



HEIDENHAIN

Technical Manual

TNC 360

Valid for the NC software types

259 90 up to version 06

260 02 up to version 17

280 49 up to version 17

260 060 up to version 17

280 610 up to version 17

Preface

This Technical Manual is intended for all 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 TNC 360.

Whenever HEIDENHAIN improves the hardware or software in these 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 A4 format.

No manual 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 giving us your ideas.

DR. JOHANNES HEIDENHAIN GmbH
Department E/P
PO Box 1260
83292 Traunreut
Germany

Contents Technical Manual TNC 360

Update Information	1
Introduction	2
Mounting and Electrical Installation	3
Machine Integration	4
Machine Parameters	5
Markers and Words	6
PLC Programming	7
Data interfaces	8
Original Equipment Manufacturer's (OEM) Cycles	9
Positioning Module	10
Appendix	11

Update Information No. 8

In mid February 1996 **software version 17** was released for the NC software types 260 02 and 280 49, and for the newly introduced software types 260 060 (for 1 MB EPROMs) and 280 610 (for 2 MB EPROMs). The new software types were introduced for the Polish conversational language.

Software version 17 contains the following additions:

- The software now supports the new handwheel **HR 410**.

The HR 410 is a portable electronic handwheel with:

- Keys for selection of five axes
- Keys for traverse direction
- Keys for three pre-programmed feed rates for latched traverse
- One key for actual-position capture
- Three keys for machine functions determined by the machine manufacturer
- Two permissive buttons
- Emergency stop button
- Magnetic holding pads

With **MP7640 = 6** you can activate the functions for the HR410 handwheel.

MP7645.0 determines whether the keys on the handwheel are assigned to the NC or the PLC.

MP7645.0 = 0
NC key assignment

X		IV
Y		V
Z		ACTUAL POSITION CAPTURE
LOW FEED RATE	MEDIUM FEED RATE	HIGH FEED RATE
—		+
O109 I173	O110 I174	O111 I175

Handwheel
control panel

MP7645.0 = 1
PLC key assignment

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

With the exception of the A, B and C function keys, all keys are assigned to the NC. MP7670.x determines the interpolation factor for low, medium and high speeds. MP7671.x determines the low, medium and high speed values. The speed is given as a percentage factor of the manual feed rate (MP1020.x).

All keys are assigned to the PLC. Handwheel axis and handwheel interpolation are set by module 9036. With W766 you can change the feed rate by pressing the axis direction keys.

MP7670 Interpolation factor for handwheel
Input: 0 to 10
MP7670.0 Interpolation factor for low speed
MP7670.1 Interpolation factor for medium speed
MP7670.2 Interpolation factor for high speed

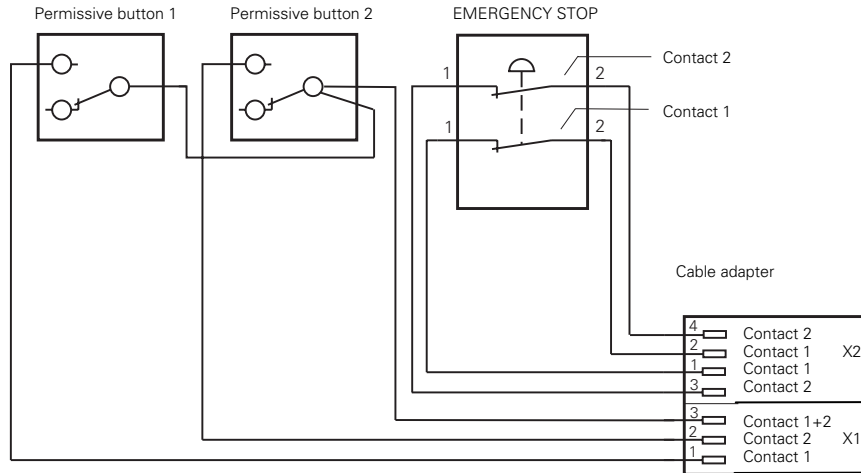
MP7671 Manual feed rate in "handwheel" operating mode with HR 410
Input: 0 to 1000 [% to MP1020]
MP7670.0 Low speed
MP7670.1 Medium speed
MP7670.2 High speed

A dummy plug (Id.-Nr. 271 958 03) is available for an EMERGENCY STOP.
There are connecting terminals on the adapter for the contacts from the emergency stop button and the permissive buttons (maximum load 1.2 A)

Connector layout:

VL ID number 281 429..			KA ID number 296 466..			VB ID number 296 467 05			HR 410 ID number 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	Connecto r (male) 18-pin		Connecto r (female) 18-pin	Connecto r (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/BR	1	1	
					WH/BR	3	Contacts 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 contacts for the HR 410 permissive buttons and the EMERGENCY STOP button:



- When machine parameters are being downloaded via the V.24 interface, comments beginning with the characters ";" or "*" can also be downloaded, either before or after the machine parameter value.
- Machine parameter MP5020, bit 9 decides whether the control sends the EOT character after receiving the ETX character.
- The error message "ERROR IN TRANSFERRED VALUE" is displayed after a timeout is exceeded during transfer via the V.24 interface.
- PLC Module 9036 expanded
With Module 9036, handwheel interpolation factors of all or of individual axes, and the values for the jog positioning can be transferred from the PLC to the NC.

Calling the module:

PS B/W/D/K PLC status information type
 PS B/W/D/K PLC status value
 CM 9036
 PL B/W/D Messages that are stored in the STACK:
 0: PLC status information was transferred
 1: Incorrect PLC status information type
 2: Transferred PLC status value incorrect
 3: Writing inhibited (e.g. by MP)

The marker M3171 is set if there is an error in transfer.

NC status information type:

PLC status value:

0:	Handwheel interpolation factor for X-axis	0 ... 10
1:	Handwheel interpolation factor for Y-axis	0 ... 10
2:	Handwheel interpolation factor for Z-axis	0 ... 10
3:	Handwheel interpolation factor for 4th axis	0 ... 10
4:	Handwheel interpolation factor for all axes	0 ... 10
5:	Handwheel interpolation factor for all axes	0 ... 10
6:	Select handwheel axis; X-axis	0
	Y-axis	1
	Z-axis	2
	4th axis	3

7 to 9 reserved

10:	Limit on jog increment	0 ... 50000 μm
	-1 =	Remove limit, activate last jog increment
	-2 =	Remove limit, activate minimum from PLC limit on jog increment and jog increment input via NC

Update Information No. 7

The new edition of the Technical Manual for TNC 360 incorporates the data from Update Information issues No. 1 to 6 in the appropriate chapters and sections. The description of TNC 355 has been deleted since this unit is no longer in our product program.

Please replace the complete contents of the manual.

We will continue to send new information on the hardware and software of the TNC 360. Please file the Update Information issues here.

Update Information No. 6

New PLC Input/Output Unit PL 410 B

In December the PLC I/O unit PL 410 will be replaced by the PL 410 B.

The PL 410 B provides the same number of switching inputs/output as on the PL 410. The connector layout is compatible with the PL 410. The dimensions have changed slightly near the connections X15 to X22.

In contrast to the PL 410, the PL 410 B will be available in two versions. The following version can be connected to the TNC 360:

PL 410 B

64 PLC inputs
31 PLC outputs
1 output "Control is operational"

The logic unit must be connected to the PL 410 B with a new connecting cable.

Connecting cable LE 360C/ PL 410 B: Id.-Nr. 289 111 ..
Max. 20 m

Update Information No. 5

For NC software types 260 02 and 280 49, **software version 13** was released in mid-June 1994, **software version 14** at the end of June, and **software version 15** at the beginning of July.

Software version 13 contains the following enhancements:

The input range for machine parameter MP1350 (type of reference mark approach) was expanded. An input value of 3 selects encoders with distance-coded reference marks, and the direction of traverse reverses when the trip dog for the reference end position is crossed.

Data transfer in blockwise mode (ACK/NAK protocol) could be interrupted if the control characters STX or ETB were transmitted incorrectly. Since the TNC did not know the cause of the interruption, it sends NAK (after a delay in which no further data are transferred) if a block was not completely recognized due to a faulty STX or ETB.

If ACK is not received within a certain time, the error message TRANSFERRED DATA INCORRECT N is generated.

If bit 3 in machine parameter MP7641 is set, the electronic handwheel is selected in each operating mode (initially before REF traversing) to be able to interrogate the inputs of the handwheel keys (except axis keys) in the PLC. The axis keys on the handwheel and the handwheel impulses do not become active until handwheel mode is selected. After an error the handwheel is not selected again until the handwheel key is pressed.

If the operating voltage of the control is outside the limit values, the blinking error message PROCESSOR CHECK ERROR M is displayed.

The input range of machine parameter MP4220 has been expanded to 0 ... 65535.

The memory for the executable PLC program has been increased from 28K bytes to 32K bytes.

The value for the analog voltage of the analog input at connector X8 is transferred to word W392.

The meaning of machine parameter MP7225 (automatic block generation with the capture actual position key or with PLC marker M2829) has been changed as follows:

MP7225 = 1 : Block generation with the capture actual position key

MP7225 = 2 : Block generation with PLC marker M2829

MP7225 = 0 : No block generation

The resolution of the feed rate display depends on the programmed feed rate:

Feed rate ≤ 31 999 mm/min (previously 29 999 mm/min): display step 2 mm/min

Feed rate > 31 999 mm/min (previously 29 999 mm/min): display step 20 mm/min

Software version 14 was released to correct the following error:

If a STOP and an M function for PLC positioning were executed in one NC block, in the following block an M function that becomes effective at the beginning of the block (e.g., M3) was not executed!

Software version 15 was released to correct the following error:

If, in an OEM cycle that was run from the EPROM, the feed rate was defined via Q parameter, a greatly excessive feed rate was output.

Update Information No. 4

In late February 1994, software version 12 was released for the NC software 260 02 and 280 49.

Note:

In software version 11, the deceleration ramp is too flat when the feed rate is changed at constant contour transitions during operation with feed precontrol. This error was corrected in software version 12.

Software version 11 must be replaced by software version 12!

No new features were introduced in software version 12.

Update Information No. 3

Software version 11 for software types 260 02 and 280 49 was released at the end of January 1994. Software version 10 was not released for general distribution.

The following improvements were made since version 9:

- Machine parameter 7411 defines whether during execution of the TOUCH-PROBE block the current compensation values for probe length and radius should be taken from the central tool file or from the calibration process.

MP7411 = 1 : Probe length and radius values from the tool file

MP7411 = 0 : Probe length and radius values from the calibration process

MP 7411 is also effective for tool length compensation in the digitizing cycles. This machine parameter was already available in version 08, but it was not documented.

- Machine parameter MP7225 was introduced for automatic generation of NC blocks in the PROGRAMMING AND EDITING mode. With the "actual position capture" key (teach-in) it is now possible to generate a positioning block in plain language dialog (not in ISO mode) with a maximum of 3 axes and without feed rate, radius compensation or M-functions. This positioning block is inserted below the currently addressed block in the selected NC part program. The current actual position values become the nominal position values. The axis is selected in the dialog "AXIS SELECTION =" in the MOD operating mode. Here up to 3 out of 4 axes can be selected by pressing the corresponding axis keys.

MP7225 = 1 : Block generation with the actual-position-capture key (teach-in)

MP7225 = 2 : Block generation with the actual-position-capture key or with the PLC marker M2829 (the marker is reset by the NC)

MP7225 = 0 : No block generation or axis selection possible.

- If the machine parameters are erased, the RS-232-C/V.24 interface is now preset to FE mode.
- PLC inputs I160 to I175 (HR 332 handwheel) now generate signal edges. The positive edge is assigned to markers M1660 to M1675, the negative flank to markers M1860 to M1875.
- If machine parameter MP7641, Bit 2 = 1, the interpolation factor for the handwheel can be set both from the keyboard as well as from PLC module 9036. If Bit 2 = 0, the machine parameter functions as before (input either from the keyboard or from PLC module 9036).
- The NC saves the code number entered in the MOD mode in Doubleword D276.
- Machine parameter MP810 defines the modulo value for reducing the position values of the auxiliary axes. Machine parameter MP7470 has no effect on auxiliary axes. Rotary axes as NC axes are always reduced to the range 0° to 359.999°. Machine parameter MP810 was already available in version 08, but it was not documented.

- The new PLC module 9124 makes it possible to set a feed rate override value for secondary axes. The override value can lie between 0% and 100% (resolution 0.01 %) and must be transferred as a whole number (0 to 10 000). It can be set before the beginning of a movement or during the movement of an auxiliary axis. If the control is reset, the NC presets an override value of 100%.

Call :

PS B/W/D/K <Axis> (0..3 for X/Y/Z/4)

PS B/W/D/K <Override value> (0..10 000)

CM 9124

PL B/W/D <Error code>

0: Override was set

1: Invalid axis was entered

2: Axis is not defined as auxiliary axis

3: Override value is invalid

Error status after call: M3171

= 0 : Override was set

= 1 : Error condition see above

If more than one of the PLC modules 9120/9121/9123 for controlling the movement of auxiliary axes is called within one PLC scan, only the last called PLC module is executed. The module 9124 can be called in addition to the above modules within one PLC scan, but it will always become effective after the other module.

- In PLC module 9036 (transferring PLC status information to the NC) the interpolation factor can be set for all axes by transferring the value 4 or 5.
- The PLC can inhibit the reference pulse for specific axes through Word W608 (bit-coded,...4zyx).
Bit = 1 : Reference pulse is not evaluated
Bit = 0 : The next reference pulse is evaluated
- By setting marker M2615 the reference mark of the spindle is evaluated again. The marker is reset by the NC.
- Marker 2510 fixes the spindle potentiometer setting at 100%. Marker M2511 has the same effect on the feed rate potentiometer.
- CC blocks in OEM cycles are effective only locally and are not transferred into the calling program.

Update Information No. 2

In early September 1993 the software version 09 was released for the software types 260 02 and 280 49.

The new version contains the following changes:

- If marker M2612 (Suppress position exchange in the tool table) is set before a T strobe is set (M2046), the position numbers are not exchanged. The PLC acknowledges the T strobe without having exchanged in the tool magazine and resets the PLC marker M2612. The new tool number is shown inverted and the associated tool data (length and radius) are activated. The tool number is shown inverted in the status display until the tool has physically been changed. If the control is switched off and on in this condition, the last exchanged tool becomes active again.
- During execution of cycle 13 (spindle orientation) the NC sets the new marker M2408. This marker can be evaluated by the PLC for the spindle orientation and should afterwards be reset by the PLC.

Update Information No. 1

1 Software

In July 1993 the following software was released:

260 02x 08 for insertion of 1M byte EPROMs
and **280 49x 08** for insertion of 2M byte EPROMs

The software numbers 280 49x 01 to 07 were not delivered. The new software version became necessary because new logic units can also accommodate 2M byte EPROMs (see below in Section 2 "Hardware").

The new software version contains the following changes:

- The PLC can limit the maximum feed rate through the doubleword D596. In order to ensure compatibility to previous PLC programs the doubleword D596 is preset with the value 300 000 mm/min after control switch-on or after interruption of the PLC run.

The new feed rate is effective immediately!

The doubleword D596 has no effect with the new cycles "Tapping" (Cycle 2) and "Rigid Tapping" (Cycle 17).

- In the newly introduced machine parameter MP60, axes can be defined as auxiliary axes. These axes cannot be moved by an NC program. They are controlled exclusively by the PLC.

All auxiliary axes work independently of each other. Auxiliary axes always move in trailing mode. The following modules are available for the PLC to control the axes:

Module 9120: Positioning of auxiliary axes

Module 9121: Stopping the auxiliary axes

Module 9122: Status inquiry of auxiliary axes

Module 9123: Traversing the reference point of an auxiliary axes

- Spindle orientation can be suppressed at the beginning of Cycle 17 "Rigid Tapping" with machine parameter MP7160. At the beginning of the cycle, the spindle voltage is decelerated with the ramp from MP3410.1 to the value 0 volts.

In this case it is not possible to cut into the same thread several times!

- Up to 20 NC "tool def" blocks can be read-in during blockwise transfer without central tool memory, whereby the tool def block must be read-in before the associated "tool call". When a "PGM call" NC block is read-in or when a user cycle is called, the NC blocks with "tool def" are considered up to the above mentioned limit and provided that the proper order is followed. A violation of these preconditions releases the error message "TOO MANY TOOLS".

- If bit 2 is set in machine parameter MP7300, the last inserted (programmed) tool is automatically activated during switch-on.
- By setting the static PLC marker M2612 the PLC can now prevent the exchange of pocket numbers in the central tool memory during a P output.
- Function FN19 was introduced, with which two numerical or Q parameter values with an accuracy of 1/1000 (i.e., three places after the decimal point) are transferred into the PLC doublewords D280 and D284. A value of 2.5, for example, is filed in the doublewords as 2500.

The unit of measure of the calling NC program is set in marker M2150 (millimeter=0/inch=1).

During transfer the NC sets the strobe marker M2149. The transfer must be acknowledged by the PLC with the marker M2611.

- The number of PLC labels was increased to 1024.
- Marker M2614 was introduced. Setting this marker blocks the output of PLC functions (M/S/T/Q output) by the NC. The marker is set and reset by the PLC; it is read by the NC.
- Marker M2827 was introduced. It is set by the PLC and causes the following behavior in case of an external EMERGENCY STOP and erasable positioning error:
Machining is not aborted ("control in operation" symbol off, strobe signal reset). Instead, machining is merely interrupted as in an NC STOP (control-in-operation symbol blinks). This permits machining to be resumed with NC START after the error has been corrected. This marker functions only for the output of M/S/T/Q strobes.
- Now a maximum of 32 Q parameters can be transferred for user cycles. To do this, the 'DLG-DEF' or 'DLG-CALL' blocks must be programmed several times in the user cycle, whereby in the third DLG block only the first five entries can be evaluated.
- The following Q parameters were introduced:
Q114 current tool length
Q115 to Q118 measured values of the 4th axis after a programmed probing cycle
- The type of tool compensation (R0/RR/RL/R+/R-) is stored in Q parameter Q123:
Q123 = 0 means R0
 = 1 means RR
 = 2 means RL
 = 3 means R+
 = 4 means R-
- Application as positioning module:
If machine parameter MP 4010 = 1 is programmed (PLC program from RAM), no machine parameters will be taken from the PLC EPROM when the positioning module is switched on. If the CRC sum of the machine parameters is incorrect they will be taken from the PLC EPROM.

- The software now supports PLC subprograms stored in the PLC-EPROM (translated PLC code). With the PLC.EXE programming software from HEIDENHAIN such programs can be written and on an external computer and filed in the PLC EPROM.
- The incremental jog positioning can be activated or deactivated (dialog "JOG-INCREMENT: ...") in the 'ELECTRONIC HANDWHEEL' operating mode by pressing the 'ELECTRONIC HANDWHEEL' key, provided that marker M2498 is set.
- In the 'PROGRAMMING AND EDITING' operating mode the electronic handwheel can be used to move the axes, provided that in machine parameter MP7641 bit 1 = 1. The interpolation factor (regardless of handwheel model) and the handwheel axis (for HR130) can be change only in the 'ELECTRONIC HANDWHEEL' operating mode. A handwheel axis, once chosen, remains in effect even when the operating mode is changed.

Simultaneous operation of the handwheel interface and the RS-232-C interface at differing baud rates (38 400 and 19 200 baud) results in the error message "BAUD RATE NOT POSSIBLE".

- The current feed rate in mm/min is now available in the PLC.
- Module 9150
During an active M/S/T output the PLC can use module 9150 to define an NC block, which is then executed after the M/S/T strobe is acknowledged, before the control continues the NC program. An NC block can also be defined if no program is being run. The block is then executed immediately.

Call:

```
PS B/W/D/K <Instruction code>
PS B/W/D/K <Address of the parameter block in the B/W/D range>
CM 9150
PL B/W/D <Error code>
```

Error code: 0 = NC block was inserted
 1 = NC program started, but no M/S/T strobe
 2 = Unknown instruction code
 3 = Incorrect address in B/W/D range

Error status after call: M3171 = 0 NC block was inserted
 = 1 error condition see above

At present the instruction code <0> is implemented for TOOL CALL

Parameters :

```
B<Adr+0> active elements bit-coded
Bit 0 =1: Tool number, otherwise modal
Bit 1 =1: Tool axis, otherwise modal
Bit 2 =1: Spindle speed, otherwise modal
Bit 3 =0
Bit 4 =0
B<Adr+1> Tool axis (0/1/2/3 = X/Y/Z/IV)
W<Adr+2> Tool number
D <Adr+ 4> Spindle speed (Format 0.001 rpm)
```

- Module 9120
Positioning an auxiliary axis

The positioning of an axis is started by presetting a target position (in the reference system), a feed rate and a flag register. The axis is positioned without regard to other processes in the control. There is no contour interpolation with other axes.

Conditions:

The given axis must be activated via MP10 and configured as an auxiliary axis via MP60.

The values for rapid traverse, analog voltage for rapid traverse, acceleration, etc., must be properly set in the machine parameters.

For axes with automatic reduction (modulo value in MP810.x) the axis is always moved in the shortest direction to the target position, unless the target position is entered as an incremental value.

There is no checking for violations of the limit switch ranges!

The axis must be stationary. If the axis is already moving, the positioning must be terminated beforehand with module 9121.

If the axis was in the reference point traversing mode, this state is canceled. The positioning always builds on the momentary counter contents.

If the modules 9120, 9121 and 9122 are called several times during a PLC scan, only the last instruction is executed.

If a "positioning error" status was set in this axis it is erased.

Potential errors:

A non-existent axis was transferred.

An axis was transferred that was not identified as an auxiliary axis in MP10 and MP60.

The axis is already moving.

Call:

PS	B/W/D/K <Axis>	(0 to 3 for X/Y/Z/4)
PS	B/W/D/K <Target position>	(in the reference system, Format 0.001mm)
PS	B/W/D/K <Feed rate>	(mm/min)
PS	B/W/D/K <Flag register>	Bit 0 = 1: incremental target position = 0: absolute target position

CM9120

PL	B/W/D	<Error code>
		0: Positioning was started
		1: A non-existent axis was transferred
		2: Axis is not configured as an auxiliary axis
		3: The axis is already moving
		4: Absolute position outside of the modulo range

Error status after call: M3171	= 0: Positioning was started correctly
	= 1: Positioning was faulty

- **PLC MODULE 9121**
Stopping a positioning with an auxiliary axis

A positioning started beforehand with modules 9120 or 9123 can be canceled at any time with module 9121.

Conditions:

The given axis must be activated via MP10 and configured in MP60 as an auxiliary axis.

If modules 9120, 9121 and 9122 are called several times for the same axis during a PLC-scan, only the last instruction will be executed.

Potential errors:

A non-existent axis was transferred.

An axis was transferred that was not configured in MP10 and MP60 as an auxiliary axis.

The given axis is already stationary.

Call:

PS B/W/D/K<Axis> (0 to 3 for X/Y/Z/4)

CM9121

PL B/W/D <Error code>

0: Positioning is canceled

1: A non-existent axis was transferred

2: Axis is not configured as an auxiliary axis

3: Axis was already stationary

Error status after call: M3171

= 0: Positioning was stopped

= 1: Faulty execution

- **PLC MODULE 9122**
Inquiring the status of an auxiliary axis

For a certain axis a bit-coded status word is transferred that contains information on the momentary operating state of this axis.

Conditions:

Status changes caused by commands that the PLC sends to control the auxiliary axes (modules 9120, 9121, 9123) are not recognized until the next PLC scan.

After switch-on, bit 1 (axis over reference point) is erased.

It is possible to position the axis without traversing the reference point first.

Potential errors:

A non-existent axis was transferred.

Call:

PS B/W/D/K <Axis> (0 to 3 for X/Y/Z/4)

CM9122

PL B/W/D <Status>

Bit 0: 1= Axis is auxiliary axis

Bit 1: 1= Axis has traversed the reference point

Bit 2: 1= Axis is positioned

Bit 3: 1= Direction of motion is negative

Bit 4: 1= A positioning error has occurred

Error status after call: M3171

= 0: Status was transferred

= 1: Faulty execution

- **PLC MODULE 9123**

Passing over the reference point of an auxiliary axis

The module starts positioning in a preset direction, which continues until a reference point is found or the positioning is canceled with module 9121.

Conditions:

The given axis must be activated via MP10 and configured as an auxiliary axis via MP60.

The values for rapid traverse, analog voltage for rapid traverse, acceleration, etc., must be set in the machine parameters.

There is no checking for violations of the limit switch ranges!

The axis must be stationary. If the axis is already moving, the positioning must be canceled beforehand with module 9121.

The feed rate override is not included in the calculation.

The state "reference point not yet traversed" is set for the axis.

A reference point that already exists in this axis is erased, but not the numerical value of the axis. This is not reinitialized until the reference point is found.

If modules 9120, 9121 and 9122 are called several times during a PLC scan, only the last instruction is executed.

If a "positioning error" status was set in this axis it is erased.

The positioning is stopped as soon as the reference point is reached. Since the axis must decelerate before it can stop, it comes to rest behind the reference point in the direction of motion.

Potential errors:

A non-existent axis was transferred.

An axis was transferred that was not configured in MP10 and MP60 as an auxiliary axis.

The axis is already moving.

Call:

PS	B/W/D/K <Axis>	(0 to 3 for X/Y/Z/4)
PS	B/W/D/K <Feed rate>	(mm/min)
PS	B/W/D/K <Flag register>	Bit 0 = 1: negative traverse direction = 0: positive traverse direction

CM9123

PL	B/W/D	<Error code>
		0: Positioning was started
		1: A non-existent axis was transferred
		2: The axis is not configured as an auxiliary axis
		3: The axis is already moving

Error status after call: M3171	= 0: Positioning was started
	= 1: Faulty execution

2 Hardware

The part numbers of the LE 360C logic units that can accommodate 2M byte EPROMs are:

Id.-Nr. 270 641 3x for BE 212

Id.-Nr. 270 642 3x for BF 110

The 2-M byte EPROMs occupy only the sockets IC-P1 and IC-P2. IC-P3 and IC-P4 remain vacant. The jumper located next to IC-P1 should then be inserted in the setting 2M!

You will receive Update Information issues on the TNC 360/TNC 335 hardware and software whenever developments warrant. This information will then be included in Supplementary Issues that you can integrate into the appropriate chapters of the Technical Manual.

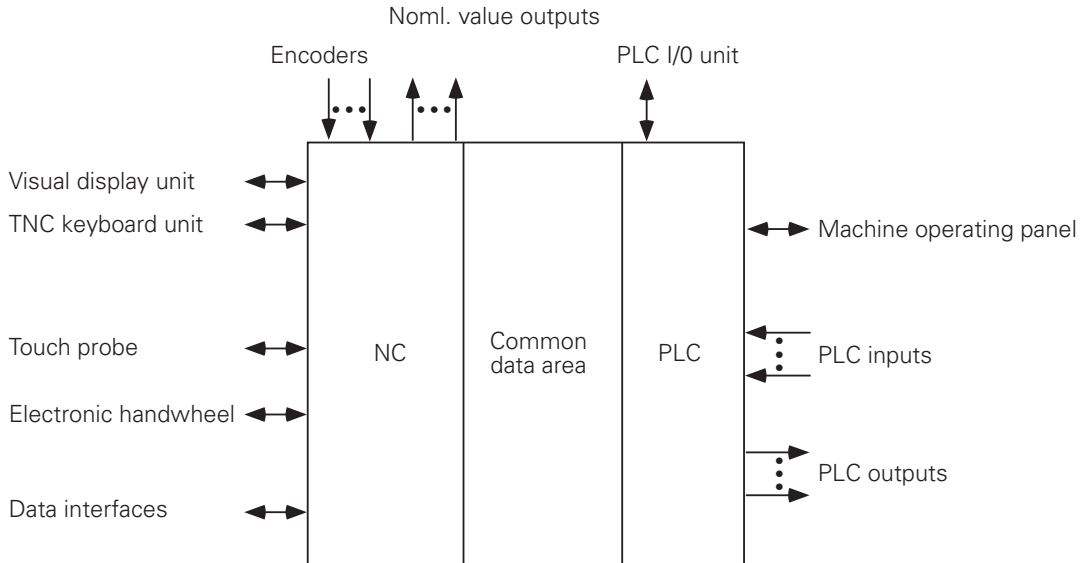
Introduction – Contents

1	Hardware Concept	2-2
2	Features and Specifications	2-3
2.1	TNC 360	2-3
3	Software	2-6
3.1	NC Software	2-6
3.1.1	Software and hardware versions	2-7
3.1.2	Software option	2-8
3.2	PLC Software	2-8
3.3	EPROM sockets	2-9

1 Hardware Concept

The HEIDENHAIN TNC 360 contouring control is designed for use with drilling and milling machines.

The HEIDENHAIN TNCs consist of several units. The principle 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 Features and Specifications

2.1 TNC 360

Components	• Logic unit LE 360 • Keyboard TE 355 A, TE 355 B • Color graphics VDU BE 212 (12 inch, 512 x 256 pixels) • Flat luminescent screen BF 110 (192 mm x 120 mm, 640 x 400 pixels)
Control modes	• Contouring control for 4 axes and spindle orientation • Linear interpolation in 3 of 4 axes • Circular interpolation in 2 of 4 axes
Program memory	Buffered RAM memory (approx. 70 KB) for 32 NC programs, central tool file, PLC program (if not filed in EPROM), EPROM memory (128 KB) for PLC program, OEM cycles, dialogues for OEM cycles, PLC error messages
Tool memory	99 tools
Operating modes	• Manual operation • Electronic handwheel /jog positioning • Positioning by manual data input • Program run, single block • Program run, full sequence • Programming and editing • Test run (logical and graphical)
Programming	In HEIDENHAIN conversational mode and according to ISO
Entry and display resolution	1 µm, 5 µm, 10 µm, 50 µm, 100 µm

Programmable functions	• Nominal position (absolute or incremental dimensions) in Cartesian or polar coordinates • Straight lines • Circular arcs • Helical interpolation • Corner rounding, chamfering • Tangential approach and departure from a contour • Tool number, tool length and radius compensation • Spindle speed • Rapid traverse • Feed-rate • Program call from inside other programs • Subprograms and repetition of program sections • Fixed cycles for peck drilling, tapping (without floating tap holder), slot milling, rectangular pocket milling, circular pocket milling • Cycles for milling pockets with a free contour (SL cycles) • Shifting and rotation of the coordinate system, mirroring, scaling, dwell time, miscellaneous functions M, program stop • Spindle orientation (to be implemented by the machine manufacturer) • OEM specific cycles (to be defined by the manufacturer of the machine)
Parameter-programming	Mathematical functions ($=$, $+$, $-$, x , $:$, $\sin$, $\cos$, angle α of $r \sin \alpha$ and $r \cos \alpha$, $\sqrt{}$, $\sqrt{a^2 + b^2}$), parameter comparison ($=$, $\neq$, $>$, $<$), output of parameter values via the data interface
Digitizing	• With TS 120 and TNC software expansion option • Optional evaluation software for PCs
Maximum traverse	$\pm 30\,000$ mm (1181 in.)
Maximum traversing speed	300 m/min (11 810 ipm)
Data interfaces	• RS-232-C/V.24; data transfer rates up to 38 400 baud
Cycle times	Block processing time: 40 ms (for 3D straight lines without radius compensation and with 100% PLC utilization) Control loop cycle time: 6 ms PLC cycle time: 24 ms
Position feedback	Incremental HEIDENHAIN linear and angular encoders, preferably with distance-coded reference marks, or incremental HEIDENHAIN rotary encoders

Control inputs	• 4 inputs for position measuring systems (4 sinusoidal inputs) • 1 measuring system input for spindle orientation (square-wave input signal) • 1 input for electronic handwheel • 1 input for 3D-touch probe system • 55 PLC inputs + 1 control-is-ready input • Additionally 64 PLC inputs on optional PLC I/O board PL 410 B
Control outputs	• 5 analogue outputs for the spindle and axes • 31 PLC outputs + 1 control-is-ready output • Additionally 31 PLC outputs on optional PLC I/O board PL 410 B
Integrated PLC	• Programming in the form of a list of instructions, max. 4000 PLC commands • Entry by HEIDENHAIN keyboard or data interface
Supply voltage	NC: 24 Vdc (See Chapter 3, Section 4.1) PLC: 24 Vdc
Power consumption	NC: approx. 27 W (with BE 212 connected) PLC: approx. 48 W (See Chapter 3, Section 4.1) PL 410B: approx. 480 W (See Chapter 3, Section 4.1) BF 110: approx. 33 W
Environmental temperature	• Operating: LE/BE 0 to 45° C (32 to 113° F) BF 110: 0 to 40° C (32 to 104° F) • Storage: -30 to 70° C (-22 to 158° F)
Approximate weight	LE 360 C: 8.0 kg TE 355: 1.6 kg BE 212: 11.0 kg BF 110: 1.7 kg PL 410 B: 1.5 kg HRA 110: 0.7 kg

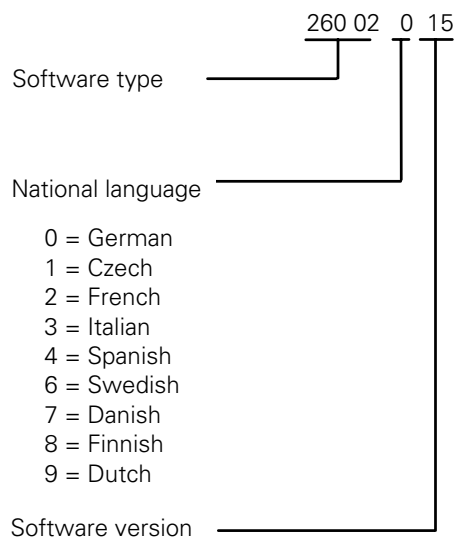
3 Software

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

After switching on the control, the NC and PLC 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

The 8-digit NC software number identifies the type of software, the dialogue language (language of the country) and the software version.



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

3.1.1. Software and hardware versions

HEIDENHAIN has manufactured several different hardware versions of the logic units LE 360 and LE 360 C. The following table shows which software is compatible with which hardware version:

Software Type /Version	Id.-Nr. LE 360			Id.-Nr. LE 360 C	
	258 991 99	264 660 99	264 085 99	270 641 39 (BE 212)	270 642 39 (BF 110)
259 90x	02 to 05	02 to 05	02 to 05	06	-
260 02x (1-MB EPROM)	-	-	from 07	from 07	from 07
280 49x (2-MB EPROM)	-	-	-	from 08	from 08

Only the software types 260 02 and 280 49 will continue to be developed.

Software Releases

HEIDENHAIN releases new versions of NC software in irregular intervals.

NC Software Version	Release
259 90x 02	1/91 (Introduction)
259 90x 03	7/91
259 90x 04	3/92
259 90x 05	3/92
259 90x 06	7/92
260 02x 04	3/92 (Introduction)
260 02x 05	3/92
260 02x 06	7/92
260 02x 07	10/92
260 02x 08	7/93
260 02x 09	9/93
260 02x 10	Never released
260 02x 11	1/94
260 02x 12	2/94
260 02x 13	6/94
260 02x 14	6/94
260 02x 15	7/94
260 02x 16	3/95
280 49x 08	7/93 (Introduction)
280 49x 09	9/93
280 49x 10	Never released
280 49x 11	1/94
280 49x 12	2/94
280 49x 13	6/94
280 49x 14	6/94
280 49x 15	7/94
280 49x 16	3/95

3.1.2 Software option

HEIDENHAIN offers the "Digitizing 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 LE 360 logic unit has the variant xxx xxx 79, while the LE 360 has xxx xxx 34. If the software module is installed, the option number 262 351 01 is indicated on the screen under the NC and PLC software numbers.

The **"Digitizing with TS 120"** software option can be retrofitted. The kit is available under Id. Nr. 265 310 01.

