is automatically appended at the end of the PLC program.
The programmer must ensure that the PLC program starts with a logic sequence.

5.2 Compatibility Markers

The table below lists the PLC markers that have only been retained for compatibility with the TNC 355. However the various PLC functions should be programmed with the TNC 407 and TNC 415 using the appropriate PLC words.

Marker	Function	Set	Reset	TNC 407/415
M2032	T-Code 1. Bit (lsb)	NC	NC	
M2033	T-Code 2. Bit	NC	NC	
M2034	T-Code 3. Bit	NC	NC	
M2035	T-Code 4. Bit	NC	NC	W262
M2036	T-Code 5. Bit	NC	NC	W264
M2037	T-Code 6. Bit	NC	NC	
M2038	T-Code 7. Bit	NC	NC	
M2039	T-Code 8. Bit (msb)	NC	NC	
M2051	Mode: Manual	NC	NC	
M2052	Mode: Electronic handwheel	NC	NC	
M2053	Mode: Position with manual input	NC	NC	
M2054	Mode: Program run/Single block	NC	NC	W272
M2055	Mode: Program run/Full sequence	NC	NC	
M2057	Mode: Traverse reference marks	NC	NC	
M2064	S-Code 1. Bit (lsb)			
M2065	S-Code 2. Bit			
M2066	S-Code 3. Bit			
M2067	S-Code 4. Bit			
M2068	S-Code 5. Bit			W258
M2069	S-Code 6. Bit			
M2070	S-Code 7. Bit			
M2071	S-Code 8. Bit (msb)			
M2072	M-Code 1. Bit (lsb)			
M2073	M-Code 2. Bit			
M2074	M-Code 3. Bit			
M2075	M-Code 4. Bit			
M2076	M-Code 5. Bit			W260
M2077	M-Code 6. Bit			
M2078	M-Code 7. Bit			
M2079	M-Code 8. Bit (msb)			
M2080	minimum speed from MP3020 1. Bit (lsb)			
M2081	minimum speed from MP3020 2. Bit			
M2082	minimum speed from MP3020 3. Bit			W1008
M2083	minimum speed from MP3020 4. Bit			
M2084	minimum speed from MP3020 5. Bit			
M2085	minimum speed from MP3020 6. Bit			
M2086	minimum speed from MP3020 7. Bit			
M2087	minimum speed from MP3020 8. Bit (msb)			
M2088	Increment from MP3020 1. Bit (lsb)			
M2089	Increment from MP3020 2. Bit			—
M2090	Increment from MP3020 3. Bit			
M2091	Increment from MP3020 4. Bit (msb)			

Marker	Function	Set	Reset	TNC 407/415
M2104	G-Code S-analogue 1st Bit (lsb)			W256
M2105	G-Code S-analogue 2nd Bit			
M2106	G-Code S-Analogue 3rd Bit (msb)			
M2112	T number (P number) decade 1 (lsb)			W262 W264
M2113	T number (P number) decade 1			
M2114	T number (P number) decade 1			
M2115	T number (P number) decade 1 (msb)			
M2116	T number (P number) decade 2 (lsb)			
M2117	T number (P number) decade 2			
M2118	T number (P number) decade 2			
M2119	T number (P number) decade 2 (msb)			
M2176	Mode code (lsb)			W272
M2177	Mode code			
M2178	Mode code			
M2179	Mode code (msb)			
M2186	Code number 84159 entered	NC	PLC	D276
M2192 to M2239	Markers controllable by MP4310.0, MP4310.1 and MP4310.2	NC	NC	W976 to W980
M2452	Activate PLC positioning X axis	PLC	NC	M2704
M2453	Activate PLC positioning Y axis	PLC	NC	M2705
M2454	Activate PLC positioning Z axis	PLC	NC	M2706
M2455	Activate PLC positioning axis 4	PLC	NC	M2707
M2468	Complement PLC positioning X axis	NC	PLC	—
M2469	Complement PLC positioning Y axis	NC	PLC	
M2470	Complement PLC positioning Z axis	NC	PLC	
M2471	Complement PLC positioning axis 4	NC	PLC	
M2522	Activate PLC positioning axis 5	PLC	NC	M2708
M2527	Activate spindle orientation	PLC	NC	M2712
M2538	Complement PLC positioning axis 5	NC	PLC	—
M2543	Complement spindle orientation	NC	PLC	
M2560 to M2589	BCD numerical values for PLC positioning, tool number, spindle orientation and Q parameters	PLC	PLC	D528 to D544
M2597	Tool number: output mode, 0 = binary, 1 = BCD	PLC	PLC	—
M2800 to M2807	Key-Code	PLC	PLC	W516
M2808	Strobe for key code	PLC	NC	M2813
M2809	Activate numerical value transfer PLC to NC	PLC	PLC	M2713
M2817	Q-number (lsb)	PLC	PLC	W516
M2818	Q-number (msb)	PLC	PLC	W516
M2819	Activate datum correction	PLC	NC	M2716
M2832 to M2839	Key code of the operated disabled key	NC	NC	W274

Marker	Function	Set	Reset	TNC 407/415
M3200 to M3263	Values from MP4310.3 to MP4310.6	NC	NC	W 972 to W 988

5.3 Incompatibility

It has not been possible to maintain compatibility in all areas of the PLC programs. This is because the TNC 355 differs considerably from the TNC 407 and TNC 415 both in memory organization and ergonomics.

5.3.1 PLC Macros

The following macro programs are available in the TNC 355 for controlling the tool changer.

M3264	Convert tool number or pocket number to binary code
M3265	Increment actual value of pocket number
M3266	Decrement actual value of pocket number
M3267	Compare actual and nominal values of pocket number

These macro programs are no longer available in the TNC 407 and TNC 415. The following PLC subroutines can be used in place of the macro programs.

Convert tool number or pocket number to binary code

```
M2032 to M2039 -> M3024 to M3031
LBL 200          ;M3264 (TNC 355)
L       W262     ;T code in binary
B=      M3024    ;file in bit area
==      K+0
=       M3043    ;T code == 0
LB      M3200    ;low byte from MP4310.3
=       B255     ;load MP4310.3
LB      M3024
>       B255
=       M3044    ;Tcode > MP4310.3 1 byte
EM
```

Increment actual value of pocket number

```
(M3032 to M3039) = (M3032 to M3039) + 1
LBL 210          ;B265 (TNC 355)
LB      M3200    ;low byte from MP4310.3
=       B255     ;number of mag. pockets
LB      M3032    ;actual value
+       K+1      ;increment by 1
B=      M3032    ;
<=      B255     ;less than or equal to actual value
```

```
JPT     211
L       K+1     ;load 1 as actual value
B=      M3032   ;
LBL 211
EM
```

Decrement actual value of pocket number

```
(M3032 to M3039) = (M3032 to M3039) - 1
LBL 220         ;M3266 (TNC 355)
LB      M3200   ;low byte from MP4310.3
=       B255    ;number of mag. pockets
LB      M3032   ;actual value
-       K+1     ;decrement by 1
B=      M3032   ;
>        K+0
JPT     221
L       B255    ;load number of mag. pockets
B=      M3032   ;as actual value
LBL 221
EM
```

Compare actual and nominal values of pocket number

```
LBL230          ;M3267 (TNC 355)
LB      M3032   ;read actual value from markers
=       B250    ;actual value
LB      M3024   ;read nominal value from markers
=       B252    ;nominal value
LW      M3200   ;read reduction and tool max
=       W254    ;high byte reduc. byte 254, low byte max. T byte 255
L       B250
==      B252    ;test actual/nominal for parity
=       M3040   ;actual== nominal
JPT     231     ;if actual==nominal then end
L       B250    ;actual - nominal=>B248
-       B252
=       B248
>       K+0
JPT     232
L-      B248
=       B248    ;B248:=Abs(actual - nominal)
LBL232
;       Compute reduction
L       B255
-       B254
<=      B248
O[
L       B248
<=      B254
]
=       M3041   ;Reduction reached
;       Compute shortest path
```

```
L       B255    ;Max. tool's
>>      K+1     ;DIV 2
=       B255    ;Max. tool's DIV 2

L       B252    ;nominal value
>       B250    ;actual value
A[
L       B248    ;Abs (actual - nominal)
<=      B255    ;Max. tool's Div 2
]
O[
L       B252    ;nominal value
<       B250    ;actual value
A[
L       B248    ;Abs (actual - nominal)
>=      B255    ;Max. tool's Div 2
]
]
=       M3042   ;direction marker
LBL231
EM
```

5.3.2 PLC Error Messages

The TNC 407 and TNC 415 contain dedicated files for error messages in the first dialog language and for English error messages. File .EE1 (.ER1) or file .EEE (.ERE) is accessed depending on machine parameters MP7230 (dialog language) and MP4010 (PLC program from RAM or EPROM). (See "PLC Programming" Register, Chapter 1.3).

This means that, to select the appropriate error message, it is no longer necessary to interrogate the dialog language with PLC marker M2041 as was the case with the TNC 355.
PLC marker M2041 has no function in the TNC 407 or TNC 415.

5.3.3 Mode Code

The code for the "Programming & Editing" and "Test Run" modes is no longer displayed because a foreground and a background mode can be active in the TNC 407 and TNC 415 at the same time. (M2176 to M2179 and W272)

5.3.4 Non-Implemented Markers

Marker	Function
M2020	Spindle speed under-ranged
M2021	Nominal/actual speed difference over-ranged
M2024	Touch probe ready
M2049	Mode : Background Programming
M2050	Mode: Programming & Editing
M2056	Mode: Test program
M2060	DIN/ISO Programming
M2062	"Code number" dialog
M2063	Central tool file
M2188	Checksum error for non-volatile Q parameters
M2288	X axis was moving when program aborted
M2289	Y axis was moving when program aborted
M2290	Z axis was moving when program aborted
M2291	Axis 4 was moving when program aborted
M2292	Axis 5 was moving when program aborted
M2504	Axis clamping after constant contour transition
M2509	Activate a % factor for feed rate over-ride
M2510	Deactivate spindle over-ride
M2511	Deactivate feed rate over-ride
M2526	Switch X/Y or Z to 4th axis
M2542	Complement for M2526
M2585	PLC positioning axis S (lsb)
M2586	PLC positioning axis S
M2587	PLC positioning axis S
M2588	PLC positioning axis S
M2589	PLC positioning axis S (msb)
M2590	Define the axis to be switched to 4th axis (lsb)
M2591	Define the axis to be switched to 4th axis (msb)
M2592	Tool number from M2560 to M2567
M2593	Go to Hirth grid after NC STOP
M2594	Central tool file cannot be edited
M2595	Output tool number in addition to pocket number
M2596	Select central tool file during active program run mode
M2598	Transfer tool number
M2599	Transfer tool number
M2602	Define reference point traverse
M2603	Axis sequence for reference point traverse (lsb)
M2604	Axis sequence for reference point traverse
M2605	Axis sequence for reference point traverse
M2606	Axis sequence for reference point traverse
M2607	Axis sequence for reference point traverse (msb)
M2657	Display a second auxiliary function
M2664	No standstill monitoring X axis
M2665	No standstill monitoring Y axis
M2666	No standstill monitoring Z axis

Marker	Function
M2667	No standstill monitoring axis 4
M2668	No standstill monitoring axis 5
M2810	Data format of a numerical value in M2560 to M2576
M2811	Data format of a numerical value in M2560 to M2576
M2812	Data format of a numerical value in M2560 to M2576
M2820	Update the central tool file
M2821	Strobe for updating the central tool file
M2822	Strobe for % factor for spindle voltage
M2823	Select ramp pairs for S analogue

5.3.5 PLC Cycle Time

The contouring controls of the TNC 355, TNC 407 and TNC 415 have different PLC cycle times. This must be remembered when using the timers and counters.

Contouring Control	PLC Cycle Time
TNC 355 without extended memory	20 ms
TNC 355 with extended memory	22 ms
TNC 407	24 ms
TNC 415	20 ms

Data Interface – Contents

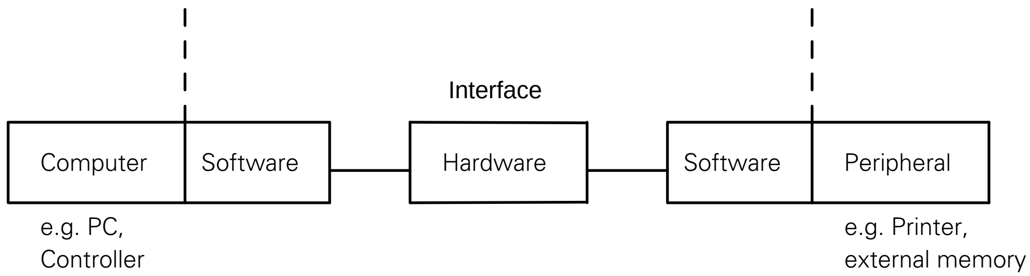
1 Introduction

When operating a computer system (PC, Controller), a wide variety of peripherals, such as printers, external memories (floppy-disk drives; hard disks) or other computer systems, can be installed in addition the Central Processing Unit (CPU).

Communication between the CPU and the peripherals is made possible by using a data interface.

Communication requires facilities for transferring data to the peripherals and of course, physical connection via a transmission line. Peripheral device control and communication, via the interface, is generally the responsibility of the computer system. The computer system therefore has to meet certain requirements.

The interfaces, which primarily consist of the physical links between the computer system and the peripherals, need appropriate software in order to control the transfer of information between the individual units. The relationship between hardware and software, which fully defines an interface, is illustrated by the following diagram:



The "hardware" in the diagram covers all the physical components, such as circuit construction, pin layout, electrical characteristics, etc. The "software" includes, for example, the drivers for the output modules, which are part of the operating software both of the computer system and the peripherals.

Due to the wide variety of computers, controllers and peripherals, standard interfaces have been introduced, which, in an ideal situation, enable extremely varied devices to be connected to each other.

Such standards include, for example, the RS-232-C/V.24 and RS-422/V.11 interfaces, which are described in detail later.

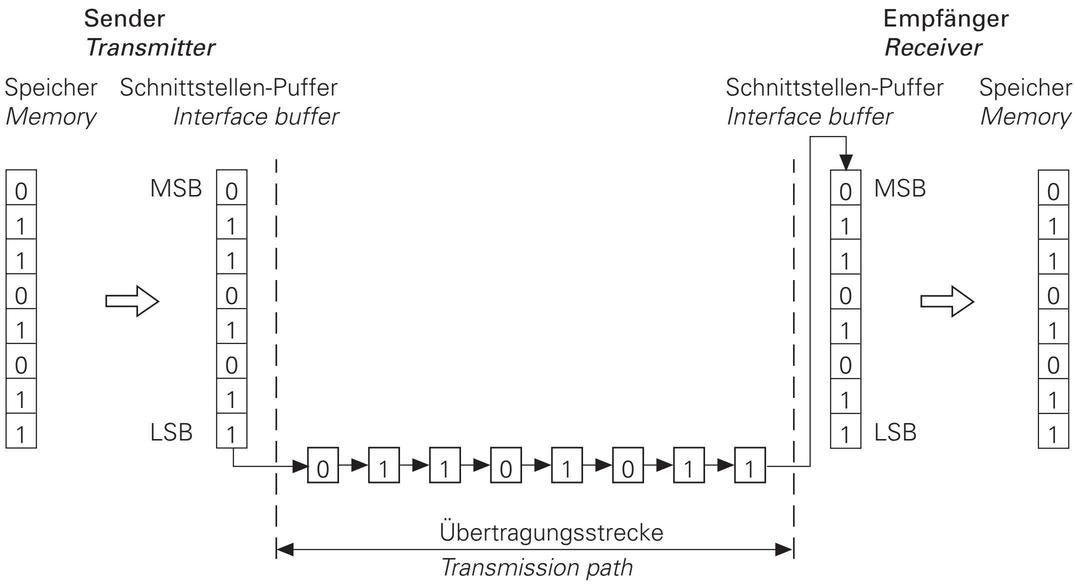
1.1 Principles of data transfer

Since all information is conveyed as data, one first needs to become familiar with a few of the principles of data transfer. The term "Data" is used to describe all of the information which the computer is capable of collecting and processing.

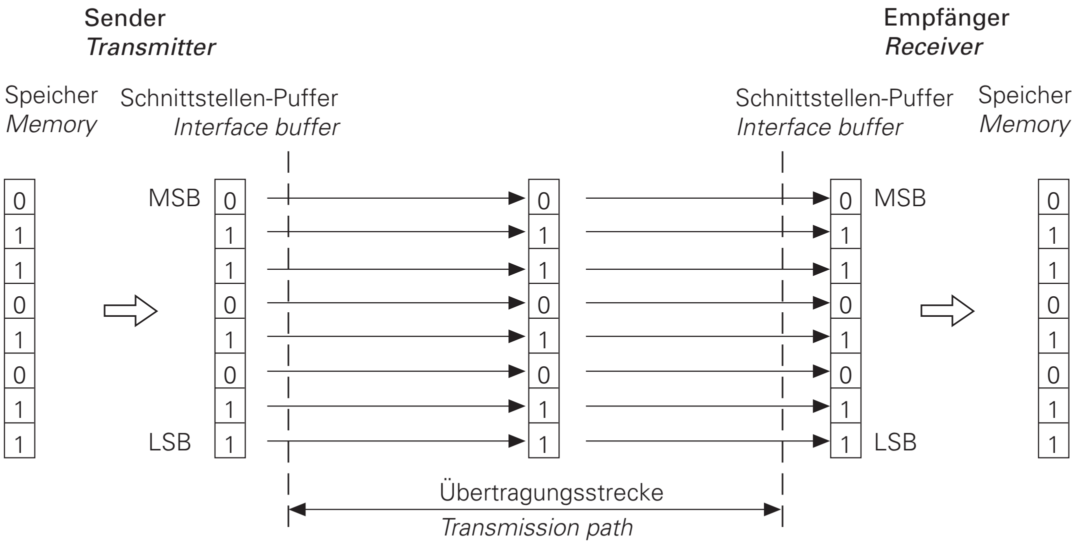
1.1.1 Serial/parallel

Data can be transmitted in either serial or parallel format.
Basically, data is coded in the computer system, e.g. as bytes (8 bits) and supplied to the interface in parallel.

In the case of serial data transmission, the parallel information from the computer system has to be converted into a serial data-flow by using an USART (Universal Synchronous/Asynchronous Receiver/Transmitter).
The receiver accepts the serial data-flow and converts it back again into parallel information.



A parallel interface, on the other hand, does not need a USART: just a line driver. Typically, the connection between the computer system and a peripheral consists of a 36-way ribbon cable. Its maximum length is generally about 3 metres.



One obvious advantage of serial data transmission becomes apparent when long distances have to be covered. With parallel transmission, the cost of the cable increases with every additional bit which has to be transmitted. In addition, the effect of interference on adjacent wires from sharp signal edges and electrical coupling is far greater over long lines than it is with serial transmission which is relatively slower and uses fewer wires.

The comparatively slow speed of serial data transmission is, at the same time, its greatest drawback. Since the individual bits are sent along the line one after the other and each transfer takes a specified time, it takes far longer to send a binary word to the receiver than it would if conveyed by parallel transmission. As it happens, most peripheral devices work fairly slowly and cannot in fact cope with data transmitted at high speed. Serial data transmission is generally adequate for devices such as external memories or mechanical printers, especially as such devices have a large internal buffer for incoming characters.

1.1.2 Asynchronous data format

In order for communication to be established between two devices involved in data interchange, they have to use a common language.
In the field of computer engineering, this language consists of digital coding of letters, figures and control characters.

One of the most common codes is the ASCII code (American Standard Code for Information Interchange) which codes all characters with seven bits.
In all, it is possible to code 2^7 = 128 characters. According to the ASCII code, the control character "Line Feed" or <LF> is coded with the following combination of bits:

0 0 0 1 0 1 0 = 10 dec = 0A hex
MSB LSB

The letter 'z' is represented by the following combination of bits:

1 1 1 1 0 1 0 = 122 dec = 7A hex

MSB LSB

i.e. when the letter "z" is transmitted via a serial interface, the appropriate bits are sent one after the other. The ASCII code is shown in full in the Appendix.

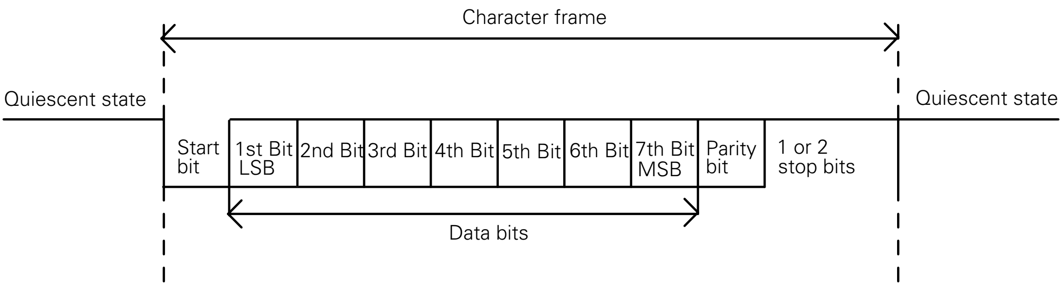
Proper data transmission requires the device concerned to interpret incoming data correctly and, in particular, to determine the start of a transmission. For this purpose, there is a synchronization process which ensures that the receiver detects the first bit of a character correctly. With an asynchronous data format, a start bit is sent before each data word and the word is then ended by one or two stop bits. One feature of this data format is that, starting from a quiescent state, transmission of a data word can begin at any time.

A quiescent state exists before switch-on and is reverted to after each transmission. Before a data bit can be transmitted this has to be communicated to the receiver. Otherwise, if the first bit of the data word has the same value as the quiescent state, the receiver will not notice any difference from the quiescent state.

A so-called "start bit" is used for this purpose:

For the duration of a single bit, the transmitter emits a logic value which clearly differs from the quiescent state and which gives the receiver an opportunity to prepare its polling logic to read in the data bit. After the start bit has been sent, the data word is transmitted, bit by bit, starting with the LSB (Least Significant Bit). After the MSB (Most Significant Bit) of the data word, a so-called "parity bit" is inserted (see paragraph 1.1.3 "Checking data").

The parity bit is followed by one or two stop bits. These final stop bits ensure that the receiver has enough time to recognise the transmitter again before the start of the next character. Synchronization is repeated before each character. The complete word is referred to as a character frame.



1.1.3 Checking data

With an asynchronous character frame, transmission errors can be detected by using a parity-check procedure. A parity bit is sent in addition to the data bits. The evaluation of this bit enables the receiver to check the parity of received data.

The parity bit can take three different forms; the same form of parity must be set at both interfaces.

– No parity check
Error detection is dispensed with.

– Even parity
The transmitter counts bits with a value of 1. If the number is odd, the parity bit is set to 1, otherwise it is cleared to 0. The sum of the set data bits and the parity bit is therefore always even. Upon receiving a word, the receiver counts all of the set bits, including the parity bit. If this count yields an odd number, there is a transmission error and the data word must be repeated, or an error message will be displayed.