The kit contains:

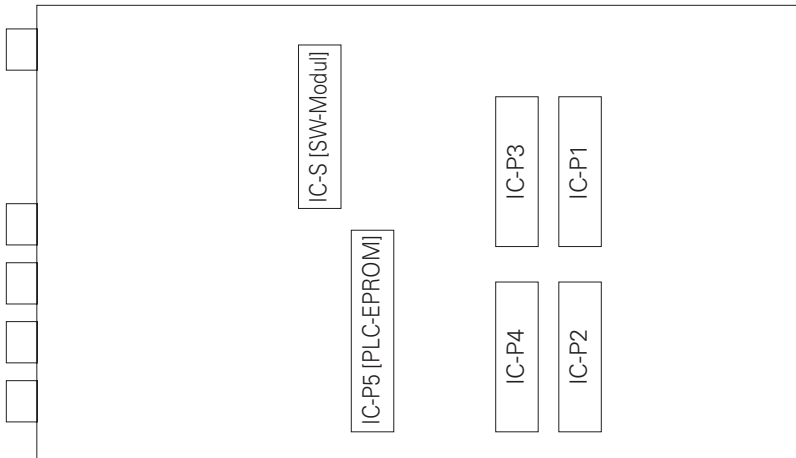
- Software module (EPROM Id.-Nr. 262 351 01)
- Printed circuit board
- Sponge rubber
- ID label
- Mounting Instructions

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.

3.3 EPROM sockets

Sockets for the processor board:



If the 2-MB EPROMs are used, only the sockets IC-P1 and IC-P2 are needed. IC-P3 and IC-P4 remain vacant. The jumper located next to IC-P1 must then be set to position 2M.

Mounting and Electrical Installation – Contents

1	Hardware components TNC 360	3-4
1.1	Changes in the ID-number	3-6
2	Assembly Hints	3-8
2.1	Electrical noise immunity	3-8
2.2	Heat generation and cooling	3-8
2.3	Humidity	3-9
2.4	Mechanical vibration	3-9
2.5	Mounting position	3-9
2.5.1	Logic unit	3-10
2.5.2	Visual display unit (VDU)	3-11
2.5.3	PLC Input/Output board PL 410	3-11
2.6	Degree of protection	3-11
3	Summary of Connections	3-12
4	Power Supply	3-14
4.1	Overview	3-14
4.1.1	NC power supply	3-14
4.1.2	PLC power supply	3-15
4.1.3	Buffer battery	3-16
4.2	Power supply for the visual display unit	3-17
5	Encoders	3-19
5.1	Linear encoders	3-19
5.2	Angular encoders	3-19
5.3	Encoder inputs for sinusoidal signals (7 to 16 mApp)	3-20
5.3.1	Connector assignments	3-20
5.3.2	Connecting cable	3-20
5.4	Encoder inputs for square-wave signals	3-21
5.4.1	Connector assignments	3-21
5.4.2	Connecting cable	3-21
6	Nominal Value Output	3-23
6.1	Connector assignment	3-23
6.2	Connecting cable	3-23
7	Visual Display Unit (VDU)	3-26
7.1	Connector assignment	3-26
7.2	Connecting cable	3-27

8	Touch Probe System Input	3-28
8.1	Connector assignment	3-28
8.2	Connection of the touch probe system	3-29
8.2.1	TS 120	3-29
8.2.2	TS 511	3-30
9	RS-232-C/V.24 Data Interface	3-32
10	Handwheel Input	3-33
10.1	Pin layout	3-33
10.2	Portable handwheel HR 330	3-33
10.3	Integral handwheel HR 130	3-34
10.4	Portable handwheel HR 332	3-35
10.5	HRA 110 handwheel adapter	3-37
11	PLC inputs/outputs	3-40
11.1	Technical data	3-40
11.2	Connector assignment	3-41
11.2.1	PLC inputs	3-41
11.2.2	PLC output	3-42
11.3	Connecting cable	3-43
11.4	PLC I/O expansion-board	3-45
11.4.1	PL 400 connection	3-45
11.4.2	PLC inputs/outputs on the PL 400	3-47
11.5	PL 410 PLC I/O expansion-board	3-49
11.5.1	PLC inputs/PLC outputs on the PL 410	3-50
12	Machine control panel	3-53
12.1	Pin layout	3-53
12.2	Connecting cable	3-54
13	TNC keyboard	3-56
13.1	Pin connections	3-56
13.2	Connecting cable	3-57
14	Dimensions	3-58
14.1	LE 360 C	3-58
14.2	Keyboard units	3-59
14.2.1	TE 355 A	3-59
14.2.2	TE 355 B	3-60
14.3	Visual display units	3-61
14.3.1	BE 212	3-61
14.3.2	BF 110	3-62

14.4	Input/Output units	3-63
14.4.1	PL 410 B	3-63
14.4.2	PL 410	3-64
14.4.3	PL 400	3-65
14.5	Panel-mounted handwheels	3-66
14.5.1	HR 130	3-66
14.5.2	HR 150	3-67
14.5.3	HRA 110 handwheel adapter (for HR 150)	3-68
14.5.4	Handwheel knobs	3-69
14.6	Cable adapter	3-72
15	Grounding Diagram	3-75
16	Cable Overview	3-77

1 Hardware Components TNC 360

The TNC 360 consists of the following hardware components:

- LE 360 C (Logic unit),
- TE 355 A or TE 355 B (TNC keyboard),
- BE 212 or BF 110 (Visual display unit),
- if desired, PL 410 B

The TNC 360 is not subject to export restrictions. An export version is not necessary.

Logic unit

TNC 360 and TNC 335:

LE 360 C

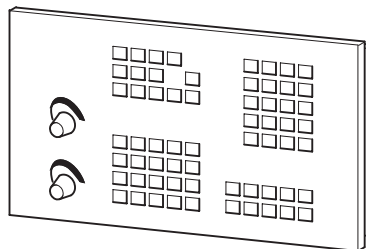
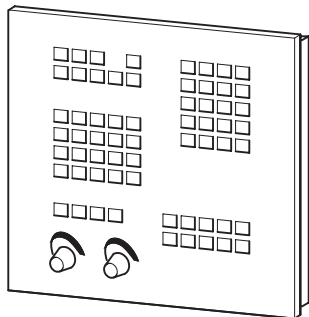
Id.-Nr. 270 641 .. (for connecting the BE 212)

Id.-Nr. 270 642 .. (for connecting the BF 110)

TNC keyboard

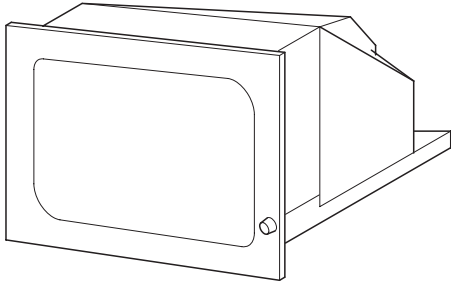
TNC 360: TE 355 A (Id.-Nr. 255 015 01)

TE 355 B (Id.-Nr. 255 016 01)

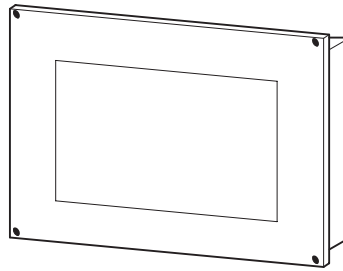


Visual display unit

BE 212 (Id.-Nr. 242 370 01)

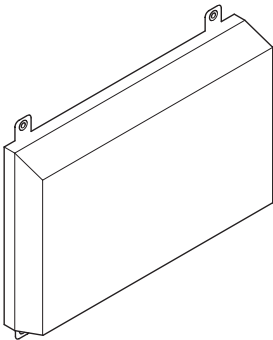


BF 110 (Id.-Nr. 267 209 01)



PLC Input/Output Unit (Option)

PL 410 (Id.-Nr. 263 371 01)



1.1 Changes in the ID Number

LE 360 Logic Unit:

Id.-Nr. 258 991 99	Series introduction	since 1990
Id.-Nr. 258 991 98	Socket for "digitizing" software module	since 1991
Id.-Nr. 258 991 79	Same as Id.-Nr. 258 991 98, but with software module	since 1991
Id.-Nr. 264 660 99	New processor board	since 3/91
Id.-Nr. 264 660 79	Same as Id.-Nr. 264 660 99 but with software module	since 7/91
Id.-Nr. 264 085 99	New PLC- and graphics board	since 7/91
Id.-Nr. 264 085 79	Like Id.-Nr. 264 085 99, but with software module	since 7/91

LE 360 C Logic Unit

Id.-Nr. 270 641 29	For connecting a BE 212	since 1992
Id.-Nr. 270 641 24	With digitizing software module	
Id.-Nr. 270 642 29	For connecting a BF 110	since 1992
Id.-Nr. 270 642 24	With digitizing software module	

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 $\varnothing \geq 6 \text{ mm}^2$ (see Grounding diagram).
- Use of original HEIDENHAIN cables, connectors and couplings.

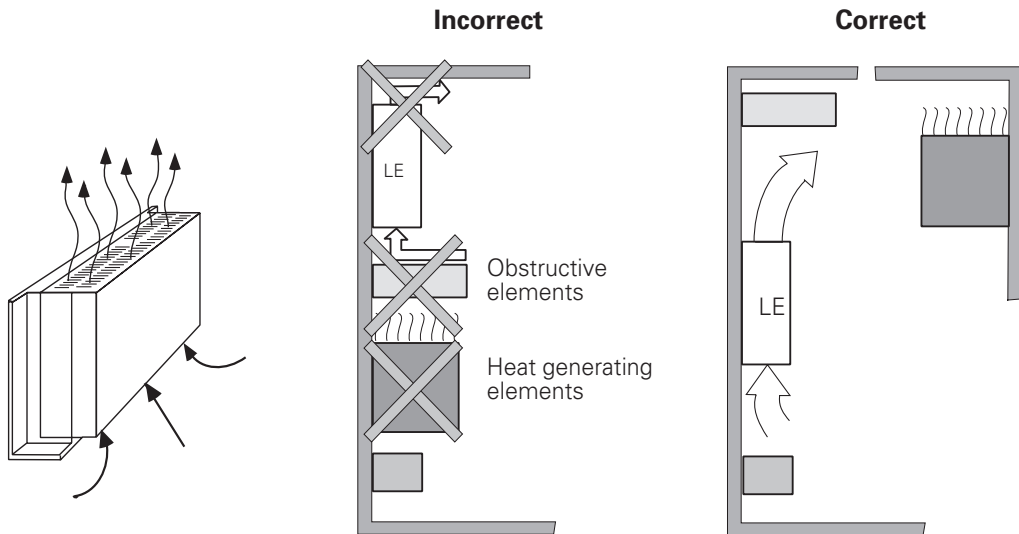
2.2 Heat generation and cooling

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

Permissible ambient temperature in operation: 0° C to 45° C (BF 110: 0° C to 40° 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, vapors 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

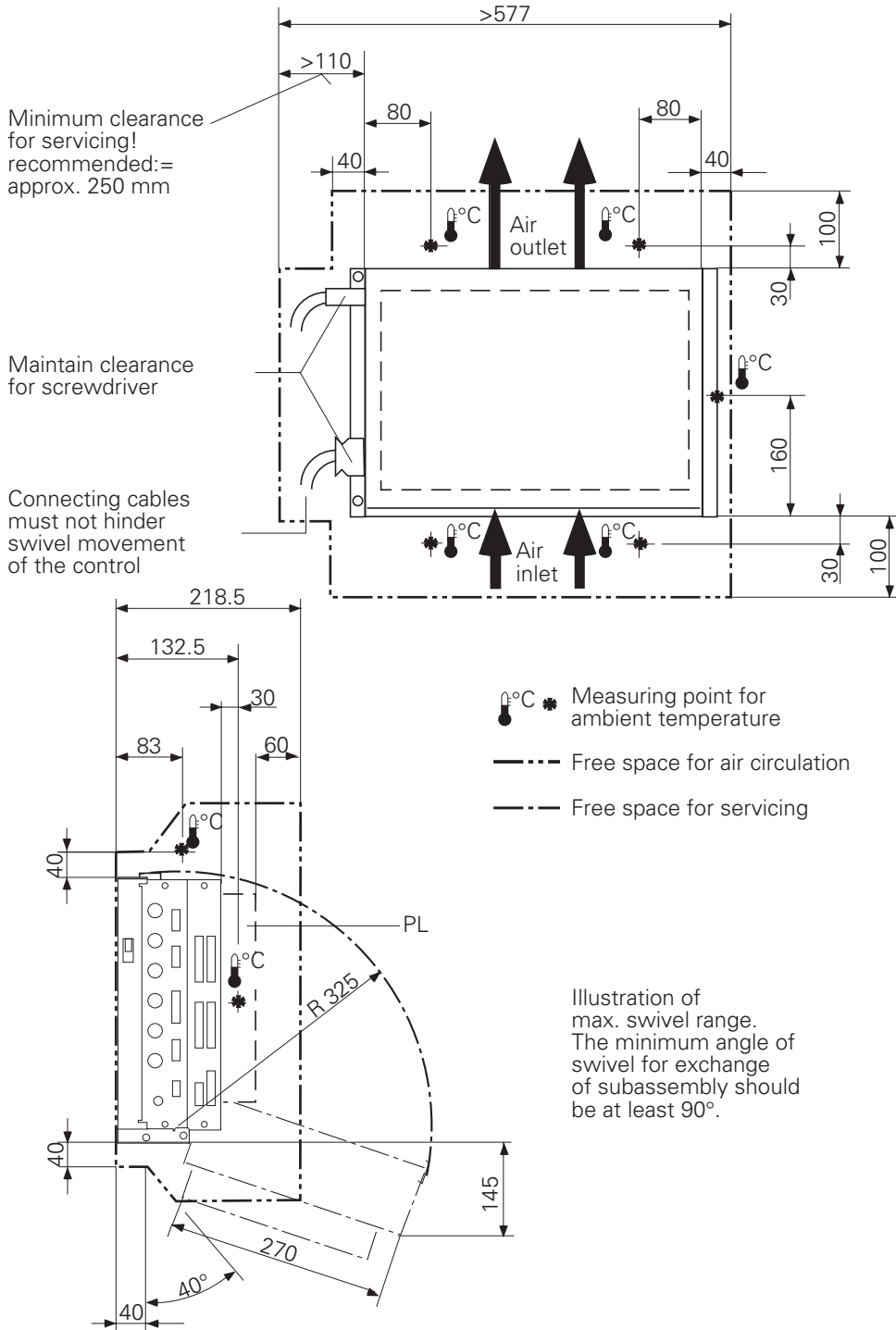
2.5 Mounting position

Note the following fundamental points on mounting:

- Mechanical accessibility
- Permissible environmental conditions
- Electrical noise immunity
- The electrical regulations in effect in your country

2.5.1 Logic unit

HEIDENHAIN recommends the following mounting position:



2.5.2 Visual display unit (VDU)

Permissible ambient temperature

BE 212 max. 45° C (113° F)

BF 110 max. 40° C (104° F)

The VDU must be installed with a minimum clearance of 25 mm to the housing. It is recommended especially, that a large area is left free above the unit for heat removal.

Temperature is measured at a distance of 25 mm to the housing. The above mentioned temperatures must not be exceeded.



Please also note for the BE 212:

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.)

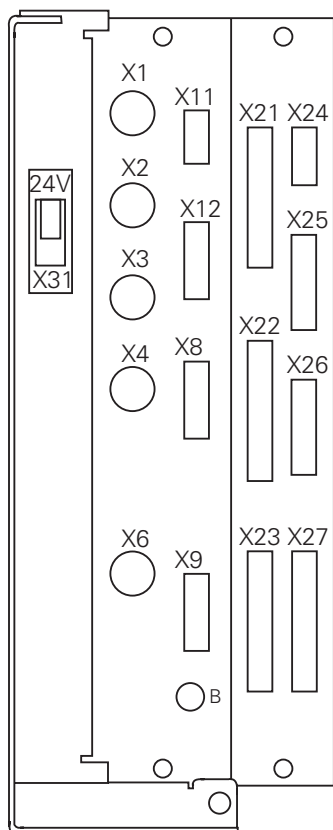
2.5.3 PLC Input/Output board PL 410

One PL 410 can be connected to the LE 360 C, if desired. There is no preferred mounting position for heat removal.

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



Control loop 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 = Touch probe system
- X8 = Nominal value outputs 1,2,3,4,S
- X9 = VDU
- X11 = HR 130/330/332 handwheels,
HRA 110

PLC and graphics board

- X21 = PLC output
- X22 = PLC input
- X23 = TNC keyboard (TE)
- X24 = Power supply 24 V for PLC
- X25 = Data interface RS-232-C/V.24
- X26 = Input/Output board PL 410
- X27 = Machine operating panel

X31 = Power supply 24 V for NC

B = Signal ground



Danger to internal components!

Do not engage or disengage any connections while the unit is under power.

4 Power Supply

4.1 Overview

The supply voltages must meet the following specifications:

Unit		Supply voltage	Voltage range dc average	Max. current consumption	Power consumption
LE	NC	24 Vdc (VDE 0160, 5.88 low-voltage electrical separation)	Lower limit 20.4 V $\overline{\text{---}}$	LE 360 1.5 A	28.8 ... 36 W (also supplies the BE 212)
	PLC	24 Vdc (VDE 0160, 5.88 base insulation)	Upper limit 31 V $\overline{\text{---}}$ ¹⁾	2 A when half ³⁾ of the outputs are switched on simultaneously	Approx. 48 W when half ³⁾ of the outputs are switched on simultaneously
	PL 410 B PL 410 ⁴⁾ PL 400 ⁴⁾			20 A when half ³⁾ of the outputs are switched on simultaneously	Approx. 480 W when half ³⁾ of the outputs are switched on simultaneously
HRA 110		(VDE 0160, 5.88 low-voltage electrical separation)		0.2 A	Approx. 5 W
BF 110 ²⁾				Approx. 1 A with full display	24 W typical 32 W max.

¹⁾ Voltage surges up to 36 V $\overline{\text{---}}$ permissible for $t < 100$ ms.

²⁾ The BF 110 has its own line power connection, which also powers the internal ventilation.

³⁾ No more than half the outputs can be switched on simultaneously.

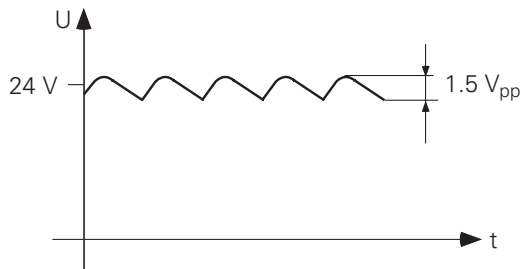
⁴⁾ No longer included in product program

4.1.1 NC power supply

Connection terminal X31

Pin Number	Assignment
1	+ 24 Vdc
2	0 V

The NC and the HRA 110 must not be supplied from the machine control voltage supply! It requires an individual, external and separately generated supply voltage according to VDE 0551. Use 24 Vdc with a permissible ac component of 1.5 Vpp (recommended filter capacitor 10 000 μ F/40 Vdc).



Danger to internal components!

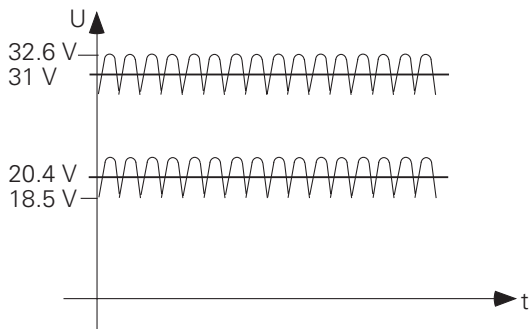
The internal fuses of the power supply assembly must be exchanged only by HEIDENHAIN personnel.

If the operating voltage of the control (5V on the processor board) is outside the limit values a blinking error message "PROCESSOR CHECK ERROR M" is displayed.

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.

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). This means an absolute upper voltage limit of 32.6 V and an absolute lower voltage limit of 18.5 V.



X24 power supply for the PLC at the LE

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



Danger to internal components!

Use only original replacement fuses.

Power supply for the PL 410 B

The PLC outputs are powered in groups.

Terminal	Assignment	PLC output
X9	0V	
X10	+24 V power for logic and for "Control is operational"	
X11	+24 V power for outputs	O32 to O39
X12		O40 to O47
X13		O48 to O55
X14		O56 to O62

Power supply for the PL 410

Connections as with PL 410 B.

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

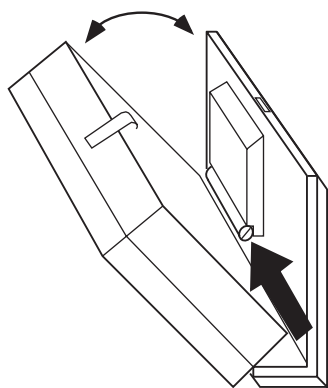
4.1.3 Buffer battery

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 Power supply for the visual display unit

BE 212

The BE 212 visual display unit is powered through the LE (connector X9).

BF 110

X1 power supply

Pin Number	Assignment
1	+ 24 V
2	0 V



The BF 110 must not be supplied with 220 V!

5 Encoders

The HEIDENHAIN contouring controls are designed for the installation of incremental linear and angular encoders.

The control controls the actual position with a measuring step of 0.001 mm or 0.001°. Encoders with a graduation period of 0.001 mm or 0.001° to 1 mm or 1° may be used.

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

5.1 Linear encoders

Measurement of length is best performed by a linear 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 encoders:

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 Angle encoders

For direct angular measurement in the A, B or C axes the following incremental angle encoders 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 Encoder inputs for sinusoidal signals (7 to 16 μA_{pp})

The LE can have encoders with sinusoidal inputs (7 μA_{pp} to 16 μA_{pp}).
Maximum input frequency is 30 kHz.

5.3.1 Connector assignments

X1, X2, X3, X4 encoder 1, 2, 3, 4

Flange socket with female connector insert (9-pin)

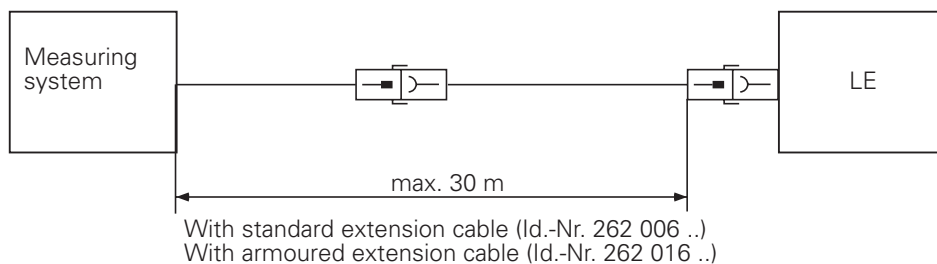
Pin No.	Assignment
1	I_{1+}
2	I_{1-}
5	I_{2+}
6	I_{2-}
7	I_{0+}
8	I_{0-}
3	+ 5 V (U_P)
4	0 V (U_N)
9	Inner shield
Housing	Outer shield = unit housing



The interface complies with the recommendations in VDE 0160, 5.88 for separation from line power.

5.3.2 Connecting cable

Please use only HEIDENHAIN encoder cables, connectors and couplings. Standard HEIDENHAIN extension cables cover a maximum distance of 30 m.



A connecting cable with power supply lines $\varnothing 1 \text{ mm}^2 \text{ PUR } [3 (2 \times 0.14) + (2 \times 1.0) \text{ mm}^2]$,
Id.-Nr. 244 955 01 can have a maximum length of 60 m.

5.4 Encoder inputs for square-wave signals

One encoder with square-wave signals can be connected at the input X6. Maximum input frequency is 300 kHz.

5.4.1 Connector assignments

X6 encoder 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	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



The interface complies with the recommendations in VDE 0160, 5.88 for separation from line power.

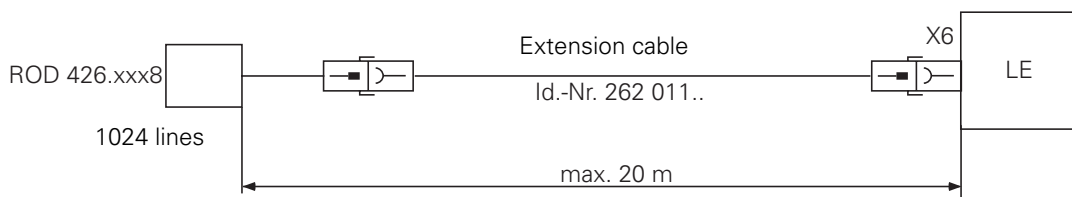
5.4.2 Connecting cable

Please use only HEIDENHAIN encoder cables, connectors and couplings.

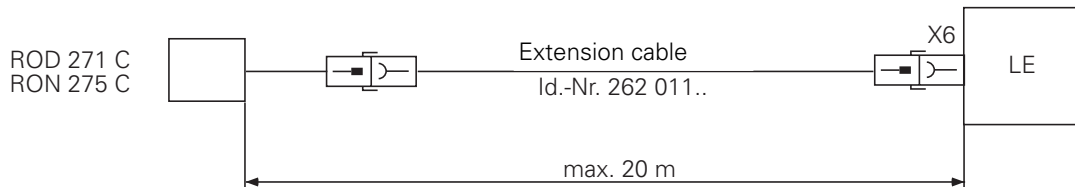
In order to be able to connect an encoder to the square-wave signal input of the logic unit, the sinusoidal signal from the encoder must be converted to a square-wave signal. This conversion is performed by the interpolation and digitizing electronics (EXE). The interpolation and digitizing electronics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

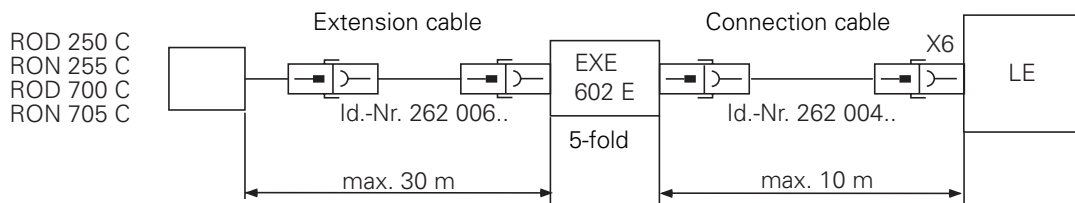
Spindle orientation:



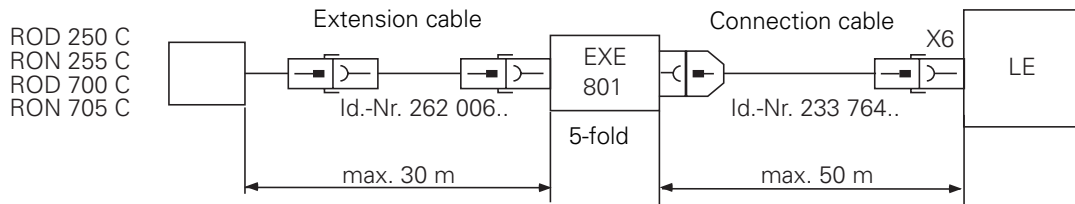
Angle encoders:



or:



or:



If necessary, linear encoders can also be connected to the LE via interpolation and digitizing electronics.

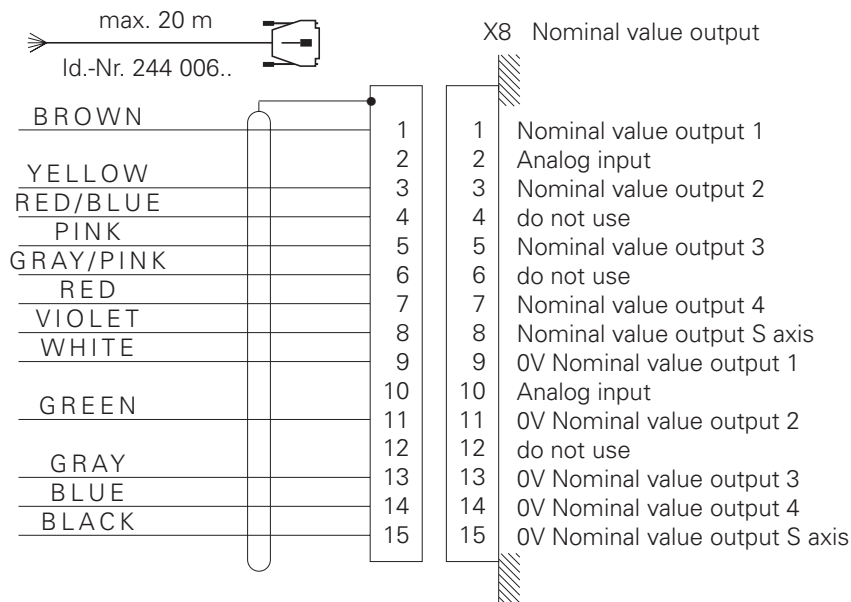
6 Nominal Value Output

The 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

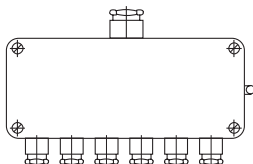


The interface complies with the recommendations in VDE 0160, 5.88 for separation from line power.

6.2 Connecting cable

HEIDENHAIN offers a connecting cable with a connector at one end (Id.-Nr. 244 006 ..).

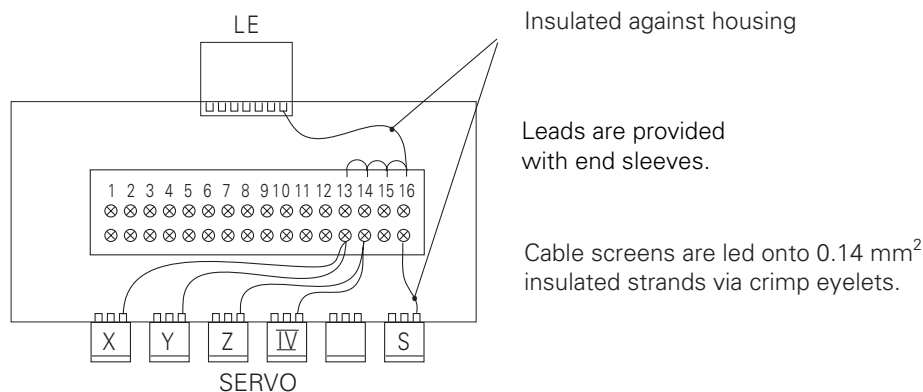
The connecting cable to the nominal value outputs may not have more than one intermediate terminal clamp. The clamp must be made in an earthed connection box. This is necessary when the cable must branch to physically separate servo inputs. It is only possible to earth the screening of the servo leads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



Connection box

The housing 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 $\geq \varnothing 6 \text{ mm}^2$, see also under "Grounding diagram").

Suggested solution for connecting and wiring the screening in the connection box:



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

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

7 Visual Display Unit (VDU)

The LE 360 C is factory-prepared for connecting the BE 212 CRT screen or BF 110 flat panel display. The status is indicated by the Id.-Nr. on the ID label (Id.-Nr. 270 641 for BE 212 and Id.-Nr. 270 642 for BF 110).

7.1 Connector assignment

Both display units are connected to the socket X9 on the LE 360 C, but with different cables.

X9 Visual display unit

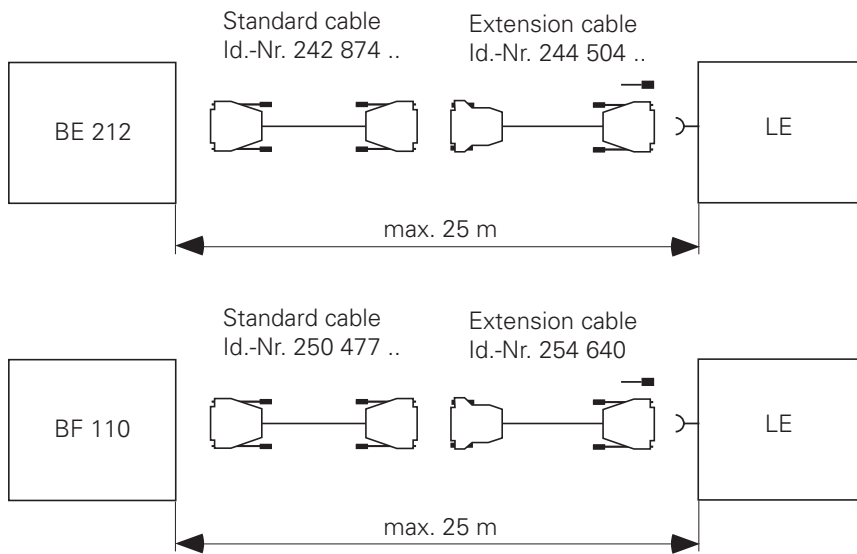
D-sub connection (female 15-pin)

Pin Number	BE 212	BF 110
1,8	0 V power supply	–
2,4	+ 12 V power supply	–
3, 5, 6	Do not use	Do not use
7	–	Video
9	V SYNC	V SYNC
10	H SYNC 1	–
11	0 V Signal	0 V Signal
12	–	–
13	Video	–
14		H SYNC 2
15		CLOCK
Housing	Outer screen = Unit housing	



The interface complies with the recommendations in VDE 0160, 5.88 for separation from line power.

7.2 Connecting cable



8 Touch Probe System Input

The 3D touch probe systems from HEIDENHAIN can be delivered in two versions:

- 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

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

8.1 Connector assignment

X12 Touch probe system TS 120/TS 511

Flange socket with female connector insert (15-pin)

Pin Number	Signal designation
1	Inner screen (0 V)
3	Ready/standby
4	Start
5	+15 V $\pm$ 10 % (U _P)
6	+ 5 V $\pm$ 5 % (U _P)
7	Battery warning
8	0 V (U _N)
9	Trigger signal
10	Trigger signal ²
2, 11 to 15	Do not use

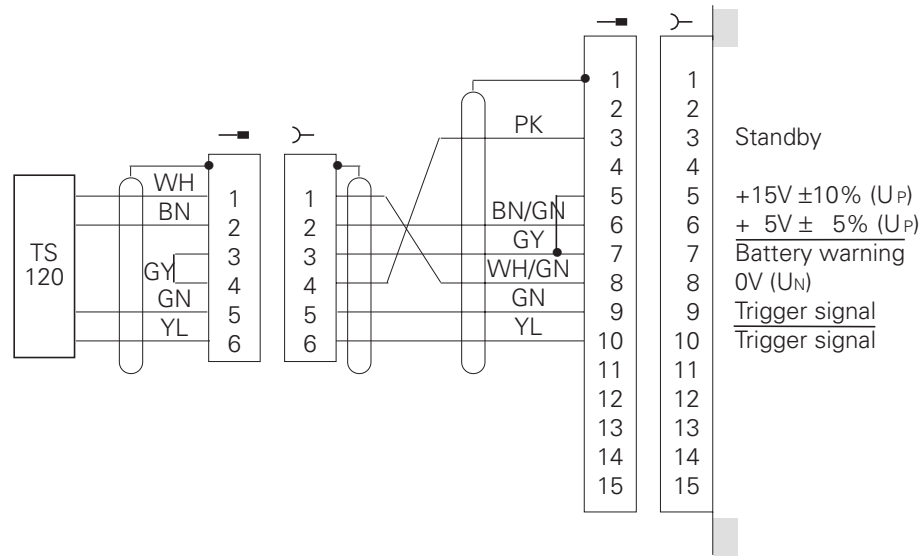
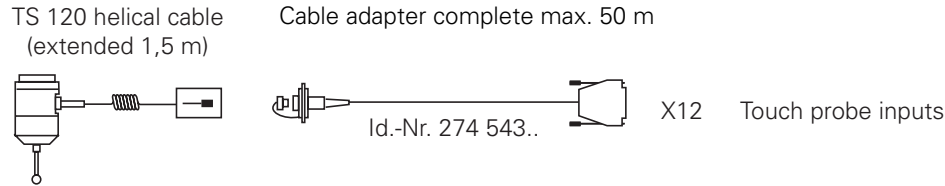
² Stylus in rest position = signal high

8.2 Connection of the touch probe system

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

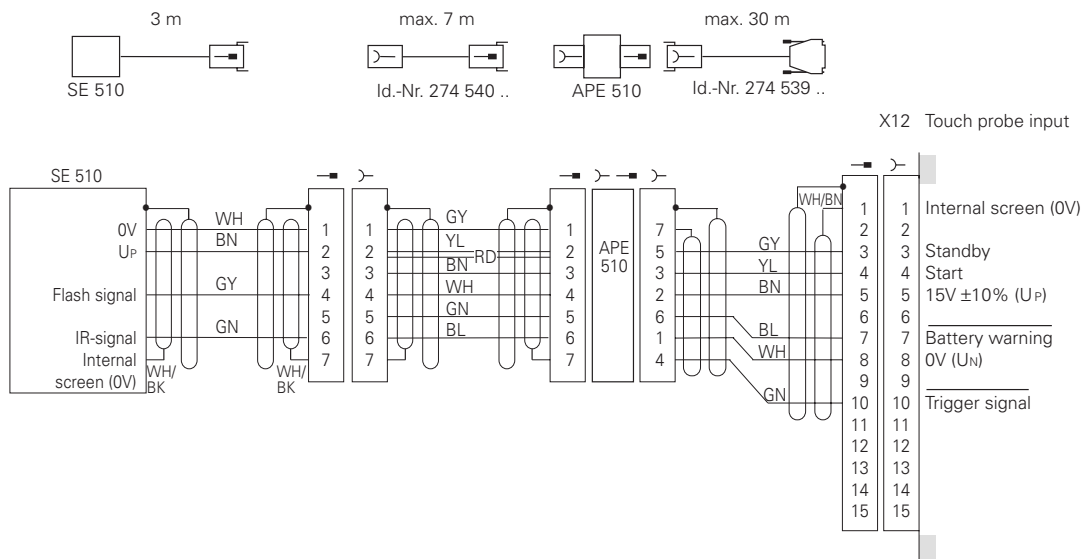
8.2.1 TS 120

The touch probe system TS 120 is connected directly to the logic unit via a cable adapter. See also Section "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 the operating instructions for the 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 ($\geq \varnothing 6 \text{ mm}^2$).

See also Section "Earthing plan".

9 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.

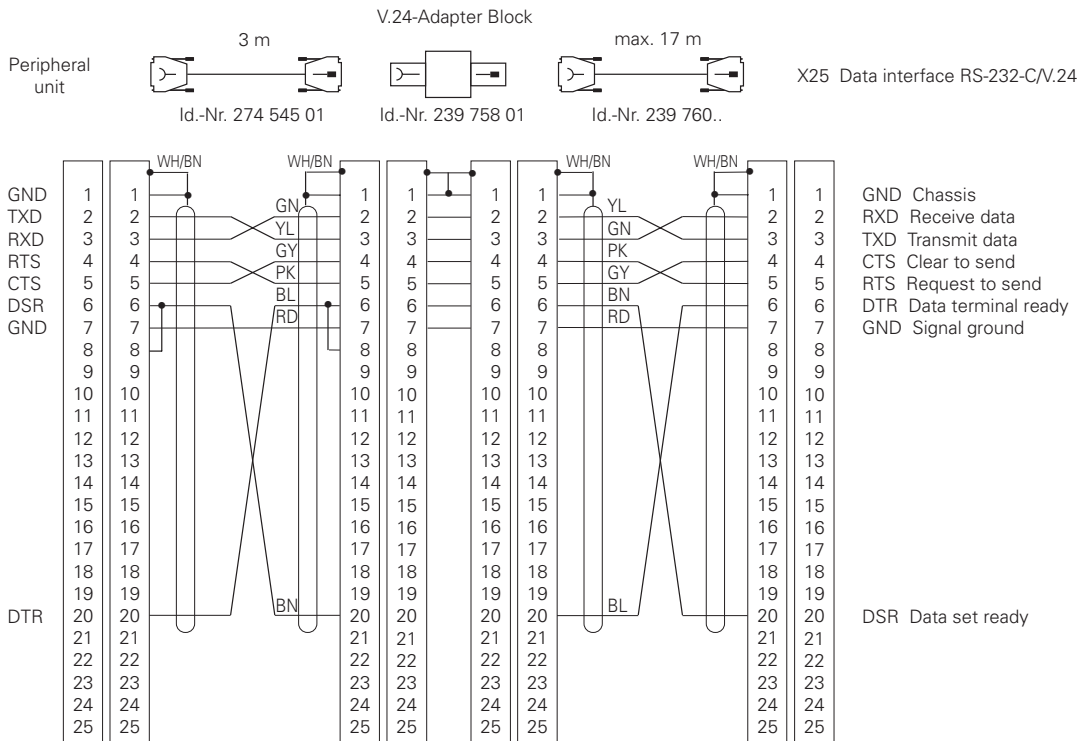
The connection to the peripheral unit is made via a cable adapter which is attached to either the operating console or the control cabinet. See also the section "Mounting dimensions". For connection to the peripheral unit, HEIDENHAIN offers a standard connecting cable (Id.-Nr. 274 545 01) , length 3 m.

X25 V.24/RS-232-C

D-sub connection (female 24-pin)

Pin Number	LE 360 C
1	Chassis GND
2	RXD
3	TXD
4	CTS
5	RTS
6	DTR
7	Signal GND
20	DSR
9 - 19, 21-24	Do not use

The interface complies with the recommendations in VDE 0160, 5.88 for separation from line power.



10 Handwheel Input

The handwheel HR 130 or HR 330 or HR 332 or the handwheel adapter HRA 110 can be attached to the HEIDENHAIN contouring control TNC 360. See also Chapter "Machine Integration".

10.1 Pin layout

X11 Handwheel input

D-sub connection (female insert, 9-pin)

Pin Number	LE 360 C
1	CTS
2	0 V (U_N)
3	RTS
4	+ 12 V (U_P)
5	Do not use
6	DTR
7	TxD
8	RxD
9	DSR



The interface complies with the recommendations in VDE 0160, 5.88 for separation from line power.

10.2 Portable handwheel HR 330

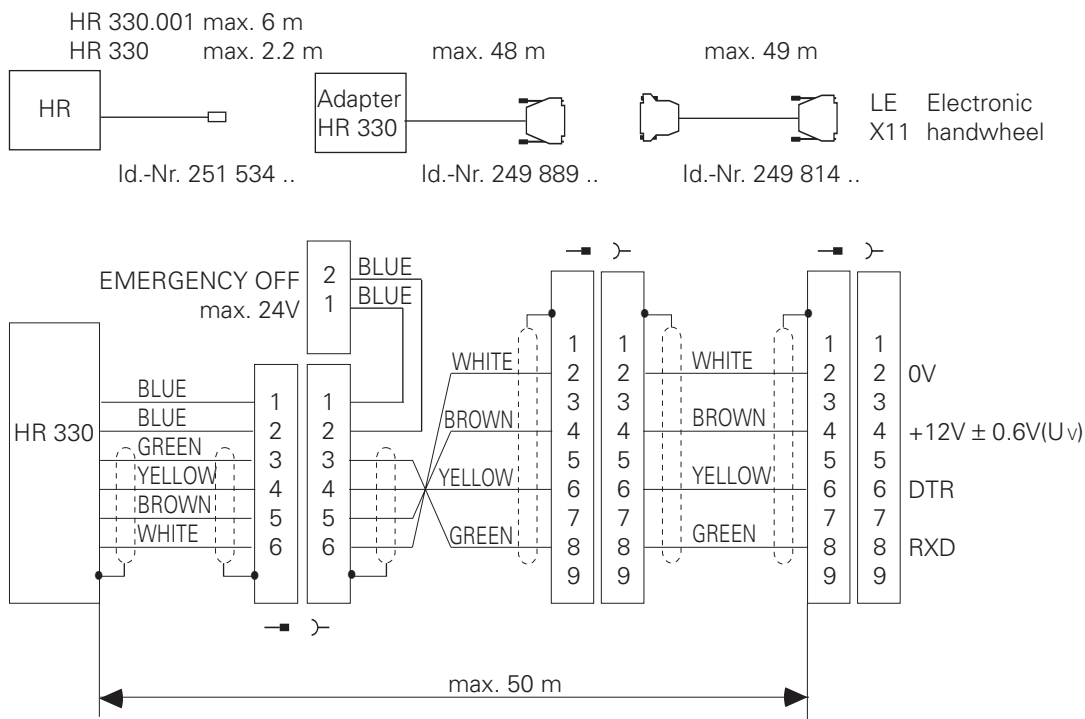
The HR 330 is a portable handwheel with keys for the 4 axes, rapid traverse, direction of traverse and EMERGENCY STOP.

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

The HEIDENHAIN extension cable Id.-Nr. 249 814 .. 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 02
– 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 for the logic unit, and two terminals for connecting the 24 V of the EMERGENCY STOP control circuit (max. load 1.2 A).

See also section "Mounting dimensions".

10.3 Integral handwheel HR 130

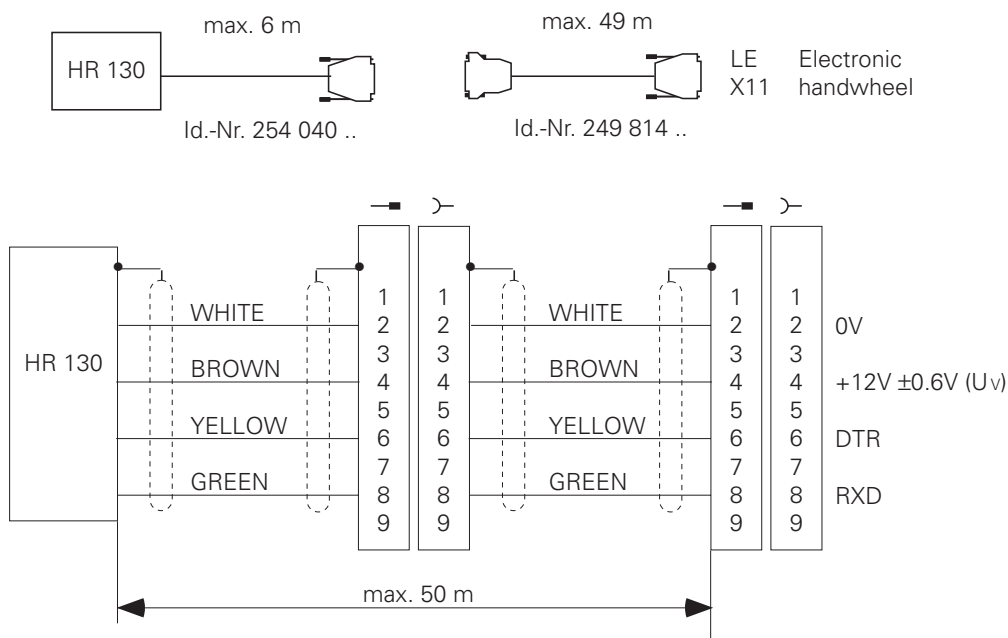
The HR 130 is the integral version of the HR 330, without the keys for the axes, rapid traverse etc.

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

Mounting dimension drawings are provided at the end of this chapter.



10.4 Portable handwheel HR 332

The HR 332 handwheel features two permission buttons and a keypad that can be evaluated via PLC (see Section "Handwheel" in Chapter "Machine Integration"). There are several variants of the handwheel with different labeling. Please contact HEIDENHAIN for more information.

The HR 332 is connected to a cable adapter on the panel through a connecting cable of 5 m length. The connecting cable is available with or without metal armor tubing. The maximum cable length from the HR to the LE is 50 m.

HR 332

Id.-Nr. 266 064 21

Connecting cable to HR 332

Id.-Nr. 272 291 ..

Connecting cable to HR 332, with metal armor

Id.-Nr. 272 292 ..

Cable adapter HR 332/ LE

Id.-Nr. 274 556 ..

Extension cable for cable adapter

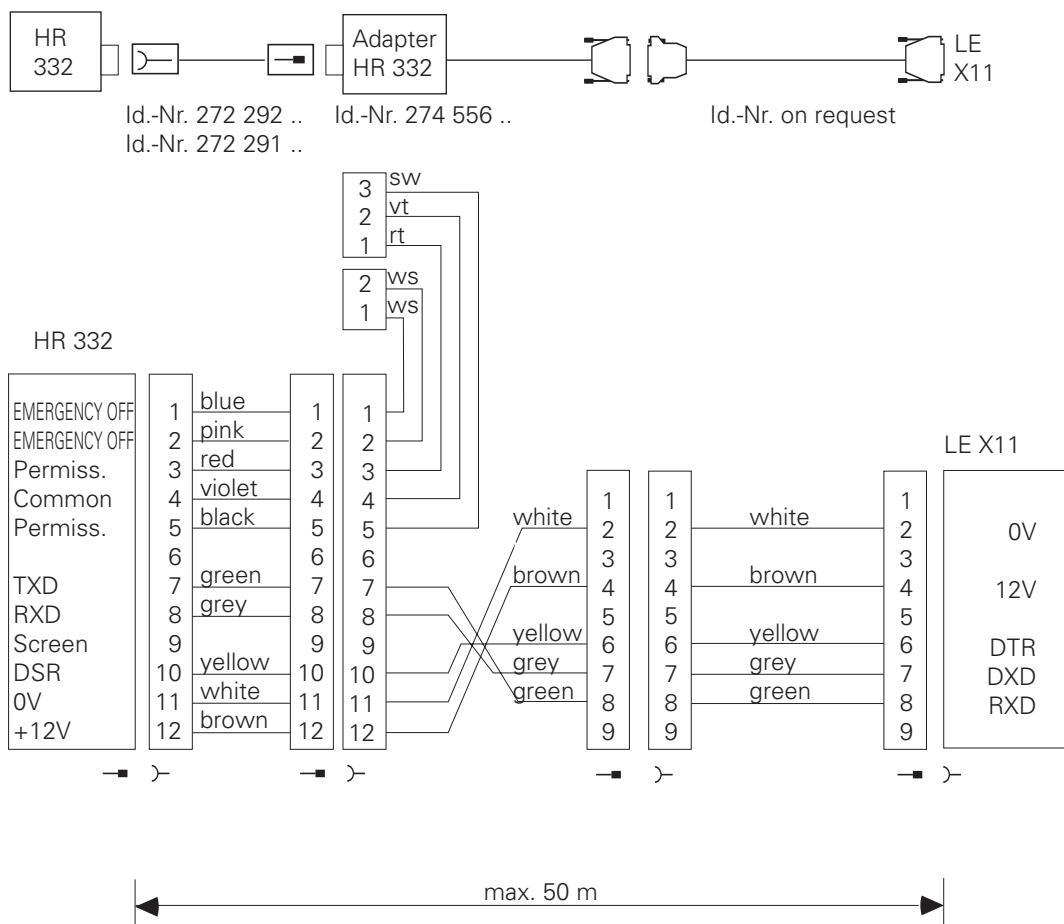
In development

Dummy plug for EMERGENCY STOP circuit

Id.-Nr. 271 958 01

Handwheel holder

Id.-Nr. 268 268 03

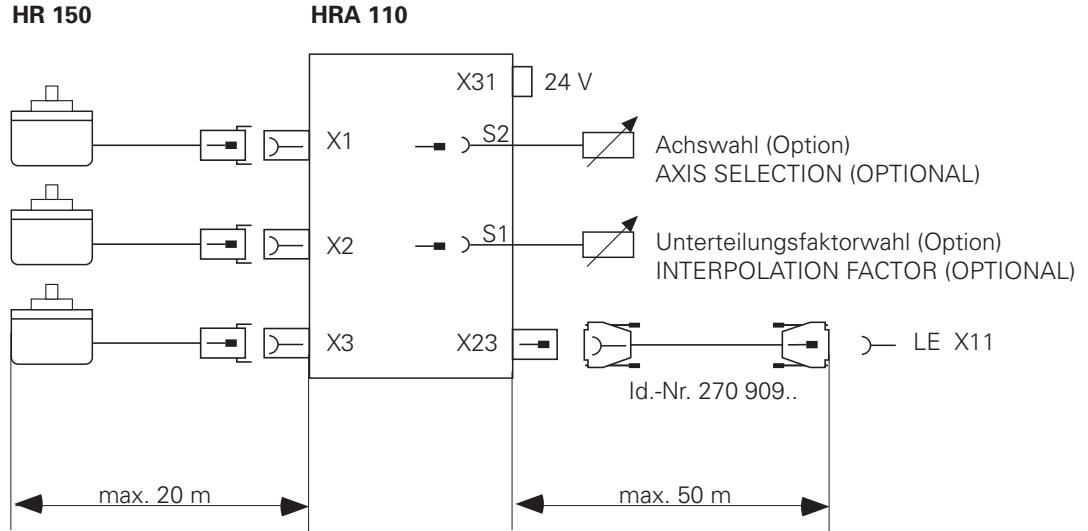


The adapter has a cable with 9-pin connector for the logic unit, two connecting terminals for the 24 V of the EMERGENCY OFF circuit of the control (max. load 1.2 A) and 3 connecting terminals for the permission switch circuit. The permission keys are make-contacts (24 V/1.2 A).

10.5 HRA 110 Handwheel Adapter

The HRA 110 permits two or three HR 150 handwheels to be connected to an LE.

The first two handwheels are permanently assigned to the X and Y axes. The third handwheel can be assigned to the X, Y, Z or IV axes either through an optional step switch or via machine parameter. (See Section "Handwheel" in Chapter "Machine Integration").



Another step switch (optional) is available for selecting the interpolation factor of the handwheels. The interpolation factor of the switch must be evaluated in the PLC and is displayed on the TNC screen, but it cannot be changed via TNC keyboard. If the handwheels are mounted without the step switch, however, the interpolation factor is selected for each axis via TNC keyboard.

Pin layout

Handwheel inputs X1, X2, X3

Pin Number	Assignment
1	I ₁ +
2	I ₁ -
5	I ₂ +
6	I ₂ -
7	I ₀ +
8	I ₀ -
3	+ 5 V (U _P)
4	0 V (U _N)
9	Inner screen
Housing	Outer screen

Handwheel adapter output X23

Pin layout

Pin Number	Assignment
1	RTS
2	0 V (U_N)
3	CTS
4	+12V (U_P)
5	Do not use
6	DSR
7	RxD
8	TxD
9	DTR

Direct current connection X31

Pin layout

Pin Number	Assignment
1	+ 24 V
2	0 V

The 24 V power supply of the PLC must not be connected to the handwheel adapter, since this would cancel the electrical isolation of the PLC inputs and outputs. The handwheel adapter must be supplied by the 24V supply of the NC (low voltage electrical separation according to VDE 0160, 5.88).

11 PLC inputs/outputs

The HEIDENHAIN contouring control TNC 360 C has a capacity of max. 119 PLC inputs and 62 PLC outputs.

55 PLC inputs and 31 PLC outputs can be connected directly to the logic unit. One PLC I/O board (model PL 410 B or PL 410), with 64 PLC inputs and 31 PLC outputs, can be connected to the logic unit.

11.1 Technical data

PLC inputs

	Logic unit/ PL 410	PL 410 B	PL 400
Potential range			
"1" signal: U_i	13 V to 30.2 V	13 V to 30.2 V	16.5 V to 30 V
"0" signal: U_i	-20 V to 3.2 V	-20 V to 3.2 V	-20 V to 4 V
Current range			
"1" signal: I_i	3.8 mA to 8.9 mA	2.5 mA to 6 mA	6.2 mA to 12.6 mA
"0" signal: I_i	1.0 mA at $U_i = 3.2$ V	0.65 mA at $U_i = 3.2$ V	1.6 mA at $U_i = 4$ V

PLC outputs

Transistor outputs with current limiting

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

Permissible load: resistive load, inductive load only with a quenching diode parallel to the inductance. No more than half of the outputs may be switched on at simultaneously.

It is not permissible to short-circuit several outputs from the logic unit simultaneously. A **single** short circuit does not cause an overload.

11.2 Connector assignment

11.2.1 PLC inputs

The PLC inputs I128 to I151 are on connector X27 for the machine control panel.

X22 PLC inputs

D-sub connector (37-pin female insert)

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
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 is operational" output are on connector X21 (PLC output). The PLC outputs O0 to O7 are also to be found on the connector for the machine control panel (X27).

See also Section "Machine control panel".

X21 PLC output

D-sub connector (37-pin female insert)

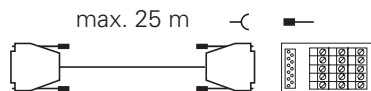
Pin Number	Assignment
1	O0
2	O1
3	O2
4	O3
5	O4
6	O5
7	O6
8	O7
9	O8
10	O9
11	O10
12	O11
13	O12
14	O13
15	O14
16	O15
17	O16
18	O17
19	O18
20	O19
21	O20
22	O21
23	O22
24	O23
25	O24 ¹⁾
26	O25 ¹⁾
27	O26 ¹⁾
28	O27 ¹⁾
29	O28 ¹⁾
30	O29 ¹⁾
31	O30 ¹⁾
32	Do not use
33	0 V (PLC) ¹⁾
34	Control operational
Housing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

¹⁾ Cannot be switched off via EMERGENCY STOP

11.3 Connecting cable

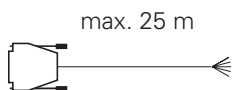
Please use only HEIDENHAIN connecting cables.

The length of cable between the logic unit and the element to be switched is limited to 25 m. HEIDENHAIN recommends building a transfer unit with a terminal strip into the switch cabinet. This transfer unit can be connected to the logic unit with the HEIDENHAIN connecting cable Id.-Nr. 244 017 ..



Id.-Nr. 244 017..

If operation is to be without transfer unit, then the HEIDENHAIN connecting cable Id.-Nr. 244 005 .. can be used.



Id.-Nr. 244 005..

Terminal color assignments:

1 = GY/RD	14 = GN/BL	27 = YL/BK
2 = BN/BK	15 = YL	28 = WH/YL
3 = WH/BK	16 = RD	29 = GY/BL
4 = GN/BK	17 = GY	30 = PK/BL
5 = BN/RD	18 = BL	31 = PK/RD
6 = WH/RD	19 = PK	32 = BN/BL
7 = WH/GN	20 = WH/GY	33 = PK/GN
8 = RD/BL	21 = YL/GY	34 = BN
9 = YL/RD	22 = GN/RD	35 = YL/PK
10 = GY/PK	23 = WH/PK	36 = VI
11 = BK	24 = GY/GN	37 = WH
12 = PK/BN	25 = YL/BN	
13 = YL/BL	26 = GY/BN	

If the connector is to be assembled in the field, a 37-pin solderable connector is available from HEIDENHAIN (Id.-Nr. 243 937 ZY).

11.4 PLC I/O unit

One PL 400 board with 63 PLC inputs, 31 PLC outputs and the "Control is operational" output can be connected to the logic unit. The PL 400 can be mounted directly on the logic unit. See Section "Power supply" for the power connection.

11.4.1 PL 400 connection

Pin connections on the logic unit

X26 PLC I/O board (PL 400)

D-sub connector (24-pin male insert)

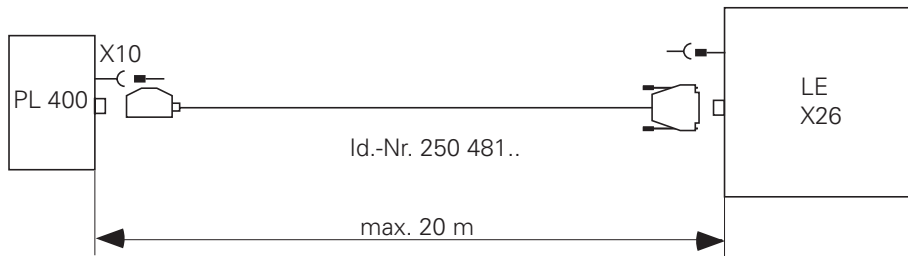
Pin Number	Assignment
1, 2, 3	0 V
4	Serial IN 2
5, 6, 17, 18	Do not use
7	$\overline{\text{RESET}}$
8	$\overline{\text{WRITE EXTERN}}$
9	$\overline{\text{WRITE EXTERN}}$
10	$\overline{\text{A5}}$
11	$\overline{\text{A3}}$
12	$\overline{\text{A1}}$
13	Screen
14, 15, 16	+ 12 V (from PL 400)
19	Serial IN 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

Pin connections on the PL 400

X10 Connections on the logic unit

Pin connections as for connector X26 on the logic unit

Connecting cable



11.4.2 PLC inputs/outputs on the PL 400

The PLC inputs/outputs on the PL 400 are distributed among nine connectors. Assignments as follows:

X1

Pin Number	Assignment PL 400
1	O32
2	O33
3	O34
4	O35
5	O36
6	O37
7	O38
8	O39
9	O40
10	O41
11	O42
12	Do not use

X2

Pin Number	Assignment PL 400
1	O43
2	O44
3	O45
4	O46
5	O47
6	O48
7	O49
8	O50
9	O51
10	O52
11	O53
12	Do not use

X3

Pin Number	Assignment PL 400
1	O54
2	O55
3	O56 ²⁾
4	O57 ²⁾
5	O58
6	O59 ²⁾
7	O60 ²⁾
8	O61 ²⁾
9	O62 ²⁾
10	Control is operational
11	Do not use
12	+24 V cannot be switched off via ext. EMERGENCY OFF ¹⁾

X4

Pin Number	Assignment PL 400
1	I126
2	I74
3	I73
4	I72
5	I71
6	I70
7	I69
8	I68
9	I67
10	I66
11	I65
12	I64

¹⁾ +24 V must always be connected, even if the outputs are not used.

²⁾ Outputs cannot be switched off via EMERGENCY STOP

X5	
Pin Number	Assignment PL 400
1	I86
2	I85
3	I84
4	I83
5	I82
6	I81
7	I80
8	I79
9	I78
10	I77
11	I76
12	I75

X6	
Pin Number	Assignment PL 400
1	I98
2	I97
3	I96
4	I95
5	I94
6	I93
7	I92
8	I91
9	I90
10	I89
11	I88
12	I87

X7	
Pin Number	Assignment PL 400
1	I110
2	I109
3	I108
4	I107
5	I106
6	I105
7	I104
8	I103
9	I102
10	I101
11	I100
12	I99

X8	
Pin Number	Assignment PL 400
1	I122
2	I121
3	I120
4	I119
5	I118
6	I117
7	I116
8	I115
9	I114
10	I113
11	I112
12	I111

X9	
Pin Number	Assignment PL 400
1	Do not use
2	Do not use
3	Do not use
4	I125
5	I124
6	I123

The connections to the PLC inputs/outputs must be made with leads with a conductor cross-section $\geq \varnothing 0.14 \text{ mm}^2 \text{ Cu}$. The maximum permissible lead length is 20 m.

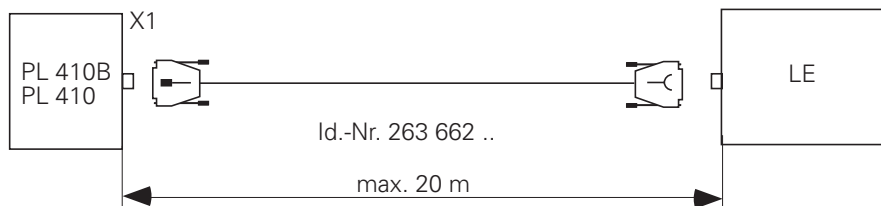
11.5 PL 410 B / PL 410 PLC I/O unit

The PL 410 B / PL 410 has 64 PLC inputs, 31 PLC outputs and one "control-is-operational" output.

In its interface to the LE and its load capacity of inputs and outputs it is identical to the PL 400.

The analog inputs (X15 to X22) of the PL 410 cannot be evaluated in the LE 360 C! The DIL switch on the PCB of the PL 410 must therefore be set to OFF (as is stated on the housing). The PLC outputs are powered in groups and are therefore switched off via EMERGENCY STOP in groups.

Connecting cable



11.5.1 PLC inputs/PLC outputs on the PL 410

The PLC inputs and outputs on the PL 410 are distributed over 6 switches. The 16-pin connectors are arranged in vertical pairs.

PLC inputs

Pin layout

X3

Pin Number	Assignment
1	I64
2	I65
3	I66
4	I67
5	I68
6	I69
7	I70
8	I71
9	I72
10	I73
11	I74
12	I75
13	I76
14	I77
15	I78
16	I79

X5

Pin Number	Assignment
1	I96
2	I97
3	I98
4	I99
5	I100
6	I101
7	I102
8	I103
9	I104
10	I105
11	I106
12	I107
13	I108
14	I109
15	I110
16	I111

X4

Pin Number	Assignment
1	I80
2	I81
3	I82
4	I83
5	I84
6	I85
7	I86
8	I87
9	I88
10	I89
11	I90
12	I91
13	I92
14	I93
15	I94
16	I95

X6

Pin Number	Assignment
1	I112
2	I113
3	I114
4	I115
5	I116
6	I117
7	I118
8	I119
9	I120
10	I121
11	I122
12	I123
13	I124
14	I125
15	I126
16	I127

PLC outputs

Assignment of the grouped power supply:

Terminal	Assignment PL 410 /PL 410 B	
X9	0 V	
X10	+24V PL supply and "Control is ready" (cannot be switched of via EMERGENCY STOP)	
X11	+24 V supply	O32 - O39
X12	+24 V supply	O40 - O47
X13	+24 V supply	O48 - O55
X14	+24 V supply	O56 - O62

Pin Layout

X7

Pin Number	Assignment PL 410/PL 410 B
1	O32
2	O33
3	O34
4	O35
5	O36
6	O37
7	O38
8	O39
9	O40
10	O41
11	O42
12	O43
13	O44
14	O45
15	O46
16	O47

X8

Pin Number	Assignment PL 410/PL 410 B
1	O48
2	O49
3	O50
4	O51
5	O52
6	O53
7	O54
8	O55
9	O56
10	O57
11	O58
12	O59
13	O60
14	O61
15	O62
16	"Control is operational"

12 Machine Control Panel

A separate 37-pin female connector (X27) is mounted on the logic unit for the connection to the manufacturer's proprietary machine control panel. This connector includes the PLC inputs I128 to I151, 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 I151 may be connected only with the power supply from pins 36 and 37, since this power supply is internally secured as required.

12.1 Pin layout

X27 Machine control panel

D-sub connector (37-pin female insert)

Pin Number	Assignment
1	I128
2	I129
3	I130
4	I131
5	I132
6	I133
7	I134
8	I135
9	I136
10	I137
11	I138
12	I139
13	I140
14	I141
15	I142
16	I143
17	I144
18	I145
19	I146
20	I147
21	I148
22	I149
23	I150
24	I151
25	Do not use
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⁾

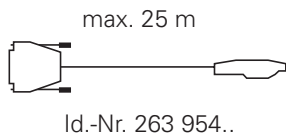
¹⁾ Externally available PLC reference potential for the outputs O0-O7

²⁾ Externally available (via fuse) PLC supply voltage for the inputs.

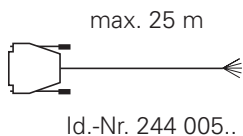
12.2 Connecting cable

Please use only HEIDENHAIN connecting cables.

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



If the machine control 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 Section "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 355 A/B is connected to the logic unit by a connecting cable.

13.1 Pin connections

On the logic unit

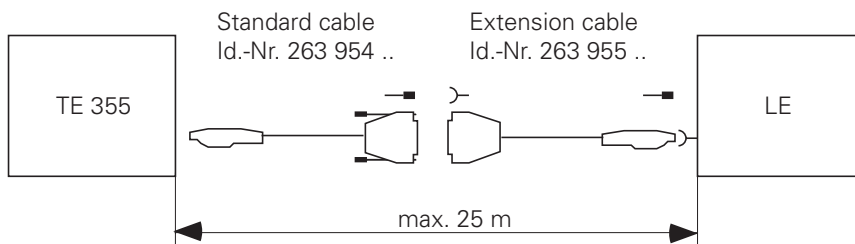
X23 TNC keyboard (TE 355 A/B)

D-sub connector (37-pin female insert)

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

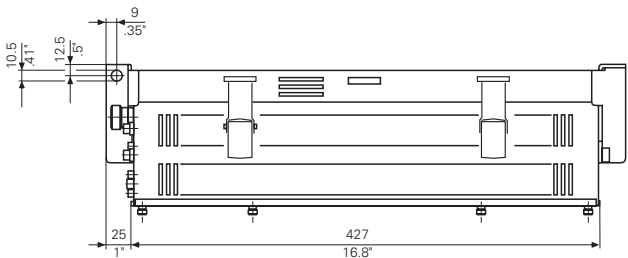
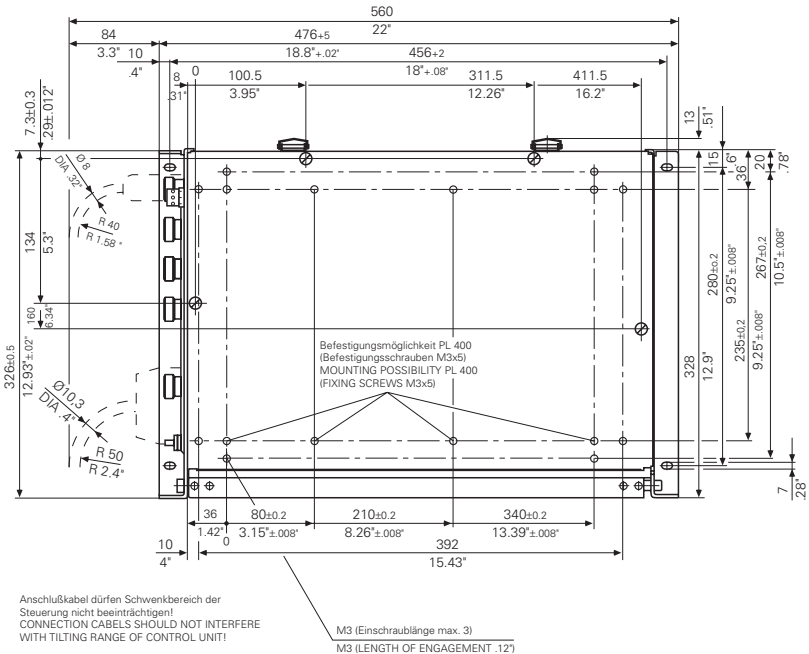
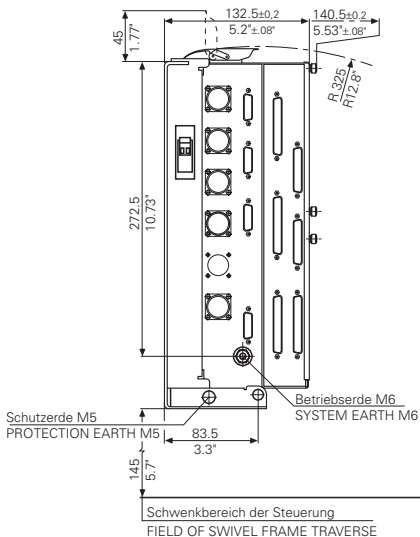
13.2 Connecting cable

Please use only HEIDENHAIN connecting cables.