– Odd parity
In this case, the parity bit is so chosen by the transmitter that the total number of all the set bits is odd. In this case, an error will be detected if the receiver observes an even number of set bits in its evaluation.

Example:

Parity bit

1 Even parity

Letter: "z" 1111010

0 Odd parity

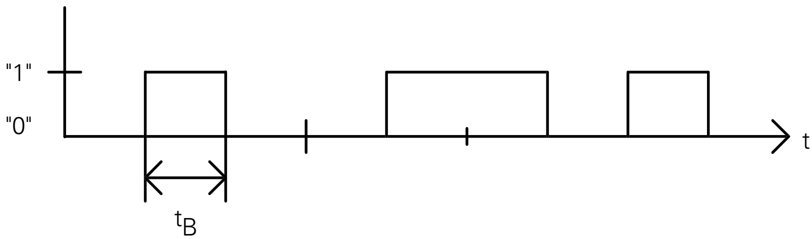
1.1.4 Data transfer rate

The data transfer rate of an interface is given in 'BAUD' and indicates the number of bits of data transmitted in one second.

$$1 \text{ baud} = \left[1\ \frac{\text{Bit}}{\text{s}}\right]$$

Common baud rates are:
110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200 and 38400 baud.

The time taken to transmit one bit (t_B) can be calculated from the baud rate:



$$t_B = \frac{1}{\text{Baud rate}[\frac{\text{bits}}{\text{s}}]}$$

For example, a baud rate of 19,200 baud will have a bit duration of t_B = 52.083 s.

The number of characters transmitted can be calculated from the baud rate and the transmission format:

$$\text{Characters per second} = \frac{\text{Baud rate}[\frac{\text{bits}}{\text{s}}]}{\text{Number of bits per character}}$$

Example:
With a transmission format of seven data bits, one start bit, two stop bits and a data transfer rate of exactly 300 baud:-

$$\frac{300 \text{ Baud}}{10 \text{ bits}} = 30 \text{ characters per second}$$

will be transmitted.

1.2 Handshaking

A "handshake" procedure is often used in connection with interfaces. This means that two devices are, as it were, working "hand in hand" in order to control data transfer. A distinction is drawn between "software handshaking" and "hardware handshaking".

Either hardware or software handshaking can be chosen for communication between two units.

1.2.1 Hardware handshaking

With this procedure, control of data transfer is executed by electrical signals. Important information, such as "Clear To send" (CTS), "Clear to receive", "Start transmission" and "Stop transmission", is signalled by the hardware.
For example, when a computer character is to be transmitted, the CTS signal line (see Section 2.2 "RS-232-C/V.24 interface") is checked to see whether it is active (ON). If it is, the character is transmitted. Otherwise the computer will delay transmission until the CTS line is switched to active.
Hardware handshaking requires, as a minimum, two data lines - TxD and RxD, the RTS control line and the CTS signal line and a ground connection.

1.2.2 Software handshaking

With software handshake, control of data transfer is achieved by appropriate control characters transmitted via the data line. One such handshake is the XON/XOFF method, which is in widespread use on the RS-232-C/V.24 interface. The meaning "XON" is assigned to an ASCII code control character (DC1) and the meaning "XOFF" to another (DC3). Before transmitting a character, the computer checks whether the receiving unit is transmitting the XOFF character. If it is, it delays transmission until it receives the character XON, indicating that the connected unit is ready to receive further characters.
Apart from the data lines (TxD, RxD), and ground, no other lines are needed for software handshaking.

2 TNC data interfaces

2.1 General

Besides the standard RS-232-C/V.24 interface, the TNC 407/415 has another data interface, the RS-422/V.11, which is used for long transmission distances.

The two interfaces differ only in the construction of their hardware (signal lines, signal levels and pin layout).The data format and transmission protocol are the same for both interfaces.

The HEIDENHAIN FE 401 floppy disk unit, ME 101 magnetic tape unit (no longer in production) and external devices with appropriate data interfaces (computers, printers, readers, punches) can be connected via either RS-232-C/V.24 or RS-422/V.11 interfaces.

The two interfaces of the TNC can also be operated in parallel; for example, a program can be read in via the RS-232-C/V.24 interface and simultaneously executed, while a data file could be read in via the RS-422/V.11 interface.

However, an external interface for level conversion must be used when connecting a floppy disk unit and magnetic tape unit to the RS-422/V.11 interface.

Two transmission protocols are available for data transfer:

– Standard data transmission protocol
– Data transfer with Block Check Character (BCC)

The data format and control characters concerned are freely configurable within the framework of these protocols.

2.2 RS-232-C/V.24 interface

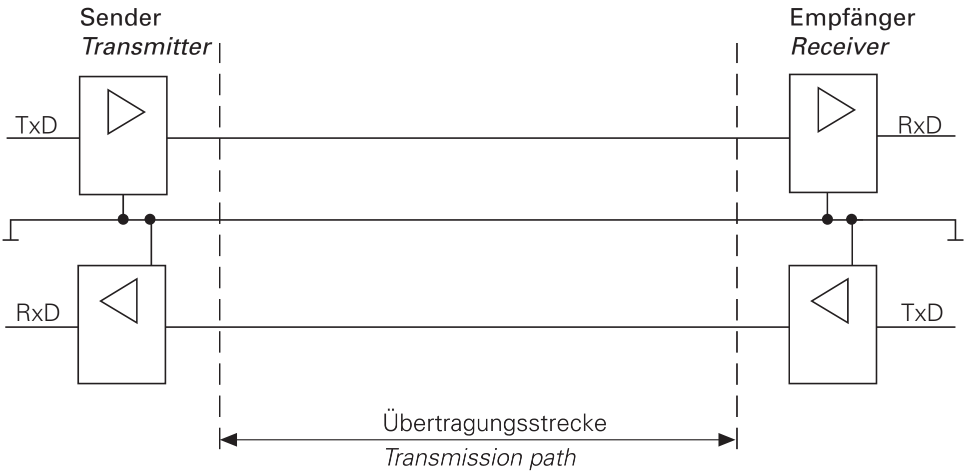
RS-232-C is the designation of a serial interface based on the EIA standard of the same name and can be used for transmission rates up to 19,200 bits/sec. Data transfer is executed asynchronously, with a start bit before each character and one or two stop bits after each character. The interface is designed for transmission distances of up to 30 metres.

The RS-232-C interface has been adopted with slight modifications and has been introduced into Europe as the V.24 interface. The relevant German standard is DIN 66020.

2.2.1 Hardware

The physical connection between two RS-232-C/V.24 interfaces is an asymmetrical line, i.e. the common ground connection between transmitter and receiver is used as a return wire.

Physical connections:



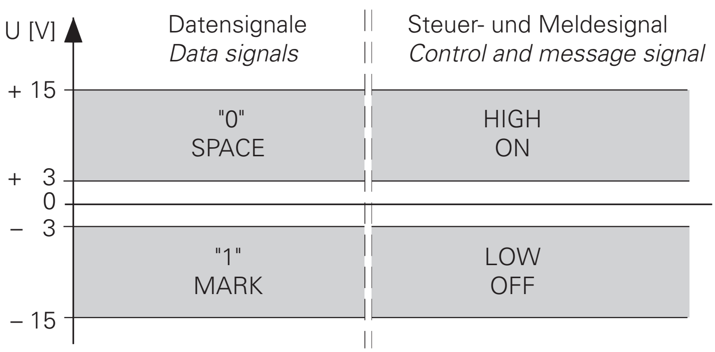
2.2.2 Signal levels

The RS-232-C/V.24 interface must differentiate between two different signal lines and their levels.

Data lines:
The data signals are defined as being logic "1" (MARK) over the range -3V to -15V and as logic "0" (SPACE) over the range +3V to +15V.

Control and signal lines:
These signals are defined as being ON (High) over the range +3V to +15V and as OFF (Low) over the range from -3V to -15V.

For all of the signals, the voltage range from -3V to +3V is not defined as a logic level and can therefore not be evaluated.



2.2.3 Signal designations

The RS-232-C/V.24 interface distinguishes between data lines, control/signal lines and the earth conductor.

Data lines:

TxD Transmitted data
RxD Received data

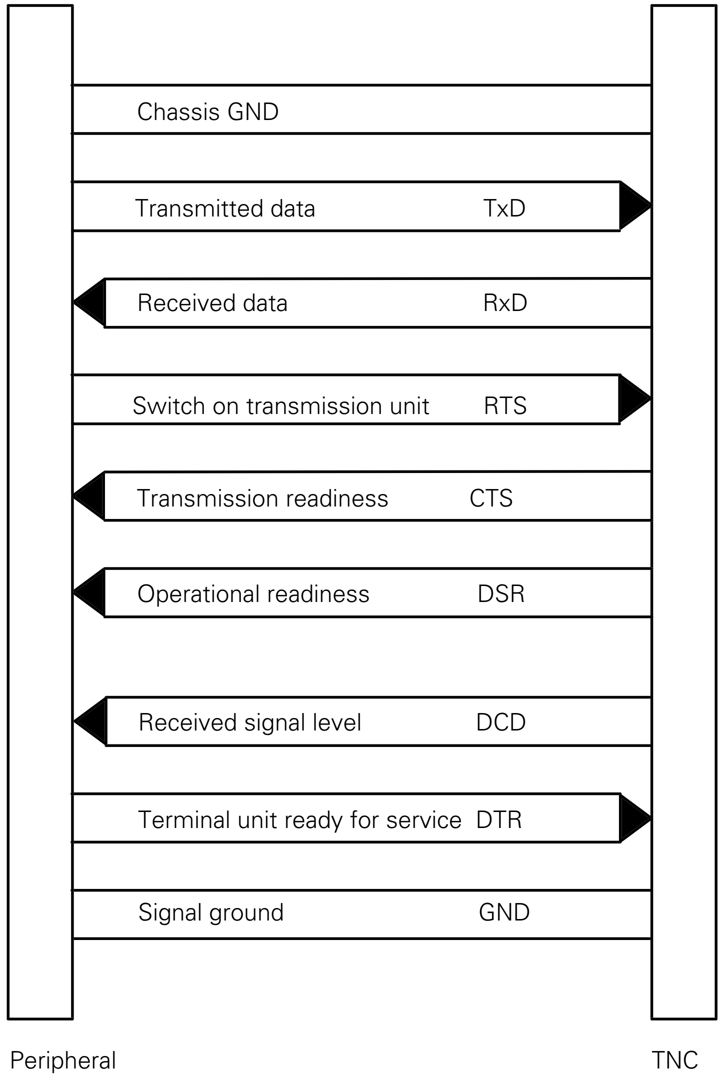
Control/signal lines:

DCD (Data Carrier Detect):	Received signal level. The DCD signal indicates to the transmitter that the information received at the receiver lies within the defined level.
	The DCD signal (pin 8) is not used by the TNC, i.e. the TNC delivers no signal from pin 8.
DTR (Data Terminal Ready):	This signal shows that the TNC is ready for service (e.g. receiving buffer full => DTR = Low).
DSR (Data Set Ready):	Peripheral ready for service.
RTS (Request to Send):	Switch transmission unit on. TNC wishes to transmit data.
CTS (Clear to Send):	Readiness for transmission. The peripheral is ready to receive data.

Earth conductor (cables for power supply):

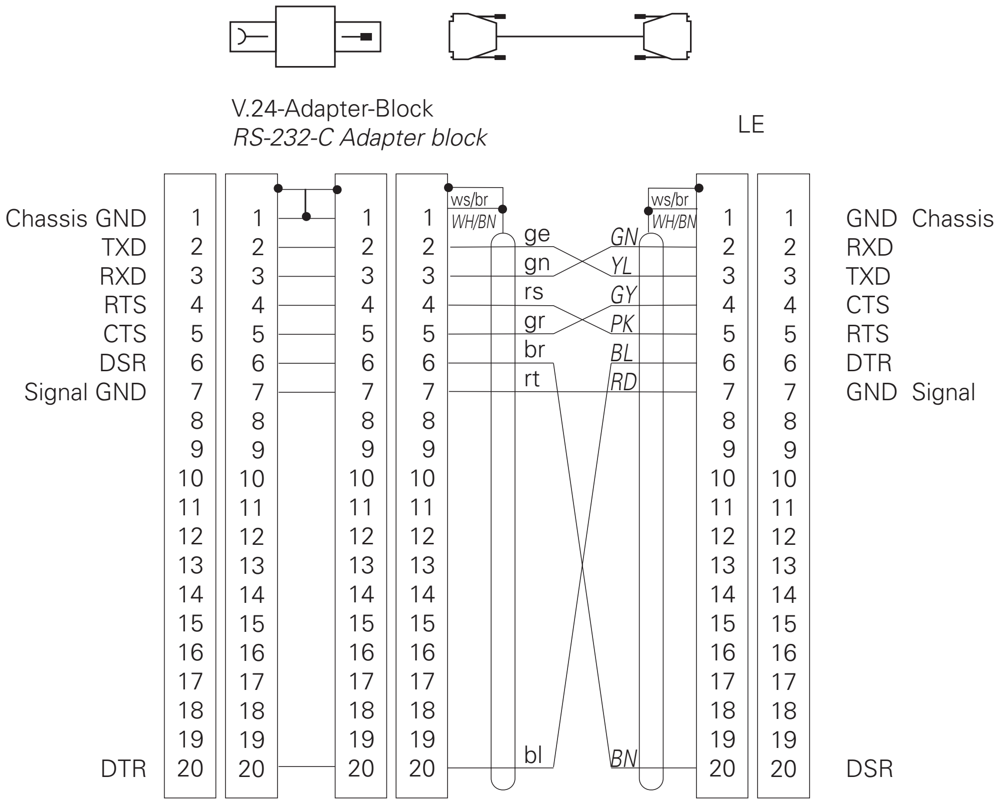
Chassis GND:	Casing connection
Signal GND:	0-Volt lines for all signals

The lines of the RS-232-C/V.24 serial interface:



2.2.4 Pin layouts

The differences between the pin layouts of the logic unit and the adapter block of the TNC should be noted! The corresponding pin layouts are shown below (see the "Installation and Electrical Connection" Manual).



A 9-pin plug on a PC should have the following pin layout:

Pin	Allocation
1	Not in use
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	RI (Ring Indicator)

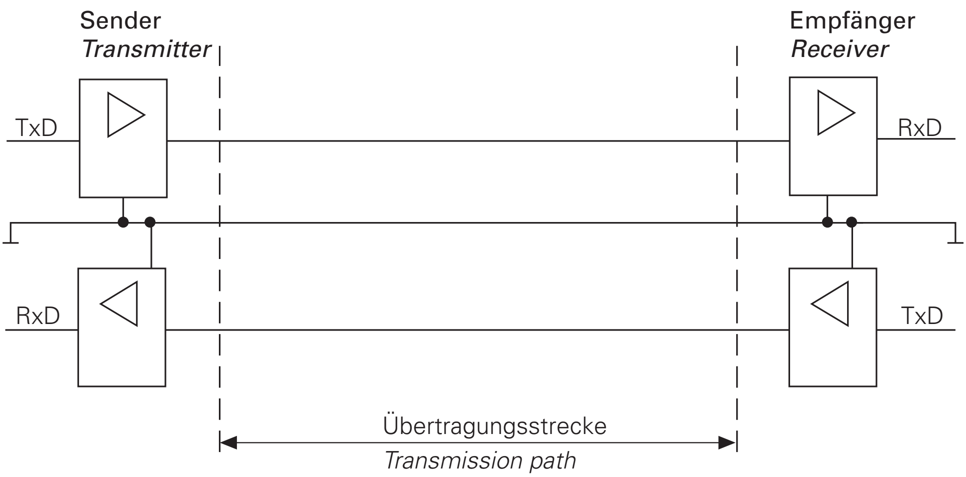
2.3 RS-422/V.11 interface

Due to the limited capabilities of the RS-232-C/V.24 interface, the RS-422/V.11 interface was developed . This interface is also standardized and works symmetrically. The RS-422/V.11 interface is suitable for data transfer speeds up to 10 Mbit/sec. Since the interface module of the TNC can only cope with a rate of up to 38,400 baud, a transfer rate of 38,400 baud must also be set at the RS-422/V.11 interface.
However, at this baud rate it is possible to transmit over a cable one kilometre long.

2.3.1 Hardware

The standard RS-422/V.11 works with differential voltages. The advantage of this method is that, on the transmission path, radiated interference acts simultaneously and by the same amount on both signal lines. At the receiver only the differential voltage of the two signal lines is evaluated and therefore the radiated interference is unimportant.
Considerably longer lines can therefore be used and, because of the suppression of interference, the transfer speed can be considerably higher.

Physical connections:



2.3.2 Signal levels

On a RS-422/V.11 interface, the signals are both transmitted and received as differential voltage. A positive differential voltage corresponds to logic "0" (OFF), and a negative differential voltage to logic "1" (ON).

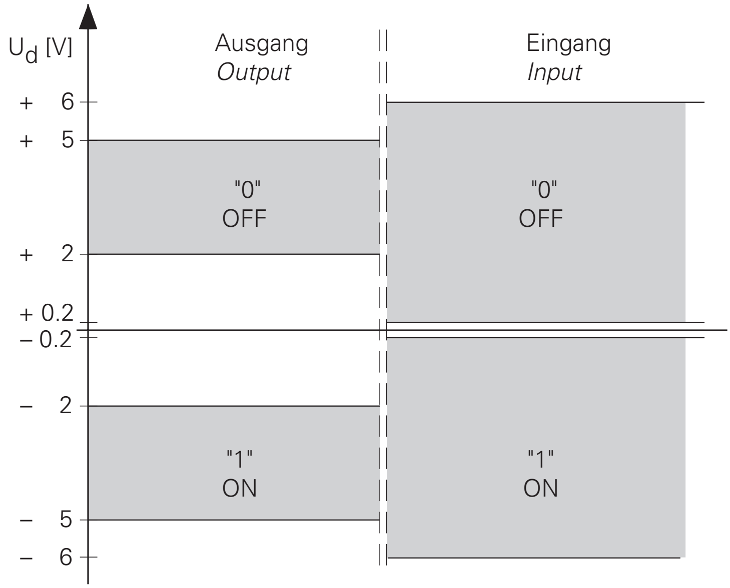
Differential voltages between

V_{dmin} = 2V and V_{dmax} = 5V

are transmitted and the control unit detects the differential voltages between

V_{dmin} = 0.2V and V_{dmax} = 6V

as a logically defined level.



2.3.3 Signal designations

With a RS-422/V.11 interface, the following signals are transmitted as differential signals:

Data signals:

TxD, $\overline{\text{TxD}}$	RxD, $\overline{\text{RxD}}$

Control and verification signals:

RTS, $\overline{\text{RTS}}$	CTS, $\overline{\text{CTS}}$
DSR, $\overline{\text{DSR}}$	DTR, $\overline{\text{DTR}}$

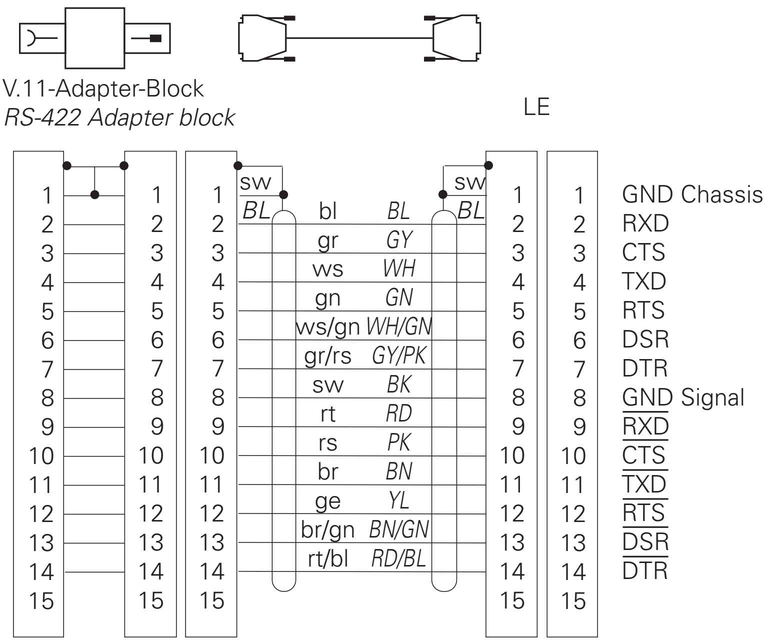
These signals perform the same functions as those on the RS-232-C/V.24 interface.

The transmission protocol is absolutely identical for the RS-232-C/V.24 and RS-422/V.11 interfaces.

In addition, the protective earth connects the transmitter and receiver casings. The signal GND represents the differential voltage reference conductor.

2.3.4 Pin layouts

On the TNC 407/415, the following pin layout is adopted at the logic unit and adapter block (see the "Installation and Electrical Connection" Manual)



2.4 Data interface functions

The data interfaces on the TNC can be used to save data and files and read them back in again, to output programs to external devices (e.g. printers), to read in programs and simultaneously execute them and to carry out data transfer (communication) between TNCs.

2.4.1 Saving/reading files

The following table lists all the files which can be saved to external memory (floppy disk unit, magnetic tape unit and PC) and can be read back in from them.

File	File extension	Code
HEIDENHAIN Dialog NC program	.H	H
DIN/ISO NC program	.I	D
Tool table	.T	T
Pallet table	.P	L
Datum table	.D	N
Machine parameters	.MP	M
Compensation table	.COM	V
Compensation Assignment	.CMA	S
PLC program	.PLC	P
Text file	.A	A
Error messages (1st language)	.ER1	A
Error messages (English)	.ERE	A
Dialogs (1st language)	.DI1	A
Dialogs (English)	.DIE	A
Pocket table	-	R
Help files	.HLP	J
Pocket tables	.TCH	R
Help files	.HLP	J
Point tables	.PNT	U

After the appropriate key numbers for the PLC, the machine parameters and the correction table have been entered, these files can be written-to or read-from via the data interfaces.

Data transfer is initiated with the EXT key as usual.

Current values of Q parameters, PLC error messages and dialogs can also be outputted via the two interfaces (NC program: FN 15: PRINT).

The magnetic tape unit is only suitable as an external data carrier to a limited extent, because only one file can be stored per cassette side. However, this file can contain several programs.

Using the floppy disk unit, up to 256 programs (approximately 25,000 program blocks) can be stored. This represents a storage capacity of approximately 790 kilobytes.

When transmitting and receiving a file the appropriate code file is outputted and read in again complete with a Block Check Character (BCC).

If the file is stored in an external computer using HEIDENHAIN's TNC.EXE data transfer software, a new file extension is generated. This extension consists of the code and the letters NC.

Example:
If a pallet-table is stored, it is given the file extension *.LNC.

When outputting the file, the error messages, dialogs and text files are all output as ASCII files with the code A.*. Consequently, if they are sent on an external data carrier, these files must have different file names so that they are not overwritten by files with similar names.
If the files are read in again, they are first of all stored as ASCII files and have to be converted back to the original file-type by the user.
The menu functions "Read all files", "Read files with acknowledgement" and "Show ext. directory" only work with files which contain the name concerned at the start of the file (NC programs and tables). The remaining files (PLC, error, dialog and ASCII files) have to be read in individually.