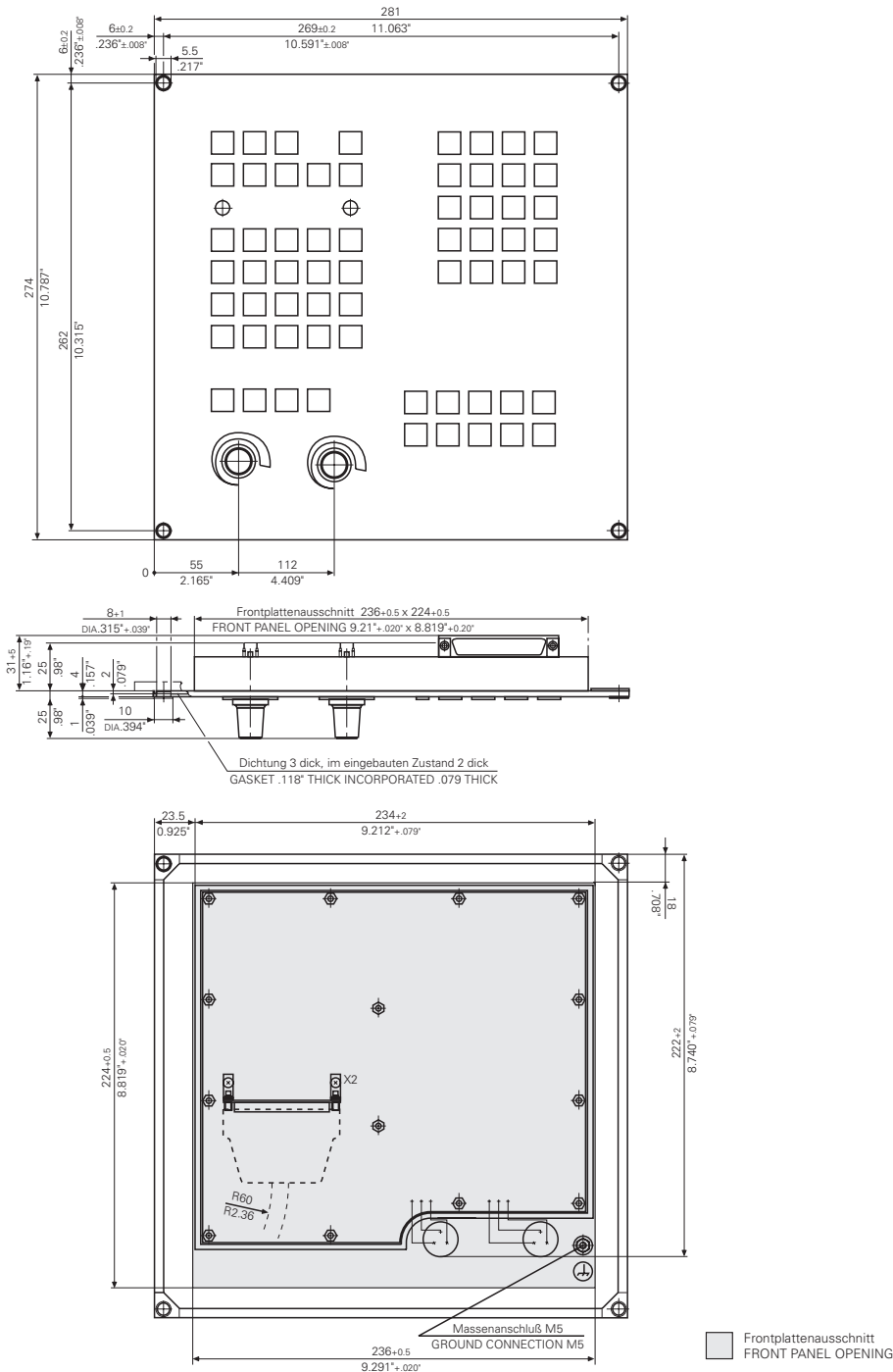
14 Dimensions

14.1 LE 360 C

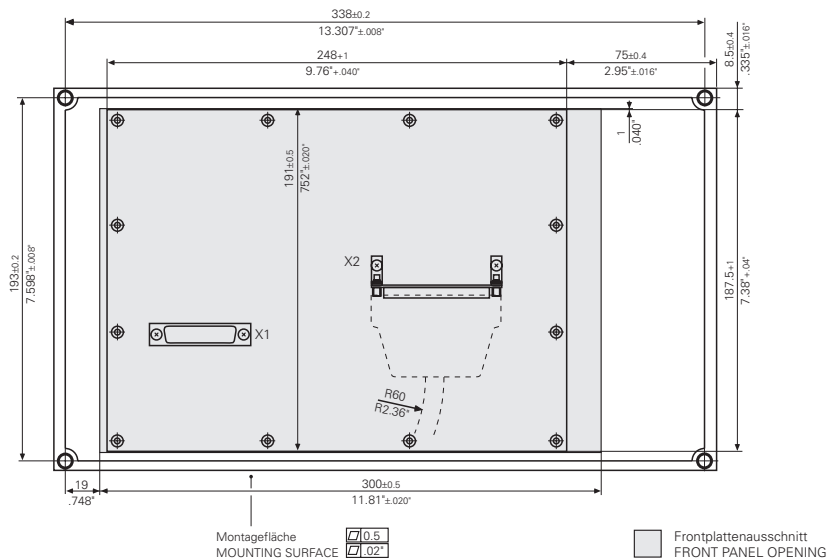
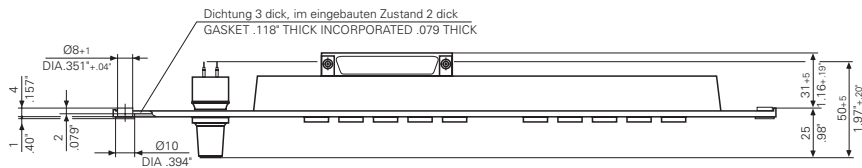
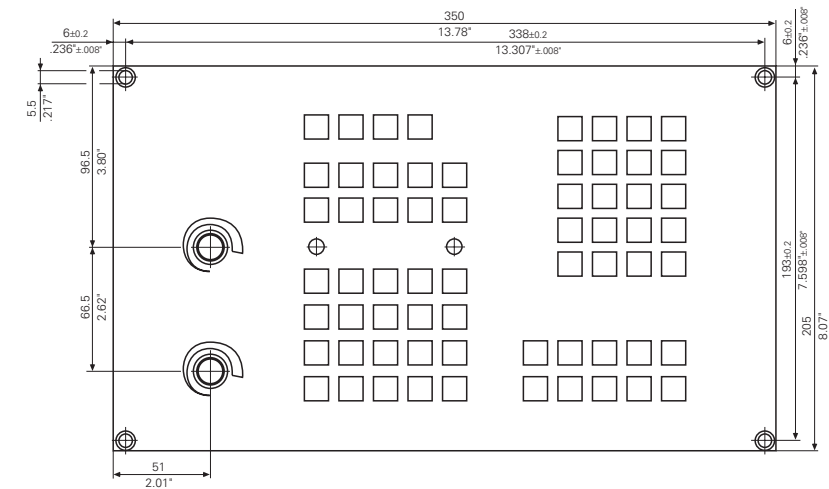


14.2 Keyboard units

14.2.1 TE 355 A

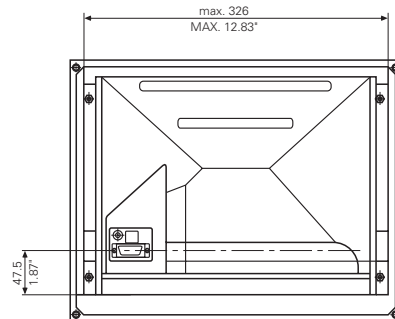
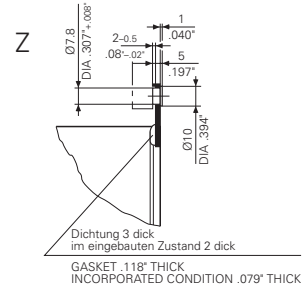
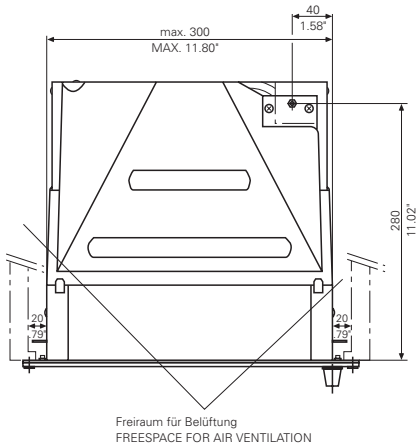
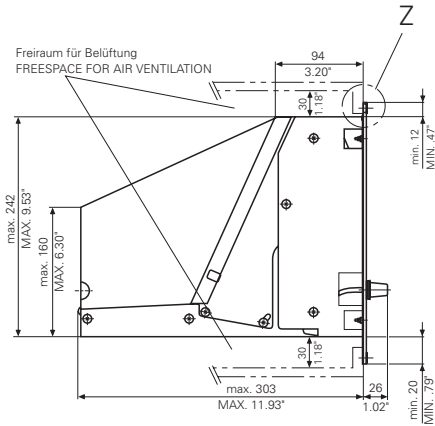
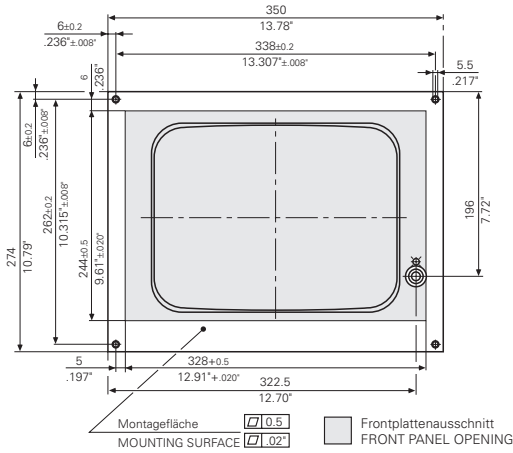


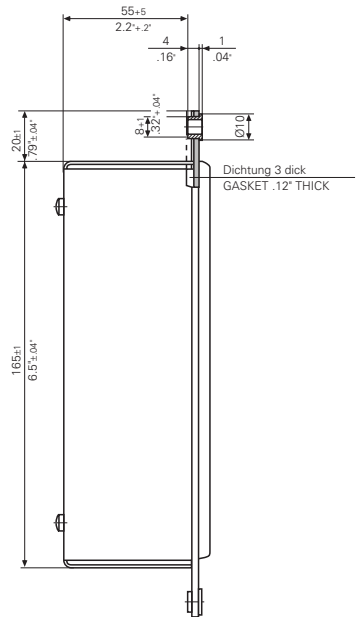
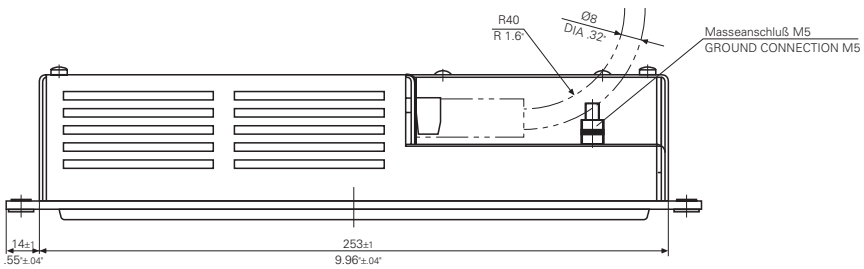
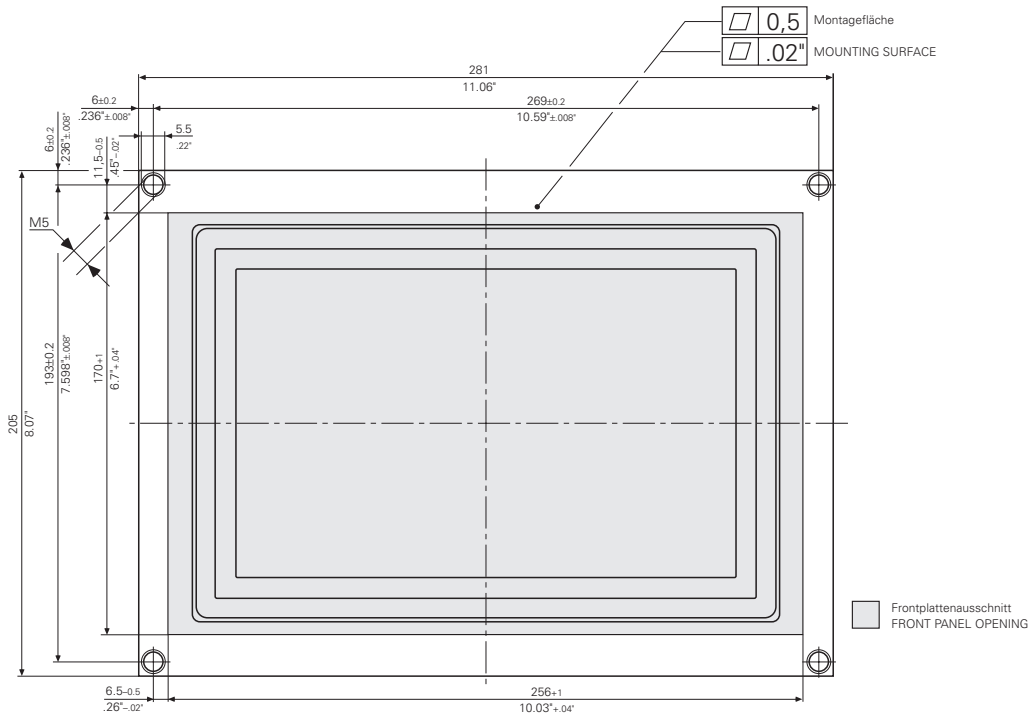
14.2.2 TE 355 B



14.3 Visual display units

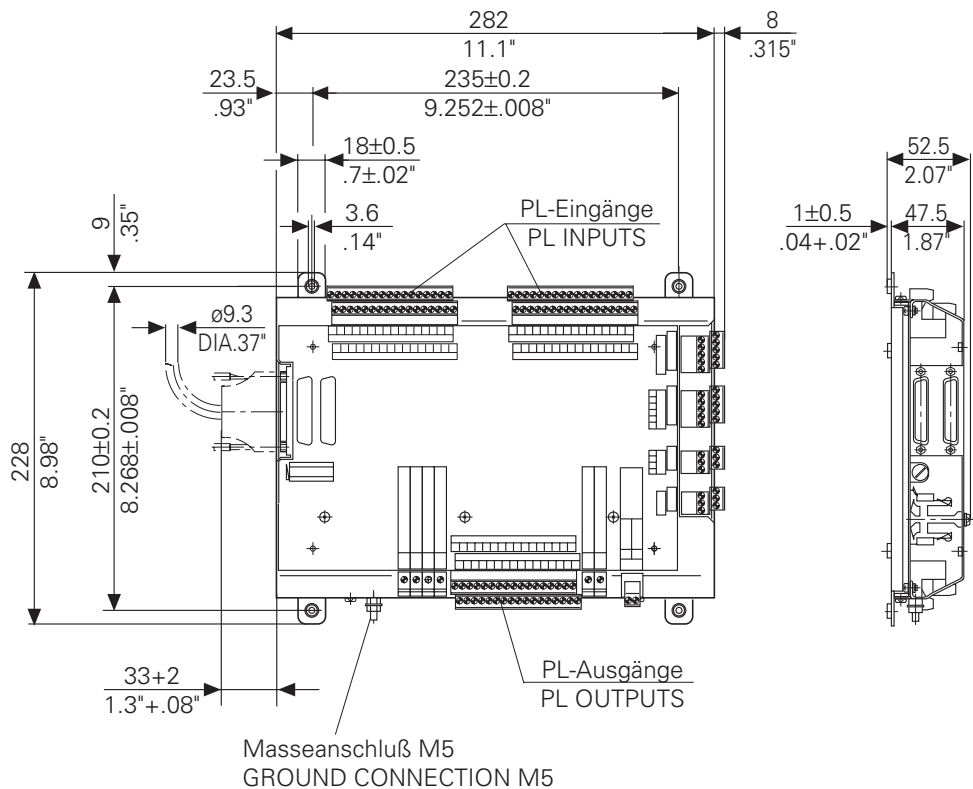
14.3.1 BE 212



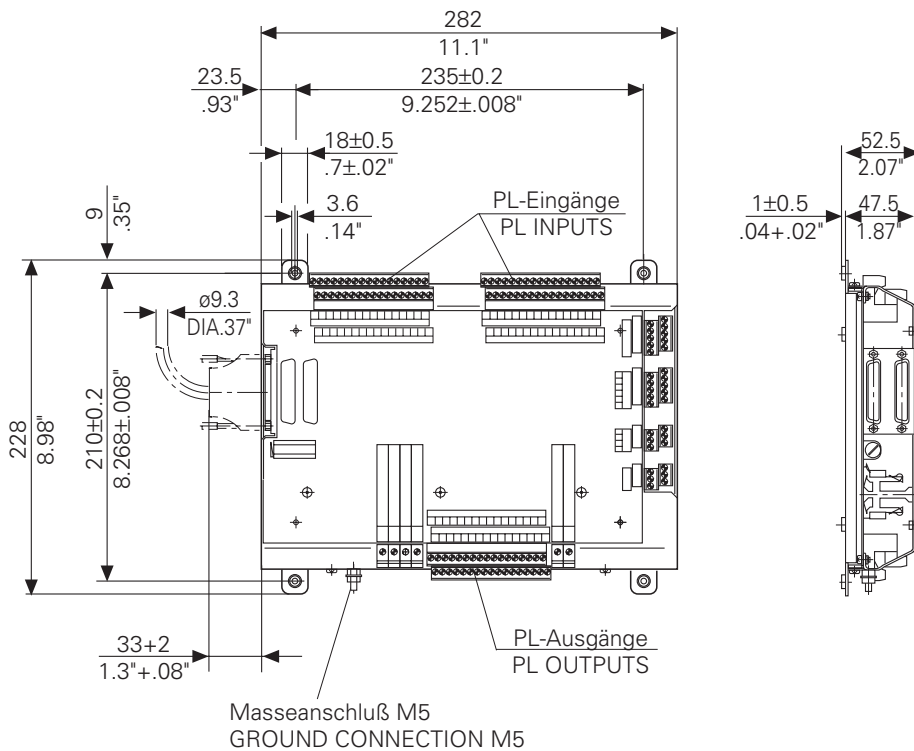


14.4 Input/Output units

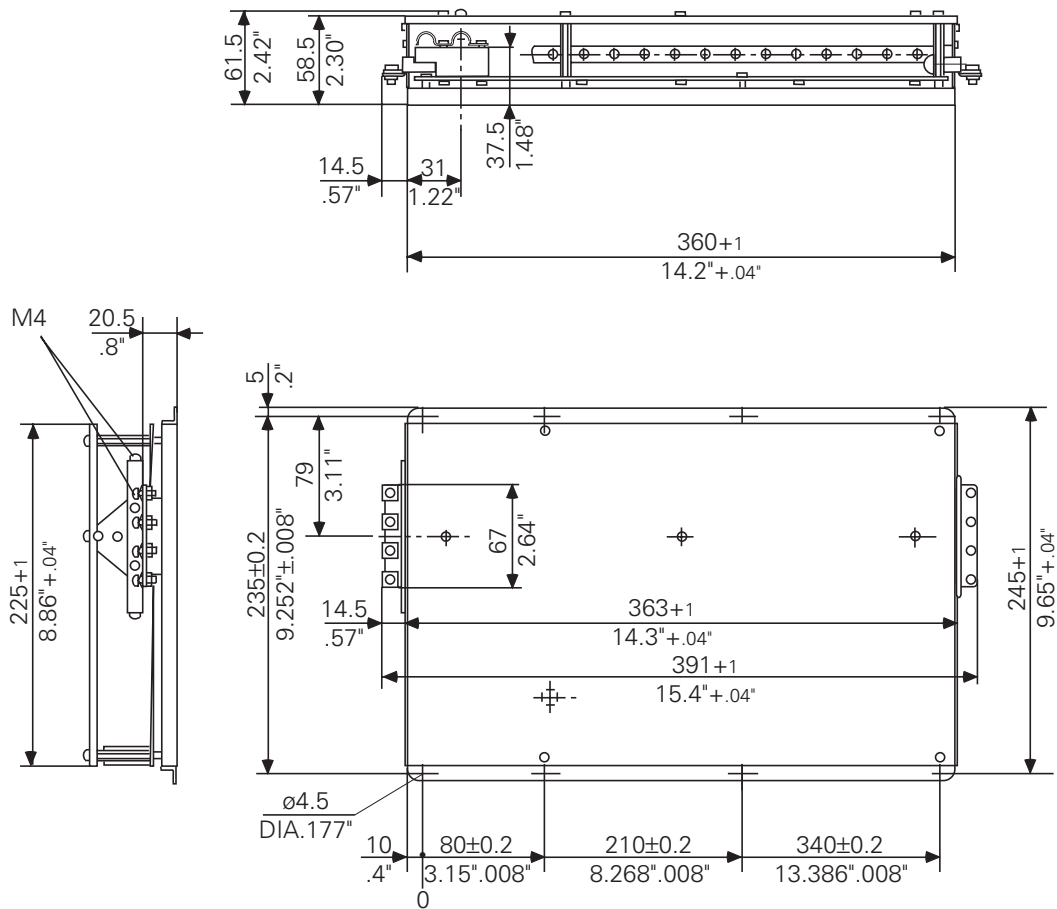
14.4.1 PL 410 B



14.4.2 PL 410

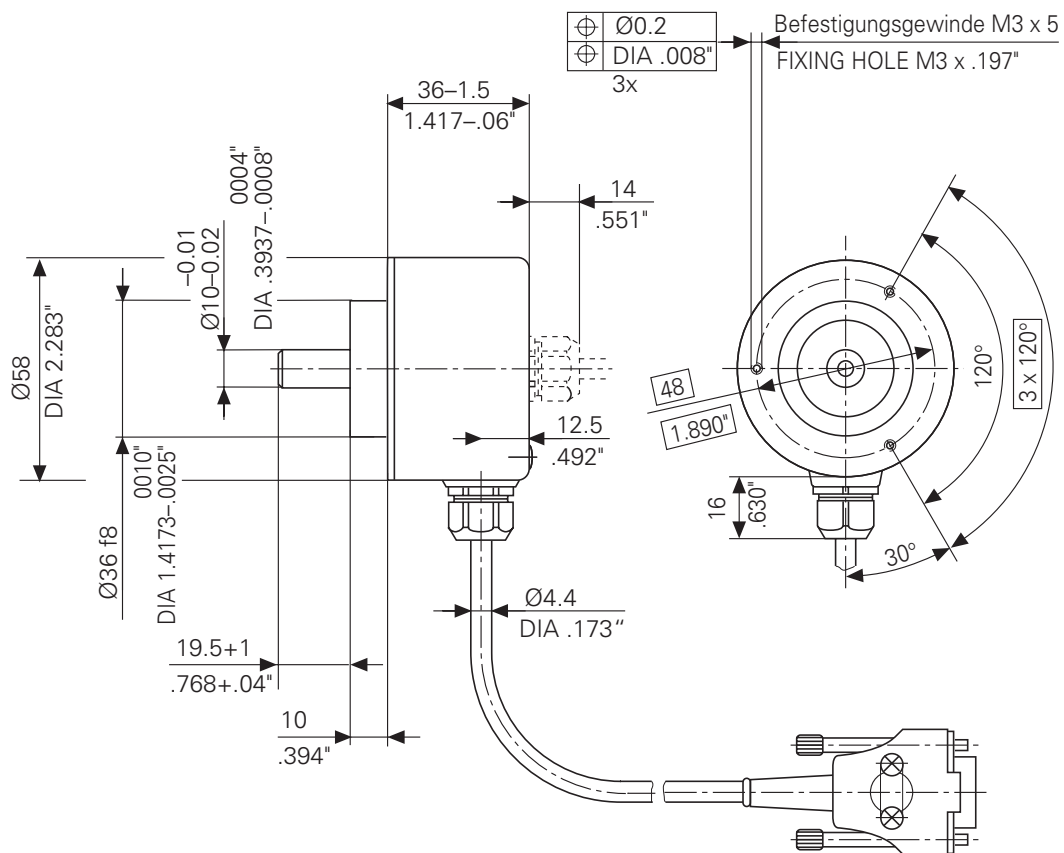


14.4.3 PL 400

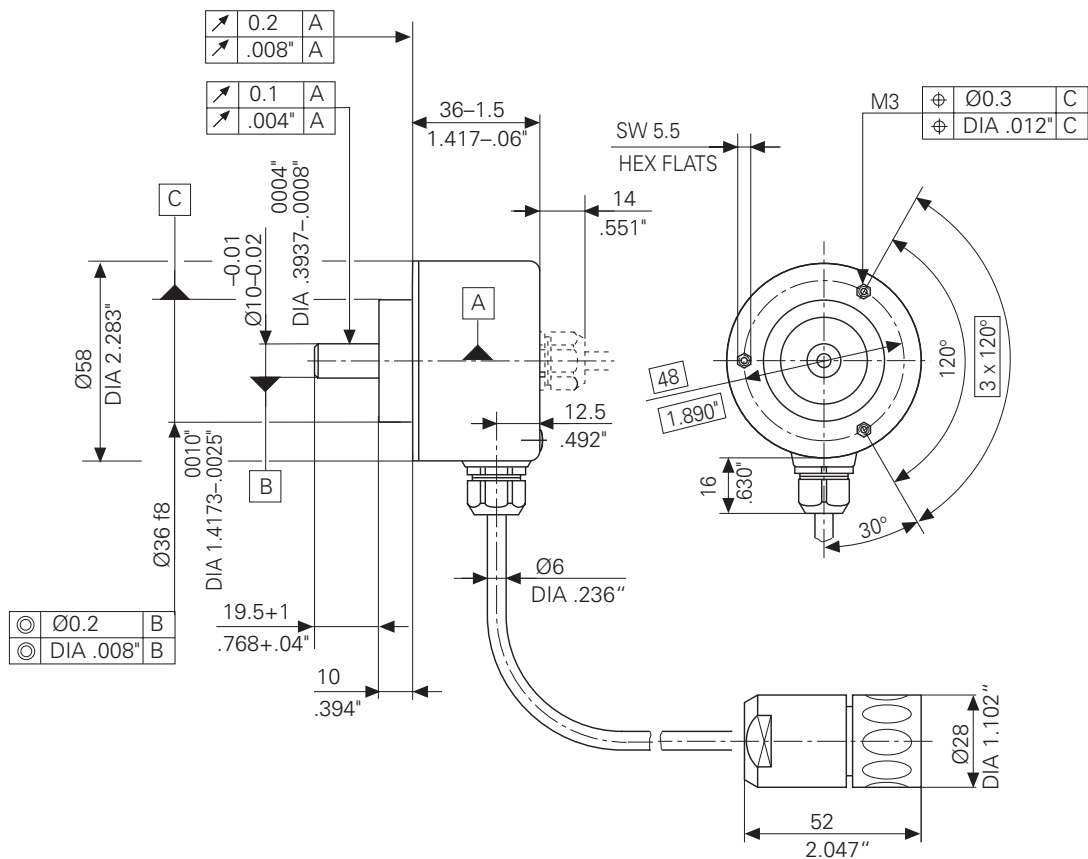


14.5 Panel-mounted handwheels

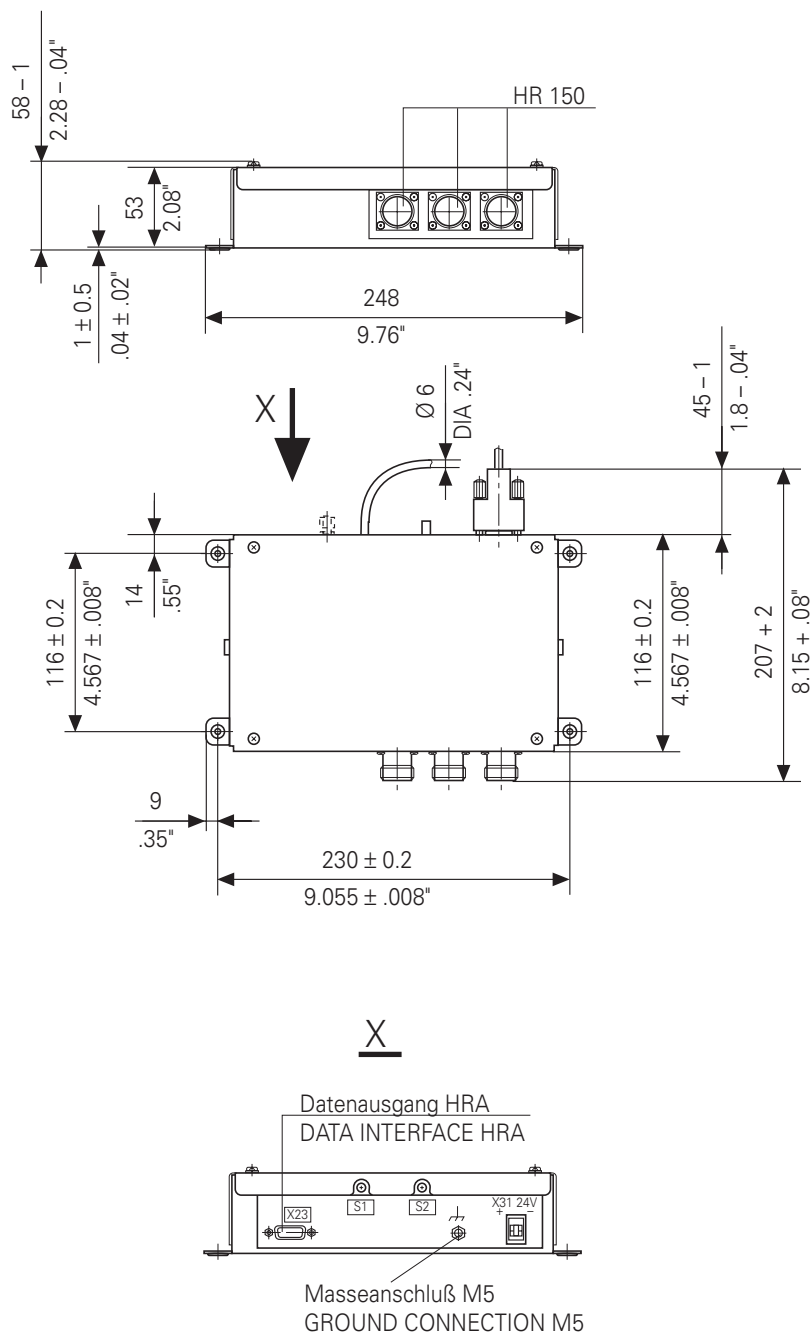
14.5.1 HR 130



14.5.2 HR 150

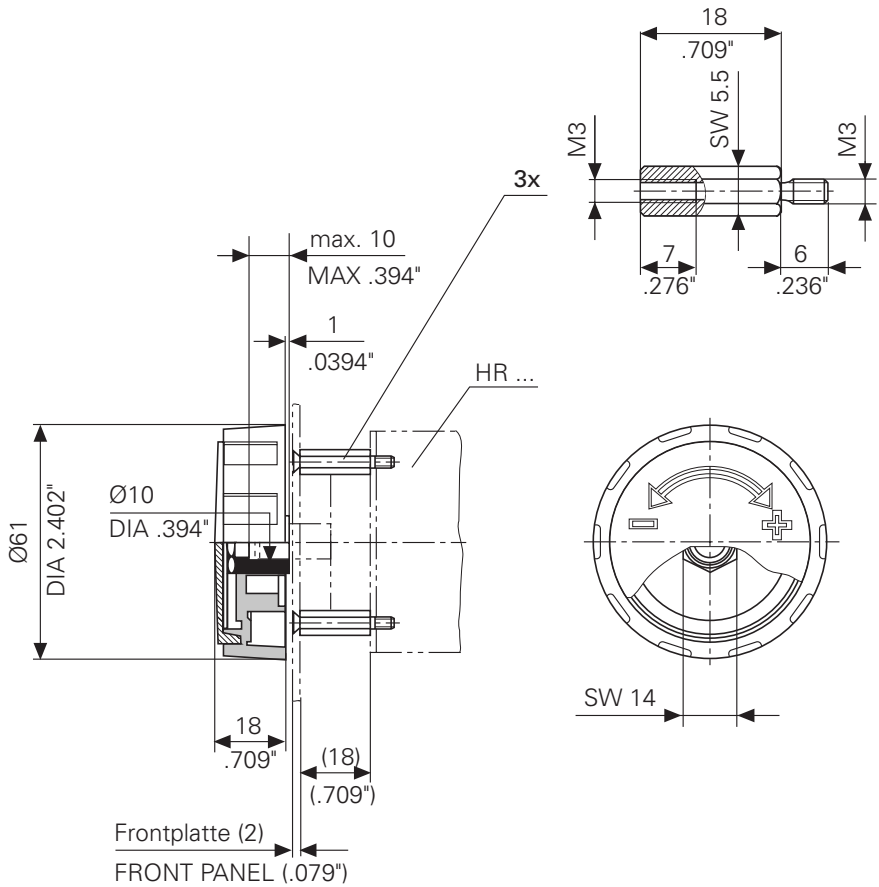


14.5.3 Handwheel adapter HRA 110 (for HR 150)

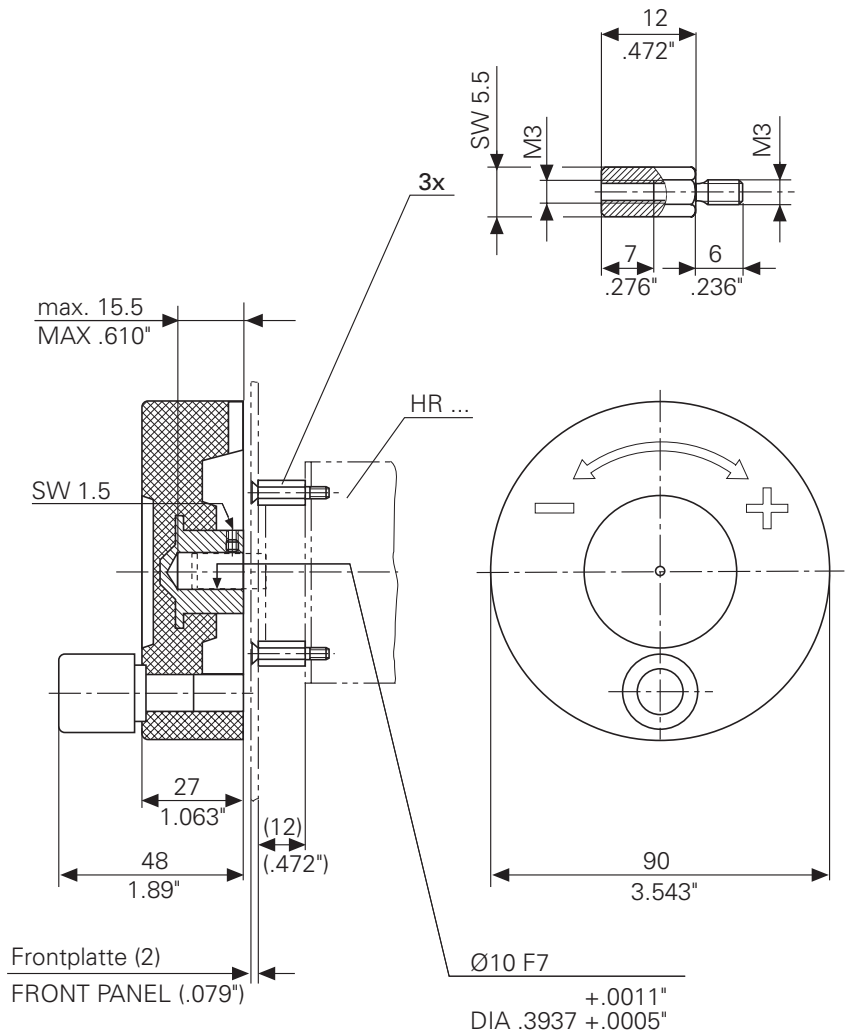


14.5.4 Handwheel knobs

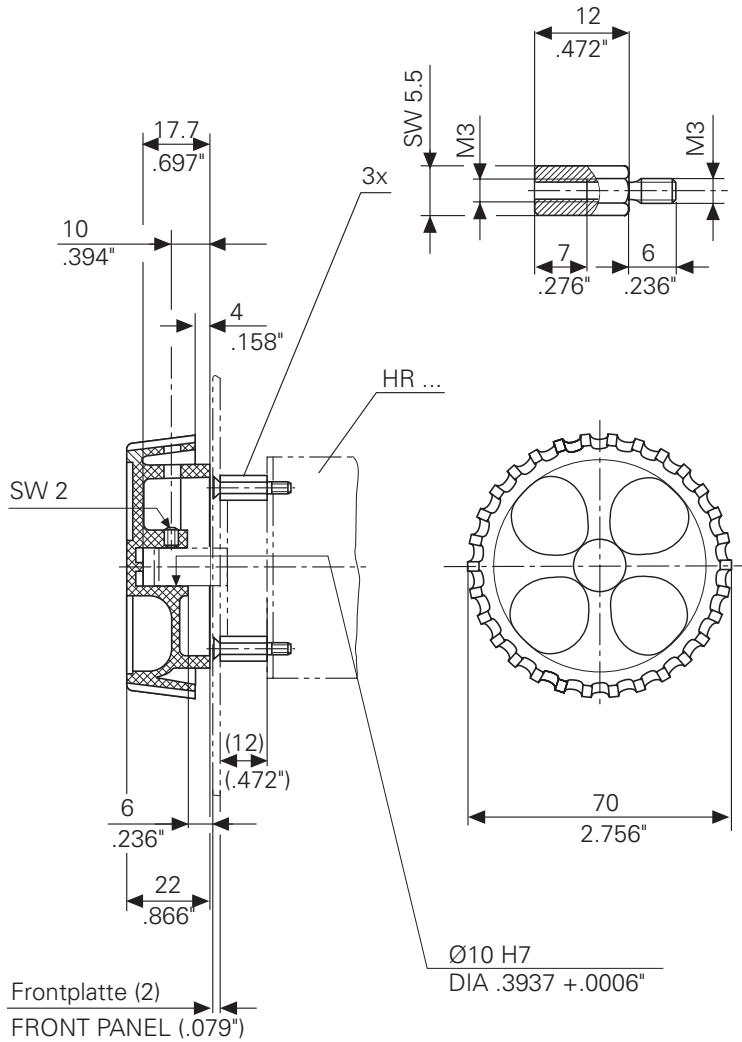
Knob, small



Knob, large

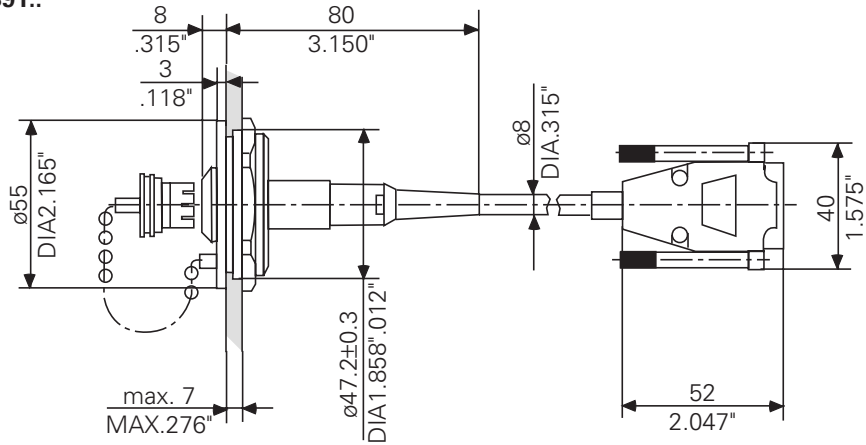


Knob, ergonomic

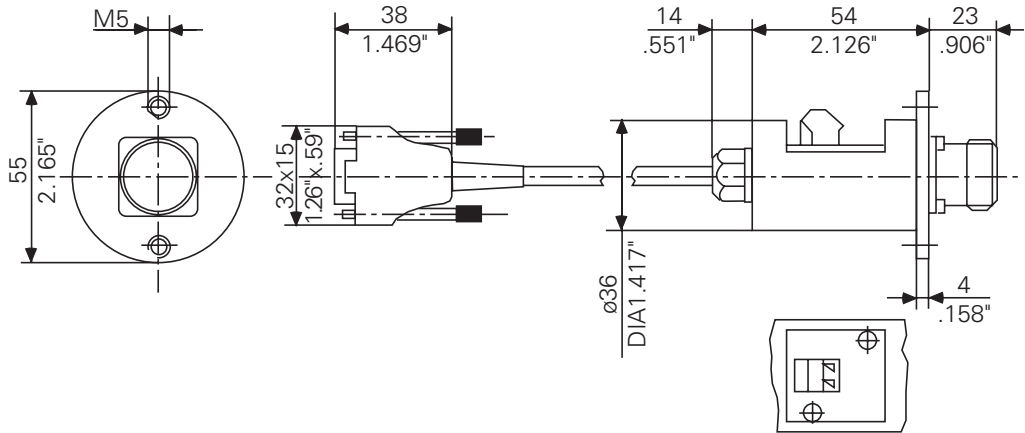


14.6 Cable adapter

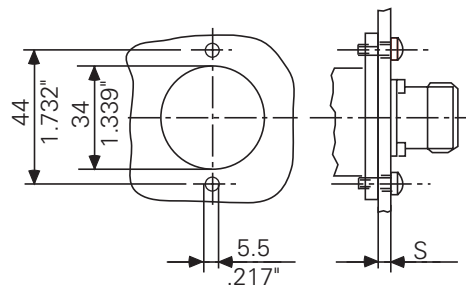
Cable adapter for TS 120
Id.-Nr. 244 891..