2.4.2 Output to external devices

Any external device, e.g. computers, printers, readers and punches, can be addressed via either of the two interfaces.
For this purpose, the TNC has three freely configurable interface modes (EXT1/EXT2/EXT3) which, within certain limits, permit any setting of the data format and control characters of the required data transmission protocol.

The setting selected at the external devices must of course match the TNC. On printers, this is done by setting the DIP-switches or adjusting the transmission parameters.

If data transfer to a computer is desired, appropriate data-transfer software must be installed. To help in this, HEIDENHAIN offer their TNC.EXE data transfer software, which permits transfer between TNC and a PC using a fixed transmission protocol.

2.4.3 Reading in and simultaneously executing programs (DNC operation)

In RUN PROGRAM mode with "Block transfer", machine programs can be transmitted from an external memory or the FE floppy disk unit via serial data interfaces and be executed simultaneously (DNC operation). This allows machine programs which exceed the storage capacity of the control unit to be executed.

After the program has started, executed blocks are erased and further blocks are continuously called from the external memory.

The minimum and maximum memory requirement for the drip feed mode is defined with machine parameters MP7228.0 and MP7228.1 respectively. The maximum memory requirement can also be set to leave enough spare for parallel programming during execution.
The error message "Program Memory Overflow" is output if the minimum storage is not enough.

If the program blocks are read into the TNC faster than they can be executed, the entire NC memory is initially fully loaded up to the maximum memory limit (MP7288.1). If it is full, transfer is suspended until a storage space of 256 bytes is vacated by execution. Only then is transfer resumed, until once again the memory is filled.

MP7228	Storage requirement in drip feed mode Entry range: 0 to 1024 KBytes
MP7228.0	Minimum storage
MP7228.1	Maximum storage

2.4.4 Communication between TNCs

For certain applications, it is necessary for TNCs to be able to exchange data or to communicate with each other. This is made possible by both RS-232-C/V.24 and RS-422/V.11 interfaces.

The simplest form of data exchange is the transfer of files (e.g. NC programs) from one TNC to another. To do this, the same transmission format must be set at both control units and transfer started. It must be ensured that the reading-in control unit is the first to start.

For the application "Positioning module", i.e. further NC axes are required, the positioning values have to be transmitted from the master logic unit to the slave logic unit. On this basis, data can be transferred to another TNC via the data interface, using PLC modules at the PLC level (see Section 5 "Data transfer via PLC").

2.5 Configuration of interfaces

2.5.1 Selection of interfaces

Either data interface may be inhibited with MP 5000.

MP 5000 Inhibit data interfaces
Input range: 0 to 2

0 = no interface inhibited
1 = interface RS-232-C/V.24 inhibited
2 = interface RS-422/V.11 inhibited

If neither of the two interfaces is inhibited, the following settings can be selected with the appropriate data format and the data transmission protocol.

In addition to the three freely configurable operating modes EXT1/EXT2/EXT3 (for PLC only, see Section 5 "Data transfer by PLC"), there are three fixed modes (ME, FE1/FE2). The FE1 and FE2 modes have to be set if the HEIDENHAIN floppy disk unit or an external computer using the TNC.EXE transmission software is connected. In both these modes, the transmission protocol with Block Check Character is rigidly defined [for the difference between FE1 and FE2 operation: see Section 4 "Data transfer with block Check Character (BCC)"].

The ME operating mode matches the HEIDENHAIN ME 101 magnetic tape unit and must be selected if this is to be connected.

In this mode, the standard transmission protocol is set.

However, there is one important difference:

The character <EOT> is never sent because this character cannot be processed by the magnetic tape unit.

2.5.2 Freely configurable interfaces

The three operating modes EXT1/EXT2/EXT3 (PLC) are freely configurable via the machine parameters.

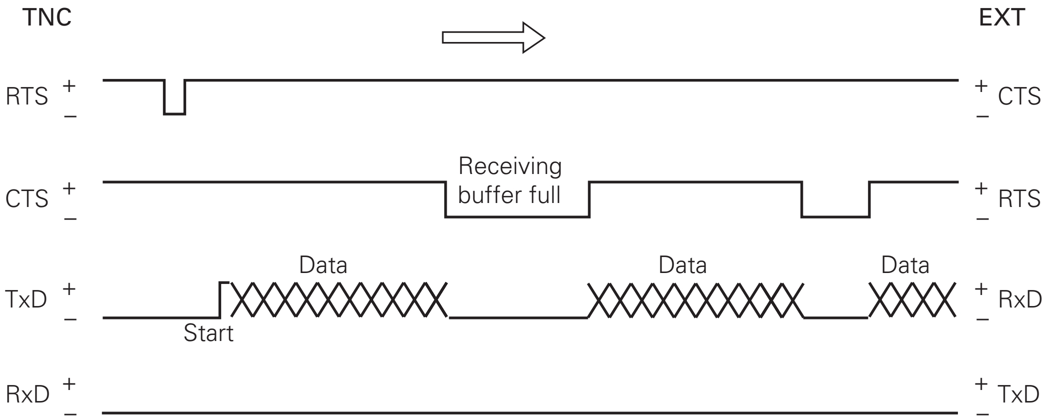
The data format and the type of handshake are set in the MP 5020.

Data bits
Bit 0 can be set to determine whether transmission is to be with 7 or 8 data bits. Transmission with 7 bits is normally used, but 8 bits are needed, especially for printer interfacing.

BCC
If calculation of the BCC produces a number less than $20 (i.e. a control character) then a "Space" character ($20) is sent in addition immediately before <ETB>. The BCC will consequently always be greater than $20 and cannot therefore be recognised as a control character.

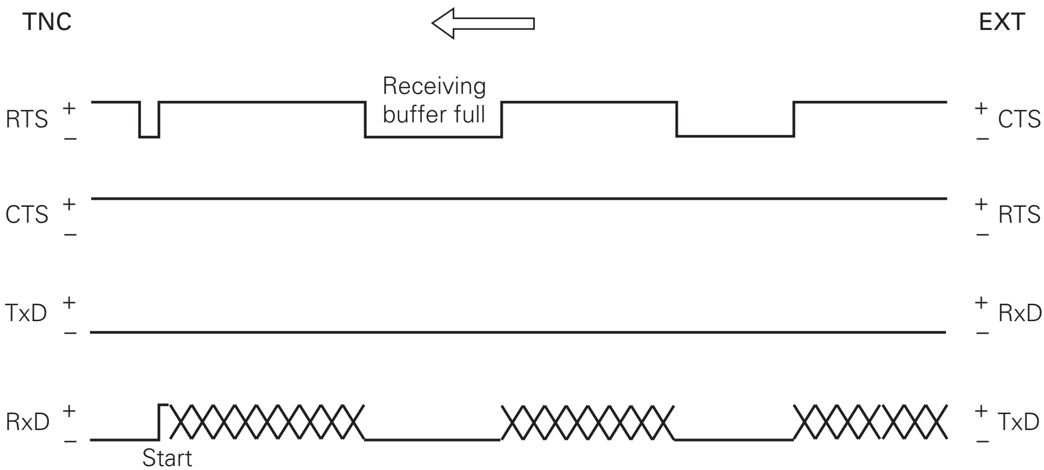
Hardware handshaking
Bit 2 can be set to determine whether the TNC stops transfer from an external device by using RTS.

– **Data output TNC -> EXT**
  When the receiving buffer is full, the external device resets the RTS signal. The TNC thereby detects that the peripheral unit receiving buffer is full because of the CTS input.



+ Positive voltage level
– Negative voltage level

– **Data input EXT -> TNC**
  When the receiving buffer is full, the TNC removes the RTS signal, which is detected by the peripheral device at its CTS input.



+ Positive voltage level
– negative voltage level

The DTR and DSR signals from the TNC indicate the operational status of the TNC and peripheral (these cannot be set via the machine parameters).

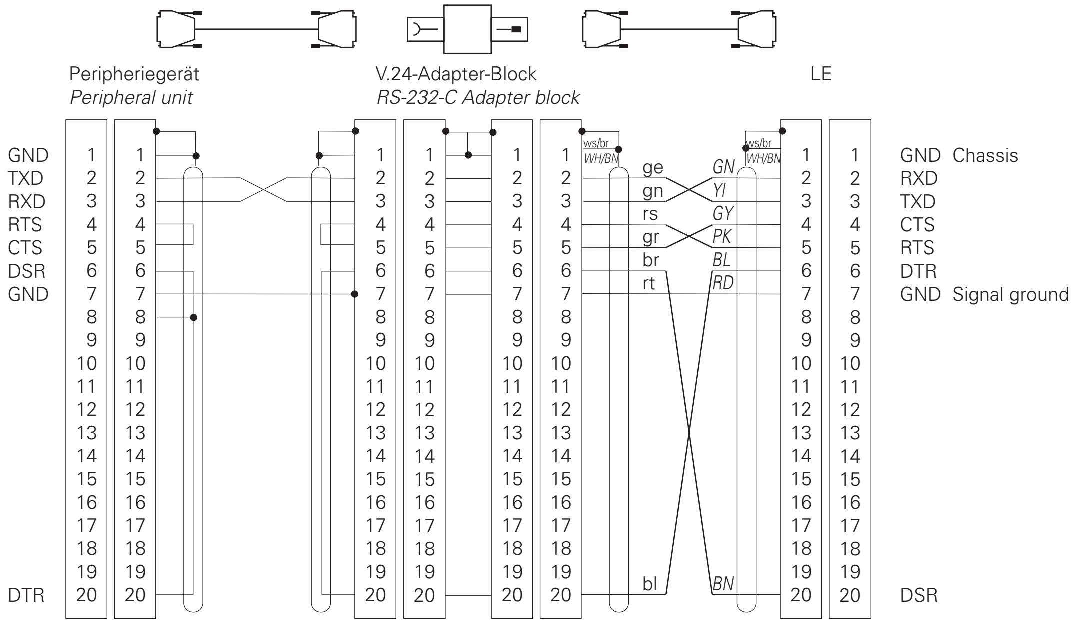
DTR: Interrogated by peripheral; it is logic "1" if TNC is ready for service.
DSR: Interrogated by TNC.
LOW level => ext. data input/output not ready.
HIGH level => ext. data input/output ready.

Software handshaking

Bit 3 determines whether the TNC stops transfer from an external device with control character <DC3>. Transfer is resumed with character <DC1>.

If transfer is stopped with character <DC3>, up to 12 characters can still be stored. The remaining incoming characters are lost. Software handshake is normally recommended when interfaces are connected to an external device.

The following pin layout is possible for the external device:



A HEIDENHAIN standard cable, ref. 242 869, is recommended.

The TNC reacts both to hardware and software handshakes, regardless of the setting in MP5020.x.

If no transmission stop is set in the MP5020.x, the TNC stops the peripheral unit with the software handshake.

If transmission stop by both RTS and by DC3 is active, the TNC stops transfer with the hardware handshake.

Character parity
Bits 4 and 5 determine the type of parity check (see paragraph 1.1.3 "Checking data").

Stop bits
Bits 6 and 7 determine the number of stop bits sent at the end of a character.

MP 5020.0 Operating mode EXT1
MP 5020.1 Operating mode EXT2
MP 5020.2 Operating mode EXT3 (PLC)
Entry: %xxxxxxxx

Bit 0	7 or 8 data bits	0 = 7 data bits 1 = 8 data bits
Bit 1	Block check character	0 = any BCC 1 = BCC not control character
Bit 2	Transmission stop by RTS	0 = not active 1 = active
Bit 3	Transmission stop by DC3	0 = not active 1 = active
Bit 4	Character parity	0 = even 1 = odd
Bit 5	Character parity	0 = not desired 1 = desired

Bit 6/7 Stop bits

		Bit 6	Bit 7
1 1/2 stop bits	Bit	0	0
2 stop bits	Bit	1	0
1 stop bit	Bit	0	1
1 stop bit	Bit	1	1

For operating modes EXT1/EXT2/EXT3, MP5030 defines the transmission protocol.

MP5030.0 Operating mode EXT1
MP5030.1 Operating mode EXT2
MP5030.2 Operating mode EXT3 (PLC)
Entry : 0 or 1

0 = "Standard data transfer"
1 = "Transfer blockwise"

For the control characters for the data transmission protocols (<SOH> <ETB>, <STX>, <ETX>, <EOT>, <ACK> <NAK>), any other ASCII characters can be chosen using the following machine parameters (for table of ASCII characters, see Appendix). If these machine parameters are loaded with a nought, the standard settings given in brackets are active (as for FE1 and FE2 operation).

MP5200	Control character for Start of Text (STX)
MP5200.0	In mode EXT 1
MP5200.1	In mode EXT 2
MP5200.2	In mode EXT 3 (PLC)
	Input range: 0 to 127
MP5201	Control character for End of Text (ETX)
MP5201.0	In mode EXT 1
MP5201.1	In mode EXT 2
MP5201.2	In mode EXT 3 (PLC)
	Input range: 0 to 127
MP5206	Control character for Start of Command Block (SOH)
MP5206.0	In mode EXT 1
MP5206.1	In mode EXT 2
MP5206.2	In mode EXT 3 (PLC)
	Input range: 0 to 127
MP5207	Control character for End of Command Block (ETB)
MP5207.0	In mode EXT 1
MP5207.1	In mode EXT 2
MP5207.2	In mode EXT 3 (PLC)
	Input range: 0 to 127
MP5208	Control character for "Transfer O.K." (ACK)
MP5208.0	In mode EXT 1
MP5208.1	In mode EXT 2
MP5208.2	In mode EXT 3 (PLC)
	Input range: 0 to 127
MP5209	Control character for "Transfer Defective" (NAK)
MP5209.0	In mode EXT 1
MP5209.1	In mode EXT 2
MP5209.2	In mode EXT 3 (PLC)
	Input range: 0 to 127
MP5210	Control character for "End of Transmission (EOT)
MP5210.0	In mode EXT 1
MP5210.1	In mode EXT 2
MP5210.2	In mode EXT 3 (PLC)
	Input range: 0 to 127

When selecting ASCII characters, it must be ensured that the control characters are not arbitrarily mixed and that no figures or letters which occur in the transferred text are used.
Example: The control character for Start of Text (MP5200.x) must not be assigned to <DC3>, otherwise transmission will stop when software handshaking is set on the peripheral.

For the file type, when transferring with Block Check Characters (see paragraph 2.4.1 "Saving/reading files"), an ASCII character must be entered for the file which is being output or input. With an input value of nought, the TNC automatically enters the correct type of file in the file header.

MP5202	ASCII character for file type for data input
MP5202.0	In mode EXT 1
MP5202.1	In mode EXT 2
MP5202.2	In mode EXT 3 (PLC)
	Input range: 0 to 127

MP5204	ASCII character for file type for data output
MP5204.0	In mode EXT 1
MP5204.1	In mode EXT 2
MP5204.2	In mode EXT 3
	Input range: 0 to 127

Example: If a DIN/ISO program is to be output in operating mode EXT2, MP5204.1 = 68 (='D') must be set (or MP5204.1=0).

The ASCII characters for input and output identification can also be freely defined.

If the machine parameters concerned are loaded with nought, the standard settings apply (E=Input/A=Output).

MP5203	ASCII character for input identification (E)
MP5203.0	In mode EXT 1
MP5203.1	In mode EXT 2
MP5203.2	In mode EXT 3 (PLC)
	Input range: 0 to 127

MP5205	ASCII character for input identification (E)
MP5205.0	In mode EXT 1
MP5205.1	In mode EXT 2
MP5205.2	In mode EXT 3 (PLC)
	Input range: 0 to 127

2.6 External programming

In the case of external programming and subsequent transfer, attention should be paid to the following:

– <CR><LF> or <LF> must be programmed at the start of the program and after each program block
– After the end-of-program block <CR> <LF> and, in addition, the end-of-text control character must be programmed (standard setting <ETX> or via MP 5201.x)
– Blank characters between the individual words can be omitted in NC programs.
– When reading in DIN blocks, '*' is not needed at the end of the block
– Comments are separated from the NC block by a semicolon ';'
– Comments located in front of the program are not stored
– Block numbers do not need to be programmed - they are generated by the TNC (only for dialog programming)

2.7 Interfacing with other equipment

Any other external devices can be interfaced with the TNC by using configurable operating modes ET1/EXT2/EXT3. For this purpose, machine parameters 5020.x to 5210.x permit relatively free adjustment of the data format, the data transmission protocol and the control characters.

Consider the example of interfacing EXT1 with a printer using a serial interface (Example: NEC P7PLUS).

The following setting is selected at the printer itself (see the Operating Manual of the printer concerned):

– Serial interface
– Data bits
– Even character parity
– XON/XOFF protocol (software handshake)
– 9,600 baud

The following settings (EXT1) are made at the TNC:

MP5000 = 0	No interface inhibited
MP5020.0 = %10101001	8 data bits
	Any BCC character
	Transfer stop by RTS not active
	Transfer stop by DC3 active
	Character parity even
	Character parity required
	1 stop bit
MP5030.0 = 0	Standard data transfer

In the "RS-232/RS-422 Setup" of the TNC, the EXT1 operating mode must still be assigned to the RS-232 interface and the baud rate set to 9,600 (see the TNC 407/415 Operating Manual).

3 Data transmission protocols

The TNC enables data and files to be transferred using different protocols (which can be selected via the interface setup or the machine parameters).

These transmission protocols can be selected in six different operating modes, as follows:

– ME	Standard transmission protocol to match HEIDENHAIN magnetic tape unit (7 data bits, 1 start bit, 1 stop bit)
– FE1, FE2	Transmission with Block Check Character and with fixed control characters (7 data bits, 1 start bit, 1 stop bit)
– EXT1, EXT2, EXT3	Freely configurable operating modes: data format, transmission protocol and control characters can be freely set via machine parameters
LSV/2	Two-way transfer for TNC diagnostics and remote operation conforms to DIN 66019. This protocol always runs in the background of the TNC and is started externally by a PC.

The following applies to data transmission protocols (except LSV/2) :

– If a file which is read in is already stored in the TNC, the message "ERASE/OVERREAD" is displayed. In this case the TNC aborts transmission with the appropriate handshake and does not continue transfer until after acknowledgement.

  In the event of an attempt to erase write-protected files, the error message "PROTECTED PGM" is displayed and the dialog "CONTINUE = ENT/END = NOENT ". In this case, either the next file can be read in or the transfer can be aborted.

– If a file has been read out and the data transfer menu has been terminated with the END key, the TNC outputs characters <ETX> and <EOT> (or ASCII characters according to setting in MP5201.x and MP5210.x in operating modes EXT1, EXT2 and EXT3).

– If a transmission is terminated with the END-key, the error message "PROGRAM INCOMPLETE" is issued.

3.1 Standard transmission protocol

3.1.1 General

This protocol is set as standard in operating mode ME and can also be optionally selected for operating modes EXT1/EXT2/EXT3 via the machine parameters. In the following, the control characters which are sent and received with this protocol are listed for the various transmission alternatives. When outputting a file, the <NUL> character is sent exactly 50 times at the start of the file. When reading in, however, the control unit ignores this character. Therefore it is of no importance how often the peripheral unit sends the <NUL> character before the file.

If, however, ME mode is set instead of EXT1/EXT2/EXT3, attention must be paid to the following:

All the programs in a file which end with the end-of-text character <ETX> are stored in the magnetic tape unit ME.

The ME transmits this file with all the programs to the control unit. The TNC then selects the appropriate program and stores it.

A further important difference between EXT1/EXT2/EXT3 mode in the standard protocol and ME mode is that the <EOT> character is never sent in ME mode because the magnetic tape unit is incapable of processing this character.

In this protocol, if an error is to be signalled to the TNC, the following sequence of instructions must be sent:

<ESC> <1> 'ERROR NUMBER'

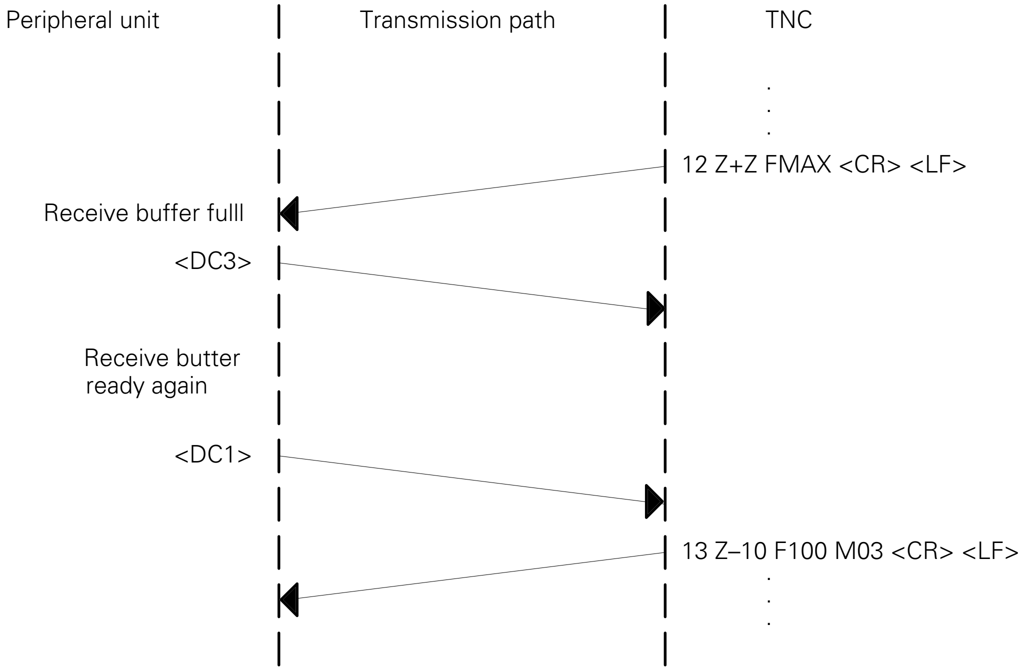
The blocks are not checked for correctness but are transmitted one after the other. If the receiver's data buffer is full it can stop the transfer and resume in one of two ways:

– Stop transfer by sending the character <DC3> (XOFF);
  continue by transmitting character <DC1> (XON) (software handshake).
– By suitable levels on the control and message lines RTS and CTS of interfaces V.24/RS-232-C or V.11/RS-422 (hardware handshake).

Example: Protocol for dialog program

<NUL><NUL><NUL><NUL><NUL><NUL><NUL>...	50 times
0 BEGIN PGM 1 MM<CR><LF>	Program block 1
1 TOOL DEF 1 L+0 R+3<CR><LF>	Program block 2
.	
.	
.	
26 END PGM 1 MM <CR><LF>	End of program
<ETX><EOT>	Close data transfer menu

Example of software handshake:



Hardware handshake (see section "Freely configurable interfaces")

3.1.2 Protocols

This section lists the transfer protocols for the various methods of data output and input.

The EXT1 mode is set:

– Control character for "End of Text": <ETX>
– Control character for "End of Transfer": <EOT>
– Software handshake

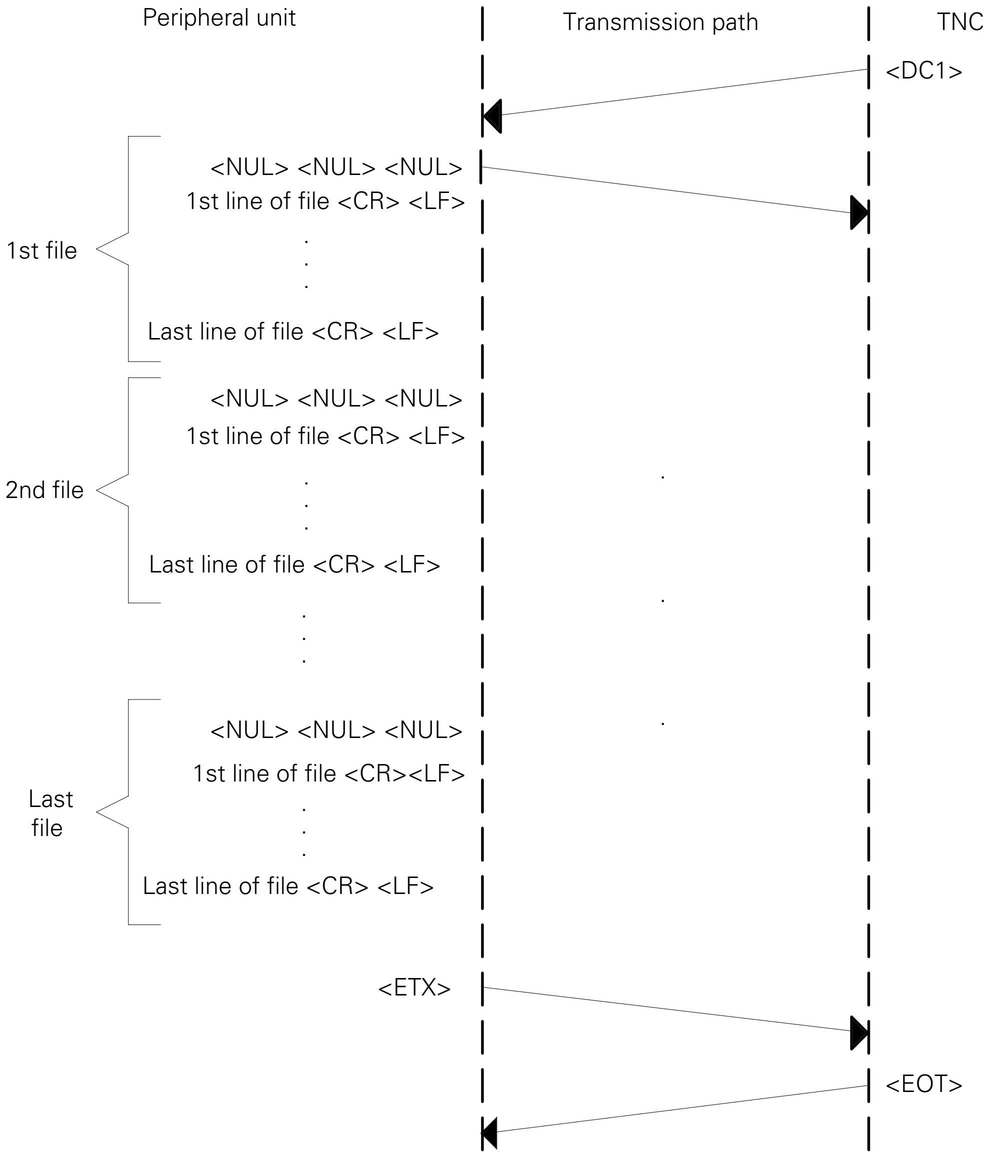
Request external directory

Using the soft key "Show ext. directory", the list of file names can be requested from an external memory and displayed in the TNC.

If the external directory is requested, the TNC sends control character <DC1>.

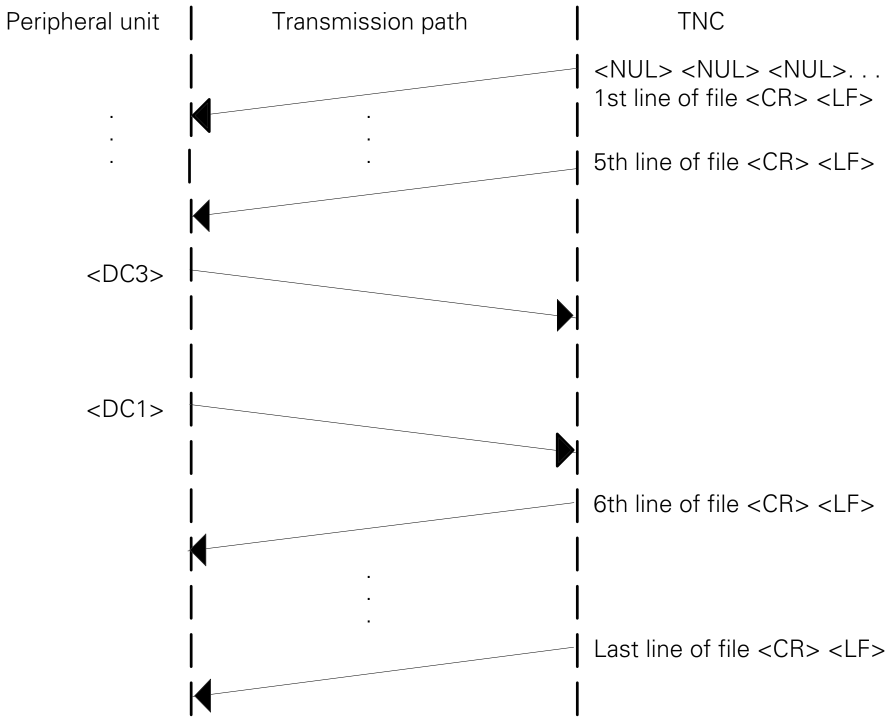
If the request is immediately interrupted with the END key, the TNC sends characters <ETX><EOT> and no directory is read in.

If the request is not interrupted, the peripheral unit sends all of the external programs in order. Their names are then shown in the TNC.



Output selected file

The TNC outputs all of the program lines in order. The peripheral unit can stop transmission with character <DC3> and start it again with character <DC1>.



Output all files

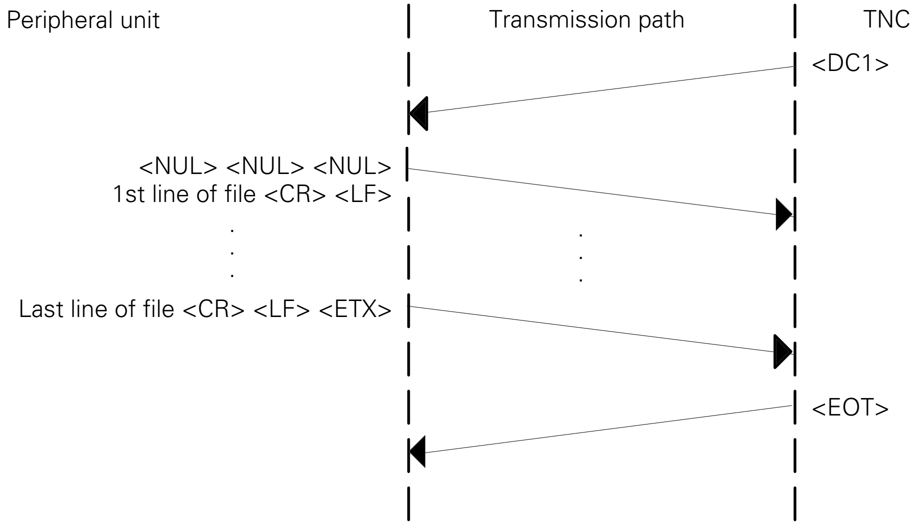
The procedure is similar to the protocol described in paragraph 3.1.2 "Outputting a selected file". The TNC arranges all the programs in order and transmits them. No control character is sent between the individual files.

Output file with acknowledgement

The programs are transmitted in the same way as described in section 3.1.3 "Outputting all files". However, an acknowledgement is expected from the user between the transmission of individual programs.

Read-in selected file

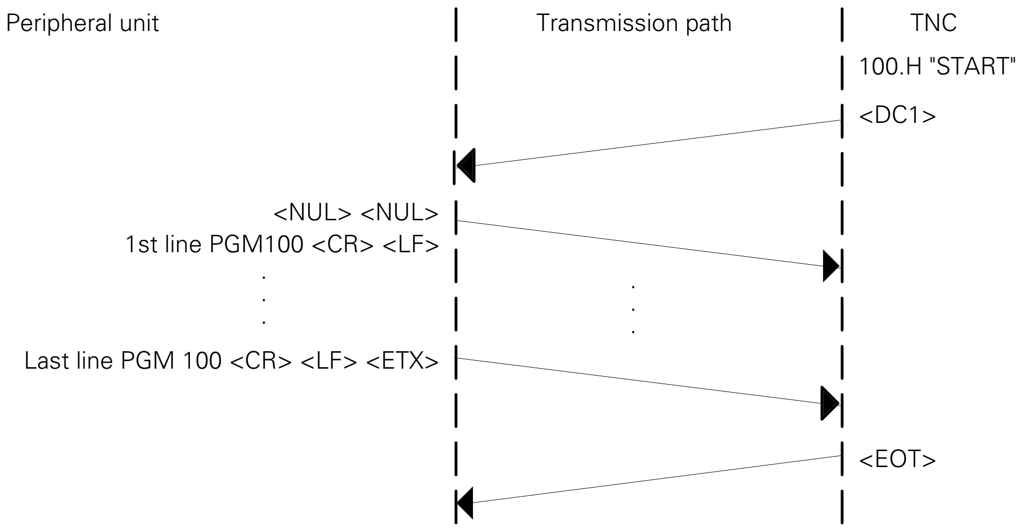
If a file is read in from a peripheral unit (e.g. a PC), the corresponding name must be indicated in the TNC and the TNC be started first, i.e. the TNC outputs the character <DC1>. Transmission of the file concerned is then initiated at the peripheral unit.
When the entire file has been transferred, the TNC sends character <EOT>.



In this transfer method, the TNC can stop transmission with <DC3> and continue it with <DC1>. If the file name in the first line of the file and the name indicated in the TNC are not identical, the TNC reads each block in and searches for the file name concerned.

If the END PGM-block has been read in, and the selected name is not known, the TNC remains static without an error message, and transfer must be terminated with the END key.

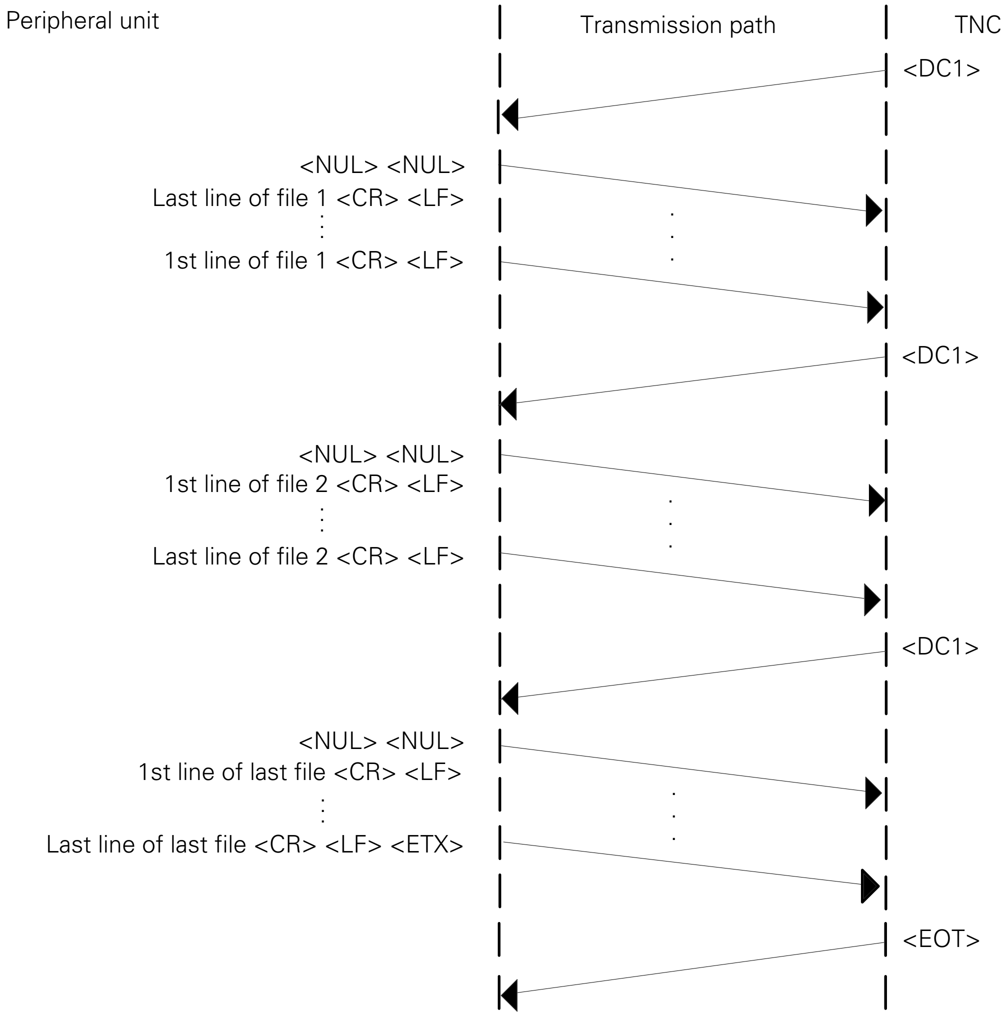
Example: Reading in program 100.H.



If, in this case, the last PGM-block ends with the <ETX> character then transfer is terminated without an error message but the data is not stored.

Read-in all files

If both the peripheral unit and the TNC have been started, the following protocol is followed:

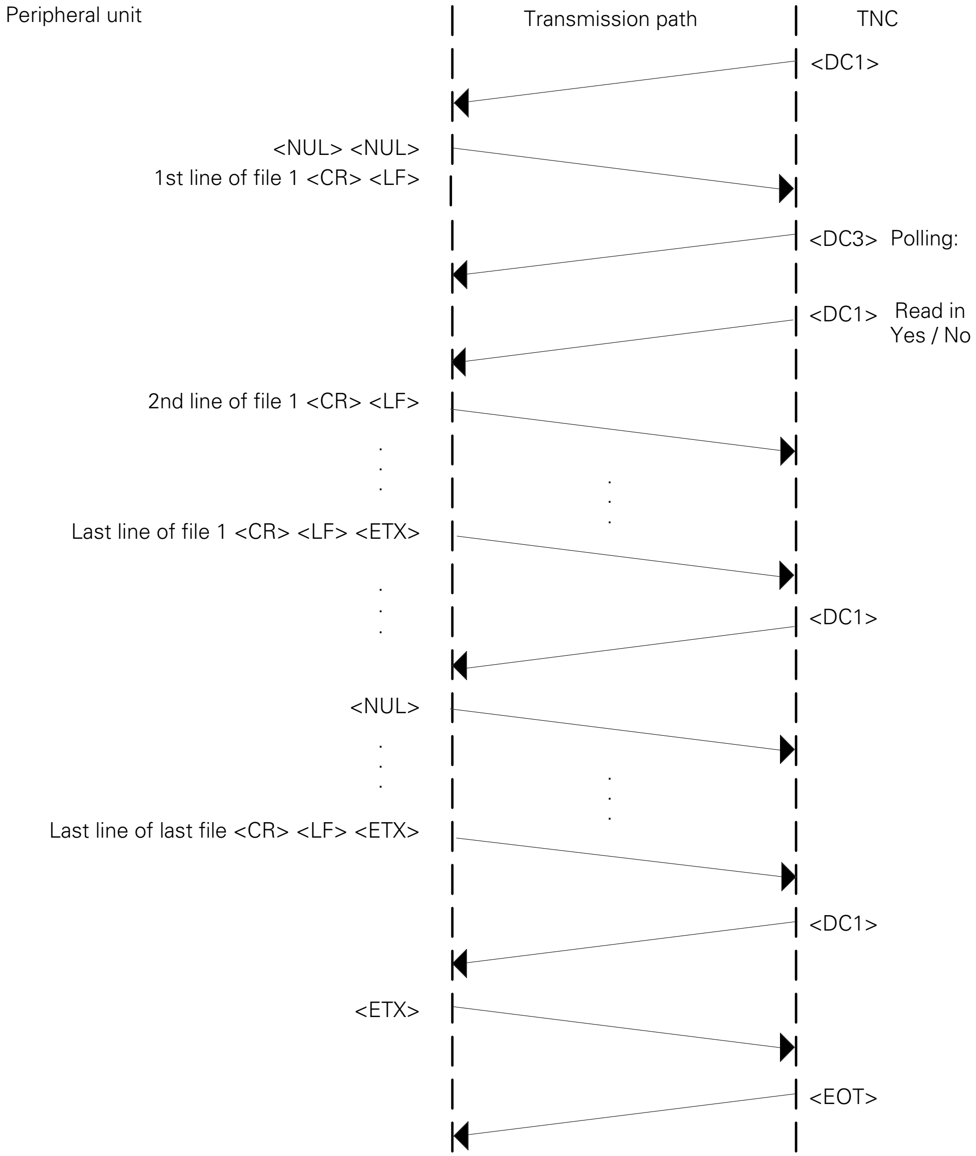


If several programs are gathered together in a file which ends with <ETX> then these programs are read in without being requested by <DC1>.

The request <DC1> is not sent until a program has ended with <ETX>.

Read-in file with acknowledgement

After commencement of transfer, the peripheral unit sends the first program module until the receiving buffer of the TNC is full. The TNC then stops transmission with <DC3> and awaits acknowledgement from the user. If the file is to be transferred, the TNC sends <DC1> and the program is read in and stored. Otherwise the file is in fact read in but not stored. If hardware handshaking is set, transfer by using the RTS signal is stopped and restarted.



3.2 Data Transfer with Block Check Character (BCC)

3.2.1 General

This protocol is specific to HEIDENHAIN and operates with different control characters and an additional data check feature when transferring.

The protocol is set with the following operating modes:

- FE1 mode
- FE2 mode
- EXT1/EXT2/EXT3 mode (selectable)

The data transfer protocol is fully identical for all these modes except for the FE1 mode in which a command sequence is automatically output at the beginning to request the contents directory from the peripheral unit.

In the freely configurable modes (EXT1/EXT2/EXT3) the following control characters (<SOH>, <ETB>, <STX>, <ACK>, <NAK>, <ETX>, <EOT>) can be defined at will as ASCII characters (see section "Freely configurable interfaces").

When a file is transferred the first block - the so-called Header - is transmitted, consisting of the following characters:

<SOH>"H" "Name" "M" <ETB>BCC<DC1>

<SOH> (Start of Header): This character identifies the beginning of the header.

The header contains the identifier "H" of the program (see section "Saving/reading files"), the program name "Name" and the transfer mode "M" (E=Input/A=Output).

The header terminates with the character <ETB> which ends a data transfer block.

The next character BCC provides an additional data check:

As well as checking the parity of the individual characters (see chapter "Checking data") the parity of the complete transferred block is also checked. The BCC (Block Check Character) always rounds the individual bits of the transferred characters in a data transfer block to even parity.

Example of BCC generation:

Character	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SOH	0	0	0	0	0	0	1
H	1	0	0	1	0	0	0
1	0	1	1	0	0	0	1
5	0	1	1	0	1	0	1
E	1	0	0	0	1	0	1
ETB	0	0	1	0	1	1	1
BCC	0	0	1	1	1	1	1

In this example program "15" that has been written in HEIDENHAIN dialog ("H") is downloaded across the data interface (E"). A parity bit is also generated for the BCC (with even parity the BCC parity bit in this example is assigned significance "1").

The character <DC1> is transmitted after the BCC. This character (XON) is necessary for certain devices in order to explicitly request an answer from them and to reactivate the transfer.

The transfer of the <DC1> character can be suppressed in modes EXT1, EXT2 and EXT3 by setting bit 3 in machine parameter 5020.X ("Transmission stop by DC3") to zero (see section "Freely configurable interfaces").

The <DC1> character is not required when reading in a file in the format with BCC.

At the end of every block the receiver checks that it has transferred correctly. To do this the receiver computes a BCC from the received block and compares it with the received BCC. If the received BCC and the computed BCC are identical, the receiver transmits the <ACK> character (= positive checkback), i.e. the data block has been transmitted without error.

If the two BCCs are not identical then the receiver transmits the <NAK> character (= negative checkback), i.e. the data block was not transmitted correctly and must be re-transmitted. This process is repeated three times, then the error message ("TRANSFERRED DATA INCORRECT E {XE "TRANSFERRED DATA INCORRECT E") is output and the transfer is aborted.

If the header is acknowledged with <ACK> however the first file block can be transmitted:
<STX>0 BEGIN PGM 1 MM <ETB> BCC <DC1>

The beginning of a data block is always identified by the control character <STX>. The other control characters in this block are identical with the control characters in the header.

If the block is acknowledged by <ACK> then the next program block is transmitted, with <NAK> the same block has to be re-transmitted etc.