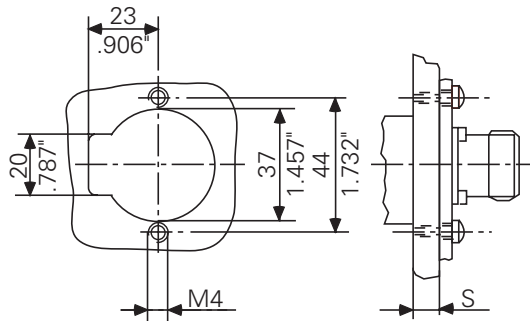
Cable adapter for HR 330 Id.-Nr. 249 889 ..
Cable adapter for HR 332 Id.-Nr. 274 556 ..



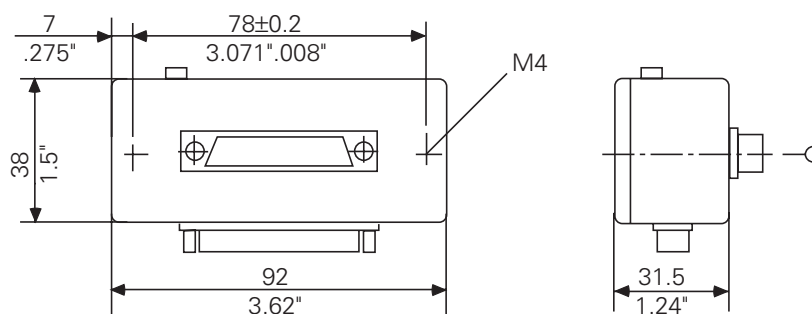
Mounting when housing thickness $S \leq 4$



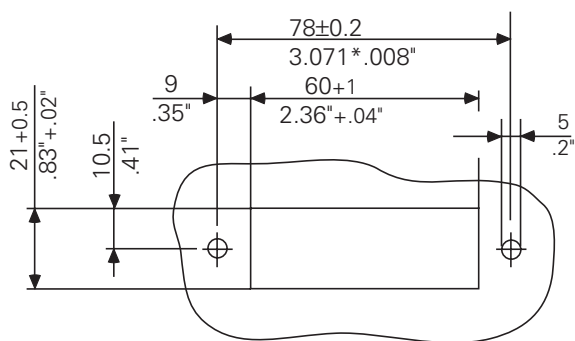
Mounting when housing thickness $S > 4$



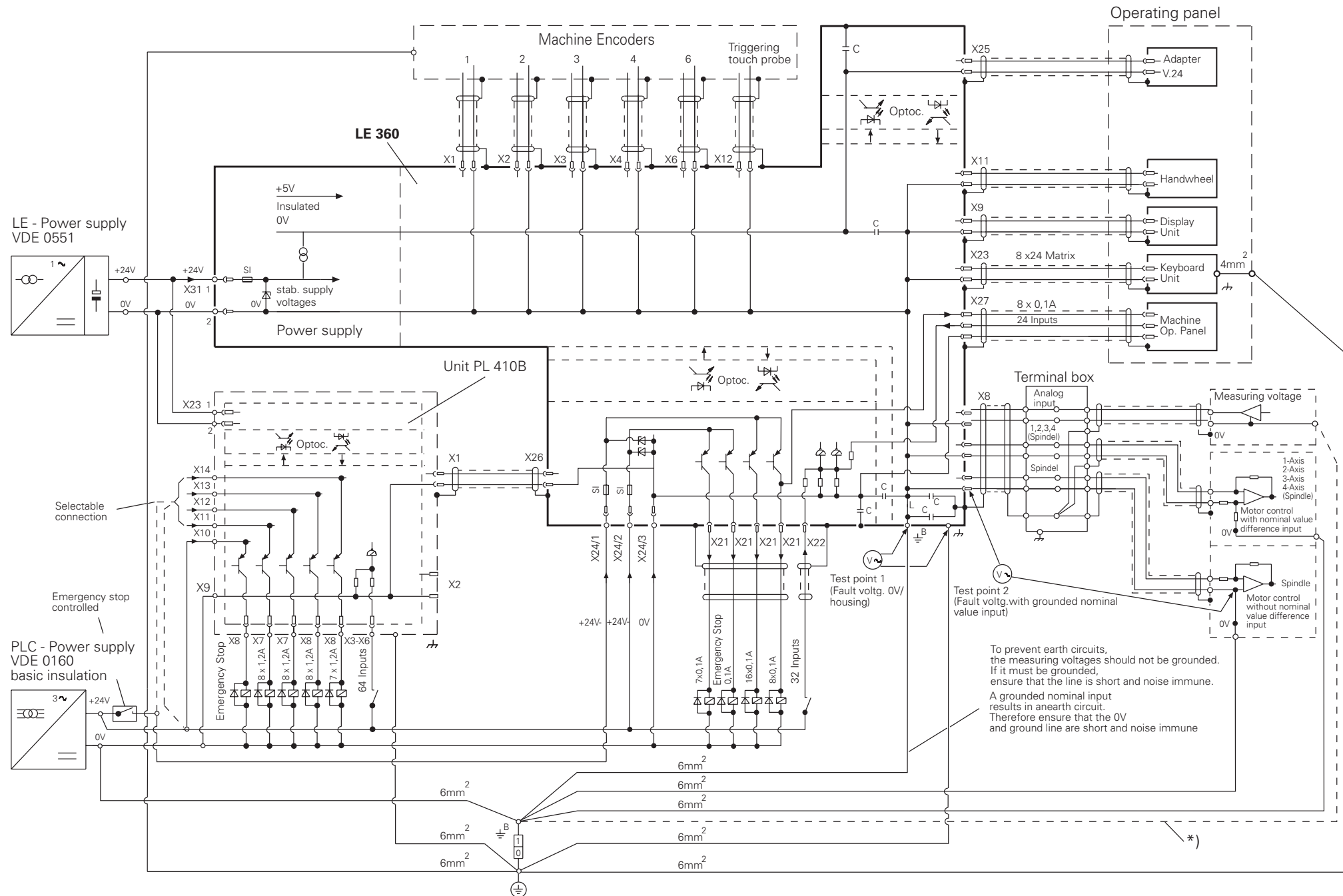
RS-232-C/V.24 Adapter Block



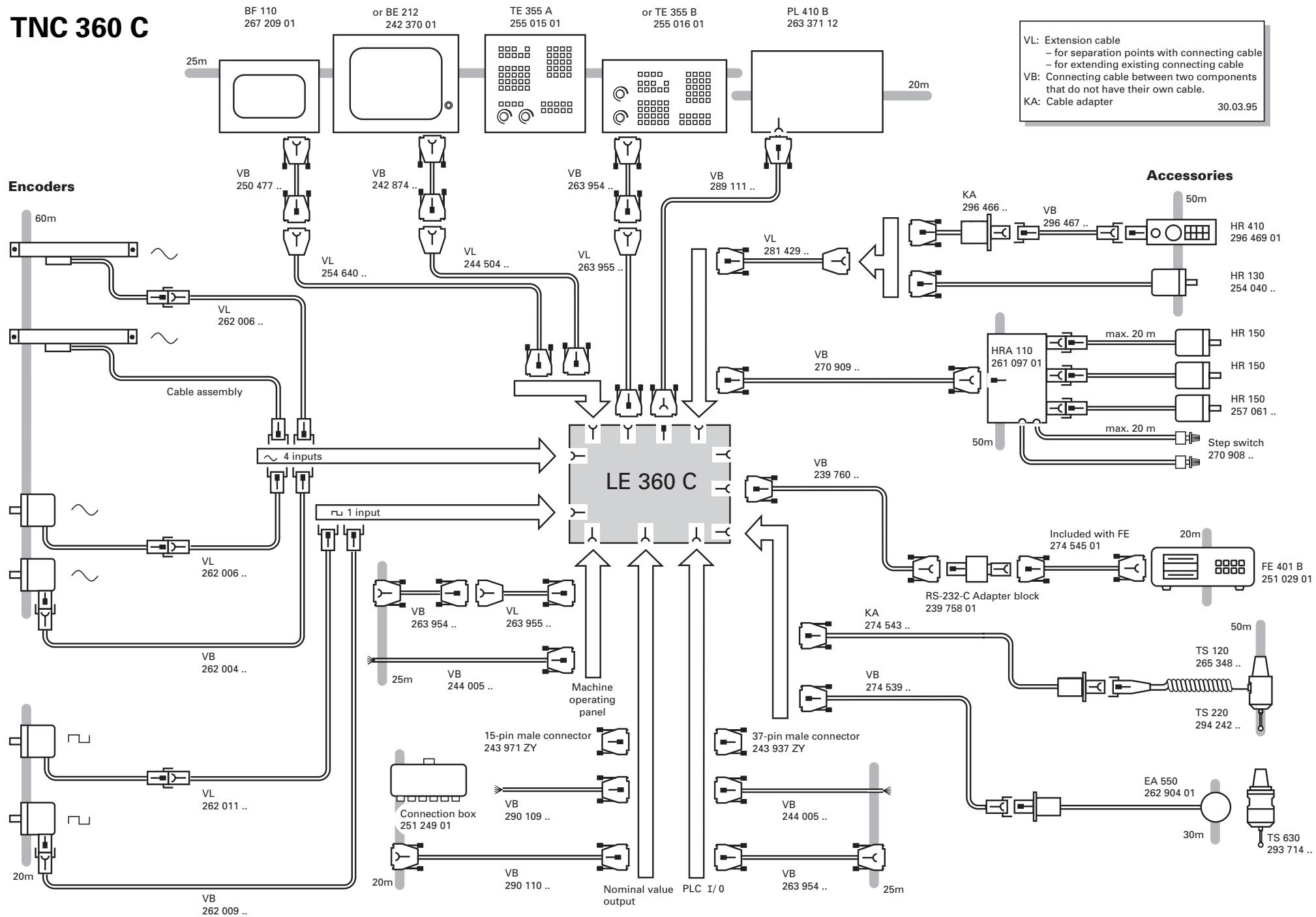
Opening for mounting the adapter



15 Grounding Diagram



TNC 360 C



Machine Integration – Contents

1	Machine axes	4-5
1.1	Measuring systems	4-5
1.1.1	Signal period	4-5
1.1.2	Direction of traverse	4-7
1.1.3	Measuring system monitoring	4-8
1.2	Axis designation	4-11
1.2.1	Assignment	4-12
1.2.2	Current tool axis	4-13
1.3	VDU display	4-13
1.4	Traverse ranges	4-15
1.5	Lubrication pulse	4-18
1.6	Axis-error compensation	4-21
1.6.1	Backlash compensation	4-21
1.6.2	Compensation for reversal errors in circular motion	4-22
1.6.3	Linear axis-error compensation	4-23
1.6.4	Nonlinear axis error compensation	4-24
1.6.5	Temperature compensation	4-29
1.7	PLC positioning	4-32
1.8	PLC Axes	4-35
2	Reference marks	4-36
2.1	Passing over the reference marks	4-38
2.1.1	Measuring systems with distance-coded reference marks	4-39
2.1.2	Measuring systems with one reference mark	4-43
2.1.3	Linear measurement via rotary encoder	4-46
2.2	Machine datum	4-49
3	Servo positioning of the NC-axes	4-51
3.1	The position control loop of an NC-machine	4-51
3.2	Servo positioning in TNC controls	4-52
3.2.1	Control with servo lag	4-52
3.2.2	Control with feed forward	4-60
3.3	Offset adjustment	4-63
3.3.1	Offset adjustment by code number	4-63
3.3.2	Automatic cyclical offset adjustment	4-63
3.3.3	Offset adjustment with integral factor	4-64
3.4	Contour behavior in corners	4-66
3.4.1	Radial acceleration	4-66
3.4.2	Constant feed rate in corners	4-66
3.4.3	Constant feed rate in corners with M90	4-67
3.5	Monitoring functions	4-68
3.5.1	Position monitoring for operation with lag	4-69

3.5.2	Position monitoring for operation with feed forward control	4-70
3.5.3	Monitoring the analog voltage	4-71
3.5.4	Movement-monitoring	4-71
3.5.5	Standstill monitoring	4-72
3.5.6	Positioning window	4-72
3.6	Controlled axes	4-74
3.6.1	Axis-enable, feed rate enable	4-74
3.6.2	Axes in position	4-75
3.6.3	Axes in motion	4-76
3.6.4	Open control loop	4-77
3.6.5	Actual - nominal value transfer	4-77
4	Spindle	4-79
4.1	Analog output of the spindle speed	4-80
4.1.1	Direction of spindle rotation	4-80
4.1.2	Gear range	4-81
4.1.3	Spindle-override	4-84
4.1.4	Gear change	4-85
4.2	BCD-coded output of spindle speed	4-91
4.3	Spindle-orientation	4-93
4.4	Tapping	4-99
4.4.1	Tapping with floating tap holder for analog spindle-speed output	4-99
4.4.2	Tapping with floating tap holder and coded spindle-speed output	4-102
4.4.3	Rigid tapping	4-103
5	EMERGENCY STOP Routine	4-106
5.1	Connection diagram	4-107
5.2	Flow diagram	4-108
6	Display and Operation	4-109
6.1	Machine datum	4-109
6.2	Graphic simulation	4-112
6.3	Status display	4-113
6.3.1	Position display	4-113
6.3.2	Display mode and traverse direction for rotary axes	4-114
6.3.3	Feed rate display, feed rate display	4-115
6.3.4	Display of M functions	4-117
6.3.5	Display of run duration	4-117
6.3.6	Control is operational	4-118
6.3.7	Cancel status display	4-119
6.3.8	Expanded spindle display	4-119
6.3.9	Generating NC blocks with the actual-position-capture key	4-121
6.4	PLC text window	4-121
6.5	Error messages	4-121

6.6	Cycles	4-126
6.6.1	Cycle inhibit	4-126
6.6.2	Pocket milling	4-128
6.6.3	Milling-cycles for pockets with free-programmed contours	4-128
6.6.4	Scaling factor	4-130
6.7	File types	4-132
6.7.1	File types – disable	4-132
6.7.2	Block numbers - step size for ISO programs	4-132
6.8	User parameters	4-132
6.9	Code numbers	4-133
6.10	Programming station	4-134
6.11	Dialog language	4-134
6.12	Decimal sign	4-134
6.13	Memory test	4-135
6.14	End of program run	4-135
6.15	Overwrite Q-parameters overwriting	4-135
7	M Functions	4-137
7.1	Program halt on M functions	4-140
7.2	Program halt on M06	4-140
7.3	M function M89	4-140
8	Key Simulation	4-141
8.1	TNC Keyboard	4-141
8.2	Machine control panel	4-149
9	Touch Probe	4-151
9.1	Interfacing the touch probe	4-151
9.2	Digitizing with TS 120	4-155
9.2.1	Scanning cycles	4-158
9.2.2	Response of the scanning sequence at corners	4-159
9.2.3	Optimizing the scanning sequence	4-161
10	Electronic Handwheel	4-166
10.1	Panel-mounted handwheel HR 130	4-169
10.2	Portable handwheel HR 330	4-169
10.3	Portable handwheel HR 332	4-169
10.3.1	Assignment of keys and LEDs to the PLC inputs and outputs	4-170
10.3.2	PLC example	4-171
10.4	Panel-mounted handwheels HR 150 with handwheel adapter HRA 110	4-172
10.4.1	Assignment of the switch setting to the PLC inputs	4-173
10.4.2	PLC Example	4-174
11	Analog Inputs	4-176

12	Incremental Jog Positioning	4-178
13	Hirth Coupling	4-183
13.1	Positioning in the manual operating mode	4-183
13.2	Positioning in controlled operation	4-183
13.3	Program example	4-183
14	Datum Correction	4-194
15	Tool Changer	4-196
15.1	Tool table	4-196
15.1.1	Special Tools	4-197
15.2	Controlling the tool changer	4-197
15.2.1	Output of tool number (Fixed pocket coding)	4-199
15.2.2	Output of pocket number (Variable pocket coding)	4-199
15.3	PLC program example	4-215
15.3.1	Program module TOOL DEF	4-217
15.3.2	Program module TOOL CALL	4-218
15.3.3	Program module STANDBY	4-219
15.3.4	Program module STANDBY BACK	4-219
15.3.5	Program module MANUAL TOOL IN	4-220
15.3.6	Program module MANUAL TOOL OUT	4-221
15.3.7	Program module MANUAL TOOL IN/OUT	4-222
15.3.8	Program module CHANGE	4-223
15.3.9	Program module COMPARE P-CODE WITH ISTREG	4-224
15.3.10	Program module COMPARE GRE1 WITH ISTREG	4-224
15.3.11	Program module COMPUTE SHORTEST PATH	4-226
16	Commissioning and Start-up Procedure	4-227
16.1	Code numbers for commissioning	4-227
16.2	Preparation of the machine	4-227
16.3	Commissioning the control	4-231
16.3.1	Entry of the provisional and pre-defined machine parameters	4-231
16.3.2	Entry of the PLC-program	4-231
16.3.3	Testing the EMERGENCY STOP routine	4-231
16.3.4	Testing the direction of traverse	4-232
16.3.5	Fixing the software limit switch ranges	4-233
16.3.6	Optimizing the control with lag (MP1390 = 1)	4-233
16.3.7	Optimizing the control with feed forward control (MP1390 = 0)	4-237
16.3.8	Optimizing the integral factor	4-242
16.3.9	Adjust the offset	4-243
16.3.10	Adjustment of the monitoring functions	4-243
17	Point-to-Point and Straight-Cut Control	4-246

1 Machine Axes

The HEIDENHAIN contouring controls TNC 360/TNC 335 permit the control of up to four machine axes and the spindle (see also section "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
		Input range: 1 to 15
Bit 0	X axis	+0 = not active +1 = active
Bit 1	Y axis	+0 = not active +2 = active
Bit 2	Z axis	+0 = not active +4 = active
Bit 3	4th axis	+0 = not active +8 = active

1.1 Encoders

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 encoders 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:

$$\text{Signal period (~)} = \frac{\text{ballscrew pitch [mm]} \cdot 1000 [\mu\text{m/mm}]}{\text{line count}}$$

Up to 3 decimal places can be entered in MP330.x. If traverse is 60 m/min, the signal period must not be less than 20 μm , because then the frequency would fall below the lower limit of 300 kHz.

In addition to the signal period, MP 310 can also set a signal subdivision that can be 1-, 2-, 4-, or 256-fold and should be entered so that in combination with the signal period (MP 330) a display step of 1 µm is achieved. For example, a 4 µm signal period and 4-fold subdivision yields a display step of 1 µm, a 20 µm signal period and 4-fold subdivision results in a display step of 5 µm. In order to attain a 1 µm display step from a 20 µm signal period, you must enter a subdivision factor of 256. The control internally rounds the result of the division $20 \mu\text{m}/256$ up to 1 µm.

The signal at the square-wave input (Spindle input X6) undergoes a control-internal 4-fold evaluation. Machine parameter MP 340 was introduced to permit including an external signal interpolation via EXE, or IBV. This information is needed in order to determine the absolute position with scales featuring distance-coded reference marks. In machine parameter MP340 must defined whether the ballscrew encoder has 2048 or 4096 lines.

Angular measurement: The signal period for angular measuring system is calculated as follows:

$$\text{Signal period } (\sim) = \frac{360^\circ}{\text{line count}} \cdot 1000$$

MP310 Signal subdivision

Entry: 0 to 3
 0 = 256-fold
 1 = 4-fold
 2 = 2-fold
 3 = 1-fold

MP 310.0 Signal subdivision axis X
 MP 310.1 Signal subdivision axis Y
 MP 310.2 Signal subdivision axis Z
 MP 310.3 Signal subdivision 4th axis

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: 0.100 to 999.999 [µm] or $\left[\frac{1^\circ}{1000} \right]$

MP330.0 Axis X
 MP330.1 Axis Y
 MP330.2 Axis Z
 MP330.3 4th axis

MP340 Interpolation factor of the EXE at X6

Entry: 0 to 5
 0 = No measuring system on X6
 1 = EXE without interpolation
 2 = Spindle encoder with 2048 lines
 3 = Spindle encoder with 4096 lines
 5 = EXE with 5-fold interpolation

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 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**
Input range: 0 to 15

Bit 0	X axis	+0 = positive +1 = negative
Bit 1	Y axis	+0 = positive +2 = negative
Bit 2	Z axis	+0 = positive +4 = negative
Bit 3	4th axis	+0 = positive +8 = negative

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**
Input range: 0 to 15

Bit 0	X axis	+0 = positive +1 = negative
Bit 1	Y axis	+0 = positive +2 = negative
Bit 2	Z axis	+0 = positive +4 = negative
Bit 3	4th axis	+0 = positive +8 = negative

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

0 = positive
1 = negative

Marker	Function	Set	Reset
M2160	Traverse direction X axis	NC	NC
M2161	Traverse direction Y axis		
M2162	Traverse direction Z axis		
M2163	Traverse direction 4th axis		

1.1.3 Encoder 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, it results in the error message:

"Measuring system <axis> defect A/B/C."

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 or IBV).

MP30 Checking the absolute position of the distance-coded reference marks
Input range: 0 to 15

Bit 0	X axis	+0 = not active +1 = active
Bit 1	Y axis	+0 = not active +2 = active
Bit 2	Z axis	+0 = not active +4 = active
Bit 3	4th axis	+0 = not active +8 = active

MP31 Checking the amplitude of the measuring system signals
Input range: 0 to 31

Bit 0	X axis	+0 = not active +1 = active
Bit 1	Y axis	+0 = not active +2 = active
Bit 2	Z axis	+0 = not active +4 = active
Bit 3	4th axis	+0 = not active +8 = active
Bit 4	Axis S	+0 = not active +16 = active

MP32**Checking the edge separation of the measuring system signals**

Input range: 0 to 31

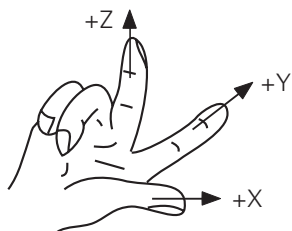
Bit 0	X axis	+0 = not active +1 = active
Bit 1	Y axis	+0 = not active +2 = active
Bit 2	Z axis	+0 = not active +4 = active
Bit 3	4th axis	+0 = not active +8 = active
Bit 4	Axis S	+0 = not active +16 = active



1.2 Axis designation

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

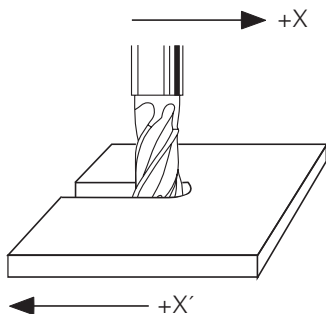
The directions of traverse may be simply determined by using the "right-hand rule."



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.

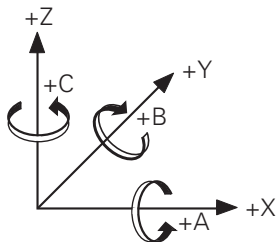
If the machine moves the workpiece, then the direction of movement and the direction of the axis are opposite to one another. The positive relative directions of movement are then designated +X', +Y' etc.



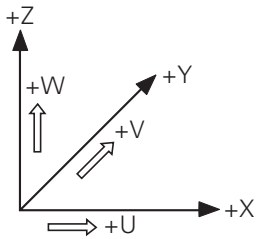
The fourth axis can be used either as an axis of rotation or alternatively as an extra linear axis.

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

A **rotational axis** is designated by the letter A, B or C. The correlation with the principle 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 principle axes and the direction of travel are also standardized in ISO 841.



MP410.3 Axis designation for 4th axis

Entry: 0 to 5

- 0 = A
- 1 = B
- 2 = C
- 3 = U
- 4 = V
- 5 = W

1.2.1 Assignment

The measuring system inputs X1 to X4 and X6 and the analog 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 four axes is only possible when the Spindle Orientation function is not utilized.

MP110 Assignment of the measuring system inputs to the axes

- Entry:
- 0 = measuring system input X1
 - 1 = measuring system input X2
 - 2 = measuring system input X3
 - 3 = measuring system input X4
 - 5 = measuring system input X6

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

MP120 Assignment of the analog outputs

Entry: 0 = output 1
 1 = output 2
 2 = output 3
 3 = output 4
 5 = output S

MP120.0 X axis
MP120.1 Y axis
MP120.2 Z axis
MP120.3 4th 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 principle axes X, Y, Z or parallel to the fourth 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		

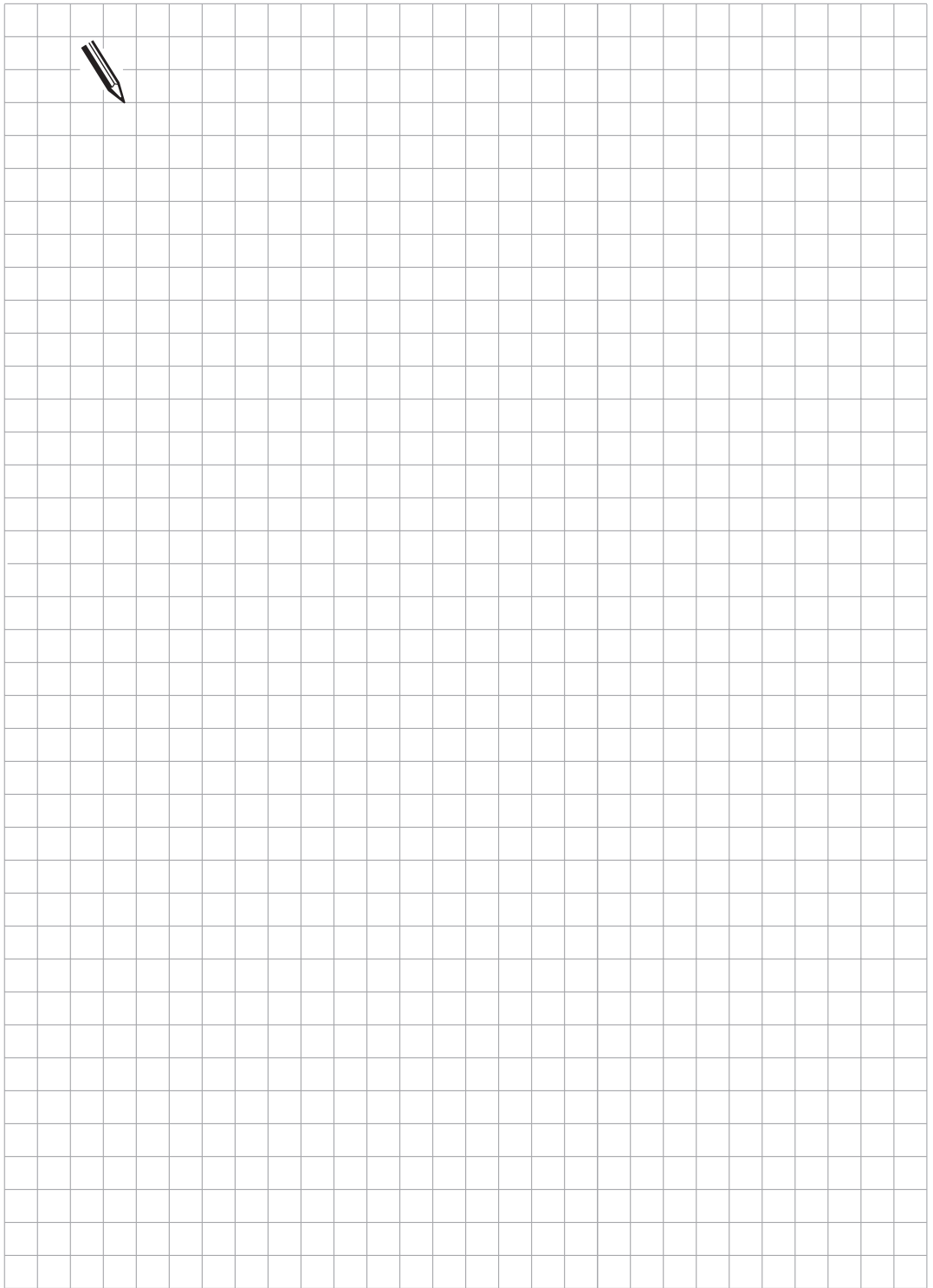
1.3 VDU display

Machine parameters can be used to select which of the active axes (MP10) should be displayed in the status window.

The display of the spindle position cannot be activated by MP40. The spindle position is displayed in the status window when the spindle control loop is closed (see section "Spindle").

MP40 VDU display Input range: 1 to 15

Bit 0	X axis	+0 = not active +1 = active
Bit 1	Y axis	+0 = not active +2 = active
Bit 2	Z axis	+0 = not active +4 = active
Bit 3	4th axis	+0 = not active +8 = active



1.4 Traverse ranges

For all four 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 referenced to the machine datum. The momentary software limit switch range is selected by the markers (M2817, M2816) and activated by the strobe marker M2824.

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

In the setting for one traverse range, the software limit switches that are entered via MOD function as well as the datum are effective for all traverse ranges that are selected through the PLC.

Traverse ranges 1 to 3:

Entry:

Linear axis: – 30 000.000 to + 30 000.000 [mm]

Rotary axis: – 30 000.000 to + 30 000.000 [°]

(Values referenced to the machine datum)

MP910 **Traverse range 1: Maximum value**

Default values after power-on;

MP920 **Traverse range 1: Minimum value**

Activated via 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+

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

MP911 **Traverse range 2: Maximum value**

Default values after power-on;

MP921 **Traverse range 2: Minimum value**

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+

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

MP912 Traverse range 3: Maximum value**MP922 Traverse range 3: Minimum value**
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+

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

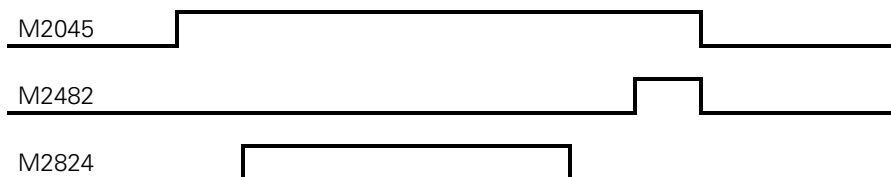
The markers M2816 and M2817 are used to define the traverse range.

M2816	M2817	Traverse range
0	0	Range 1 (MP910.x; MP920.x)
1	0	Range 2 (MP911.x; MP921.x)
0	1	Range 3 (MP912.x; MP922.x)

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.

In the "Manual" and "Electronic handwheel" modes of operation this is done immediately, in all other modes of operation it is usually done with the next M/S/T strobe. The range must be switched during an M/S/T strobes.

Example:



In the Manual and Electronic Handwheel modes, the range can be switched immediately.

		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 M2631) is set.

		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-		

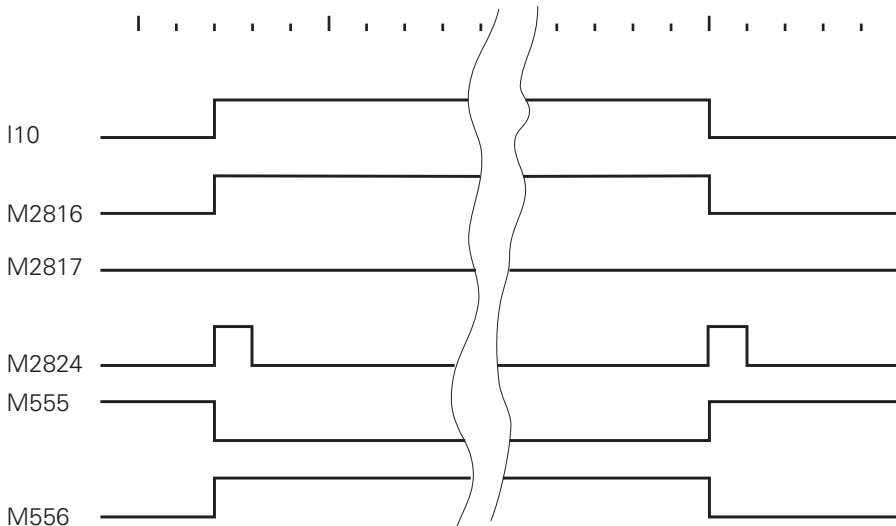
Example:

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 traveled on each axis. In 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) to "1."