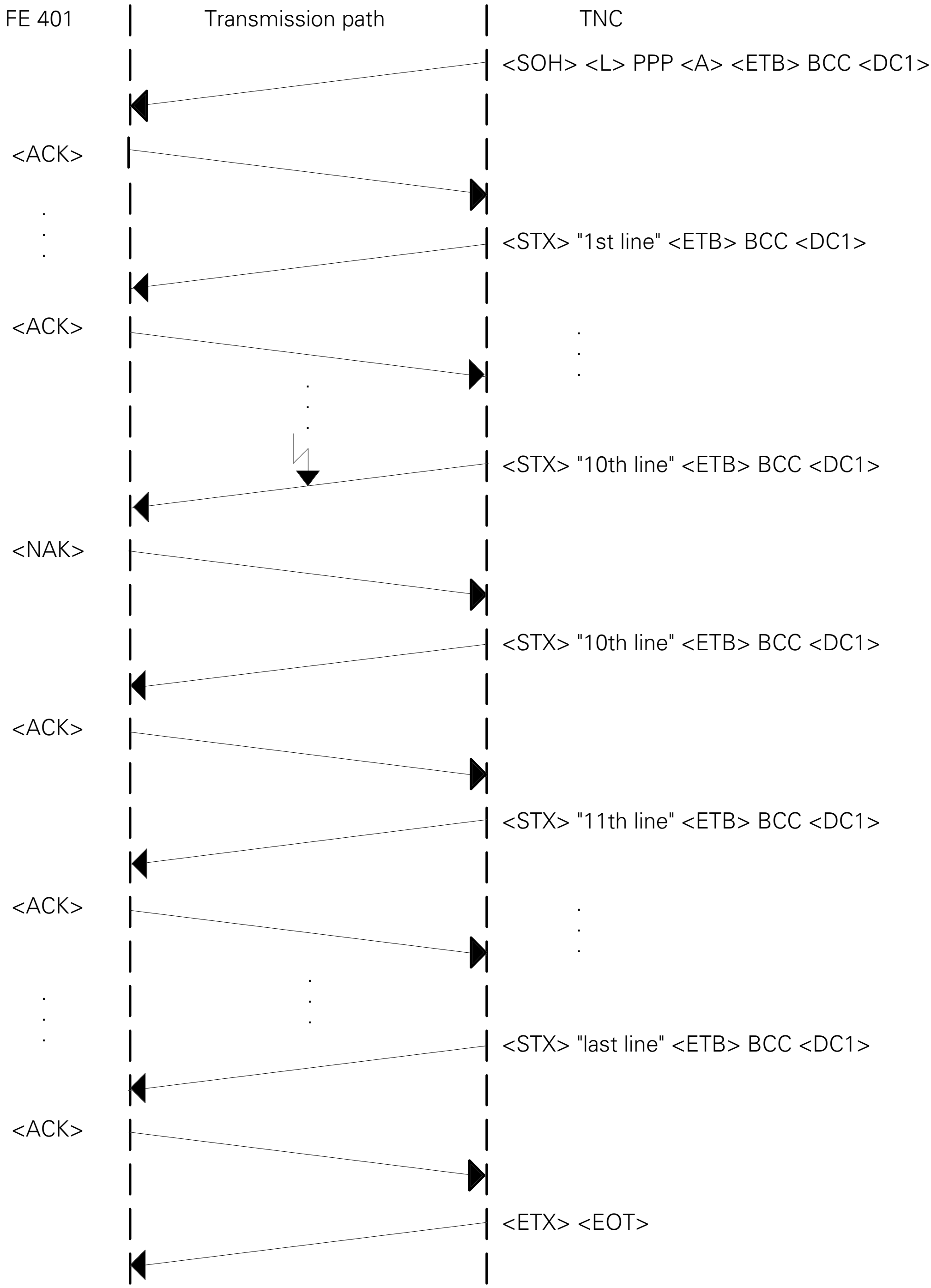
Once the last program block has transferred successfully (acknowledged by <AC>) the transmission is terminated by the characters <ETX> (end of text) and <EOT> (end of transmission).

Table of control characters:

Character	Name	Description
SOH	Start of Header	SOH identifies the beginning of the data transfer header. This is a character string that contains the program number and information about the type of program and the transfer mode.
STX	Start of Text	STX identifies the beginning of a program block.
ETB	End of Text Block	ETB terminates a data transfer block. The character that follows ETB (BCC) is used for data checking.
DC1	Start data transfer (XON)	DC1 starts the transfer of data.
DC3	Stop data transfer (XOFF)	DC3 stops the transfer of data.
ETX	End of Text	ETX is transmitted at the end of a program.
EOT	End of Transmission	EOT terminates the data transfer and establishes the idle state. The character is transmitted by the TNC at the end of a program input and to the external device in the event of an error.
ACK	Positive checkback (Acknowledge)	ACK is transmitted by the receiver when a data block has transferred without error.
NAK	Negative checkback (Not Acknowledge)	NAK is transmitted by the receiver when a data block has transferred with an error. The transmitter must re-transmit the block.

Example:

To upload a Pallet file with the name "PPP" to a peripheral device (e.g. FE 401).



The software handshake is very easy to accomplish when transmitting with BCC. The receiver transmits neither a positive (<ACK>) nor a negative (<NAK>) acknowledgement and the transmitter waits until it receives one of these characters.
When the receive buffer of the receiver unit is ready to receive again the unit transmits another <ACK> and the transmitter resumes the data transfer.

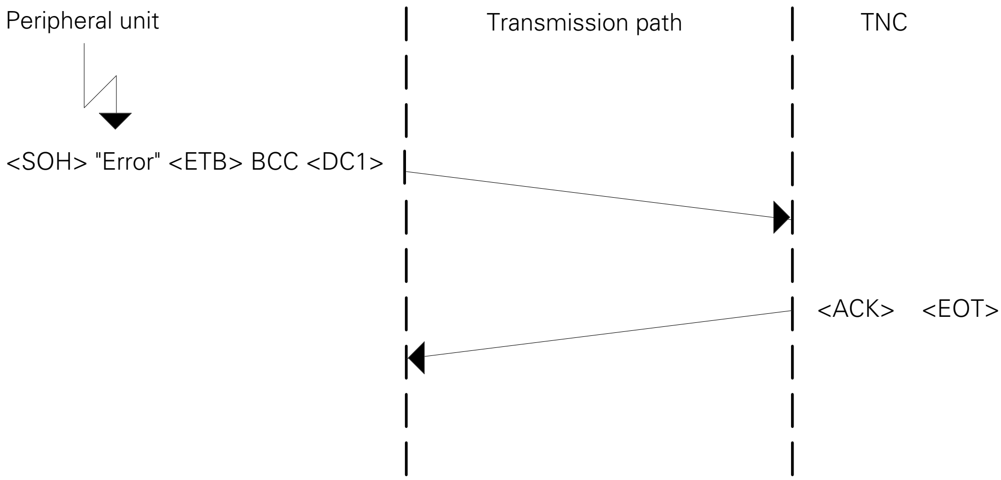
The software handshake can also be done with the aid of the control characters <DC1> and <DC3> however. If the hardware handshake (EXT1/EXT2/EXT3) was selected this is identical with the standard data transfer and BCC transfer (see section "Freely configurable interfaces").

3.2.2 Protocols

In the following, the transmission protocols are listed for the various file input and output possibilities. FE2 mode is set.

If an error occurs at a peripheral device, the following block must be sent to the TNC.

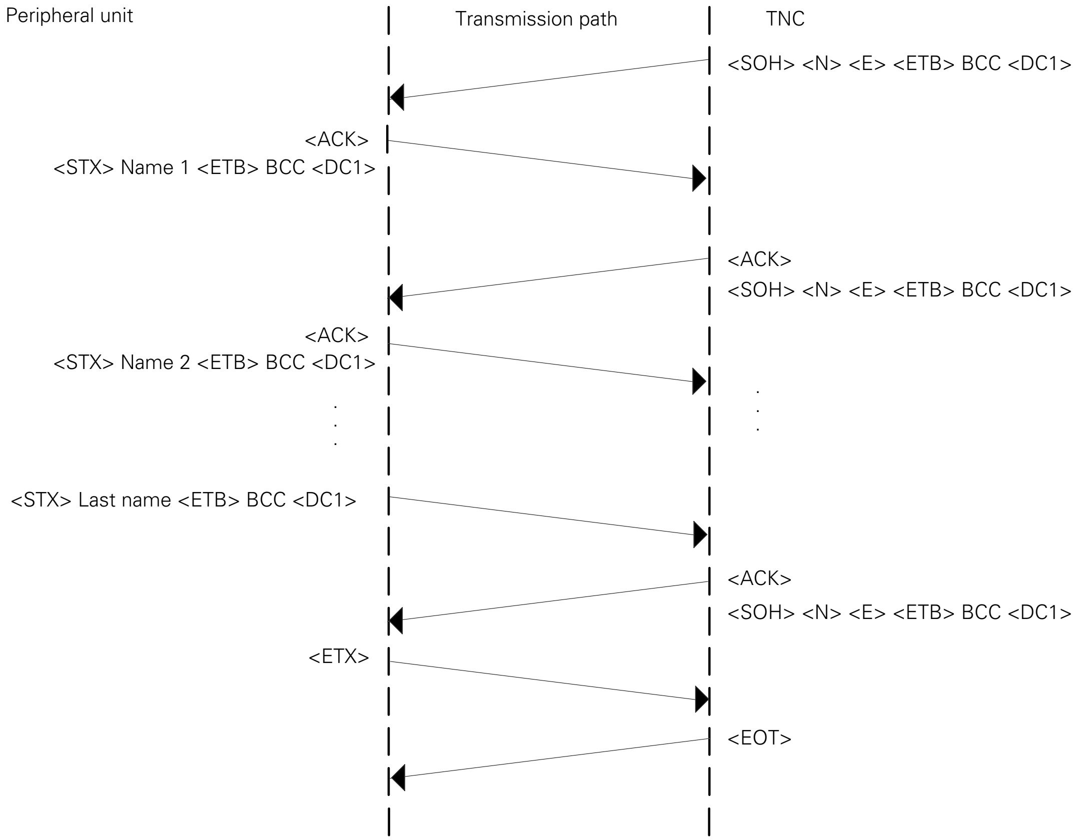
<SOH>"Error text"<ETB>BCC



The received error message is displayed in the TNC, but can be acknowledged and erased with the CE key.

Request external directory

If the soft key "Show ext. directory" is operated, the TNC outputs a header without a program name. The peripheral then starts to output the files to the TNC, but, after the first file block has been sent, the TNC immediately requests the next file.



The program names read in are displayed in the TNC.

In FE1 mode, the following 'Escape' sequence is sent to request the external directory:

<DC3><ESC><DC1><0><SP><D><CR><LF>

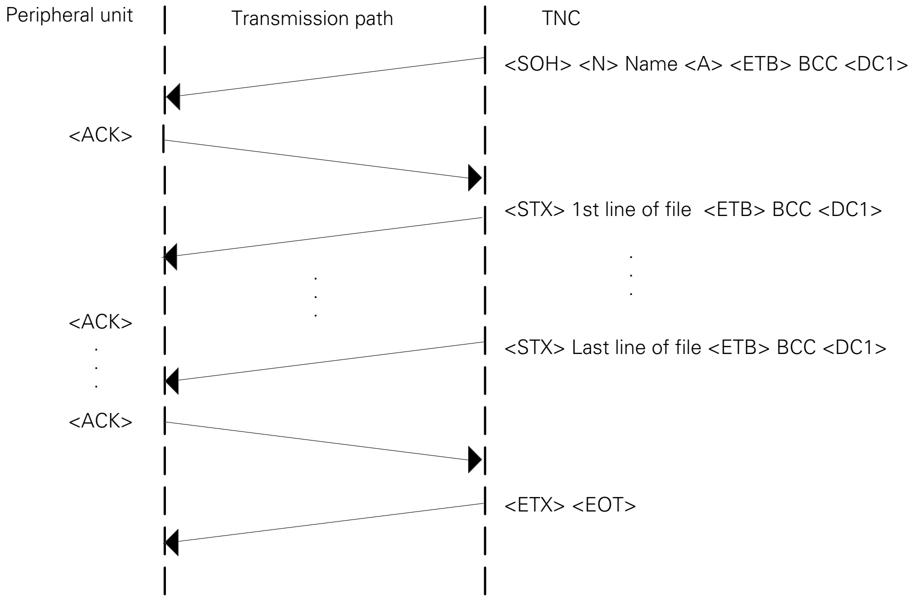
The TNC expects the following input to this request:
The first four lines, each ending in <CR><LF>, are ignored. In subsequent lines ending with <CR><LF>, only the program name and, after any number of blank characters, the number of occupied sectors are stored.

xxxxxx"Name"   "Sectors"xxxxxx<CR><LF>

If the character combination 'FREE:' is detected, only a number (= number of free sectors) will be read in.

Output selected file

The following protocol is followed:



The program name may contain up to 16 characters.

Output all files

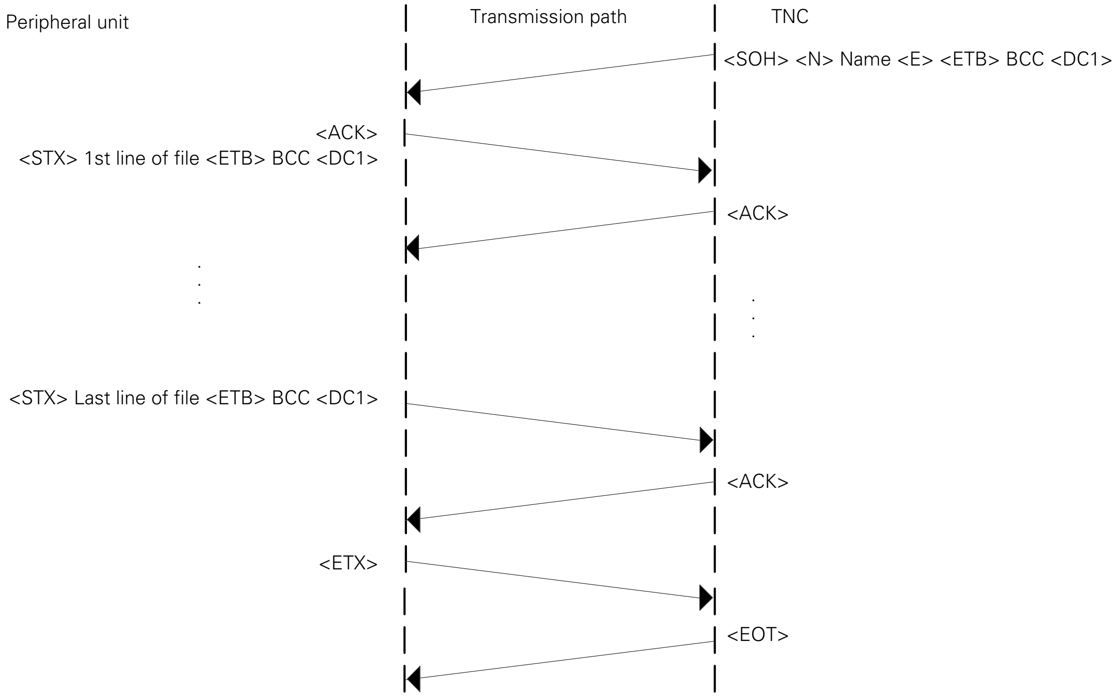
The files are output in order, as in section 4.1.2 "Outputting a selected file". Control characters <ETX><EOT> are sent to the peripheral device between files.

Output file with confirmation

In this mode, a question is asked before each file output in order to determine whether or not the file is to be output. After each file, control characters <ETX> and <EOT> are sent, as in section 4.1.3 "Outputting all files". If output of a file is not desired, the TNC immediately offers the next file in the directory for output.

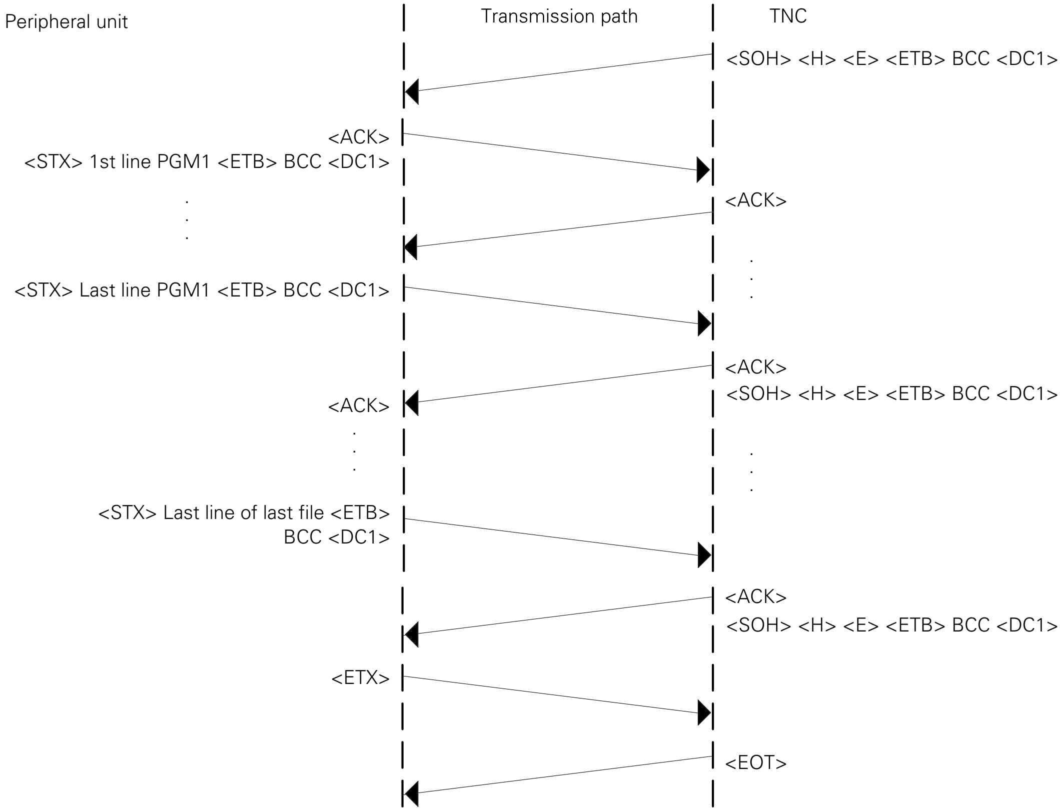
Read-in selected file

If a file is to be read in from an external memory, the TNC sends a header with the file name concerned, whereupon the peripheral sends the file.



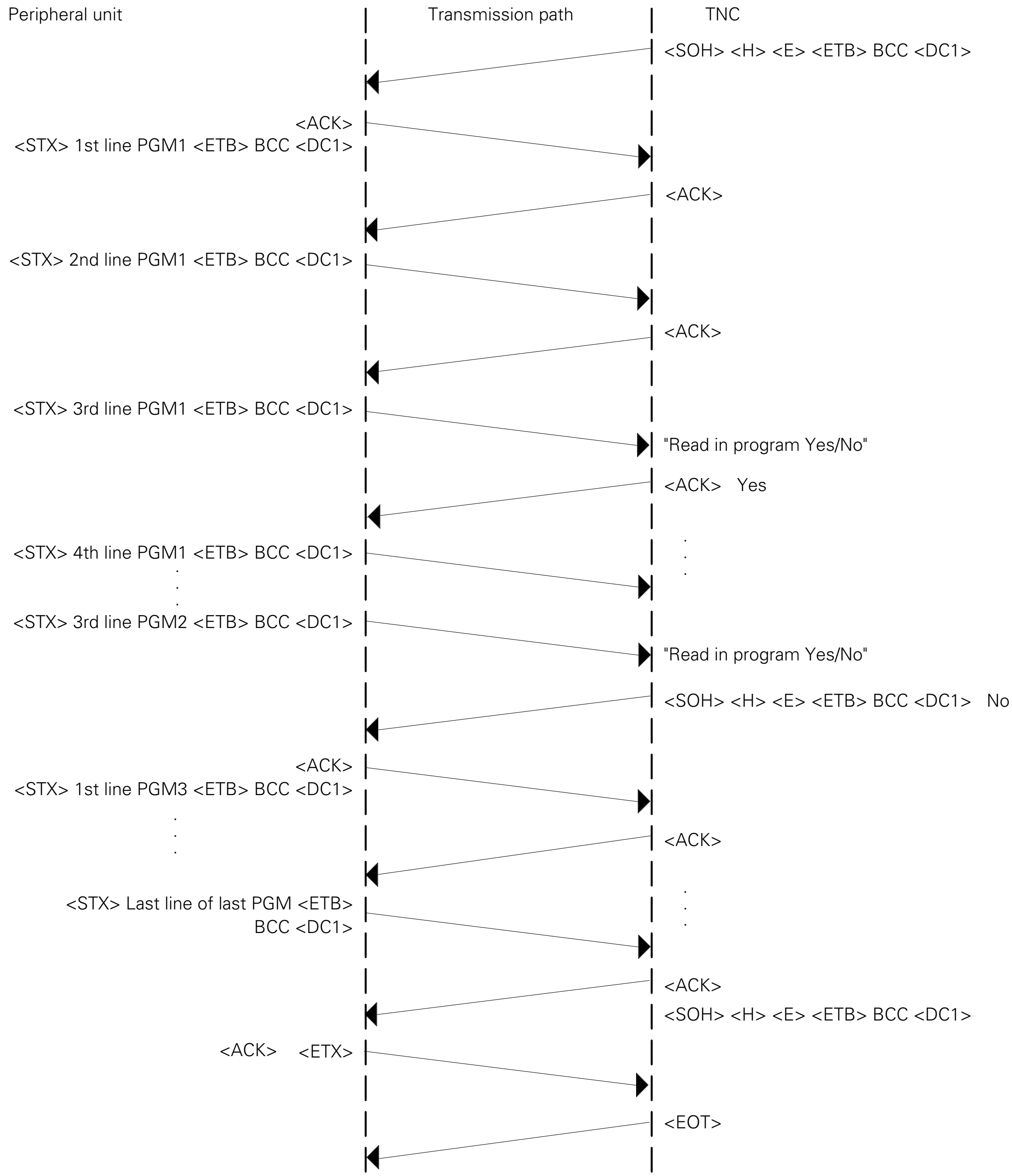
Read-in all files

In this case the TNC sends a header without a program name, and the peripheral unit sends the first file. The TNC again outputs a header without a program name, the next program is sent and so on.



Read-in file with acknowledgement

In this mode, the TNC first of all sends a header without a program name. The peripheral unit then starts data transfer, until it is stopped by the TNC. The TNC interrupts transfer by not sending a positive acknowledgement (no <ACK>) and awaits acknowledgement.
If a positive acknowledgement is given then the program is read in. Otherwise a header is immediately sent again.



The number of blocks transmitted before the TNC stops to await acknowledgement will depend on the transfer rate set. At a low data transfer rate, the TNC will stop after the first block.

3.3 LSV/2 Protocol

The LSV/2 protocol is a data transfer protocol for the two-way transfer of commands and data according to DIN 66019.
The commands and data are transferred in so-called telegrams, i.e. the data is split up into blocks (telegrams) and transmitted.
The following functions are possible:

- Data transfer
- File management (delete, copy and rename files)
- Write screen to a file (screen dump)
- Remote operation of the control functions, i.e. the control screen appears on the screen of the computer and all TNC functions can be executed from the external computer.
- Real DNC operation, i.e. starting and stopping the machine from the PC.
- Diagnosis of TNC error messages and key operations for service purposes (the last 100 events are stored in the TNC).

The LSV/2 protocol always runs in the background (independently of the interface set-up; standard baud rate 9600) and is started externally by a PC. A higher baud rate can be set in the interface set-up at the control.

HEIDENHAIN offer two software packages:

- TNC REMOTE: Software for TNC remote control. Can be run on an AT compatible PC with MS-DOS. All the above functions are available with this software.

- LSV/2 TOOL BOX: Software tools in C programming language for creating the transfer telegrams (library, executable files for telegrams, source codes, INCLUDE files for LSV/2, MAKE files).

4 Data transfer by PLC

Using PLC modules (for description see the section on "Data Transfer by PLC" in the "PLC Programming" Manual), data can be transferred by the PLC via the RS-232-C/V.24 or RS-422/V.11 data interfaces. These modules, for example, permit communication between two logic units at PLC level via the interface. One application of this type of data transfer is the positioning module, which is described in detail in the "Positioning Module" Manual.