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

MP4060 Path-dependent lubrication

Input range: 0 to 65 535 (in units of 65 536 µm)

MP4060.0	X axis
MP4060.1	Y axis
MP4060.2	Z axis
MP4060.3	4th axis

		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		
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		

Example:

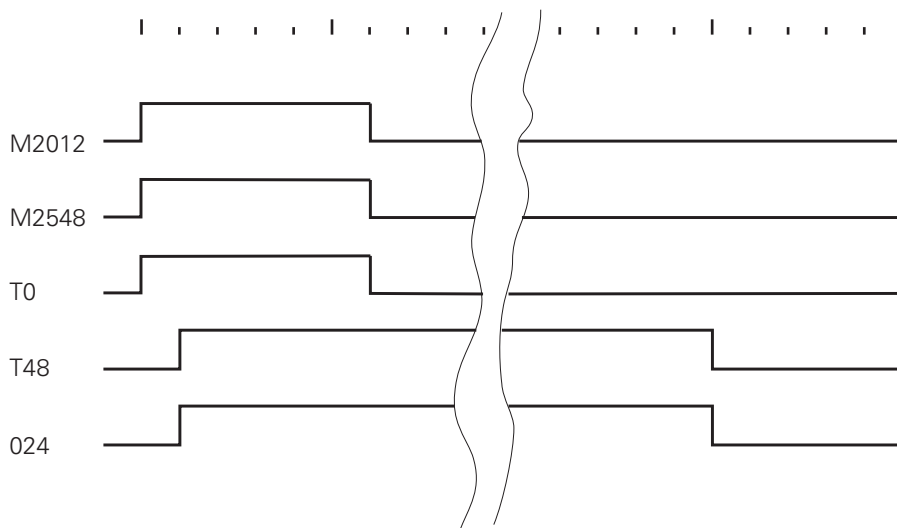
PLC program example of activating the lubrication for the X axis.

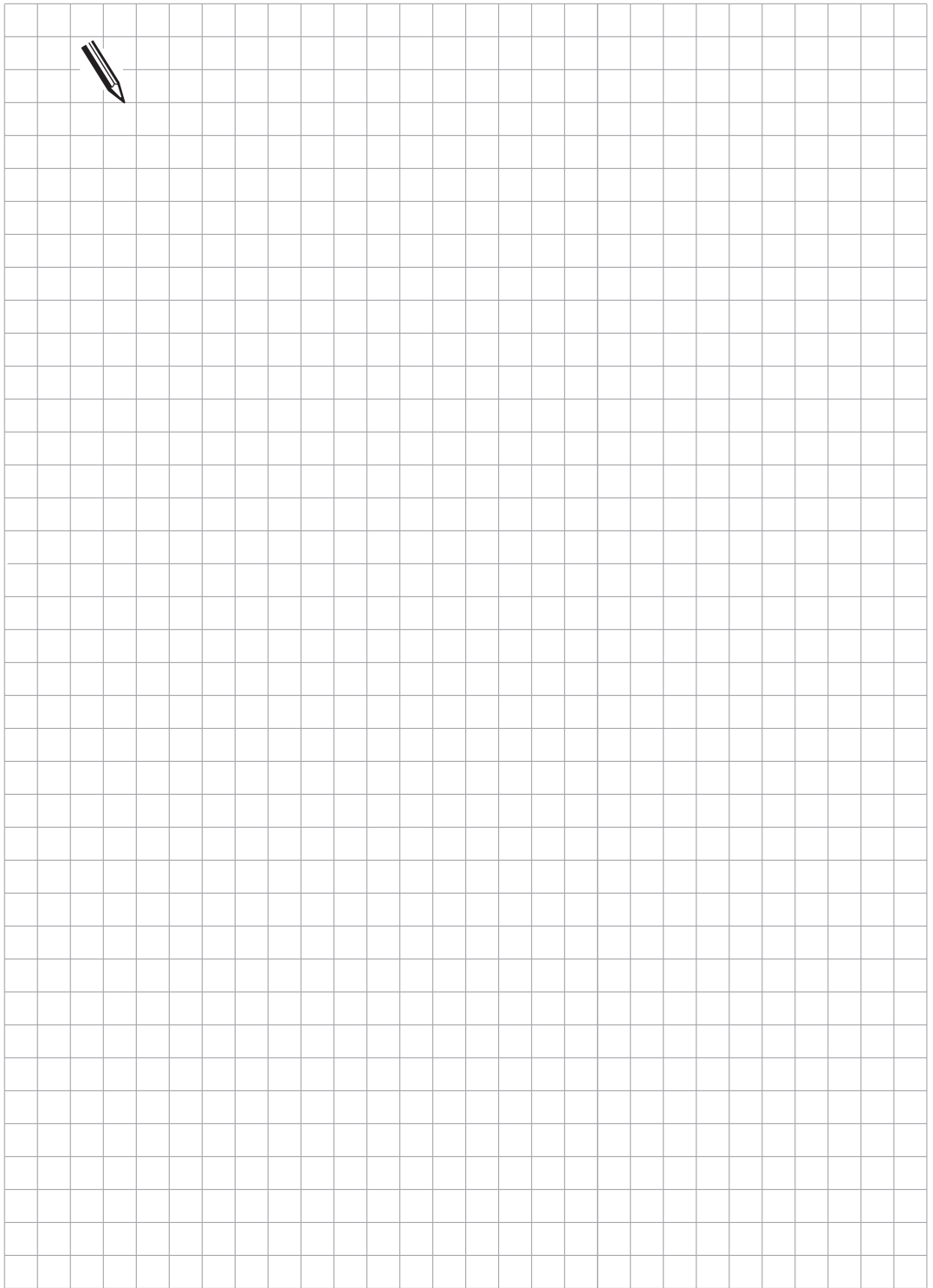
In the machine parameter MP4060.0 the traverse distance after which the X axis should be lubricated is entered. The duration of the lubrication is defined by the timer T0 (MP4110.0).

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

A 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,
- nonlinear axis-error compensation,
- compensation of thermal expansion.

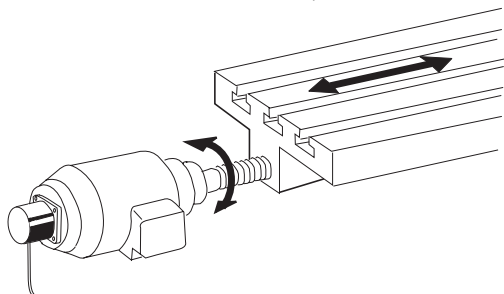
Either the linear or the nonlinear axis-error compensation can be activated. The other types of compensation can always be activated in parallel.

1.6.1 Backlash compensation

When using linear measurement by ballscrew and rotary encoder, a small amount of play can occur between the movement of the table and that of the rotary encoder.

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.

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).

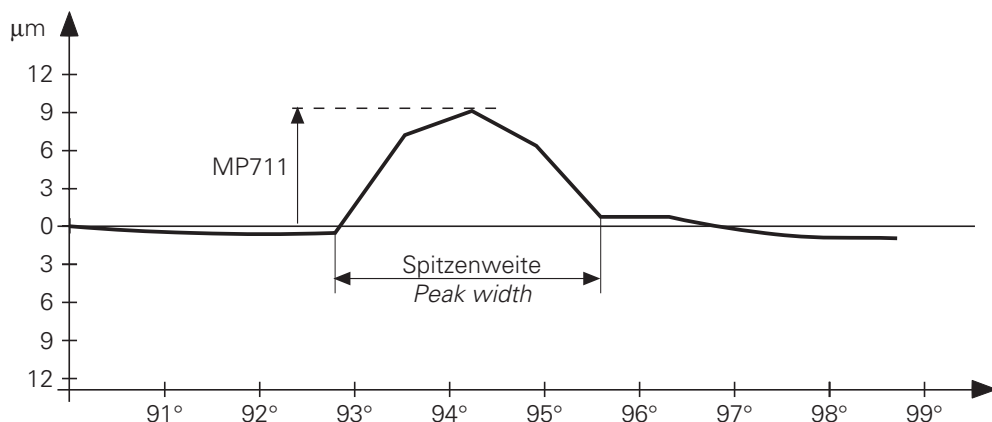
For every change in direction of traverse, the TNC adds the value from machine parameter MP710 to value from the encoder signals.

MP710	Backlash compensation
	Input range: – 1.000 to + 1.000 [mm] or [°]
MP710.0	X axis
MP710.1	Y axis
MP710.2	Z axis
MP710.3	4th 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 (DBB) 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 input value for MP712 can be calculated as follows:

- Duration of the reversal peak

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

the peak width is [°] indicated in the diagram;

feed rate [mm/min] is the programmed contouring feed rate.

- Compensation per control loop cycle time (6 ms with TNC 360)

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

The compensation value is entered in MP712.

MP711 Size of the reversal peaks in circular motion

Input range: -1.000 to +1.000 [mm]

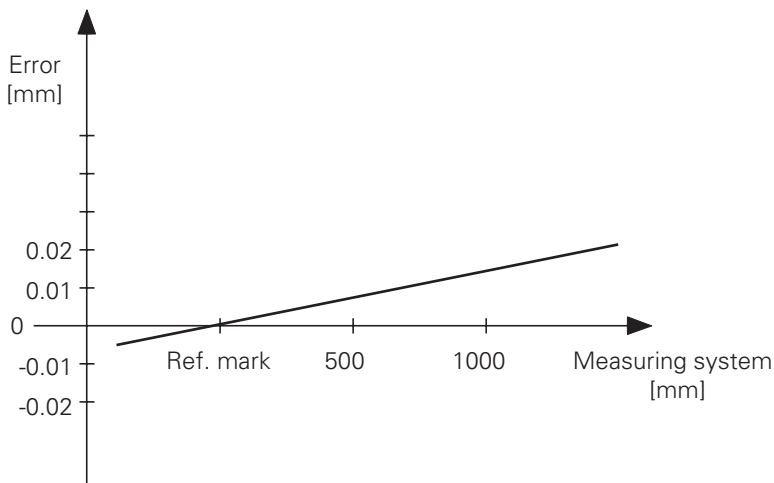
MP711.0	X axis
MP711.1	Y axis
MP711.2	Z axis
MP711.3	4th axis

MP712 Compensation value per control loop cycle time
Input value 0.000 to 1.000 [mm]

MP712.0	X axis
MP712.1	Y axis
MP712.2	Z axis
MP712.3	4th axis

1.6.3 Linear axis-error compensation

One linear axis-error can be compensated per linear axis (not rotary axes A, B or C). 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.



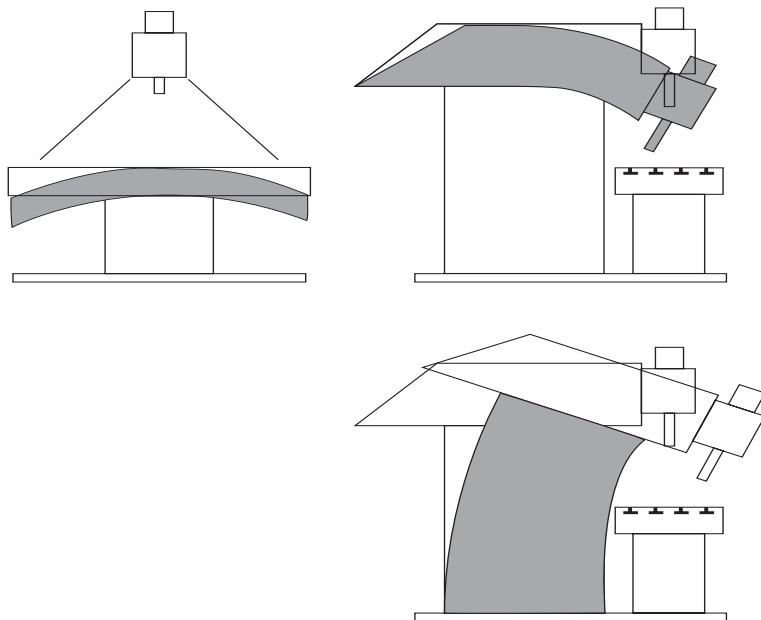
MP720 Linear axis-error compensation
Input range: - 1.000 to + 1.000 [mm/m]

MP720.0	X axis
MP720.1	Y axis
MP720.2	Z axis
MP720.3	4th axis

1.6.4 Nonlinear axis error compensation

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

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



For example, the leadscrew 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.

In the HEIDENHAIN contouring control an axis can only be corrected as a function of **one** error-related axis. So in our example either the leadscrew pitch or the sag can be compensated. For each of the four axes **one** table of corrections with 64 correction values per table can be entered. The following definitions must be fixed for this purpose:

Correlation:	Correction as a function of which axis? ($X=F(X)$; $X=F(Y)$ etc.)
Datum point:	Distance to the machine datum The error curve must always start with correction value = 0 at the datum point.
Distance:	Distance between the correction points (grid). entry of the exponent to base 2 (e.g. entry 11 = $2^{11} = 2,048$ mm). max. input value = 2^{23}

When determining the error curve with the aid of a comparator measuring instrument the above definitions must be already taken into account, whereby it is important to note that the error curve is continued in positive direction starting at the datum.

Example:

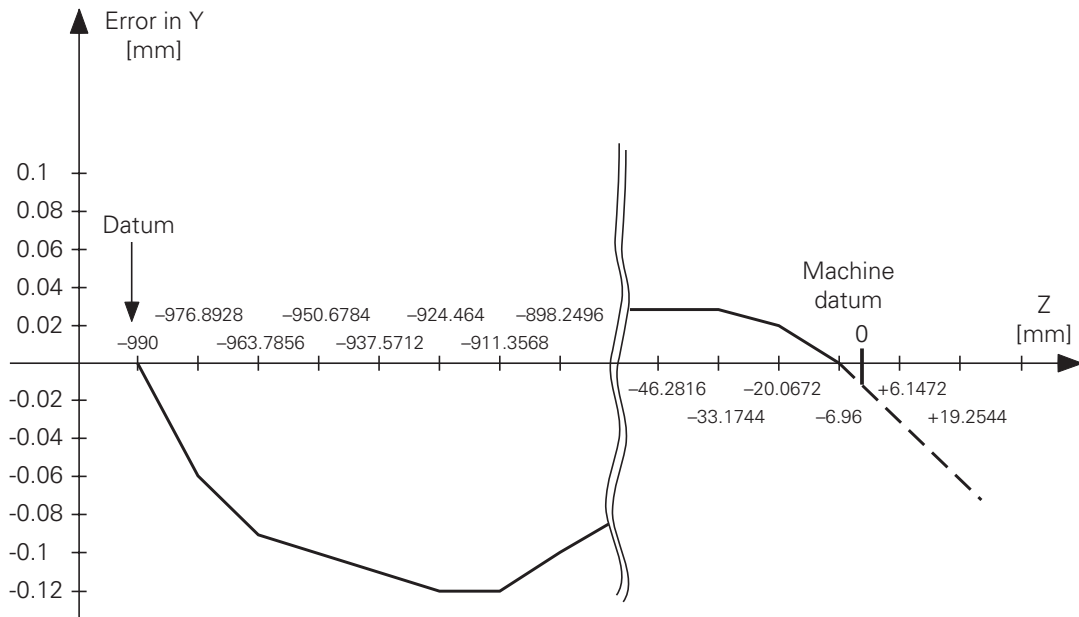
$$Y = F(Z)$$

Measurement length on Z axis = 1 000 mm

Desired distance between correction points = $1\,000\text{ mm} : 64 = 15.62\text{ mm}$

Possible exponent (base 2) = $2^{14} = 16.384\text{ mm}$

Datum point: -990



The errors which have thus been determined can be entered in the form of a table directly into the HEIDENHAIN contouring control. However, the axis-error compensation is only effective when it is enabled for a specific axis by the machine parameter MP730.

Before entering the correction table the code number 105 296 must be entered. After the code number has been entered the control will initially show the correction table for the X axis:

```

X = F(X)
DATUM POINT +0
SETUP 1
0 X+0      X+0
1 X+0.001  X+0
2 X+0.002  X+0
.
.
.

```

The orange axis keys can be used to select a different fault-related axis for the X axis. After pressing the GOTO key, the axis keys can be used to select a correction table for one of the other axes. The correction value which is entered must not exceed the maximum correction-value difference. The correction-value difference is calculated as follows:

$$\text{Max. correction-value difference} = \frac{\text{distance between correction points [mm]}}{64}$$

In our example, the result is

$$\frac{16.384 \text{ mm}}{64} = 0.256 \text{ mm}$$

Only the compensation points of the error curve should be entered. The control will automatically make a linear interpolation between the compensation points. In the example above the correction table is thus:

```
Y = F(Z)
DATUM POINT      -990
SETUP             14
0  Z-990          Y+0
1  Z-973.616      Y-0.06
2  Z-957.232      Y-0.09
3  Z-940.848
4  Z-924.464
5  Z-908.08        Y-0.12
6  Z-891.696      Y-0.12
7  Z-875.312      Y-0.1
8  Z-858.928      Y-0.09
.
.
.
```

No entry is necessary in lines 3 and 4 (press the NO ENT key), because there is no visible kink. The individual lines of the correction table are selected either by the arrow keys or with the GOTO key. Press the END ☐ key to leave the correction table.

Special case: Axis of rotation

In a correction table for a rotary axis, the entered distance between correction points must be large enough that the 64 correction points will cover a complete revolution of 360°. An angular separation of at least 5.625° (2¹⁶) is therefore necessary. For an axis of rotation, correction values will only be recognized for entries from 0 to 360°.

Input and output of the correction table via the data interface

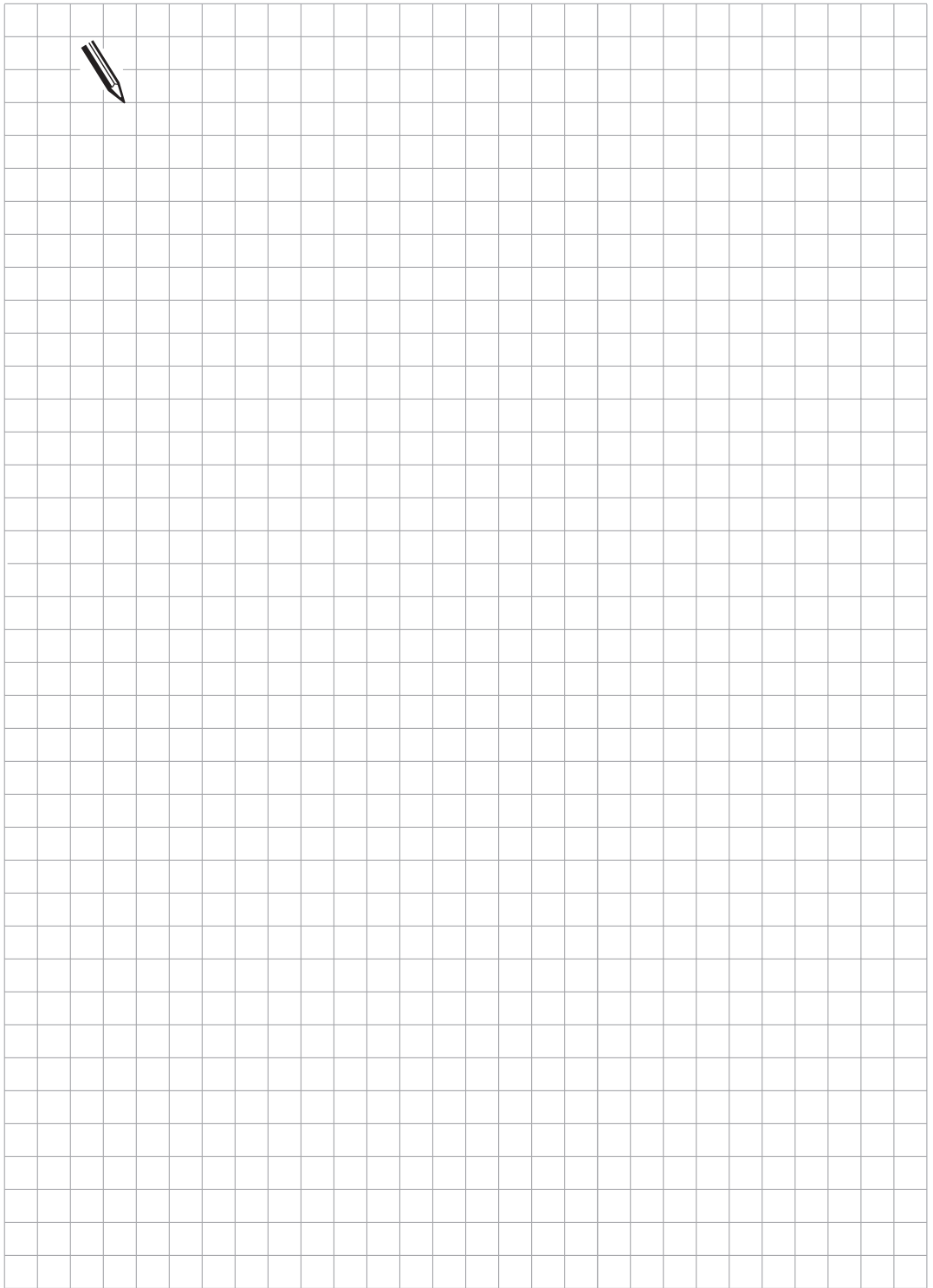
After entering the code number 105296 the correction table can be entered or read out via the data interface.

As usual, the data transfer is initiated with the EXT key. The correction table receives the file name extension ".S".

MP730

Nonlinear axis-error compensation
Input range: 0 to 15

Bit 0	X axis	+ 0 = not active + 1 = active
Bit 1	Y axis	+ 0 = not active + 2 = active
Bit 2	Z axis	+ 0 = not active + 4 = active
Bit 3	4th axis	+ 0 = not active + 8 = active



1.6.5 Temperature compensation

Exact measurements of machine thermal behavior (center of expansion in the axes, amount of expansion as a function of temperature) are necessary to compensate the effects of thermal expansion!

The words W576 to W582 were introduced for thermal expansion compensation. As soon as these words receive a value, the "follow-up tracking" becomes active.

"Follow-up tracking" means that the actual position value changes 1 µm/sec until it has changed by the full value from the words W576 to W582. This does not change the value in the actual position display.

The actual thermal expansion can be measured, for example, with a TESTOTERM unit. The values are transmitted to the PLC through the PLC inputs, further processed in the PLC program and finally transmitted to the words W576 to W582. The actual value display does not change.

	Input range: +32 767 to –32 768 [µm]
W576	Follow-up axis error compensation axis X
W578	Follow-up axis error compensation axis Y
W580	Follow-up axis error compensation axis Z
W582	Follow-up axis error compensation 4th axis

Example: Temperature compensation in the Z axis.

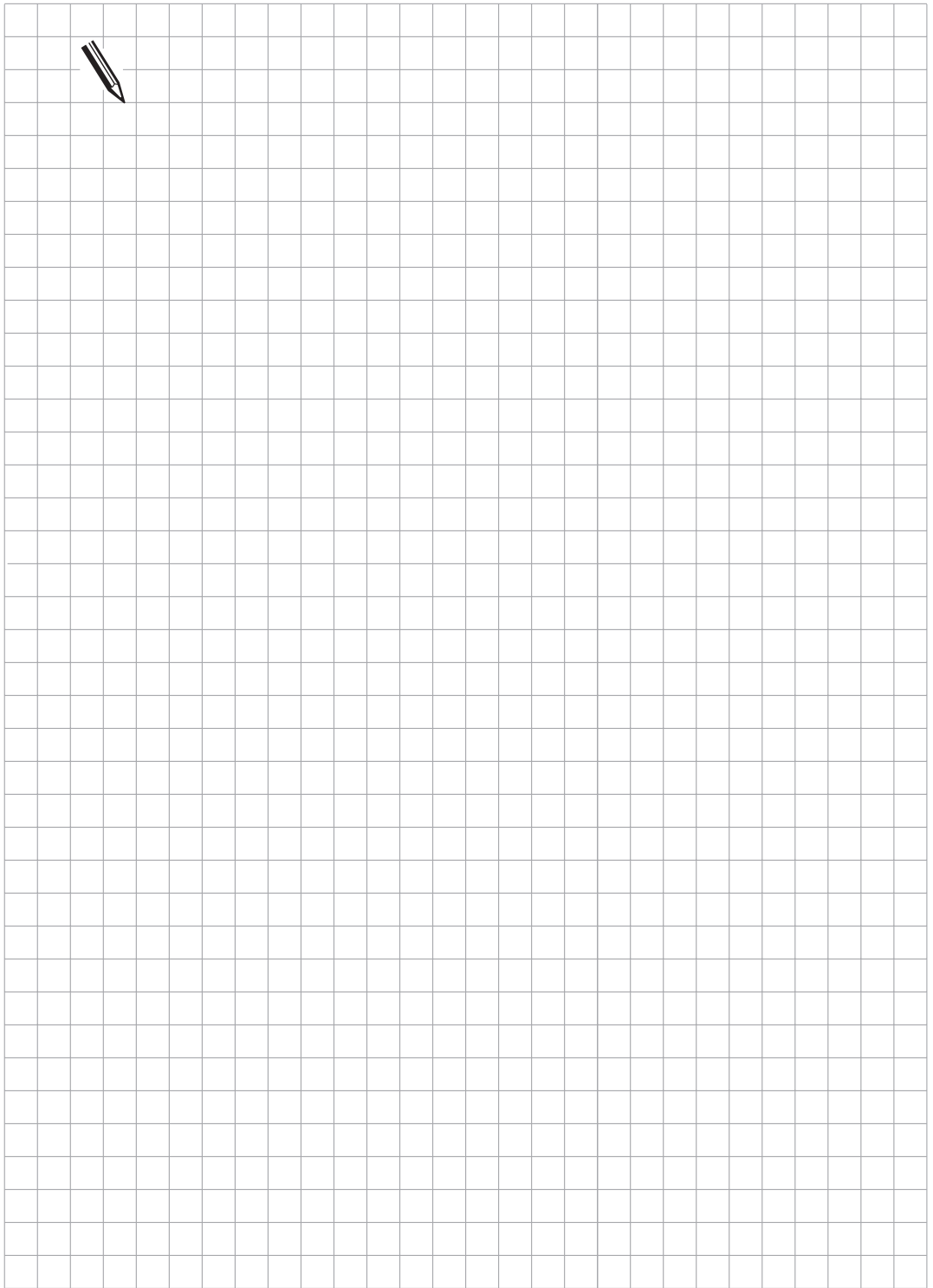
Program description

The 13 TESTOTERM-BCD outputs are transmitted at the TNC control through the PLC inputs I128 to I140 and converted to a binary data format. Then the binary value is multiplied by a correction factor (linear thermal behavior) and sent to the Z axis. The correction factor is located in MP4210.44.

Main program

L	M 0	
ON	M 0	
S	M 1	M1 is always set
LW	I 128	Read in from TESTOTERM outputs
LW	M1180	Instead of inputs, non-volatile markers
A	K+8191	Reset bits 13, 14 and 15
W=	M1200	Intermediate register TESTOTERM BCD
A	K +3840	Reset 1st, 2nd and 4th decade
>>	K +8	
X	K +100	Multiply 3rd decade by 100
=	W 252	Intermediate register TESTOTERM binary

LW	M 1200	Intermediate register TESTOTERM BCD
A	K +240	Block out 1st, 3rd and 4th decade
>>	K +4	
X	K +10	Multiply 2nd decade by 10
+	W252	Intermediate register TESTOTERM binary
=	W252	Intermediate register TESTOTERM binary
LW	M 1200	Intermediate register TESTOTERM binary
A	K + 15	Block out 2nd, 3rd and 4th decade
+	W252	Intermediate register TESTOTERM binary
=	W 248	Final register TESTOTERM binary
W=	M1220	Control register
LW	M1200	Intermediate register TESTOTERM BCD
BT	K+12	4th decade sign bit
CMT	133	Module change sign
L	W 946	Correction factor MP 4210.44 least significant word
X	W 248	Final register TESTOTERM binary
=	W580	Z axis datum shift
.		
.		
EM		
Subprogram		
LBL 133		
L-	W248	Final register TESTOTERM binary
=	W248	Final register TESTOTERM binary
W=	M1220	Control register
EM		



1.7 PLC positioning

The four axes of the control can also be positioned by the PLC. The spindle, too, can be positioned with the aid of the PLC (see section "Spindle"). The positions of the individual axes must be stored as Doublewords (D528 to D540) before activating the positioning (M2704 to M2707).

The feed for positioning the individual axes is stored in W560 to W566.

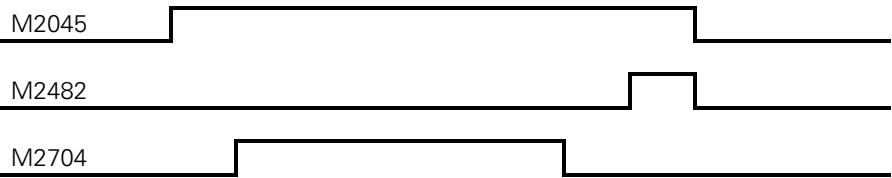
The transfer of the positions and of the feed to the PLC is carried out, for example, by machine parameters (MP4210.X, MP4220.X). All four axes can be traversed simultaneously [simultaneous activation of all the strobe markers (M2704 to M2706)]. The PLC positioning can be interrupted by resetting the strobe marker (M2704 to M2706).

The NC resets the strobe marker (M2704 to 2707) when:

- the axes have reached the given position
- NC STOP occurs in the Manual or Handwheel operating mode
- NC STOP and "internal stop" occur in the automatic operating modes
- an EMERGENCY STOP occurs
- an error message occurs that results in a stop

The strobe markers M2704 to M2707 may be set only if an M/S/T strobe is present. The M/S/T acknowledgment must not occur until the PLC positioning has been completed, i.e. markers M2704 to M2707 have been reset.

Example:



In the Manual mode of operation the strobe markers M2704 to M2707 cannot be set if an axis direction key is depressed.

The Doublewords D528 to D540 have a multiple usage. They have the following meaning for the PLC positioning:

Address	Function
D528	Position X axis [1/ 1000 mm]
D532	Position Y axis
D536	Position Z axis
D540	Position 4th axis

Feed rate for PLC positioning

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

		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		

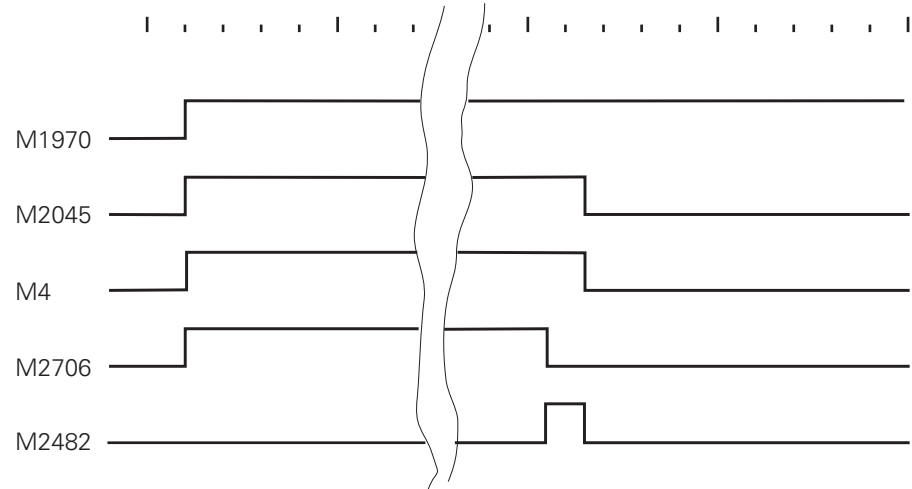


- The positions which are given are referenced to the machine datum (MP960.x).
- Software limit switches are not considered.
- Tool corrections are not calculated.
- The path compensation must be terminated before a PLC positioning.
- PLC positioning is not displayed in the test graphics.

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

```
.
.
.
67      L M0
68      ON M0
69      S M2496      ;Decoded M code output (M1900 to M1999)
70      S M2719      ;Word processing (incompatible with TNC 155/355)
71      L M1970      ;M function M70?
72      A M2045      ;Change signal M function?
73      AN M4        ;Edge marker
74      CMT 110      ;Load position and feed
75      S M2706      ;Activate PLC positioning Z axis
76      S M4         ;Edge marker
77      L M1970      ;M function M70?
78      A M2045      ;Change signal M function?
79      AN M2706      ;PLC positioning Z axis done
80      S M2482      ;Acknowledgment M function done
81      LN M2045      ;No M function?
82      R M2482      ;Acknowledgment reset
83      R M4         ;Edge marker reset
.
.
.
1210     EM          ;Main program end
1211     LBL 110      ;Load Z position and feed rate
1212     L D776        ;Load target position from MP4210.2
1213     = D536        ;Target position PLC positioning Z axis
1214     L W964        ;Load feed from MP4220.2
1215     = W564;Feed PLC positioning Z axis
1216     EM
```



1.8 PLC Axes

The controlled axes can also be assigned individually to the PLC. On a machine with three NC axes, for example, the 4th axis could be used independently from the three NC axes to position the toolchanger. Machine parameter MP60 defines which axis is to be controlled by the PLC.

The positioning of the PLC axes is commanded through a special module in the PLC program:

Module 9120: Starts the positioning of a PLC axis

Module 9121: Stops the positioning of a PLC axis

Module 9122: Polls the status of a PLC axis

Module 9123: Traverses the reference marks in a PLC axis

For a description of each module, see the chapter "PLC Programming."

If a PLC axis has been edited in the NC program, the error message "WRONG AXIS PROGRAMMED" appears during program run. PLC axes can be moved with the machine axis direction keys. PLC axes always move in servo-lag mode. Several PLC axes can be started at once. These axes cannot, however, be interpolated with each other. In MP810 a modulo value can be entered for the counting mode of the axes. An input value of 0 means that the display value is not reduced.

MP60 PLC auxiliary axes

Input: 0 to 15

Bit 0	Axis X	+0 = normal axis +1 = PLC axis
Bit 1	Axis Y	+0 = normal axis +2 = PLC axis
Bit 2	Axis Z	+0 = normal axis +4 = PLC axis
Bit 3	4th axis	+0 = normal axis +8 = PLC axis

MP810 Modulo value for reducing the position display values

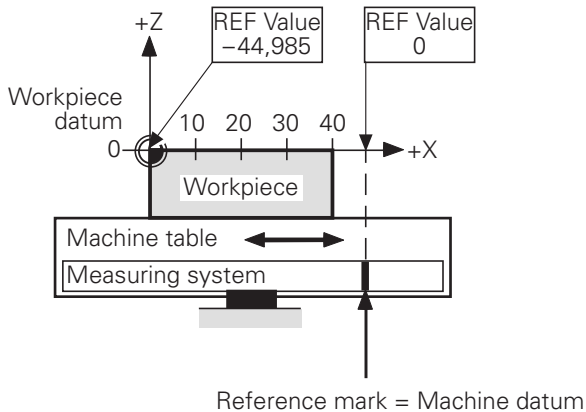
Input: 0 to 99 999.999 [mm]

MP810.0	Axis X
MP810.1	Axis Y
MP810.2	Axis Z
MP810.3	4th axis

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 encoder, this defined assignment of values to positions must be re-established after every power interruption.

The HEIDENHAIN linear encoders are therefore equipped with one or several reference marks. When a reference mark is traversed, it generates a signal that identifies the particular position as a reference point. By traversing the reference marks after a power interruption, the assignment of positional values to axis positions (and, at the same time, the fixed machine references) that were most recently determined by datum point setting are re-established.



Since it is inconvenient to restore the reference points if large traverses are required to do so, HEIDENHAIN recommends the use of encoders with distance-coded reference marks. On these measuring systems the absolute position value is available after crossing two reference marks.

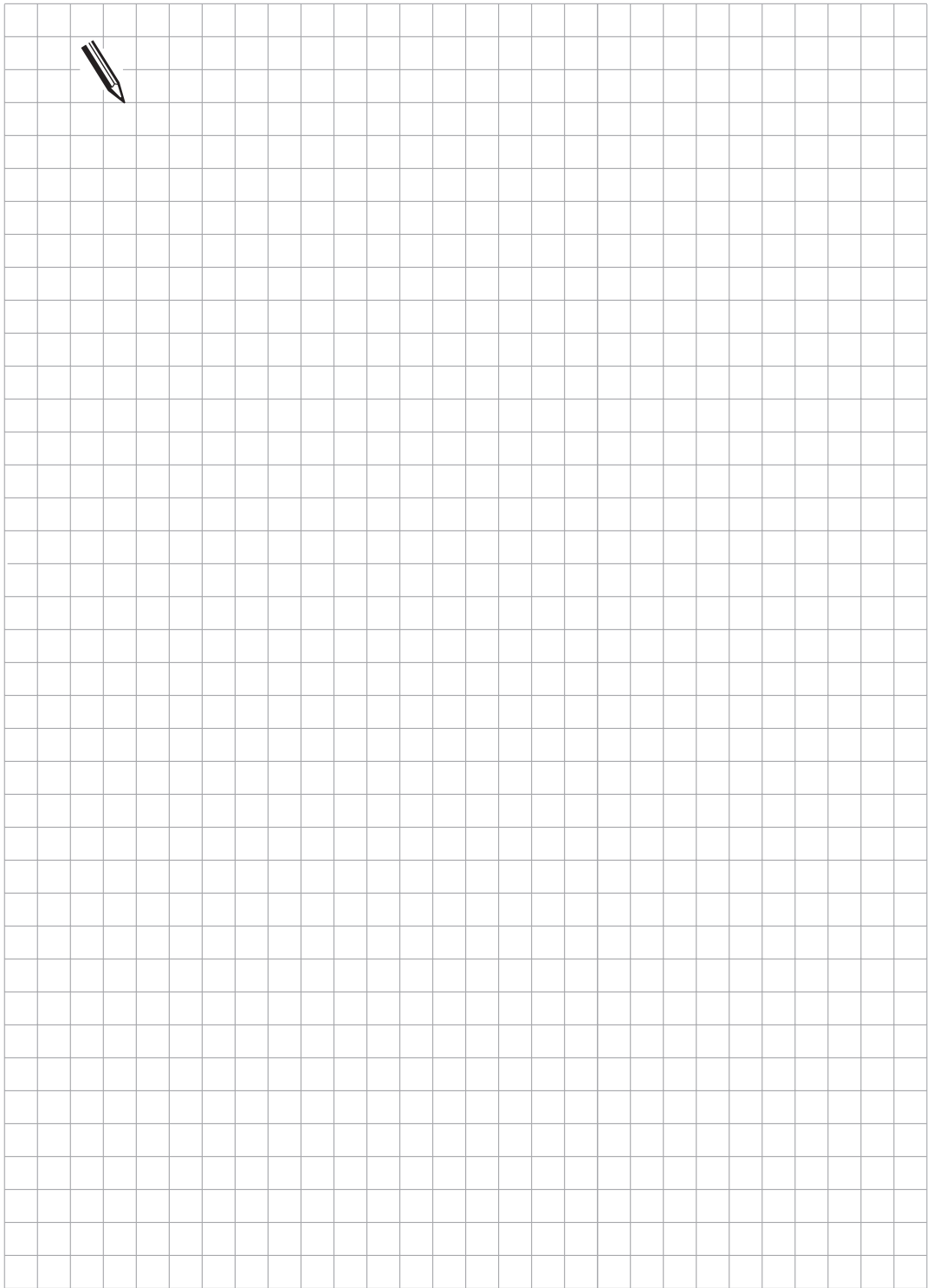
The graduation consists of an incremental grating and a parallel track for the reference marks. The absolute position of each reference mark is encoded by varying the distance between adjacent reference marks according to a mathematical algorithm.



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 traversed 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 is...

- the software limit switch activated,
- the most recently set datum point reproduced,
- PLC positioning and positioning with the miscellaneous functions M91 and M92 possible,
- the counter value set to zero for non-controlled axes.

For distance-coded measuring systems the machine datum is 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).

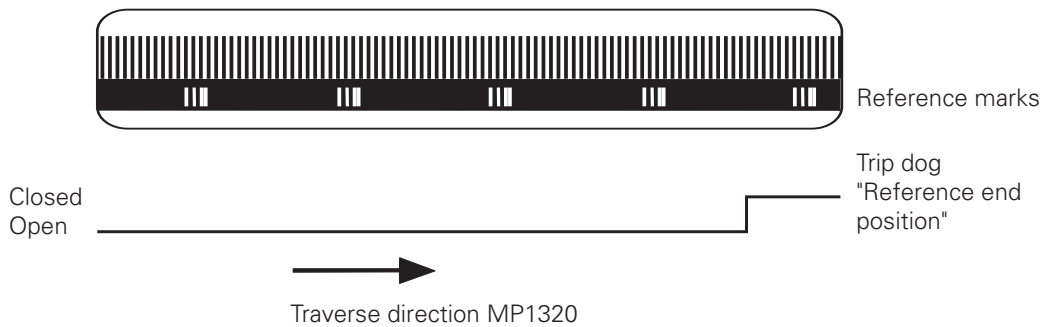
The functional sequence for passing the reference marks can be fixed specifically for each axis by machine parameters (MP1350.x).

The operating condition "PASS OVER REFERENCE MARKS" is sent to the PLC by the NC (W272).

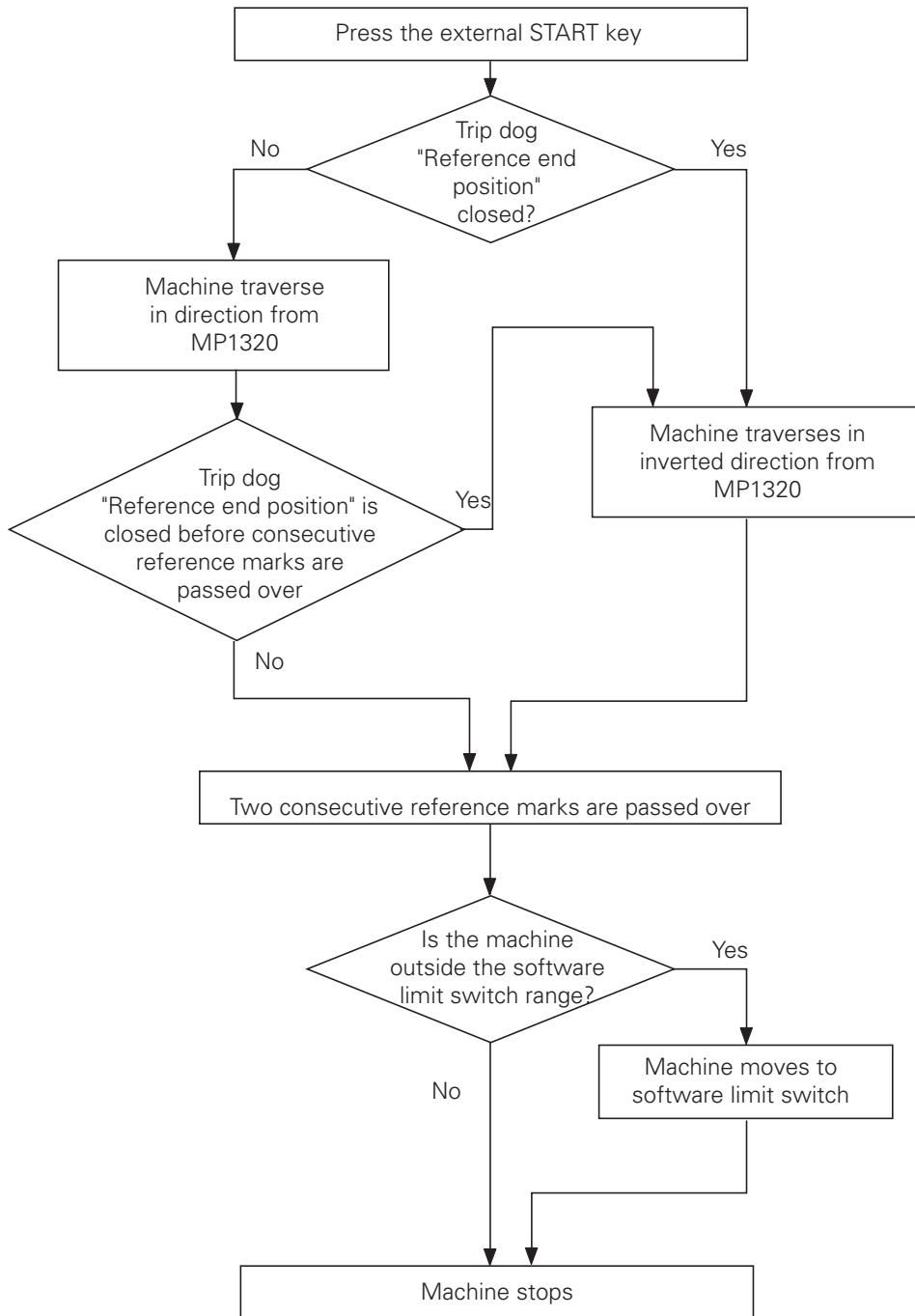
In order to avoid exceeding the traverse range when passing over the reference marks a trip dog (reference end position) is necessary. This trip dog must be fixed at the end of the traverse range by the manufacturer of the machine. The trigger signal from the trip dog is connected to an available PLC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

2.1.1 Measuring systems with distance-coded reference marks

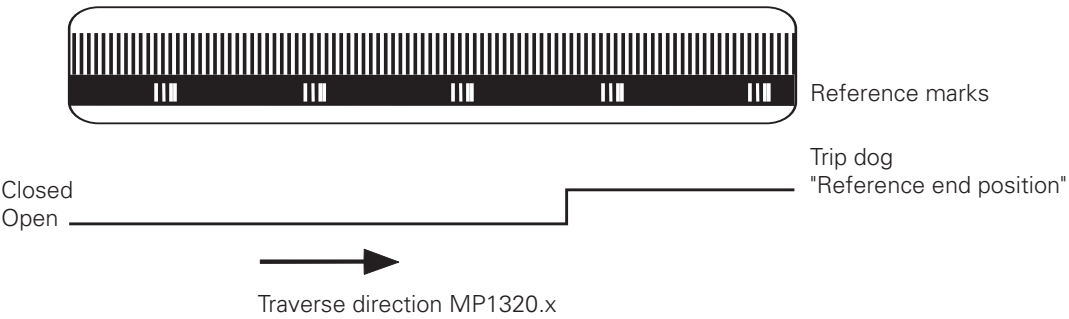
Machine parameter MP1350.x=3



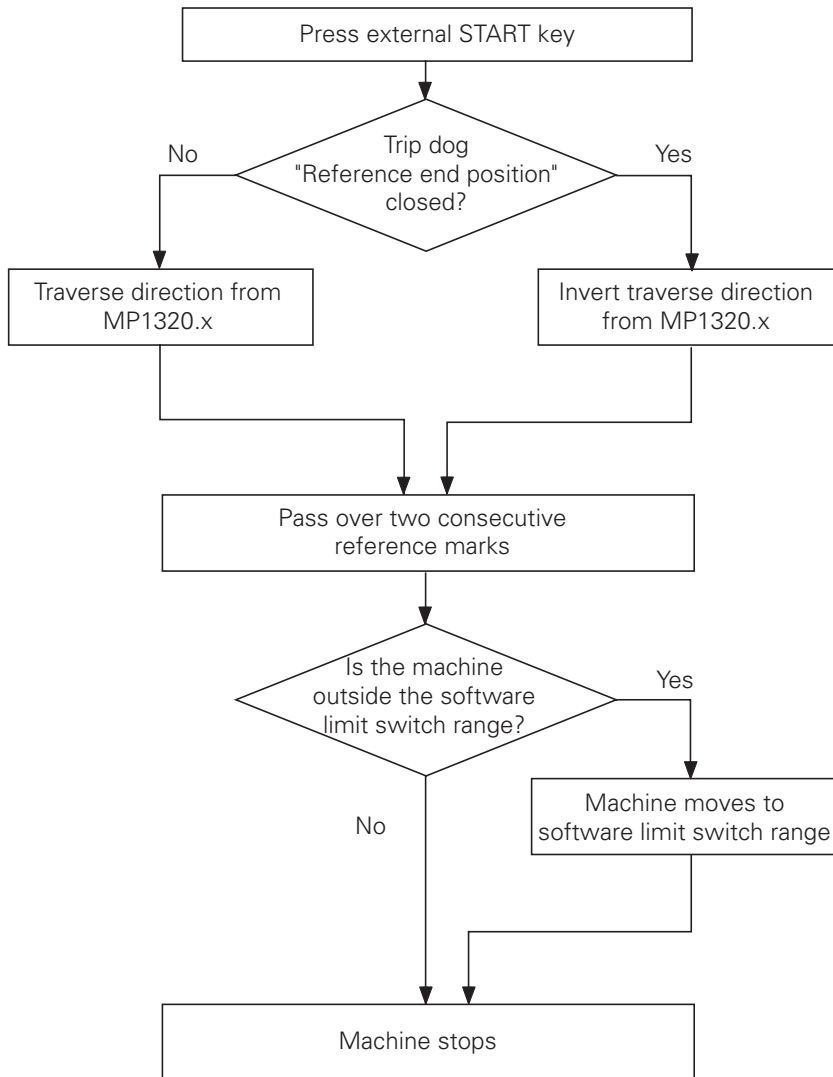
Sequence "Automatic passing over reference marks" (press the external START key).
MP1350.x = 3



Machine parameter MP1350.x=0



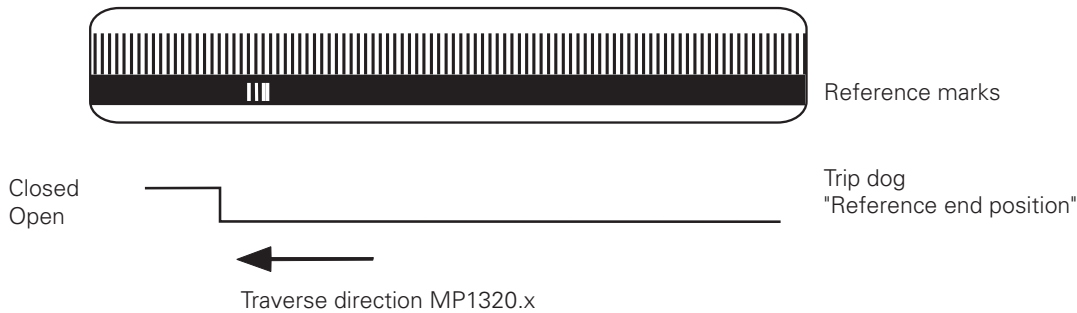
Sequence "Automatic passing over reference marks" (press the external START key).
MP1350.x = 0



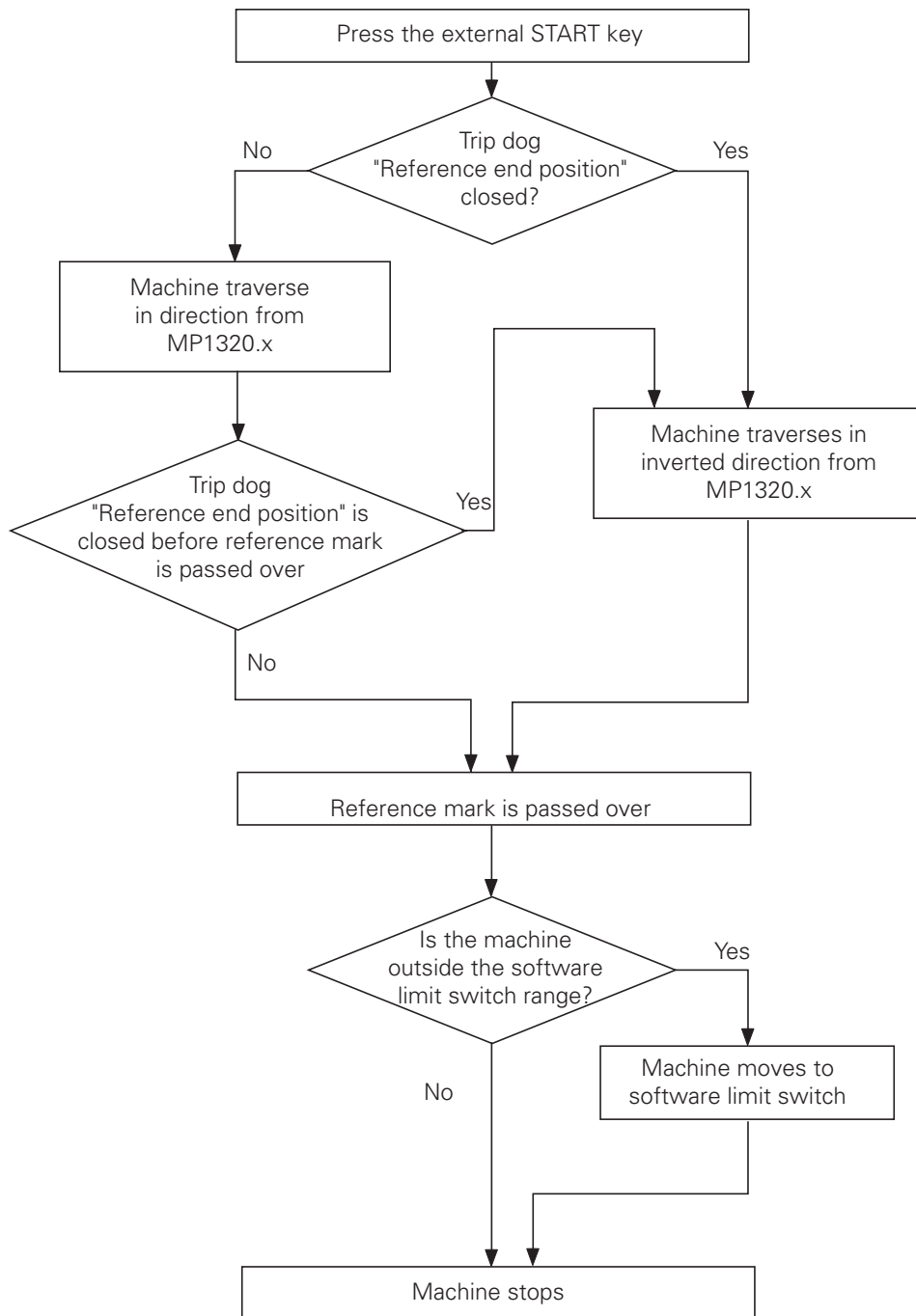
If during automatic pass-over the trip dog is not closed until it is in the "Reference end position" range, the contouring control will ignore the signal. It is therefore necessary that there be at least two reference marks in the range of the "Reference end position."

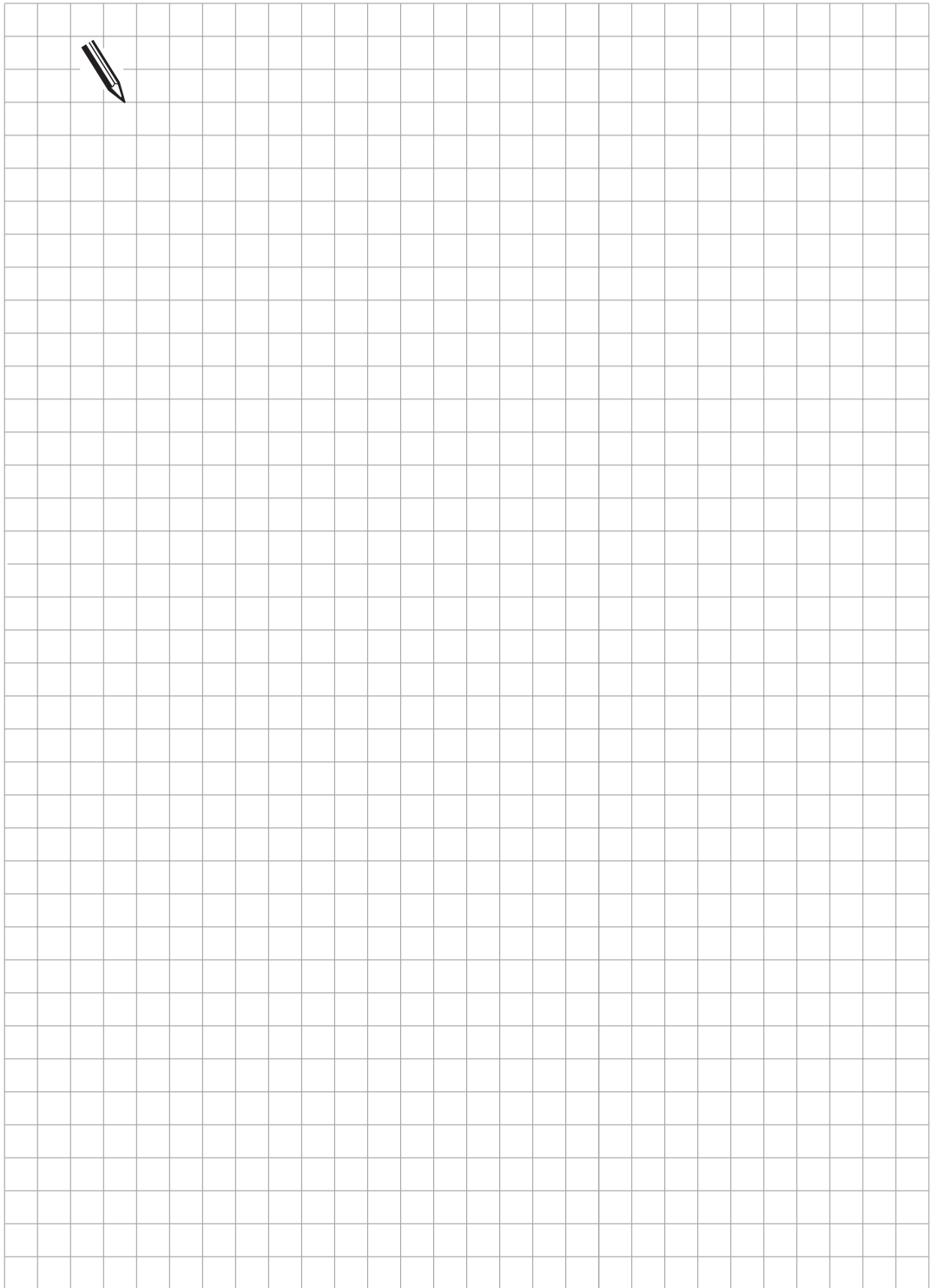
2.1.2 Measuring systems with one reference mark

Machine parameter MP1350.x=1



Sequence "Automatic passing over reference marks" (press the external START key).

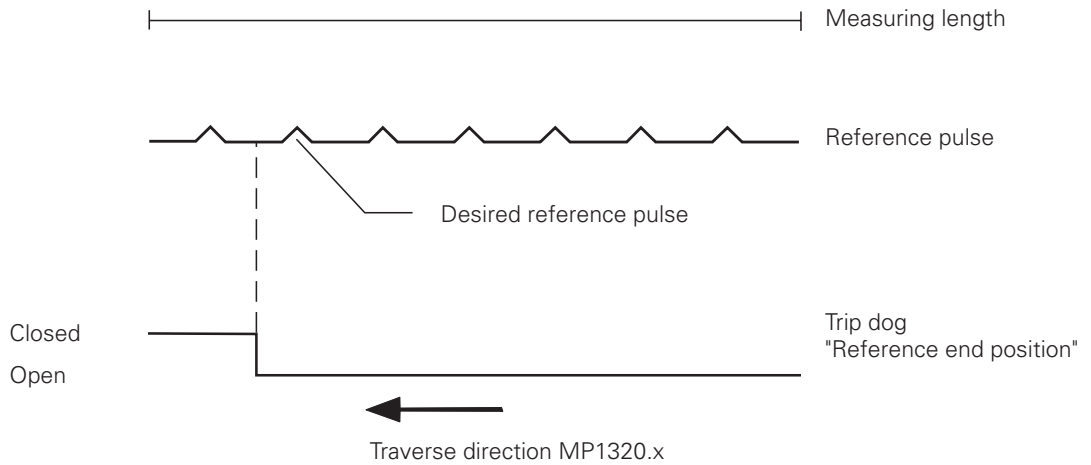




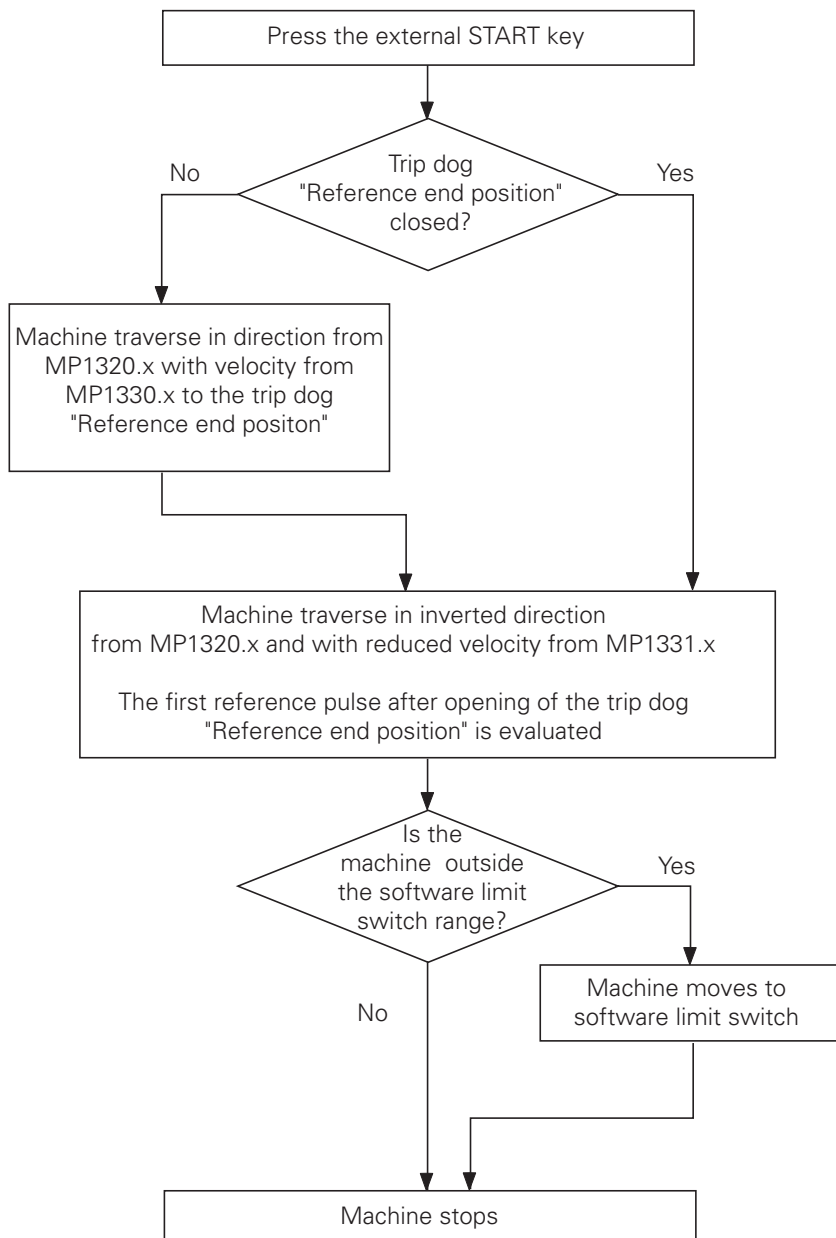
2.1.3 Linear measurement via rotary encoder

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).



MP1320 Direction for traversing the reference marks

Input range: 0 to 15

Bit 0	X axis:	+ 0 = positive + 1 = negative
Bit 1	Y axis:	+ 0 = positive + 2 = negative
Bit 2	Z axis:	+ 0 = positive + 4 = negative
Bit 3	4th axis:	+ 0 = positive + 8 = negative

MP1330 Feed rate for traversing the reference marks

Input range: 80 to 30 000 [mm/min]

MP1330.0	X axis
MP1330.1	Y axis
MP1330.2	Z axis
MP1330.3	4th axis

MP1331 Feed rate for leaving the reference end position

(only for rotary encoders MP1350=2)

Input range: 80 to 500 [mm/min]

MP1331.0	X axis
MP1331.1	Y axis
MP1331.2	Z axis
MP1331.3	4th axis

MP1340 Sequence for traversing reference marks

Entry: 0 = no evaluation of the reference mark

1 = X axis

2 = Y axis

3 = Z axis

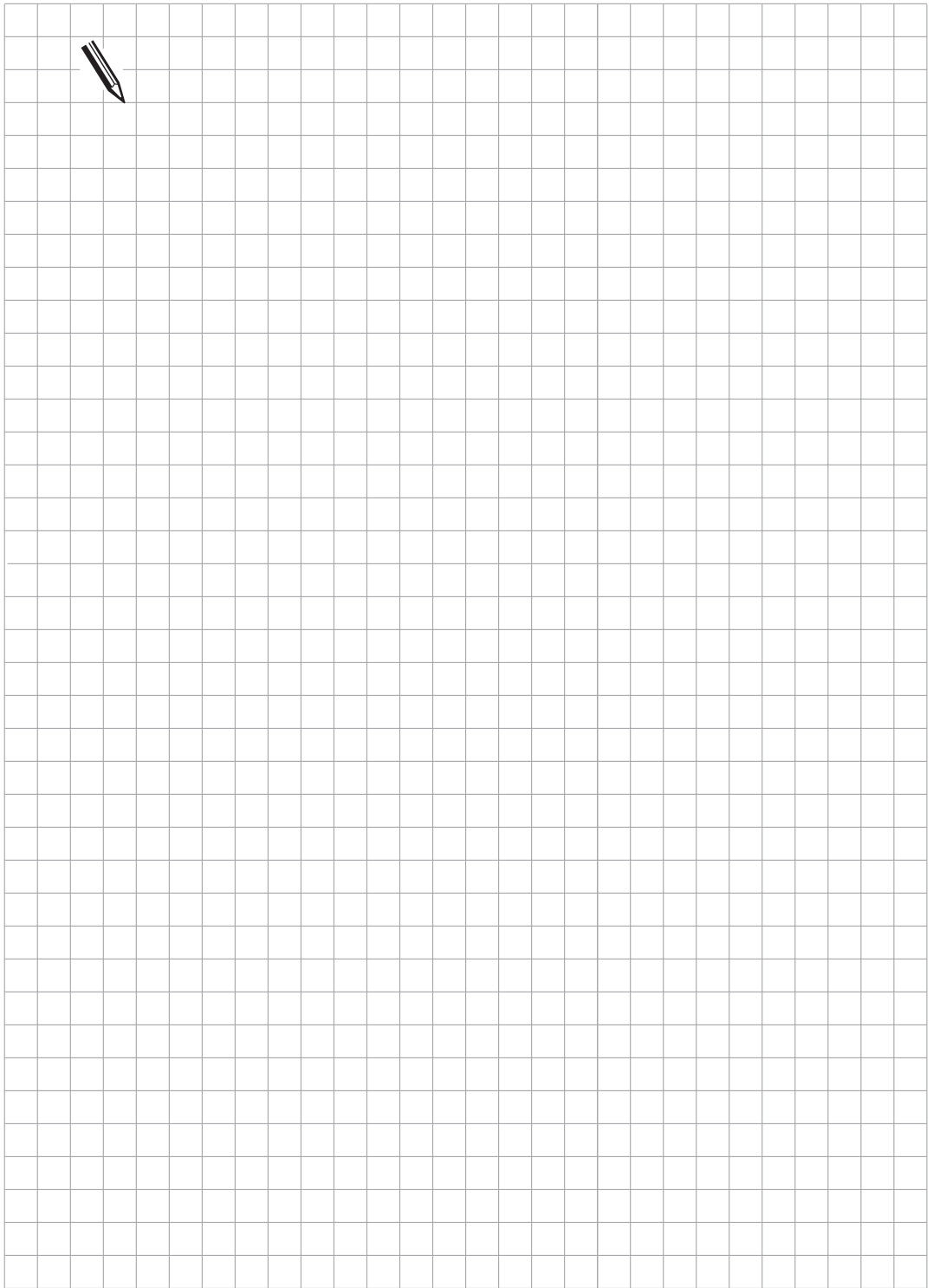
4 = 4th axis

MP1340.0	1st axis
MP1340.1	2nd axis
MP1340.2	3rd axis
MP1340.3	4th 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)	
MP1350.0	X axis		
MP1550.1	Y axis		
MP1350.2	Z axis		
MP1350.3	4th axis		
Marker	Function	Set	Reset
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 4th axis		
Address	Function	Set	Reset
W272	Operating mode	NC	NC
	0 = Programming and editing		
	1 = Manual operation		
	2 = Electronic handwheel		
	3 = Positioning with manual entry		
	4 = Program run, single block		
	5 = Program run, full sequence		
	6 = Test run		
	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. 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

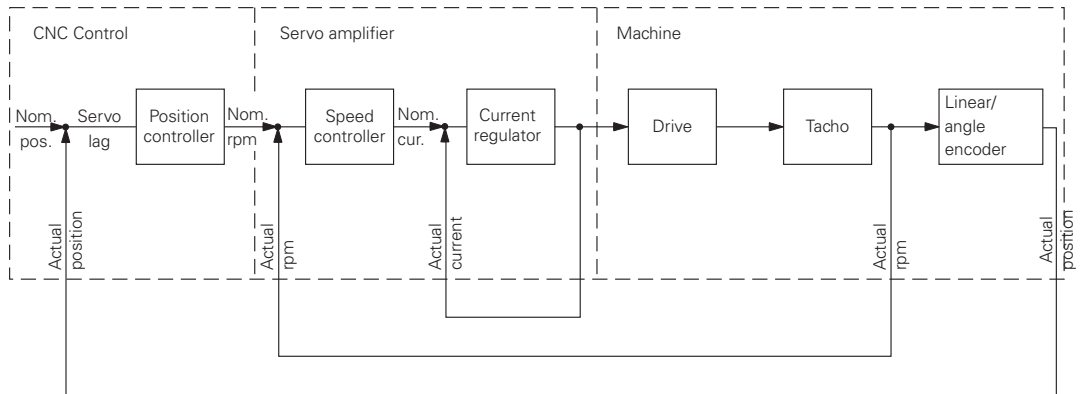
This section describes all the control functions which are important for the control and monitoring of the NC axes.

Further parameters for the NC axes can be found under "Machine axes."
The control of the spindle (S axis) is described in section "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 feed forward (section 3.2.2)

The choice of control method is determined by machine parameter MP1390.

The machine adjustment must always be carried out for both methods of control.

MP1390 Control with lag or feed forward control

Input value: 0 or 1,

1 = operation with lag,

0 = operation with feed forward control.

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.

In the operating modes "Manual operation" and "Electronic handwheel" the operation is always with servo lag, regardless of machine parameter MP1390.

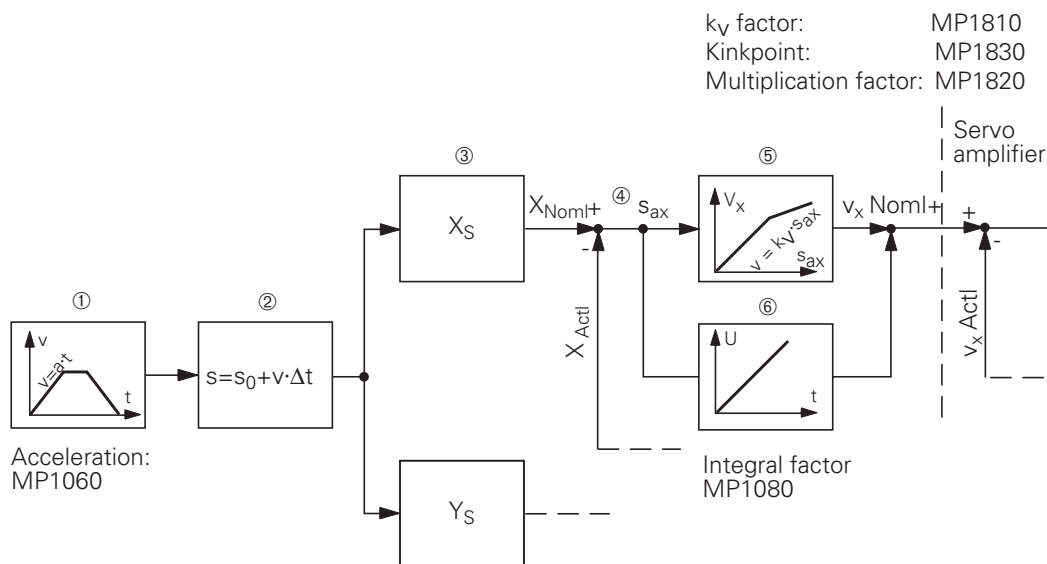
In the operating modes "Positioning with manual entry", "Program run, single block" and "Program run, full sequence" machine parameter MP1390 must be appropriately entered for lag operation.