4.1 Configuration of PLC data interface

4.1.1 General

When data is transmitted by the PLC, use of the interface is inhibited by the input/output program of the user interface. The user has the possibility of configuring the PLC interface according to the FE1, FE2 or ME mode already described, from the RS-232-/RS-422 setup, or to configure it freely. However, the following limitation applies:
A configuration at 19,200 baud is impossible if the other interface is already configured at 38,400 baud, regardless of whether this interface is assigned to the PLC or NC.

4.1.2 Free configuration

As with the interfaces EXT1 and EXT2 of the NC, the EXT3 PLC interface is also freely configurable. The appropriate control characters and the data format for EXT3 are set in the machine parameters MP5xxx.2, which are described in detail in section 2.6.2 "Freely configurable interfaces". In addition, the baud rate for transfer can be selected via MP5040.

MP5040	Data transfer rate in operating mode EXT3 (data transfer via PLC)	
	Entry : 0 to 9	
	0 = 110 baud	5 = 2400 baud
	1 = 150 baud	6 = 4800 baud
	2 = 300 baud	7 = 9600 baud
	3 = 600 baud	8 = 19 200 baud
	4 = 1200 baud	9 = 38 400 baud

5 Error messages

5.1 TNC error messages

Listed below are the error messages for data transfer, which are displayed at the TNC. In most cases the messages are self-explanatory.

General error messages:

BAUD RATE NOT POSSIBLE	One interface is configured for 19,200 baud and the other for 38,400 baud.
INTERFACE ALREADY ASSIGNED	Transfer is already taking place via interface, or data transfer has not been completed.
PROGRAM INCOMPLETE	A transfer has been interrupted or the file has not ended correctly (no END character or END block).
EXT. OUTPUT/INPUT NOT READY	Interface is not connected; peripheral unit is switched off of faulty.
TRANSFERRED DATA INCORRECT X	X can assume the values A to H, K or L (error codes).

Error codes:

E	During data transfer with Block Check Character (BCC), a (<NAK>) has been received three times in sequence.
A to H except E	The receiving component has detected an error with one of the following causes: – Same baud rate not set at TNC and peripheral unit – Parity bit wrong – Incorrect data frame (e.g. no stop bit) – Receiving component of interface faulty
K	During the transmission of an error to the TNC the <1> character was not transmitted after the <ESC> character.
L	An incorrect error number was received after the <ESC><1> error sequence (error numbers 0 to 7 are permitted).
N	An expected <ACK> or <NAK> acknowledgement was not transmitted within a certain time.
M	For data transmission with BCC, <NAK> was transmitted three times in sequence.

Error codes K and L are displayed only for transfer with standard data transmission protocol.

Error messages occurring only in ME-mode:

WRONG OPERATING MODE
TRANSFERRED DATA INCORRECT
WRONG PROGRAM DATA
ME: TAPE END
DATA MEDIUM MISSING
DATA MEDIUM EMPTY
DATA MEDIUM WRITE-PROTECTED

5.2 HEIDENHAIN peripherals' error codes

These error messages refer to the FE401 floppy disk unit and to magnetic tape unit ME 101/ ME 102. With the floppy disk unit (FE 401) connected, one of the following error codes could be outputted by the TNC:

Error code	Meaning
ERR: 001	Wrong instruction code
ERR: 002	Illegal program name
ERR: 003	Defective data transmission
ERR: 004	Program incomplete
ERR: 010	Program not on floppy disk
ERR: 011	Program protected against erasure
ERR: 012	Program storage in progress
ERR: 013	Program directory full
ERR: 014	Floppy disk full
ERR: 100	Floppy disk not formatted
ERR: 101	Sector number too large
ERR: 102	Drive not ready
ERR: 103	Floppy disk write-protected
ERR: 104	Data on floppy disk defective
ERR: 105	Sectors not found
ERR: 106	Check sum defective
ERR: 107	Disk controller faulty
ERR: 108	DMA faulty

If a magnetic tape unit is connected, the following error codes could be sent to the TNC and an appropriate error message outputted:

Error code	Error message
<ESC><1><0>	TRANSFERRED DATA INCORRECT
<ESC><1><1>	DATA CARRIER MISSING
<ESC><1><2>	DATA CARRIER WRITE-PROTECTED
<ESC><1><3>	WRONG OPERATING MODE
<ESC><1><4>	WRONG PROGRAM DATA
<ESC><1><5>	DATA CARRIER EMPTY
<ESC><1><6>	PROGRAM INCOMPLETE
<ESC><1><7>	ME: END OF TAPE

A detailed description of these peripherals can be found in the appropriate operating manual.

5.3 Data transmission software error messages

If data is transferred using the HEIDENHAIN - TNC.EXE data transmission program then the following error messages might be displayed at the TNC:

TRANSFERRED DATA INCORRECT
Attempts to transmit block to control unit has failed four times.

SEARCH FEATURE NOT ALLOWED
Search feature not included in number of acceptable characters.

INSTRUCTION NOT ALLOWED
Request instruction issued by control unit is not allowed.

PROGRAM NOT PRESENT
File requested by control unit does not exist in currently configured access path.

FILE NAME NOT PROGRAM NAME
Name of NC program and name of file do not match.

PROGRAM INCOMPLETE
NC program does not contain an end block.

PROTECTED FILE
File which is protected with read-only or hidden attribute is likely to be overwritten.

DATA CARRIER @: IS FULL
Data carrier >@:< full.

A precise description of this software is given in the Operating Manual of the transmission software.

OEM-cycles – Contents

1 Creating OEM-cycles

OEM-cycles (customized macros) are programmed in the **HEIDENHAIN dialog** as NC programs.

By using these cycles in machining programs created in the HEIDENHAIN dialog or in accordance with DIN/ISO, repetitive machining tasks or machine-specific functions can be executed via a single call. Execution of the OEM-cycles can be affected by parameter transfer.

Up to 32 different OEM-cycles, with a maximum of 128 different customer-specific dialog texts, can be produced, tested and stored in the NC program memory.

In order not to tie up the NC program memory (RAM) with the OEM-cycles and dialogs, it is possible to store this data in the PLC EPROM.

Instructions for creating OEM-cycles

OEM-cycles in the NC program memory can be called for testing by cycle 12: "Program call". It is thus also possible to test serviceability in the "Program run/single block" mode. (With a program call, all the Q parameters are globally effective!)

Permitted functions in OEM-cycles

– Tool definition with numerical values or Q parameters for tool length and radius (only in PLC EPROM).
– Tool call.
– M functions apart from M02, M30, M06, without program-run stop.
– Nesting OEM-cycles:
  Other OEM-cycles or standard cycles can be called in the OEM-cycles (nesting depth four levels).
– Call from main programs of OEM-cycles which are transmitted in "block mode" and are executed simultaneously. The OEM-cycles must be stored in the control unit memory.
– Re-approach to contour after an external STOP. With an external and internal STOP, re-approach must be via block scan (mid-program start-up) (see the "Machine Interfacing" Manual and the TNC 407/TNC 415 Operating Manual).

Functions not permitted in OEM-cycles

– M functions M02, M30, M06 with program-run stop.
– Programmed STOP block.
– Program calls with PGM-CALL.
– Definition of cycle 14 "Contour":
  Cycle 14 "Contour" must be defined in the main program.
– Repetition of sections of program with CALL LBL ... REP .../...:
  OEM-cycles with program-section repetitions stored in PLC EPROM cannot be executed. However, program-section repeats can also be programmed via the Q parameter function (IF ... GOTO LBL ...) (see the "Bolt hole circle" example).

1.1 Dialog block with DLG-DEF or DLG-CALL

Programming an OEM-cycle and, by the same token, the dialog block, is only possible if the program name is in the range 99999968.H to 99999999.H.
Each of these program names is permanently assigned a cycle number (e.g. program name 999999**68**.H represents OEM-cycle **68**).

Programming the dialog block is initiated by key "LBL SET" and then "NO ENT".

DLG-DEF
If the OEM-cycles are to be active in the NC program immediately after the definition, a "DEF-active" OEM-cycle is programmed with the "ENT" key, e.g. cycle for co-ordinate transformation.

DLG-CALL
If the OEM-cycles are to be activated later in the NC program via CYCL-CALL or M99, a "CALL-active" OEM-cycle is programmed with the "NO ENT" key, e.g. for a machining cycle.

Up to 15 dialog numbers can then be input. The first dialog number is always assigned to the designation of the OEM-cycles. The remaining numbers are assigned to the Q parameters in ascending order.
If fewer than 15 dialogs are to be programmed, the dialog block can end with "END". In this connection, see also MP7250 in the following description of the relevant machine parameters.

Example:

```
0  BEGIN PGM 99999968 MM
1  DLG-DEF 0/2/8/100
.
.
.
15  END PGM 99999968 MM
```

Using machine parameter MP7240, it is possible to inhibit program input for [Program name] = [OEM-cycles number].
If MP7240 has a value 0, no program with the program name of an OEM-cycle which is held in the EPROM can be input or read into the NC program memory.
If MP7240 has a value 1, the program-name range of the OEM-cycles can also be used when the OEM-cycles are held in the PLC EPROM. If a OEM-cycles is generated in the NC program memory and if at the same time there is a OEM-cycles with the same number in the PLC EPROM, then the OEM-cycles in the NC program memory will be executed at a cycle call.

In the NC program, when defining the OEM-cycles created with dialog support, Q parameters are assigned specified input values. The Q-parameter numbers are automatically generated by the TNC. In order to prevent the same Q-parameter numbers being generated for "DLG-DEF" cycles and "DLG-CALL" cycles, the difference between Q-parameter numbers can be specified for "DLG-CALL" and "DLG-DEF" blocks.
In the case of a "DLG-CALL" block, the input values of the OEM-cycles are assigned in ascending order of magnitude to the Q parameters Q1 to Q14. For the "DLG-DEF" block, the input values are assigned to Q parameters Q[1 + MP7250] to Q[14 + MP7250].

Example for MP7250 = 30

	Parameter number in OEM-cycle with	
	DLG-CALL	DLG-DEF
Cycle parameter 1	Q1	Q31
Cycle parameter 2	Q2	Q32
.	.	.
.	.	.
.	.	.
Cycle parameter 14	Q14	Q44

MP7251 determines whether the values of the Q parameters which are changed in the OEM-cycles by calculation or assignment are globally transferred to the called program (e.g. in the case of "nesting" of OEM-cycles).
With machine parameter MP7251, the range of Q parameters from Q[100 - MP7251] to Q99 is defined as "global".

The operation of global and local Q parameters may be shown by reference to following example:
MP7251 = 40
Q [100 – 40] = Q60 => Q60 to Q99 are global Q parameters and
Q1 to Q59 are local Q parameters

	MP7251 = 40		**MP7251 < 40**	
	Q1	Q60=global	Q1	Q60=local
BEGIN PGM 100 MM				
FN0: Q1 = +1	+1	+0	+1	+0
FN0: Q60 = +5	+1	+5	+1	+5
CYCL DEF 69.0 OEMCYCLE 1	+1	+5	+1	+5
CYCL DEF 69.1 Q1 = +2	+2	+5	+2	+5
BEGIN PGM 99999969 MM				
DLG–DEF 0/32				
FN1: Q1 = Q1 + 10	+12	+5	+12	+5
FN1: Q60 = Q60 + 10	+12	**+15**	+12	**+15**
END PGM 99999969 MM				
STOP	+2	**+15**	+2	**+5**
END PGM 100 MM				

MP7240	Inhibit program input for [Program name] = [OEM-cycle number in EPROM]. Input value 0 or 1 0 = inhibit 1 = not inhibit
MP7250	Difference between Q-parameter number for "DLG-CALL" and "DLG-DEF" block in OEM-cycles. Input range: 0 to 50
MP7251	Number of global Q parameters transferred from OEM-cycle to calling program. Input range: 0 to 100

OEM-cycles stored in the PLC EPROM can be inhibited in the PLC program via flags M2240 to M2271. Inhibited cycles cannot be defined in NC programs.
If programs with definitions of inhibited OEM-cycles are transferred to the control unit, an error message "ERROR=..." is generated in the transferred program and the program cannot be executed.

Flag	Function	Set	Reset
M2240	Inhibit OEM-cycle 68	PLC	PLC
M2241	Inhibit OEM-cycle 69	PLC	PLC
M2242	Inhibit OEM-cycle 70	PLC	PLC
.	.		
.	.		
.	.		
M2271	Inhibit OEM-cycle 99	PLC	PLC

1.2 Q parameters and functions in OEM-cycles

Q-parameter programming can be used to create OEM-cycles with variable program data, with which, for example, extremely varied drilling patterns, curves (e.g. spirals, sinusoidal, ellipse, parabola) and shaped components can be milled.

A detailed description of Q parameters and functions is given in the TNC 407/TNC 415 Operating Manual.

Q parameters with special significance

During a program run, data is transferred to the following Q parameters. When creating OEM-cycles, these can be read and also overwritten:

Q108 Tool radius of tool last called
Q109 Current tool axis
Q110 M function last output for spindle rotation direction
Q111 Coolant on/off
Q112 Overlapping factor for pocket-milling (from MP7430)
Q113 Main program contains measurements in millimetres or inches

FN-Functions with special significance

The following FN functions can be used to execute various tasks, such as error messages and data transfer NC -> PLC in the OEM-cycles.

FN14 Output of error messages and dialogs to VDU
FN15 Output of error messages, dialogs and Q-parameter values to a file or via a data interface (V.24/RS-232-C or V.11/RS-422).

FN19 Assignment of two numerical values or Q-parameter values from an OEM-cycle to PLC. (Function FN19 is described in detail in the "PLC Programming" Manual).

2 Dialogs for OEM-cycles

The dialog numbers defined in the dialog blocks of the OEM-cycles determine the text to be displayed from the dialog file concerned. Which dialog file is active will depend on the dialog language selected (MP7230) and whether dialogs are read from the EPROM or are stored in the NC program memory (see the section on "Functions for file management" in the "PLC Programming" Manual).

The dialog numbers (input: 0 to 127) of the dialog block are assigned to the first 128 lines of a dialog file.

Example:

DLG-DEF 0/1/.../127

Dialog number in OEM-cycle	Line number of dialog file	Dialog example
0	1	BOLT HOLE CIRCLE
1	2	NUMBER OF HOLES
.	.	.
.	.	.
.	.	.
127	128	PITCH

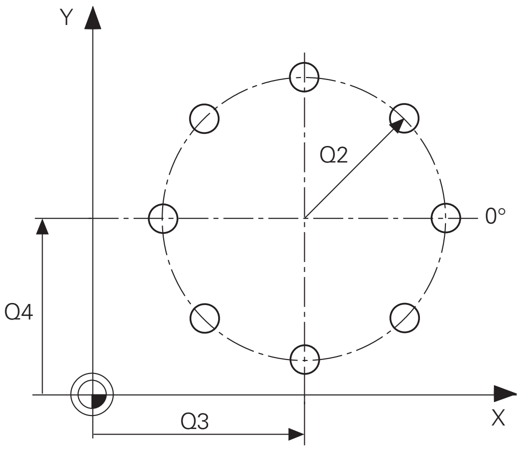
Up to 20 characters can be displayed on a dialog line. In all, a maximum of 34 characters may be input, being reproduced in full in the NC program, but only in abbreviated form on the dialog line.

3 Output in binary code

If the OEM-cycles have been fully tested and the dialogs have been specified, the two can be output in binary form for EPROM programming, together with the PLC program and the error messages. It is possible to output the files located in both the PLC EPROM and the NC program memory in binary code.
An accurate description of file output via the data interface is given in the "PLC Programming" Manual.

4 "Bolt hole circle" OEM-cycle example

The following "Bolt hole circle" program is an example of an OEM-cycle. (This cycle has not been loaded in the control unit!)
The Z-axis acts as the tool axis. The first hole in the circle is at 0°. The OEM-cycle calculates the angular position of the holes from the number of holes. The drilling positions are approached in succession in an anti-clockwise direction, and the holes are made automatically with an in-feed. Before the cycle is called, the tool is held at the safety clearance.



Input parameters:
Q1 = Number of holes
Q2 = Radius of bolt hole circle
Q3 = X co-ordinate of centre of bolt hole circle
Q4 = Y co-ordinate of centre of bolt hole circle
Q5 = Safety clearance for Z axis (negative input)
Q6 = Hole depth in Z axis (negative input)
Q7 = Drilling feed

"Bolt hole circle" OEM-cycle

0 BEGIN PGM 99999968 MM	
1 DLG-CALL 0/1/2/3/4/5/6/7	Dialog block
2 FN1: Q6 = +Q6 + +Q5	Distance traversed in Z
3 FN4: Q50 = +360 DIV + Q1	Angle increment
4 FN0: Q60 = +0	Start angle
5 CC X+Q3 Y+Q4	Centre of bolt hole circle
6 LBL 11	Jump label
7 LP PR +Q2 PA +Q60 R0 FMAX	Approach drilling position
8 L IZ +Q6 FQ7	Drilling with feed
9 L IZ –Q6 FMAX	Clear
10 FN 1: Q60 = +Q60 + +Q50	Next angle
11 FN12: IF +Q60 LT +360 GOTO LBL 11	Last bore?
12 END PGM 99999968 MM	

Dialogs for "Bolt hole circle" OEM-cycle

Dialog No.	DIALOG
0	BOLT HOLE CIRCLE
1	NUMBER OF HOLES?
2	RADIUS?
3	X COORDINATE CC?
4	Y COORDINATE CC?
5	SAFETY CLEARANCE?
6	TOTAL HOLE DEPTH?
7	IN-FEED DEPTH?

5 OEM-cycles in NC programs

OEM-cycles in the NC program memory or PLC EPROM can be defined, called and executed both in HEIDENHAIN dialog programs and also in DIN/ISO programs.

5.1 Calls in a HEIDENHAIN dialog program

In the HEIDENHAIN dialog program, OEM-cycles are defined as standard cycles (see "Dialog Programming" in the TNC 407/TNC 415 Operating Manual).

The dialog for cycle definition is initiated with the "CYCL DEF" key. The desired cycle is selected either by skimming through the pages using the vertical arrow keys or by "GOTO" and input of the cycle number (e.g. 68). The cycle is entered with the "ENT" key.

The individual parameters are input via the digital keyboard and entered with "ENT".

In the case of a "DEF-active" OEM-cycle, the cycle is effective immediately after definition. Once defined, a "CALL-active" OEM-cycle can be called and hence activated either via "CYCL CALL" or M99.

Example:

0	BEGIN PGM 1000 MM	
1	BLK FORM 0.1 Z X+0 Y+0 Z–20	Definition of blank
2	BLK FORM 0.2 X+100 Y+100 Z+0	For test/program-run graphics
3	TOOL DEF 1 L+0 R+2	Tool definition
4	TOOL CALL 1 Z S1000	Tool call
5	L Z+2 R0 FMAX M3	Approach safety clearance
6	CYCL DEF 68.0 Bolt hole circle	Definition of cycle 68 "Bolt hole circle"
7	CYCL DEF 68.1 Q1=+8 Q2=+40 Q3=+60	
8	CYCL DEF 68.2 Q4=+50 Q5=–2 Q6=–20	
9	CYCL DEF 68.3 Q7=+100	
10	CYCL CALL	Call cycle
11	END PGM 1000 MM	

5.2 Calls in a DIN/ISO program

In a DIN/ISO program, OEM-cycles are not defined via a G- function but via key "D".

The desired OEM-cycle is entered by inputting its number (e.g. 68) and "ENT". The individual parameters are input via the digital keyboard and entered with "ENT". The definition of the OEM-cycles ends with "END".

In the case of a "DEF-active" OEM-cycle, the cycle is effective immediately after definition. Once defined, a "CALL-active" OEM-cycle can be called and hence activated either via G79 or M99.

Example:

```
% 1000 G71*
N10 G30 G17 X+0 Y+0 Z–20*               Definition of blank
N20 G31 G90 X+100 Y+100 Z+0*            For test/program-run graphics
N30 G99 T1 L+0 R+2*                     Tool definition
N40 T1 G17 S1000*                       Tool call
N50 G00 G40 G90 Z+2 M3*                 Safety clearance
N60 D68 P1+8 P2+40 P3+60                Definition of cycle 68 "Bolt hole circle"
P4+50 P5–2  P6–20 P7+100*
N70 G79*                                Call cycle
N99999 % 1000 G71*
```

Positioning Module - Contents

1 PLC positioning module

1.1 Introduction

One hardware version of the LE 360 - the LE 234.003 - can be used in conjunction with the TNC407/TNC415 as a positioning module.

This means that the TNC 407/TNC 415 can be extended by up to 4 secondary axes. PLC inputs and outputs of the LE 234.003 can also be used.

The positioning module can be used for toolchangers, for pallet infeed and for controlling rotary axes and swivel heads.

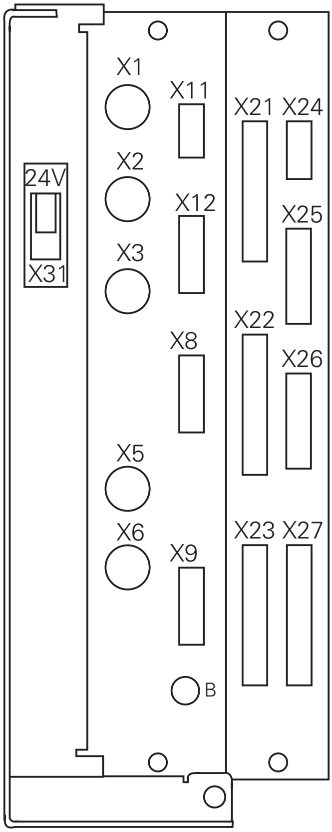
The NC software of the TNC 360 can also run on the LE 234.003, i.e. all functions of the TNC 360 are also available with the positioning module.

Detailed technical particulars are given in "Technical Manual TNC 360"

1.2 Hardware

Differences in hardware between LE 234.003 and LE 360 C:

- the X25 connector contains the RS-232-C/V.24 and the RS-422/V.11
- there is no X4 measuring system input (sinus)
- X5 measuring system input (square) is additional



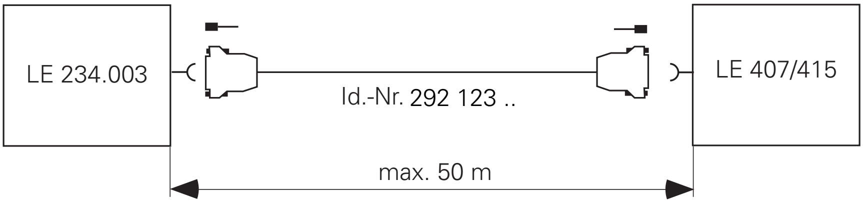
Control loop board
X1 = Measuring system 1 (~)
X2 = Measuring system 2 (~)
X3 = Measuring system 3 (~)
X5 = Measuring system 5 ( )
X6 = Measuring system S ( )
X8 = Nominal value output 1,2,3,4,S
X9 = Screen unit (for commissioning only)

B = Operational ground

X21 = PLC output
X22 = PLC input
X23 = TNC keyboard (TE)
(for commissioning only)
X24 = 24 V supply for PLC
X25 = RS-422/V.11 data interface (V.24/RS-232)
X31 = 24 V supply for LE

X11, X12 and X26 are not required.
X4 is not fitted.

Connecting cable



Pin layout

X25 data interface RS-422/V.11 and RS-232-C/V.24

Flange socket with female insert (25-pin)

Pin No.	Assignment RS-422/V.11	Assignment RS-232-C/V.24
1		Screen
2		RxD
3		TxD
4		CTS
5		RTS
6		DTR
7		GND Signal
8	spare	
9	$\overline{\text{RxD}}$	
10	$\overline{\text{CTS}}$	
11	$\overline{\text{TxD}}$	
12	$\overline{\text{RTS}}$	
13	$\overline{\text{DSR}}$	
14	$\overline{\text{DTR}}$	
15	DTR	
16, 17, 18	spare	
19	GND Signal	
20	DSR	DSR
21	RxD	
22	CTS	
23	TxD	
24	RTS	
25 Chassis	Outer screen GND Chassis	

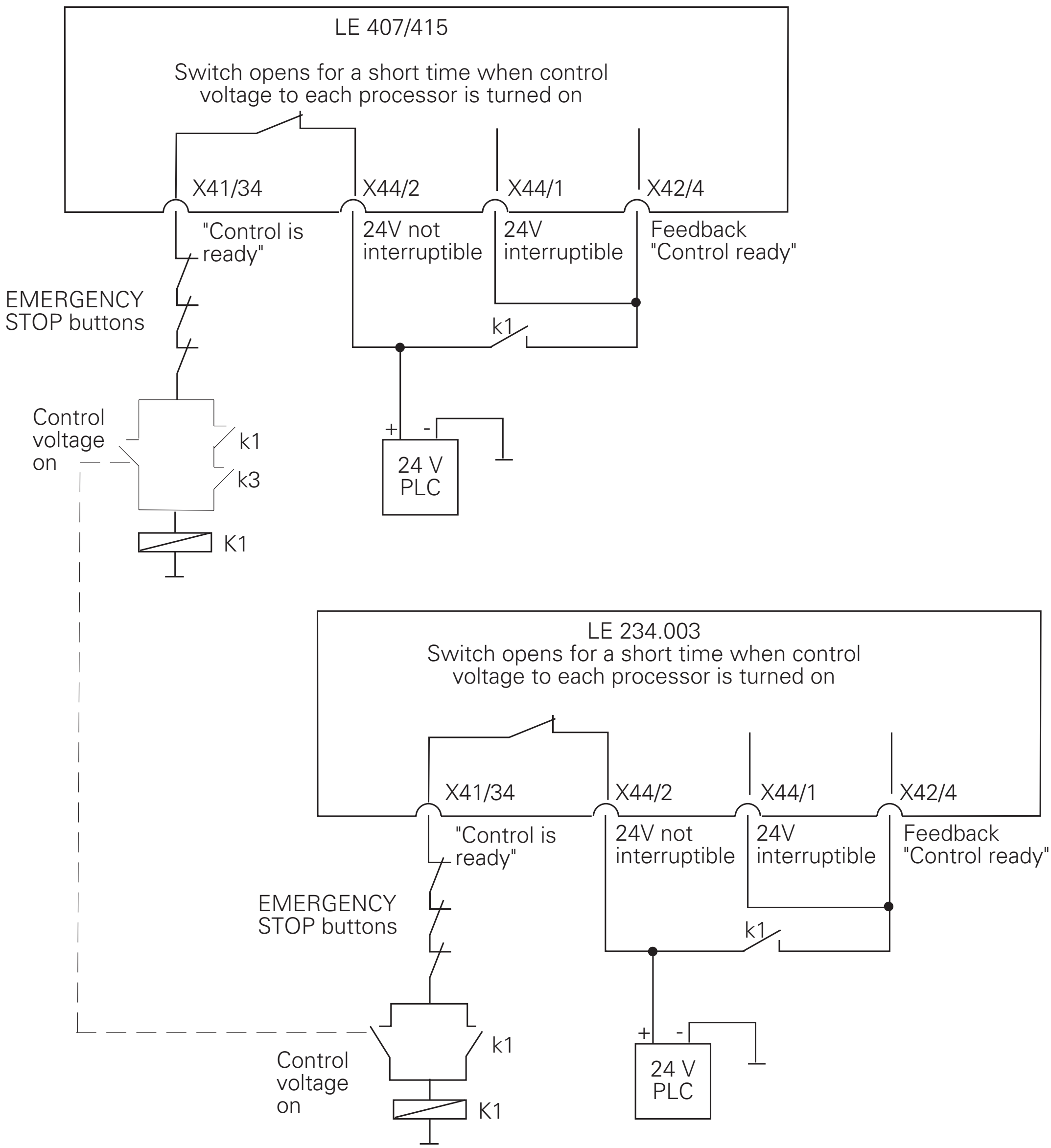
X5 Measuring system Input 5 (⊓⊔)

Pin No.	Assignment
5	U_{a1}
6	$\overline{U_{a1}}$
8	U_{a2}
1	$\overline{U_{a2}}$
3	U_{a0}
4	$\overline{U_{a0}}$
7	$\overline{U_{aS}}$
2	+ 5 V (U_P)
12	+ 5 V (U_P)
11	0 V (U_N)
10	0 V (U_N)
9 (spring)	Screen = Chassis

1.3 EMERGENCY STOP routine

The internal EMERGENCY STOP shutdown of the TNCs is tested when the supply voltage is switched on, i.e. the supply voltage to each processor is turned off for a short time.

Suitable protective circuitry must be used to ensure that the positioning module is not ready for operation if the NC test reveals an error, i.e. the voltage for the feedback is mutually interrupted. This can be achieved by the circuit illustrated. The test starts when the control voltage is switched on.



1.4 Reference signal evaluation

After reference signal evaluation of the NC axes, reference signal evaluation of the PLC axes of the positioning module must be initiated either automatically (e.g. by NC Start) or manually by special keys. PLC positioning is not possible until the reference signal is evaluated.

1.5 Installing the positioning module

The positioning module is connected to the hardware across the RS-422/V.11 interface by means of a special cable (Id.-Nr. 265 479).

Software installation is done with the PLC program of the master controller (TNC 407 or TNC 415). Modules 9100 and 9107 are used for this: they activate the RS-422/V.11 data interface and make it possible to transmit and receive binary data respectively.

These modules are described in the chapter "PLC Programming".

The PLC program for the positioning module can be written either on the LE 234.003 itself, in which case it must be connected up to screen BE 212 and control panel TE 355 A/B, or on a PC using the HEIDENHAIN PLC programming software.

Please contact HEIDENHAIN should you have any queries about the PLC programming software.

TNC 425 - Contents

1 Digital speed control

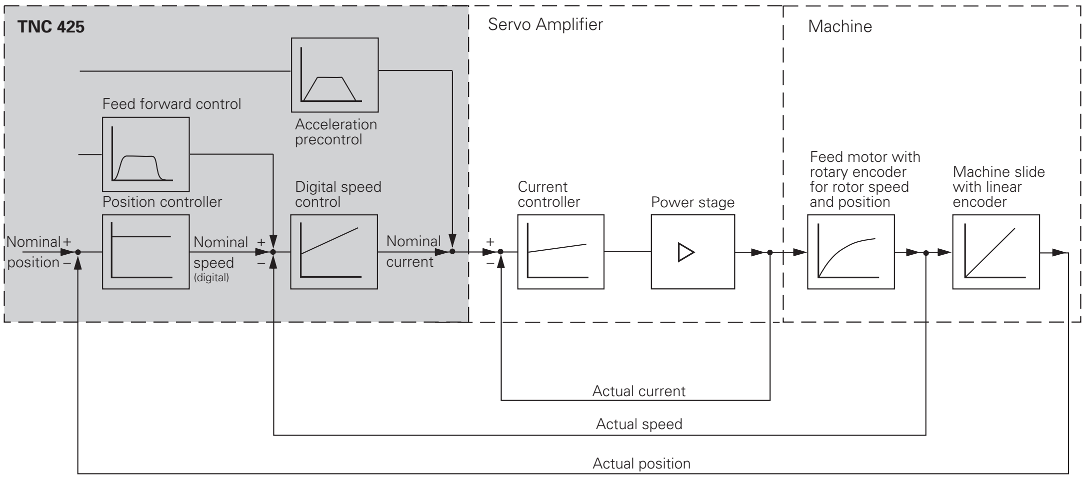
A digital speed controller has been integrated into the TNC 425.

The benefits of this concept are:

- High dynamic response
- High Kv factor
- Trailing errors are virtually "nil"
- No drift
- High geometrical accuracy with high traverse speeds
- Traverse speeds up to 60 m/min

Cycle times: Contour interpolation 3 ms
Fine interpolation 0.6 ms

Block circuit diagram of TNC 425 with integral speed controller



2 Mounting and electrical installation

The connection conditions are the same as for the TNC 415 B´ – the differences between TNC 425 and TNC 415 B are explained in the following sections.

2.1 Hardware

The TNC 425 consists of the following hardware components:

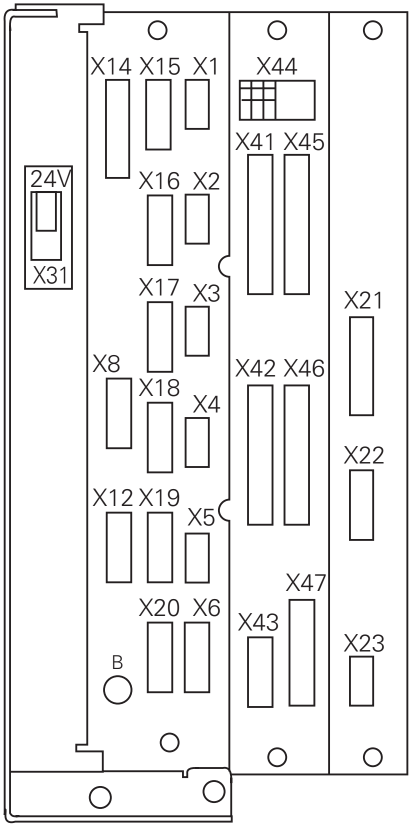
- LE 425 (logic unit)
- TE 400 (TNC control panel)
- BC 110 B (VDU)
- PL 410 (max. 2 PLC I/O units as option)

The following Identification Numbers have been allocated for the LE 425 so far:

ID Number	Logic Unit	Modification
267 214 19	LE 425	
267 214 24	LE 425	with "Digitise TS 120" module
267 214 28	LE 425 E	
267 214 38	LE 425 E	Export
267 214 39	LE 425	Standard
267 214 44	LE 425	with "Digitise TS 120" module
267 214 47	LE 425	with "Digitise TM 110" module
267 214 48	LE 425 E	Export
267 214 49	LE 425	Standard
267 214 54	LE 425	with "Digitise TS 120" module
267 214 57	LE 425	with "Digitise TM 110" module
267 214 58	LE 425 E	Export
267 214 59	LE 425	Standard

Identification number for software module "Digitizing with TS 120": 246 051 01

2.2 Summary of connections



Control loop board

X1 = Encoder 1 (~)
X2 = Encoder 2 (~)
X3 = Encoder 3 (~)
X4 = Encoder 4 (~)
X5 = Encoder 5 (~)
X6 = Encoder S ( )
X8 = Nominal value output 1,2,3,4,5,S
X12 = Triggering touch probe system
X14 = Measuring touch probe system
X15 = Encoder/speed
X16 = Encoder/speed
X17 = Encoder/speed
X18 = Encoder/speed
X19 = Encoder/speed
X20 = Spare
B = Operational ground

PLC and Graphics Board

X41 = PLC outputs
X42 = PLC inputs
X43 = Visual display unit (BC 110)
X44 = 24 V supply for PLC
X45 = TNC keyboard (TE)
X46 = Machine control panel
X47 = PLC I/O unit (PL)

Processor Board

X21 = RS-232-C/V.24 data interface
X22 = RS-422/V.11 data interface
X23 = Handwheel
X31 = 24 V DC supply for NC

2.2.1 Pin assignment

Logic unit LE 425

X1, X2, X3, X4, X5
Encoder 1, 2, 3, 4, 5,
(sine wave signal input)

Sub-D connector
(9-pin female insert)

Pin No.	Assignment	Colour
1	I_1 –	Yellow
2	0 V	White
3	I_2 –	Red
4	Inner screen	White/Brown
5	I_0 –	Pink
6	I_1 +	Green
7	+ 5 V	Brown
8	I_2 +	Blue
9	I_0 +	Gray
Chassis	Outer screen	

X6 measuring system S
(square-wave signal input)

Sub-D connector
(15-pin female insert)

Pin No.	Assignment	Colour
1	U_{a1}	Brown
2	0 V (U_N)	White/Green
3	U_{a2}	Gray
4	+ 5 V (Up)	Brown/Green
7	$\overline{U_{a0}}$	Black
9	$\overline{U_{a1}}$	Green
10	0 V (U_N)	White
11	$\overline{U_{a2}}$	Pink
12	+ 5 V (Up)	Blue
13	$\overline{U_{aS}}$	Violet
14	U_{a0}	Red
5,6,7,8,15	not assigned	
Chassis	Outer screen = device chassis	

X15, X16, X17, X18, X19
Measuring system/Speed
(sinus signal input 1 Vpp)

Sub-D connector
(15-pin female insert)

Pin No.	Assignment	Colour
1	A+	Brown
2	0 V (U_N)	White/Green
3	B+	Gray
4	+ 5 V (U_P)	Brown/Green
7	R–	Black
9	A–	Green
10	0 V (U_N)	White
11	B–	Pink
12	+ 5 V (U_P)	Blue
13	not assigned	Violet
14	R+	Red
5,6,8,13,15	not assigned	
Chassis	Outer screen = device chassis	

2.3.1 Connection of linear encoders

The LE 425 can process up to five encoders with sinusoidal output signals (7 to 16 μA_{pp}).
Max. input frequency: 50 kHz

Cable adapter, complete	(Max. 60 m)
Coupling (female) 9-pin / D-sub connector (male) 9-pin	Id.-Nr. 267 269 ..

or

Connecting cable, with one connector	(Max. 60 m)
D-sub connector (male) 9-pin	Id.-Nr. 268 371 ..

2.3.2 Connection of rotary encoders for speed control

To measure ballscrew speed, the TNC requires HEIDENHAIN rotary encoders with 1 Vpp output signals and a 5 V power supply. The signal is interpolated by a factor of 256 in the control. When selecting the line count, remember that the input frequency must no exceed the limit of 200 kHz at the LE. The line count should be selected so that the number of grating periods for speed control is at least 5 times that for position control.

Talk with your drive supplier and HEIDENHAIN about possible mounting configurations.

For various types of motors HEIDENHAIN offers the ERN 281 modular rotary encoder with a special assembly kit that enables the machine tool builder to install the rotary encoder on the motor. The mounting procedure is quite simple: the motor housing is extended by the length of the spacer in the assembly kit, and the lid is then re-fitted.

A straightforward solution with a standard encoder from the ROD series can be applied to modify the first machine when using a drive type for which no assembly kit is yet available.

Max. input frequency: 200 kHz

Cable adapter, complete	(Max. 60 m)
Connector (female) 12-pin / D-sub connector (pin) 15-pin	Id.-Nr. 267 268 ..

or

Cable adapter, complete	(Max. 60 m)
Coupling (female) 12-pin / D-sub connector (male) 15-pin	Id.-Nr. 267 267 ..

or

Connecting cable with one connector	(Max. 60 m)
D-sub connector (pin) 15-pin	Id.-Nr. 268 372 ..

2.3.3 Connection of rotary encoder for spindle orientation

The encoder for oriented spindle stop is connected at the input X6 (square-wave signals).
Max. input frequency: 300 kHz

HEIDENHAIN recommends using the ROD 426.xxx8 (1024 lines).

Cable adapter, complete	(Max. 20 m)
Connector (female) 12-pin / D-sub connector (pin) 15-pin	Id.-Nr. 267 268 ..

or

Connecting cable with one connector	(Max. 20 m)
D-sub connector (pin) 15-pin	Id.-Nr. 268 372 ..

2.4 Analogue output

The TNC 425 from HEIDENHAIN permits installation of commercially standard amplifiers and motors with AC technology. The output with the analogue nominal value voltage of the TNC's rpm controller is connected with the torque input of the servo amplifier.

This drive concept eliminates the need for tachogenerators. The rpm actual value is measured by a HEIDENHAIN incremental rotary encoder.

3 Machine interfacing

3.1 Machine parameters for digital speed control

The type of speed control - analogue or digital - can be selected for each individual axis.

The position and speed controllers receive their feedback signals from separate encoders. A linear encoder (or angle encoder for rotary axes) is the standard device for measuring distances. Speed is controlled by HEIDENHAIN rotary encoders that are fitted to the motor.

The new machine parameters for speed control make it possible to specify and optimise the control loop. Irrespective of this the controller can still be operated as before in trailing mode or with feed pre-control. The TNC 425 also positions in the manual modes in trailing mode.

The term "counting step" is one which is used in connection with the new machine parameters. Counting steps are the counting pulses from the measuring system multiplied internally by a factor of 256.

The following additional machine parameters are available for digital speed control:

MP1900	Select axes with digital speed controller
MP1910.x	Monitor speed controller
MP1920.x	Integral component for speed controller
MP1940.x	Proportional component for speed controller
MP1950	Polarity for torque signal
MP1951	Select measuring system for position control
MP1955.x	Ratio of grating period LS to ROD
MP1960	Compensation for reversal spikes on circular interpolation at the quadrant transitions for digitally controlled axes
MP1970	Motion monitor for position and speed
MP1980	Delayed shutdown of speed controller in EMERGENCY STOP

The following machine parameters are inactive for axes with digital speed control.

MP1050	Analogue voltage for rapid traverse
MP1080	Integral factor
MP1140	Movement monitoring
MP1220	Automatic cyclical offset adjustment

MP1900 Select axes with digital speed controller

Entry: %XXXXX

Bit 0	X axis	0 = Without digital speed controller
Bit 1	Y axis	1 = With digital speed controller
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

MP1910 **Monitor speed controller (drive monitoring)**
Entry: 1 to 16 777 215 [counting steps]

MP1910.0 X axis
MP1910.1 Y axis
MP1910.2 Z axis
MP1910.3 4th axis
MP1910.4 5th axis

The entry is axis-specific and made in counting steps. MP1910.x monitors the theoretically calculated output voltage of the speed controller. If this voltage exceeds the limit specified in MP1910.x the error message "Gross positioning error 3F" appears. This voltage is always limited to ± 10 V at the output.

The input value for MP1910.x can therefore be calculated as follows:

$$\text{MP 1910.x} = \frac{\text{Ulimit}}{\text{MP1920.x} * 9.7\ \mu\text{V}}$$

The speed controller integrated in the control system is a PI controller. The P and I components of the speed controller can be set in machine parameters MP1920.x and MP1940.x.

MP1920 **Integral component for speed controller (I component)**
Entry: 0 to 65 535
Recommended input range: 50 to 100

MP1920.0 X axis
MP1920.1 Y axis
MP1920.2 Z axis
MP1920.3 4th axis
MP1920.4 5th axis

MP1925 **Limiting the integral component for speed controller**
Entry: 0.000 to 65.535 [s]
Recommended input range: 0.1 to 2 s

MP1925.0 X axis
MP1925.1 Y axis
MP1925.2 Z axis
MP1925.3 4th axis
MP1925.4 5th axis

MP1940 **Proportional component for speed controller (P component)**
Entry: 0 to 65 535
Recommended input range: 50 to 200

MP1940.0 X axis
MP1940.1 Y axis
MP1940.2 Z axis
MP1940.3 4th axis
MP1940.4 5th axis

MP1945 **Factor for acceleration pre-control of the rotational speed controller**

Entry: 0.000 to 9.999 [V/(m/s2)]

MP1945.0	X axis
MP1945.1	Y axis
MP1945.2	Z axis
MP1945.3	4th axis
MP1945.4	5th axis

MP1950 **Polarity for torque signal**

Entry: %XXXXX

Bit 0	X axis	0 = positive
Bit 1	Y axis	1 = negative
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

The polarity of the nominal voltage can be modified with this MP.

MP1040 is still active but for internal voltage between position and speed controller. MP210 is only active for position encoders. The counting direction for the rotary encoder is compensated with MP1950.

MP1951 **Select measuring system for position control**

Entry: %XXXXX

Bit 0	X axis	0 = linear encoder for position control
Bit 1	Y axis	1 = motor rotary encoder for position control (for trimming speed controller only)
Bit 2	Z axis	
Bit 3	4th axis	
Bit 4	5th axis	

MP1955 **Ratio of grating period LS to ROD**

Entry: 0.1 to 100

MP1955.0	X axis
MP1955.1	Y axis
MP1955.2	Z axis
MP1955.3	4th axis
MP1955.4	5th axis

The ratio of grating periods of the linear encoder to the motor rotary encoder should be greater than 5 (see MP1970).

Example: 2000 lines per revolution for the rotary encoder; 4 mm screw pitch
2 µm grating period with ROD; 20 µm grating period with LS

Grating period of rotary encoder: $\frac{4000\ [\mu m]}{2000\ [lines]} = 2\ [\mu m]$

Entry value for MP1955: $\frac{20\ [\mu m]}{2\ [\mu m]} = 10$

MP1960 **Compensation for reversal spikes on circular interpolation at the quadrant transitions for digitally controlled axes**
Entry: –1.0000 to +1.0000[mm]

MP1960.0 X axis
MP1960.1 Y axis
MP1960.2 Z axis
MP1960.3 4th axis
MP1960.4 5th axis

For digitally controlled axes reversal spikes of the order of the entry range of MP1960.x are compensated during circular interpolation at the quadrant transitions.

MP1970 **Motion monitor for position and speed**
Entry: 0 to 300.0000 [mm]
0 => No monitoring
MP1970.0 X axis
MP1970.1 Y axis
MP1970.2 Z axis
MP1970.3 4th axis
MP1970.4 5th axis

MP1970 is used to set encoder monitoring. The position is computed from the pulses received from the position encoder (LS) and from the pulse received from the rotary encoder (ROD). The input value from MP1955 is used for this. The difference between the two results must not exceed MP1970. Movement monitoring by MP1140 is not active.

The movement monitor must be active at all times for safety reasons. It can only be deactivated by entering 0 while approach behaviour is being trimmed if the "Gross positioning error C" message appears despite the maximum value of 1 [mm] being entered.

MP1980 **Delayed shutdown of speed controller in EMERGENCY STOP**
Entry: 0 to 1.9999 seconds

In the event of a fault, braking resistors in the servo amplifier are required to stop the axes as quickly as possible to prevent danger from axes running down uncontrolled. This braking time can be reduced by delaying the shutdown of the TNC speed controller in MP1980.

At the same time, the controller outputs a braking torque which immediately brings the machine to a stop.

3.2 Optimizing the speed controller

The rotary and position encoders are trimmed in sequence. The new integrated oscilloscope makes it possible to program a step function as the output signal. The speed controller can be trimmed with this function and with the oscilloscope without the need for any additional equipment.

The position controller is automatically inactive when starting with the step function. The axes can only be traversed manually.

The controller can be trimmed in the sequence described. Provisional input values and input values for monitoring are empirical values that can vary depending on the type and size of the machine as well as on the drives, and it is therefore difficult to recommend suitable values. The characteristic curves will also vary from one type of machine to another.

Basic trimming procedure

- Assign the smallest possible value to the integral component of the speed controller (MP1920.x → 0).
- Increase monitoring of the speed controller (MP1910.x).
- Trim the proportional component of the speed controller (MP1940.x) as described below.
- Trim the integral component of the speed controller.

The speed can be trimmed with or without first evaluating the reference marks.

For traversing reference marks, the highest possible values must be entered in machine parameters MP1920.x, MP1940.x and MP1910.x to prevent the speed controller monitor responding.

MP1920.x = 200
MP1940.x = 200
MP1910.x = 300 000

Pre-setting machine parameters for the digitally controlled axes

Before trimming commences, provisional values must be entered for the corresponding machine parameters of the speed controller.

The integral component (MP1920.x) should be as low as possible so the P component can be trimmed without being unduly affected by the I component.

Since the input value for monitoring the speed controller is calculated as follows (error message: "Gross positioning error 3F"):

$$\text{MP1910.x} = \frac{\text{Vlimit}}{\text{MP1920.x} * 9.7\ \mu\text{V}}$$

a greater value for the monitor must be set when entering a low integral component (MP1920.x).

Given an ideal setting of 5 for the I component (MP1920.x) and a limit voltage of 15 V, the input value for MP1920.x is approximately 310 000.

Function	MP	Provisional input value
Axis sequence when approaching reference marks	1340.x	0
Select axes with digital speed controller	1900 Bit 0 ... 4	
Monitor speed controller	1910.x	310 000
Integral component for speed controller	1920.x	5
Proportional component for speed controller	1940.x	10
Polarity for torque signal	1950 Bit 0 ... 4	0
Select encoder for position control	1951 Bit 0 ... 4	1
Motion monitor for position and speed	1970	0 ... 0 => inactive
Delayed shutdown of position controller	1980	0

Checking polarity of nominal voltage

Cancel override and start axes with direction keys for a short period. Watch drive response, modify polarity while turning.

The monitor via MP1910 is in effect (error message: "GROSS POSITIONING ERROR 3F " if polarity is wrong).

Optimizing

The following curves should be displayed in the oscilloscope for each axis:

- Nominal speed ($\frac{mm}{min}$) : $N_{NOMINAL}$
- Actual speed ($\frac{mm}{min}$) : N_{ACTUAL}
- Output analogue voltage (mV): $V_{ANALOGUE}$

A step function is output for optimizing (see "Oscilloscope", later in this chapter). The feed rate must be selected such that the analogue voltage is < 8 V. It is now only necessary to press the appropriate axis direction keys to output a step function to the servo amplifier.

Proportional component MP1940.x

MP1940.x is increased until the step response (actual speed) shows large control fluctuations. This value is then halved to reduce overshoot to virtually nil.

Integral component MP1920.x

MP1920.x is then increased until large control fluctuations again occur for the actual speed. Half this value is then entered in MP1920.x. The resulting overshoot should not exceed 40%.

If the motor and drive spindle are linked by a toothed belt then an initial overshoot will occur in the step response – this must be disregarded for trimming purposes.

Monitoring the speed controller

Once the proportional and integral components of the speed controller have been trimmed, a suitable value must be calculated for MP1910.x.

Example: MP1920.x = 180 MP1940.x = 60

For MP1910.x with a Vlimit = 15 V, the following value is obtained:

$$\text{MP1910.x} = \frac{\text{Vlimit}}{\text{MP1920.x} * 9.7\ \mu\text{V}} = 9000$$

3.3 Optimizing the position controller

The position controller is trimmed in the same way as with the TNC 415. This rule applies for operation in the trailing error mode and with feed precontrol. Please note the description in Chapter 4, "Machine Integration", and the following information.

Evaluation of reference marks

After trimming the speed controller, the desired evaluation of the reference marks must be entered in MP1350.

Checking the direction of traverse and counting direction

The directions of traverse and counting must be checked again after the linear encoder for the position controller is activated with MP1951. The position controller is matched to the machine coordinate system with the following machine parameters:

- MP1040 Polarity of nominal speed value in positive direction of traverse
- If the counting direction of the axis is correct but motion is wrong, then MP210 (counting direction) must be modified.

Start axes with direction keys for a short period and correct the input values for polarity or counting direction if necessary.

Optimizing Kv factors for the position controller

With digital drive control the limits and/or ideal values are more dependent on the drives and the machine's mechanical systems than with previous control systems. In terms of the control principle, it is possible to work with high Kv factors and very small trailing errors. It is important to remember that the mechanical load on the machine will be greater.

Additional selectable Kv factors have been introduced for this reason, and they are stored in machine parameters MP1515.x and MP1815.x. Reduced Kv factors can be used for general milling work without great demands on accuracy at higher feed rates.

Delayed shutdown of speed controller in EMERGENCY STOP

Because the speed controller is integrated in the control system, the controlled axis runs down when the torque signal is turned off. For this reason the control system continues to be active for a certain time (MP1980) in an EMERGENCY STOP condition and outputs a torque. The time entered for this must be matched to the relevant drives and machine.

The axes must be stopped immediately by external holding brakes if the control fails and no braking torque can be output.

3.4 Oscilloscope

An oscilloscope has been integrated in the TNC 407/TNC 415 B/TNC 425 for trimming and optimizing the speed and position controllers.

The characteristic curves which the oscilloscope records can be stored in 4 channels and for all axis.

The following curves can be displayed:

Feed rate	V ACTUAL	Actual value [mm/min]
	V NOMINAL	Nominal value [mm/min]
Position and trailing error	S ACTUAL	[mm]
	S NOMINAL	[mm]
	D DIFF	Trailing error for position control [mm]
Speed	N ACTUAL	Actual speed [mm/min]
(digital speed control only)	N NOMINAL	Nominal speed [mm/min]
	N INT	Nominal/actual difference for speed controller
Nominal value	U ANALOGUE	Analogue voltage [mV]

Note on speed curves

Rotational speeds are recorded as feed rates in mm/min. The speed in rpm. can be calculated with the help of the distance covered per revolution, i.e.

$$\text{Rotational speed (rpm)} = \frac{\text{Feed rate } [\text{mm/min}]}{\text{Traverse per rev. } [\text{mm}]}$$

The data recorded for the display are stored until an NC graphic is called. Likewise a computed graphic simulation is deleted by the oscilloscope display.

A total of 5 colours can be selected in machine parameters for the oscilloscope display. They are selected in exactly the same way as the 3-plane view in NC graphic simulation.

MP7361.0	Background
MP7261.1	Grid
MP7361.2	Non-selected channels
MP7361.3	Cursor line, data, screen window
MP7361.4	Selected channel

The "Oscilloscope" mode is entered by keying in code number 688 379.

If you leave the Oscilloscope mode you can re-enter it using the MOD key and the OSZI soft key. The code number must only be re-keyed when the supply voltage has been switched off.

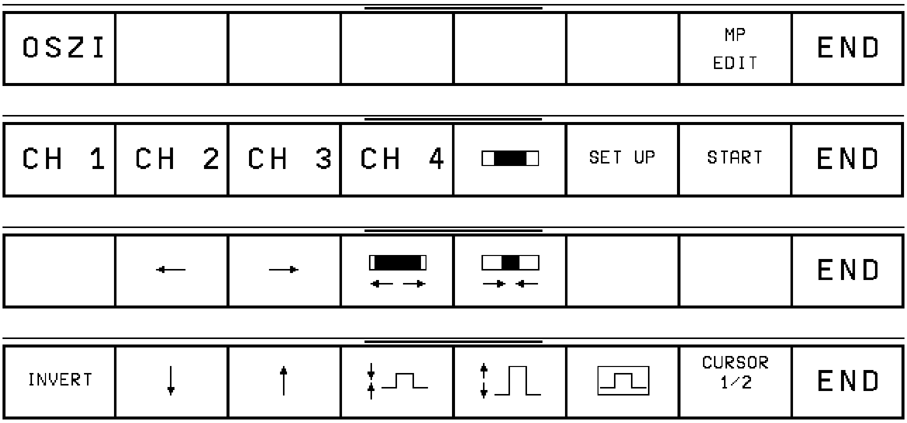
The following main display appears when the mode is called:

MANUAL OPERATION	OSCILLOSCOPE
OUTPUT	RAMP
NOML.FEED RATE	0
SAMPLE TIME	0,6 MS
CHANNEL 1 X	OFF
CHANNEL 2 X	OFF
CHANNEL 3 X	OFF
CHANNEL 4 X	OFF
TRIGGER	FREE RUN
TRIGGER THRESHOLD	+0
SLOPE	+
PRE-TRIGGER	0 %

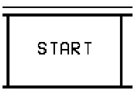
OSZI						MP EDIT	END

Use the cursor keys to move the highlight (cursor) to the desired item and to select the trigger conditions and the axes and parameters you wish to record.

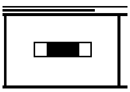
3.4.1 Soft-key rows



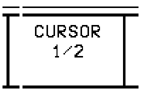
Meaning of soft keys:



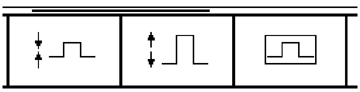
Start recording



Horizontal zoom



Activate cursor



Vertical zoom

Optimal vertical resolution, centred in screen

3.4.2 Triggers

The following entries are possible:

- FREE RUN Recording is terminated manually
- SINGLE SHOT Records a memory content - initiated by trigger condition.
- CHANNEL ... Starts recording when the trigger threshold of the selected channel is exceeded.

Trigger threshold

The trigger threshold for the selected channel is entered as a numerical value in the following units:

- Linear speed [mm/min]
- Position [mm]
- Rotational speed [mm/min]
- Trailing error [µm]
- Analogue voltage [mV]

Edge

Triggering with rising (positive) and falling (negative) edge.

Pre-Trigger

Defines the start of the recording as a % of the total recording time; possible entries are 0%, 25%, 50%, 75% and 100%. Press the ENT key to select.

3.4.3 Recording

The recording parameters to be edited are selected with the arrow keys. The values for FEED RATE and TRIGGER THRESHOLD are entered with the numerical keys. The values for all other recording parameters are selected by pressing the ENT key.

Output

In the Manual mode you can choose between the set ramp and a step function for outputting a nominal value.

The step function is only possible with digitally controlled axes and is necessary for trimming the speed controller. The step function and the oscilloscope recording facility can also be used to determine the maximum acceleration of the machine when the provisional input value is not known. In the MDI or Automatic modes acceleration is always by the set ramp.

Feed rate

With a step function as the output signal, the feed rate is entered in [mm/min]. The programmed feed rate dictates the acceleration after the ramp.

Time resolution

The recording time ranges from 2.4576 sec. to 24.576 sec. (set time x4096). The time set between 0.6 and 6 ms is the cycle time for recording curves. The recording time is overlaid beneath the grid. The beginning and end of the image are also displayed relative to the trigger point (cursor line T1).

Channels 1 to 4

A total of 4 channels can be selected for recording. The axes can be randomly assigned to the channels - this is done using the ENT key after selecting the entry position.

For each channel a characteristic curve is selected from the following variables:

Feed rate	V ACTUAL	Actual value [mm/min]
	V NOMINAL	Nominal value [mm/min]
Speed	S ACTUAL	Actual value [mm]
	S NOMINAL	Nominal value [mm]
Speed controller	N INT	Nominal/actual difference for speed controller [mm/min]
Position	N ACTUAL	Actual value [mm/min]
	N NOMINAL	Nominal value [mm/min]
Trailing error	D DIFF	Trailing error for speed control [mm]
Analogue voltage	U ANALOGUE	output analogue voltage [mV]
Channel	OFF	Channel is displayed
	SAVED	Channel is saved

Recording is activated with START (soft key). This calls a soft-key row in which the sole option is STOP. Recording can be interrupted at will. Saved channels cannot be displayed at the same time as recording is in progress because synchronization between the saved channels and the channels currently recording is not possible.

Evaluating the recorded curves with the cursor

Whereas the complete memory contents are displayed at the start, the time window selected before the start is displayed after the image is restored. The time to trigger point is shown as T1 at the left of the display. Beneath it is the absolute value in [mm/min], [mm] or [mV].

If the CURSOR 1/2 key is used to overlay a second cursor for time T2, then this cursor can also be moved using the arrow keys on the TNC control panel. The time that is displayed as T2 is the difference from T1. The numerical value overlaid underneath is also the difference from the value associated with T1. The display for T2 and the second cursor are deleted using the END or "Cursor 1/2" soft key.

Vertical zoom

The vertical grid display of each previously selected channel can be modified in fixed steps using soft keys. The value of the grid in the vertical axis is overlaid at the left hand margin beneath the channel number and the name of the recording.

Centring

Select vertical resolution to achieve an optimum display.

Now return to original vertical deflection: use the NO ENT key to restore the original display of the stored data in the vertical axis.

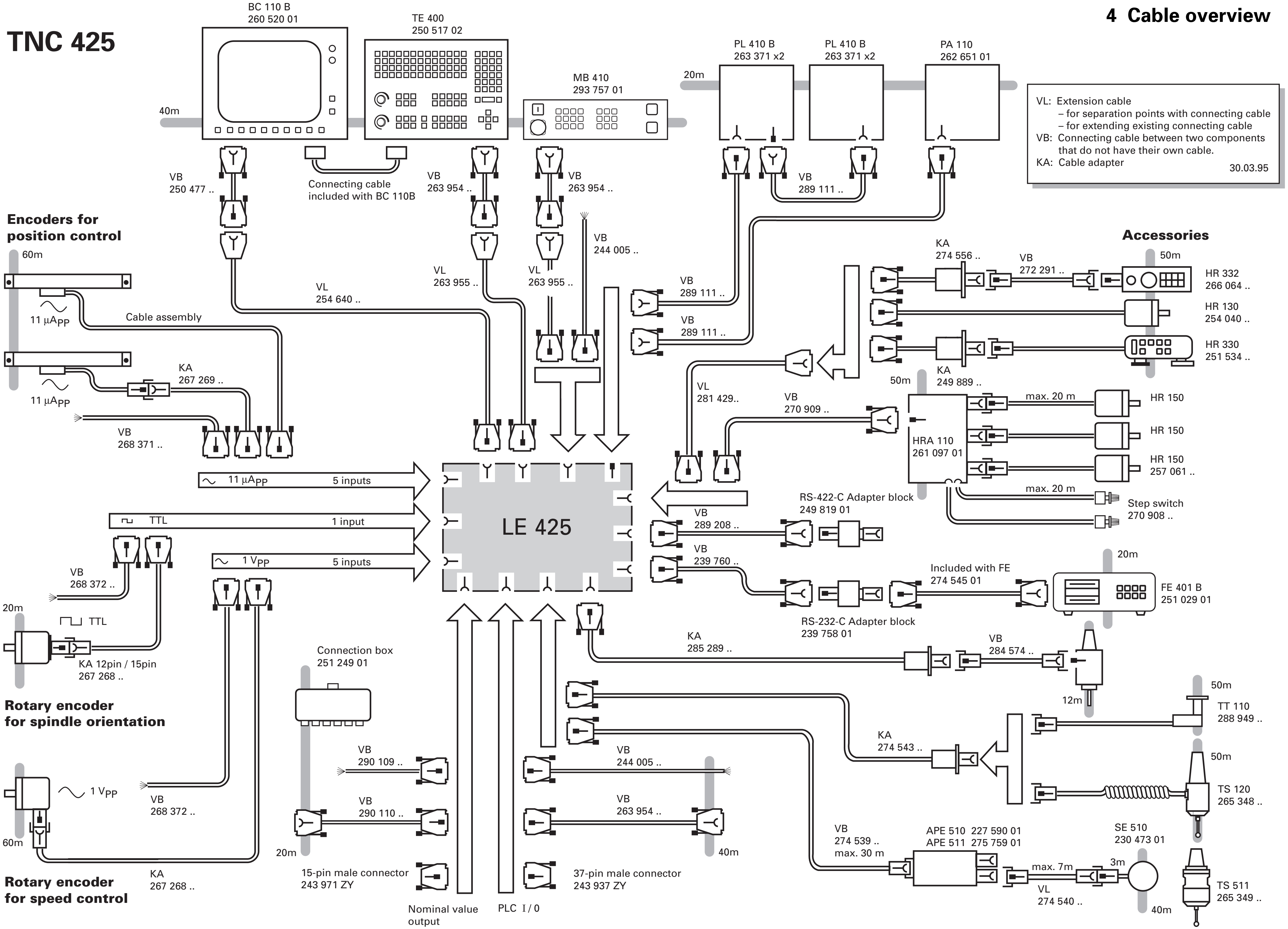
Horizontal zoom

The recording gathers 4096 evaluated data. Time resolution, i.e., the cycle time between the recorded data, can be set from 0.6 to 6 ms. The range of display stretching and condensing is limited as follows.

	Evaluated data	Data : Pixels
smallest display	4096	8 : 1
max. stretched display	64	1 : 8

The length of the displayed window and its beginning as an absolute position within the recording length of the data is displayed as a bar (scroll bar) in the status field.

TNC 425


BC 110 B
260 520 01
TE 400
250 517 02
MB 410
293 757 01
PL 410 B
263 371 x2
PL 410 B
263 371 x2
PA 110
262 651 01
40m
20m
VL: Extension cable
– for separation points with connecting cable
– for extending existing connecting cable
VB: Connecting cable between two components that do not have their own cable.
KA: Cable adapter
30.03.95
VB
250 477 ..
Connecting cable included with BC 110B
VB
263 954 ..
VB
263 954 ..
VB
289 111 ..
VB
244 005 ..
VL
254 640 ..
VL
263 955 ..
VL
263 955 ..
VB
289 111 ..
VB
289 111 ..
Encoders for position control
60m
11 µAPP
Cable assembly
KA
267 269 ..
11 µAPP
VB
268 371 ..
Accessories
50m
KA
274 556 ..
VB
272 291 ..
HR 332
266 064 ..
HR 130
254 040 ..
HR 330
251 534 ..
KA
249 889 ..
50m
VL
281 429..
VB
270 909 ..
HRA 110
261 097 01
max. 20 m
HR 150
HR 150
HR 150
257 061 ..
max. 20 m
Step switch
270 908 ..
11 µAPP
5 inputs
TTL
1 input
1 VPP
5 inputs
LE 425
RS-422-C Adapter block
249 819 01
VB
289 208 ..
VB
239 760 ..
Included with FE
274 545 01
20m
FE 401 B
251 029 01
RS-232-C Adapter block
239 758 01
VB
268 372 ..
20m
TTL
KA 12pin / 15pin
267 268 ..
Rotary encoder for spindle orientation
KA
285 289 ..
VB
284 574 ..
12m
50m
TT 110
288 949 ..
Connection box
251 249 01
KA
274 543 ..
50m
TS 120
265 348 ..
VB
290 109 ..
VB
244 005 ..
1 VPP
VB
268 372 ..
VB
290 110 ..
VB
263 954 ..
60m
20m
40m
VB
274 539 ..
max. 30 m
APE 510 227 590 01
APE 511 275 759 01
SE 510
230 473 01
3m
max. 7m
VL
274 540 ..
40m
TS 511
265 349 ..
Rotary encoder for speed control
KA
267 268 ..
15-pin male connector
243 971 ZY
Nominal value output
PLC I / 0
37-pin male connector
243 937 ZY

Appendix - Contents

1 7-Bit ASCII code

Character	DEC	OCT	HEX
NUL	000	000	00
SOH	001	001	01
STX	002	002	02
ETX	003	003	03
EOT	004	004	04
ENQ	005	005	05
ACK	006	006	06
BEL	007	007	07
BS	008	010	08
HT	009	011	09
LF	010	012	0A
VT	011	013	0B
FF	012	014	0C
CR	013	015	0D
SO	014	016	0E
SI	015	017	0F
DLE	016	020	10
DC1 (X-ON)	017	021	11
DC2	018	022	12
DC3 (X-OFF)	019	023	13
DC4	020	024	14
NAK	021	025	15
SYN	022	026	16
ETB	023	027	17
CAN	024	030	18
EM	025	031	19
SUB	026	032	1A
ESC	027	033	1B
FS	028	034	1C
GS	029	035	1D
RS	030	036	1E
US	031	037	1F
SP	032	040	20
!	033	041	21
"	034	042	22
#	035	043	23
$	036	044	24
%	037	045	25
&	038	046	26
´	039	047	27
(	040	050	28
)	041	051	29
*	042	052	2A
+	043	053	2B
,	044	054	2C
-	045	055	2D
.	046	056	2E
/	047	057	2F

Character	DEC	OCT	HEX
0	048	060	30
1	049	061	31
2	050	062	32
3	051	063	33
4	052	064	34
5	053	065	35
6	054	066	36
7	055	067	37
8	056	070	38
9	057	071	39
:	058	072	3A
;	059	073	3B
<	060	074	3C
=	061	075	3D
>	062	076	3E
?	063	077	3F
@	064	100	40
A	065	101	41
B	066	102	42
C	067	103	43
D	068	104	44
E	069	105	45
F	070	106	46
G	071	107	47
H	072	110	48
I	073	111	49
J	074	112	4A
K	075	113	4B
L	076	114	4C
M	077	115	4D
N	078	116	4E
O	079	117	4F
P	080	120	50
Q	081	121	51
R	082	122	52
S	083	123	53
T	084	124	54
U	085	125	55
V	086	126	56
W	087	127	57
X	088	130	58
Y	089	131	59
Z	090	132	5A
[	091	133	5B
\	092	134	5C
]	093	135	5D
^	094	136	5E
_	095	137	5F

Character	DEC	OCT	HEX
`	096	140	60
a	097	141	61
b	098	142	62
c	099	143	63
d	100	144	64
e	101	145	65
f	102	146	66
g	103	147	67
h	104	150	68
i	105	151	69
j	106	152	6A
k	107	153	6B
l	108	154	6C
m	109	155	6D
n	110	156	6E
o	111	157	6F
p	112	160	70
q	113	161	71
r	114	162	72
s	115	163	73
t	116	164	74
u	117	165	75
v	118	166	76
w	119	167	77
x	120	170	78
y	121	171	79
z	122	172	7A
{	123	173	7B
¦	124	174	7C
}	125	175	7D
~	126	176	7E
DEL	127	177	7F

2 Powers of 2

n	2^n
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1 024
11	2 048
12	4 096
13	8 192
14	16 384
15	32 768
16	65 536
17	131 072
18	262 144
19	524 288
20	1 048 576

Subject Index

C

D

E

F

G

H

M

N

O

P

Q

R

S

T

U

V

W