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 feed forward 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 nominal velocity value every 6 ms from the feed programmed in the NC program and the final position (X-, Y-, Z-, 4th axis) , 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 6 ms a nominal path value is calculated from the velocity nominal value.

$$s = s_o + v \cdot \Delta t$$

s = nominal path value
 s_o = previous nominal path value
 v = nominal velocity value
 Δt = cycle time 6 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_{Actl} = 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 (analog voltage).

$$v_x = k_V \cdot s_{ax}$$

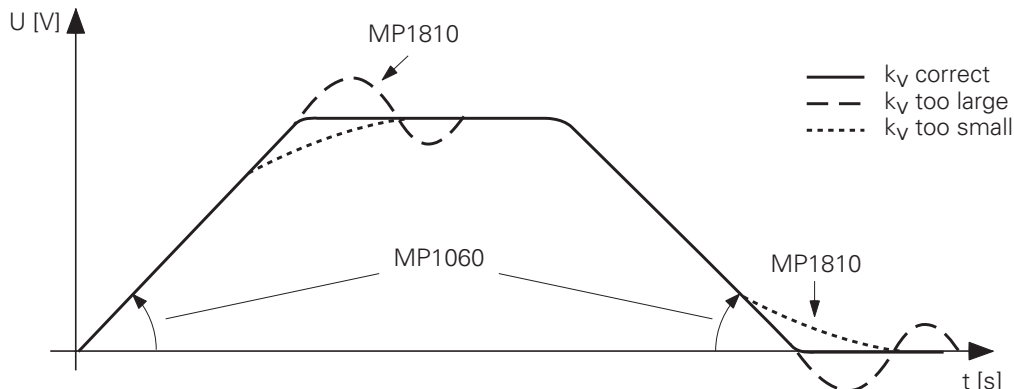
v_x = nominal velocity value for X axis

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

The k_v factor (position loop gain) determines the control loop response of the machine, and it must be matched to the machine.

If a very high k_v factor is chosen, the lag is very small. However, this can lead to an overshoot when running into a new position. If the k_v factor is too small the new position will be reached too slowly.

The optimal k_v factor must be determined by experiment (see the section "Commissioning and start-up procedure"). 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**
Input value 0.001 to 3.0 [m/s²]

MP1060.0 Acceleration X axis
MP1060.1 Acceleration Y axis
MP1060.2 Acceleration Z axis
MP1060.3 Acceleration 4th axis

The K_v factor MP1810 is, in general, determined by the rapid traverse (MP1010) of the machine and the lag, using the following formula:

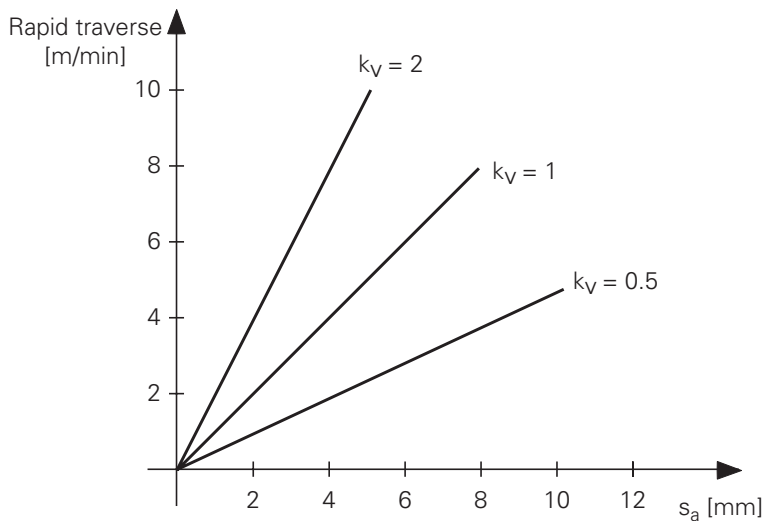
$$k_v = \frac{v_e}{s_a} \quad k_v = \text{Position loop gain} \quad \left[\frac{\text{m/min}}{\text{mm}} \right]$$

$$v_e = \text{Rapid traverse} \quad \left[\frac{\text{m}}{\text{min}} \right]$$

$$s_a = \text{Servo lag} \quad [\text{mm}]$$

or

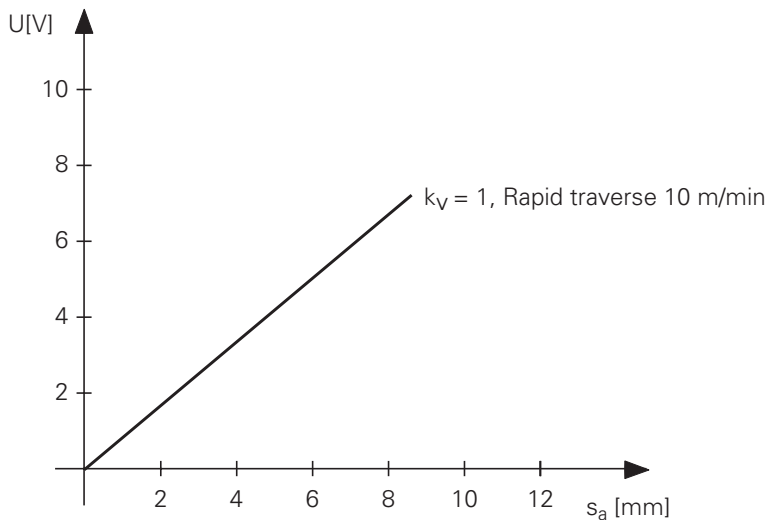
$$s_a = \frac{v_e}{k_v} [\text{mm}]$$



The rapid traverse value stored in machine parameter MP1010 can be reduced by the PLC.

The rapid traverse (maximum traversing speed) must be adjusted by the desired analog 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 analog voltage which is stored in the machine parameter MP1050.

The resulting lag error s_a thus depends on the analog 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. Unlike the rapid traverse, this has no effect on the servo behavior.

MP1810 **K_v factor for operation with lag**
Input range 0.1 to 10 [1/min]

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

MP1010 **Rapid traverse**
Input range 80 to 300 000 [mm/min]

MP1010.0	Rapid traverse X axis
MP1010.1	Rapid traverse Y axis
MP1010.2	Rapid traverse Z axis
MP1010.3	Rapid traverse 4th axis

MP1050 **Analog voltage for rapid traverse**
Input range 4.5 to 9 [V]

MP1050.0	Analog voltage X axis
MP1050.1	Analog voltage Y axis
MP1050.2	Analog voltage Z axis
MP1050.3	Analog voltage 4th axis

MP1020 **Manual feed**
Input range 80 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

Address	Function
D596	Rapid traverse from PLC

Servo accuracy:

The internal servo accuracy of the control is always 1 µm. The control has, however, a display step of 5 µm or 1 µm, selectable by machine parameter MP7290. The rounding up or down is only in the display (see section "Display and operation").

Normally a servo accuracy of 1 µm is sufficient, i.e. the control must be able to generate at least one voltage step per 1 µm positional deviation.

Calculation of the smallest voltage step:

The controls produce an analog voltage 0 to 10 V.

In the TNC 360 the 10 V analog 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 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).

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

If ΔU is divided by the smallest voltage step which can be produced (0.6 mV), the result is the number of voltage steps which are produced per 1 μm positional deviation.

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

Example:

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

$$s_a = \frac{5\,000 [\mu\text{m}]}{2} = 2\,500 [\mu\text{m}]$$

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

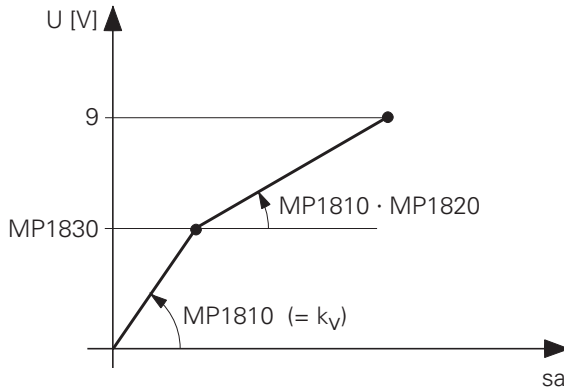
$$n = \frac{3.6 [\text{mV}/\mu\text{m}]}{0.6 [\text{mV}]} = 6 \text{ steps}/\mu\text{m positional deviation}$$

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 positioning speeds (stop, machining feed rates, rapid traverse). In such cases one can introduce a kink point, which gives the following advantages:

- a high k_v at lower positioning speeds, i.e. a larger voltage step per μm of positional deviation
- a low k_v at higher positioning speeds (above the machining rates).

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

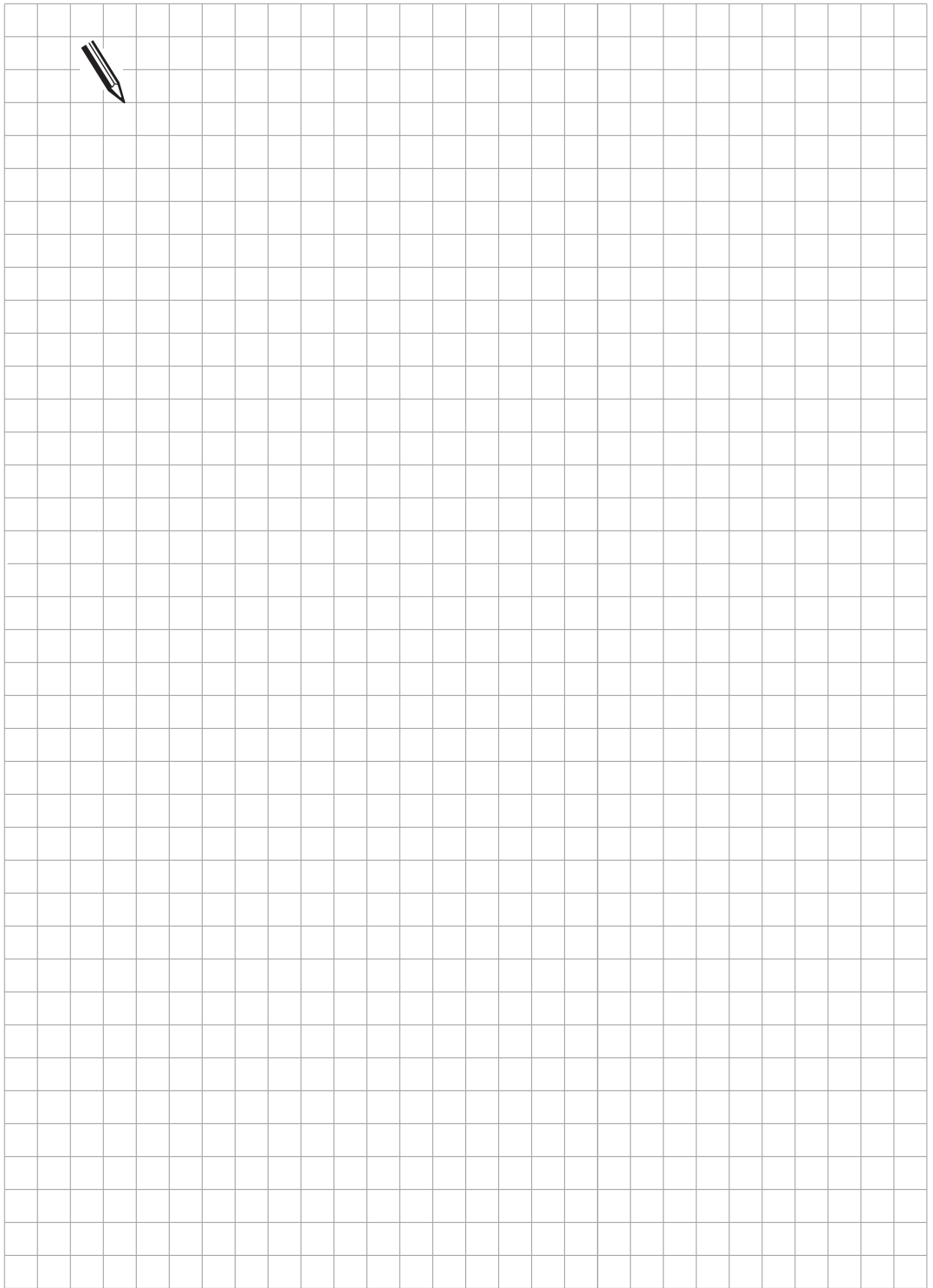


The kink point must lie above the range of machining feeds!
Under these conditions, the lag can be calculated as follows:

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

MP1820 **Multiplication factor for the K_v factor**
Input value 0.001 to 1

MP1830 **Kink point**
Input value 0 to 100 [%]



3.2.2 Control with feed forward

Feed forward 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 feed forward control has the advantage that contours can be followed very accurately at a high speed.

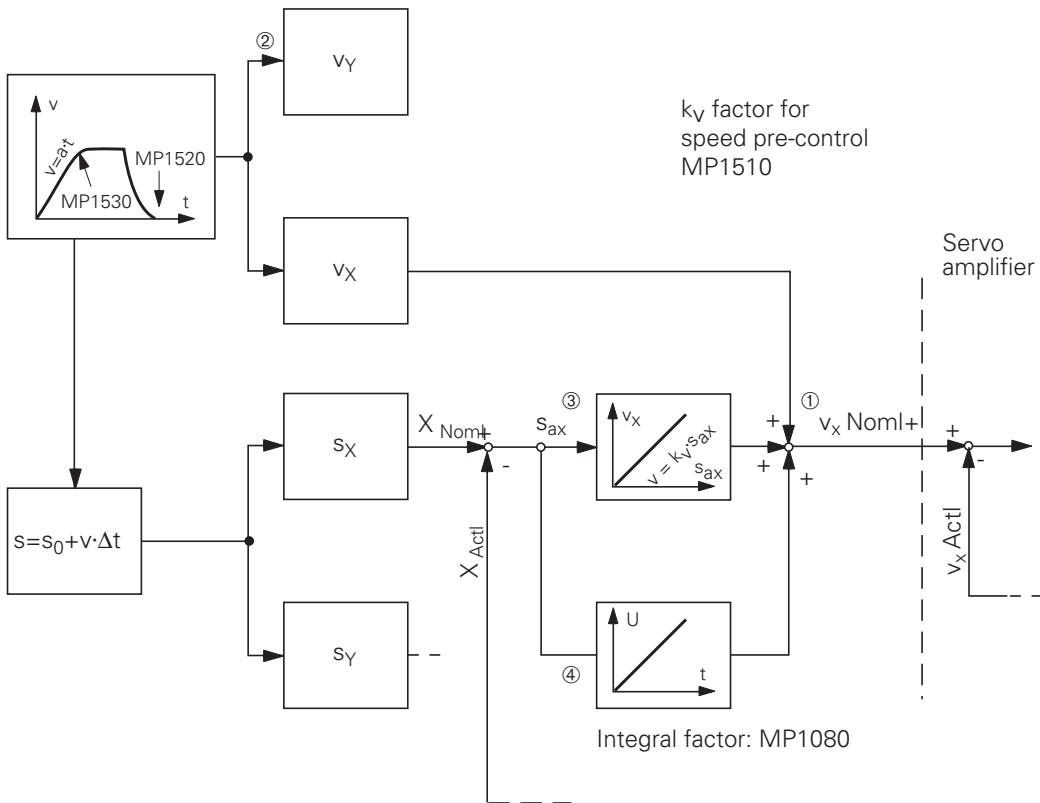
Normally, work will be carried out using feed forward control.

In the operating modes "Positioning with manual data input", "Program run, single block" and "Program run, full sequence" this method of servo control can be selected with the machine parameter MP1390.

Operation with feed forward control for the X axis can be shown in a simplified form in the following block diagram.

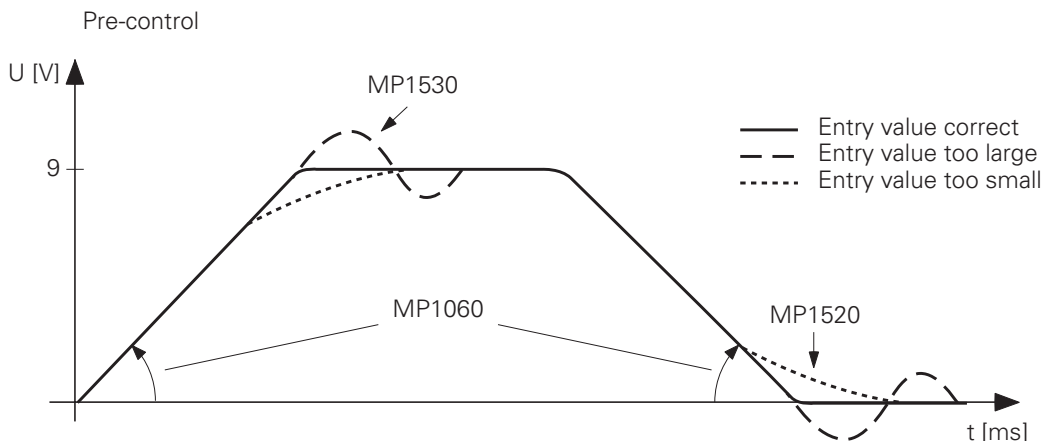
All machine parameters which influence the servo characteristic are shown here.

Acceleration: MP1060
 Transient response: MP1530
 Position approach speed: MP1520



- ① When operating with feed forward control the set velocity value (the analog potential for the drive amplifier) is formed from three components:
feed forward control,
servo lag,
integral component.
- ② The feed forward 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 feed forward control then comes into play.
- ④ If the axes are stopped, the integral factor will also be effective:
it results in an offset correction.

The feed forward control value is adjusted to the dynamics of the machine by three machine parameters.

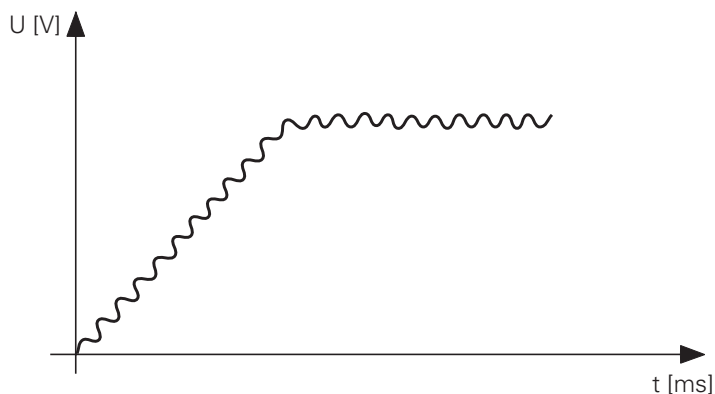


The machine parameter MP1060 determines the acceleration (= slope of the pre-controlled speed curve).

The machine parameter MP1530 determines the transient response. The larger the value which is entered, the more the system will tend to oscillate.

MP1520 determines the run-in behavior into the nominal position. The larger the value which is entered, the more the system will tend to oscillate.

The fine control of the pre-controlled speed set-value is influenced by the K_V factor for feed forward control MP1510.



If the K_V factor is too large, the system will oscillate about the pre-controlled speed set value.

The size of the K_V factor is normally the same as the K_V factor for operation with servo lag. Depending on the dynamic characteristics of the machine it may also be higher (see section "Commissioning and start-up procedure").

MP1530 Damping factor for transient response

Input value: 0.01 to 0.999

MP1520 Position approach speed

Input value: 0.1 to 10 [m/min]

MP1510 K_V factor for feed forward control

Input value: 0.1 to 10 [1/min]

MP1510.0	K_V factor X axis
MP1510.1	K_V factor Y axis
MP1510.2	K_V factor Z axis
MP1510.3	K_V factor 4th axis

3.3 Offset adjustment

The TNC 360/TNC 335 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"

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, IV, in the dialog line. The values indicate the voltage in 0.6 mV units. Thus a display of 10 means $10 \times 0.6 \text{ mV} = 6.0 \text{ mV}$. The display 0 means no offset. The displayed offset value consists of the offset values that originate in the motor controller and in the control.

On pressing the "ENT" key the offset values, which are displayed on the VDU, are automatically compensated, the control puts out an appropriate compensating voltage. Compensation only takes place if the offset voltage is $\geq 0.6 \text{ mV}$.

To switch off the automatic offset adjustment, enter the code number and press the "NO ENT" key. The offset values are stored in the control and are non-volatile.

After a control is exchanged, the offset adjustment must be re-activated via code number.

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 compensation will be in steps of 0.6 mV.

MP1220 Automatic cyclical offset adjustment

Input value: 0 to 65536 [s]

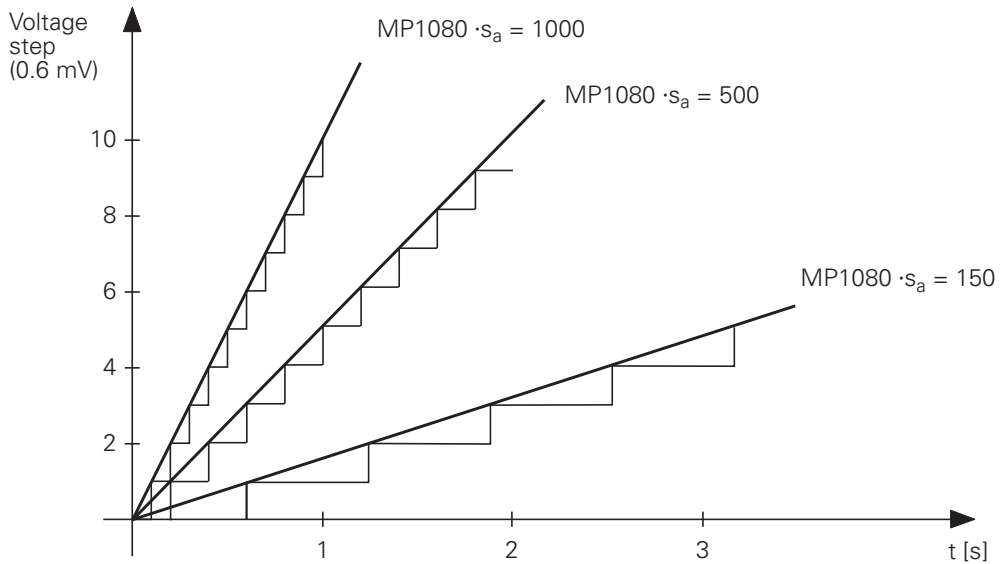
0 = no automatic adjustment

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.

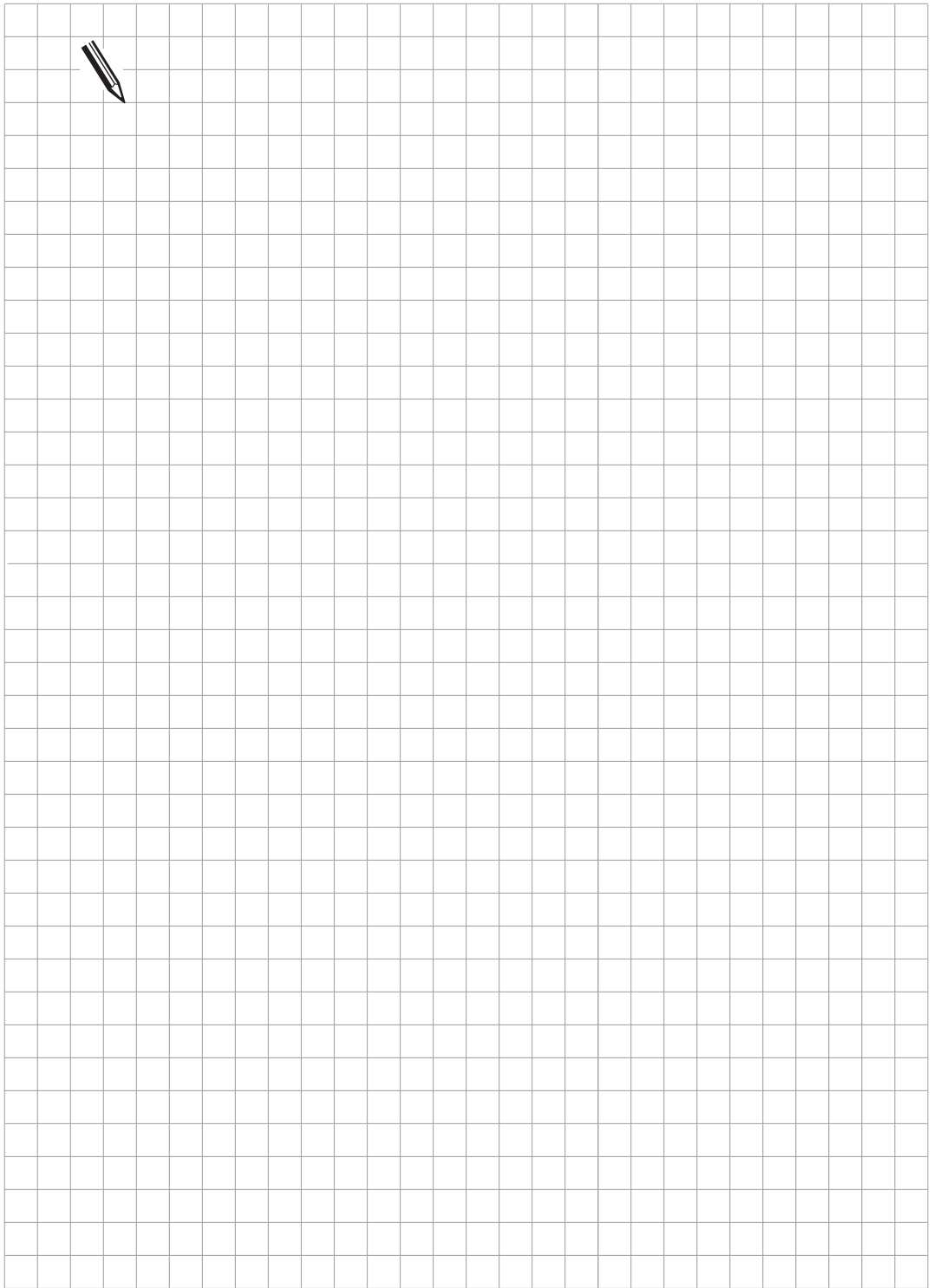


MP1080

Integral factor

Input value: 0 to 65535

MP1080.0	Integral factor X axis
MP1080.1	Integral factor Y axis
MP1080.2	Integral factor Z axis
MP1080.3	Integral factor 4th axis



3.4 Contour behavior in corners

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:

$$v = \sqrt{r \cdot \text{MP1070}}$$

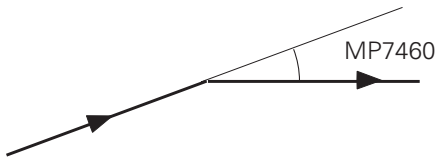
v = feed rate for circular movements [m/s]
 r = radius [m] (cutter mid-point contour)

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 feed forward control.

MP1070 **Radial acceleration**
Input value: 0,001 to 3.0 [m/s²]

3.4.2 Constant feed rate in corners

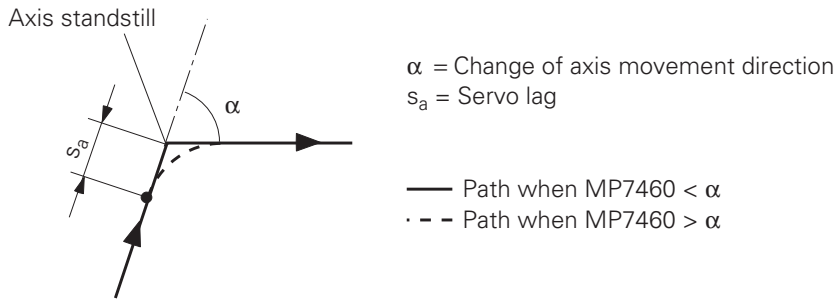
Machine parameter MP7460 defines the angle which can still be traversed with constant surface speed. This machine parameter is effective for corners without a radius compensation, for internal corners it is also effective with a radius compensation.



This machine parameter is effective for operation with feed forward 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:

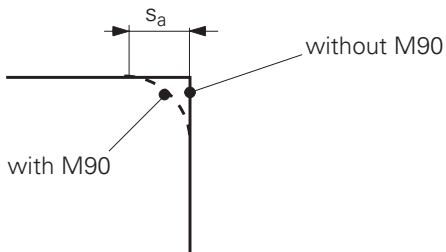


MP7460 **Constant feed rate in corners**
 Input value: 0.001 to 179.999°

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!



3.5 Monitoring functions

The NC monitors the axis positions and the dynamic behavior of the machine. If the fixed values in the machine parameters are exceeded, an error message is displayed and the machine is stopped. Position, standstill, movement and analog voltage are monitored.

If, for some reason, monitoring is not wanted at all, the following markers must be set in the PLC program:

		Set	Reset
M2688	No monitoring X axis	PLC	PLC
M2689	No monitoring Y axis		
M2690	No monitoring Z axis		
M2691	No monitoring 4th 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 a 6 ms cycle time. 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.

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

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

0 = Activation for Low level
1 = Activation for High level

		Set	Reset
W522 Bit 0	Monitoring functions suppressed if PLC input MP 4130 is activated from MP4130	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 canceled 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 canceled by switching off the control. An input 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 (cancelable) Input value: 0.001 to 1000 [mm]
---------------	---

MP1720	Position monitoring for operation with lag (EMERGENCY STOP) Input value: 0.001 to 1000 [mm]
---------------	---

3.5.2 Position monitoring for operation with feed forward control

In operation with feed forward 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

"Positioning error"

will appear, which can be canceled 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 canceled by switching off the control.

If clamped axes cause the error message "Positioning 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 canceled by "CE" before the time has elapsed, the nominal/actual value transfer is carried out and the residual voltage is switched off. This machine parameter functions in operation with lag as well as with feed forward control.

MP1410 **Position monitoring in operation with feed forward control (cancelable)**
Input value: 0.001 to 30 [mm]

MP1420 **Position monitoring in operation with feed forward control (EMERGENCY STOP)**
Input value: 0.001 to 30 [mm]

MP1150 **Delay time before switching off the residual voltage on error message "Positioning error"**
Input value: 0 to 65.535 [s]

3.5.3 Monitoring the analog voltage

Monitoring of the analog voltage is only possible in operation with feed forward 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 feed forward control as well as with lag.

During movement monitoring, the actual path traveled is compared with a nominal path calculated by the NC at short intervals (several servo cycles). If, during this period, the actual path traveled 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

Input value: 0.03 to 10 [V]

3.5.5 Standstill monitoring

The Standstill monitoring functions in operation with feed forward control and with lag. The monitoring functions 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 the axis moves in the opposite direction on beginning a positioning movement.

MP1110 **Standstill monitoring**
Input range: 0.001 to 30 [mm]

3.5.6 Positioning window

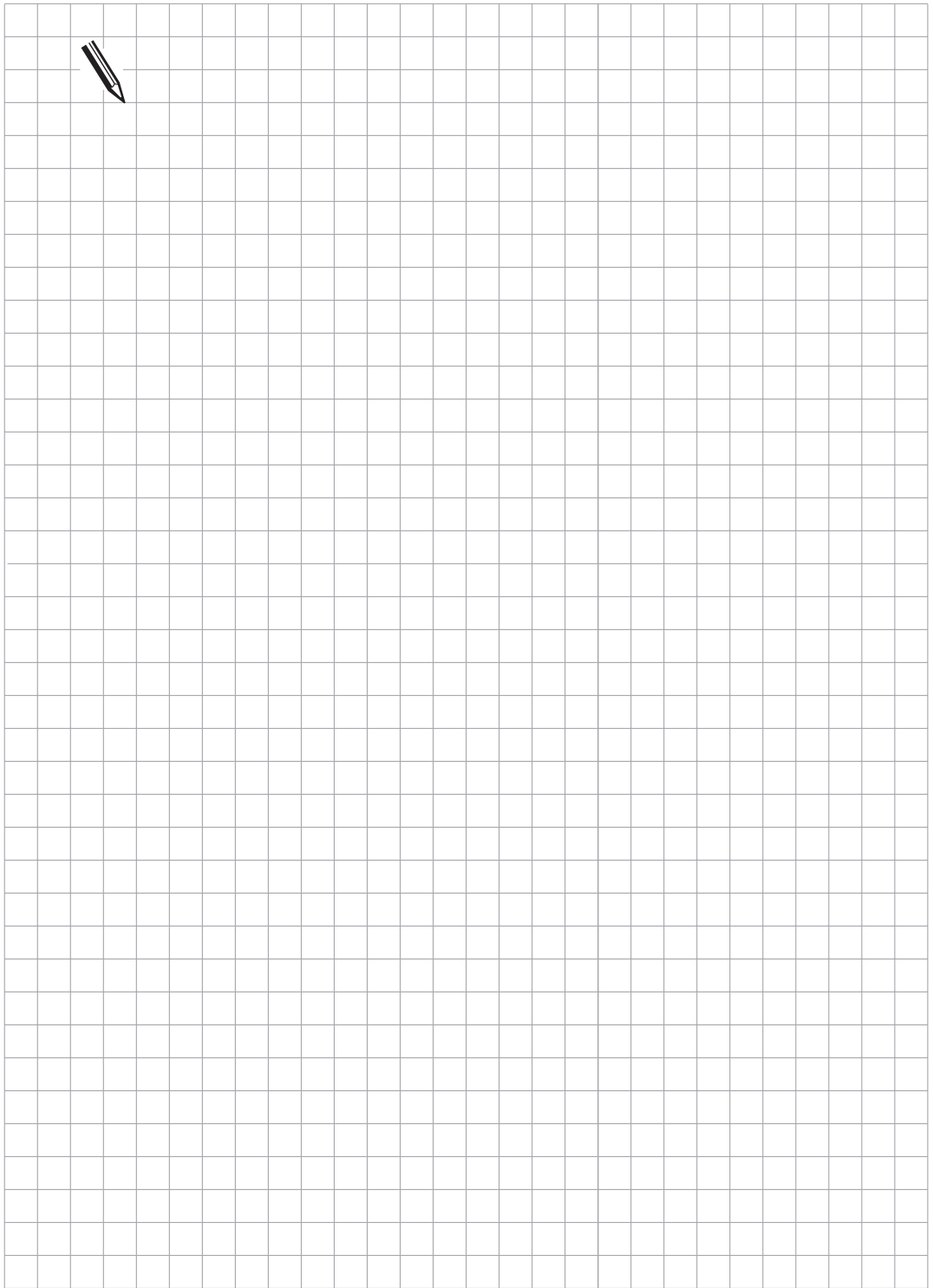
The positioning window defines the limits within which the control considers that a position has been reached. After reaching the position the control starts the execution of the next block.

The size of the positioning window is defined in machine parameter MP1030.

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**
Input range 0.001 to 2 [mm] or [°]
MP1030.0 X axis
MP1030.1 Y axis
MP1030.2 Z axis
MP1030.3 4th axis



3.6 Controlled axes

The machine parameter MP50 determines which of the four NC axes should be controlled.

MP50		Controlled axes
		Input value 0 to 15
Bit 0	X axis	+0 = not controlled +1 = controlled
Bit 1	Y axis	+0 = not controlled +2 = controlled
Bit 2	Z axis	+0 = not controlled +4 = controlled
Bit 3	4th axis	+0 = not controlled +8 = controlled

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").

		Set	Reset
M2000	Axis enable X	NC	PLC, NC
M2001	Axis enable Y		
M2002	Axis enable Z		
M2003	Axis enable 4		

The feed rate enable must be specified for all axes by M2451 and complement marker M2467. If the feed rate enable is removed, the analog voltage is output as 0 V and the axes immediately stop moving. The letter "F" is then highlighted in the status display.

		Set	Reset
M2451	Feed rate enable for all axes	PLC	PLC
M2467	Complement feed rate enable		

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.

The markers will be reset by the NC only if the axes leave the positioning window when being traversed in manual operation or automatic operation. This is also valid when passing over the reference marks. In the "Electronic handwheel" mode of operation the markers M2008 to M2011 are reset.

The markers also inform the PLC when an axis should be moved. A positioning block resets the corresponding in-position marker. This makes it possible to unclamp the axis and close the control loop.

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		

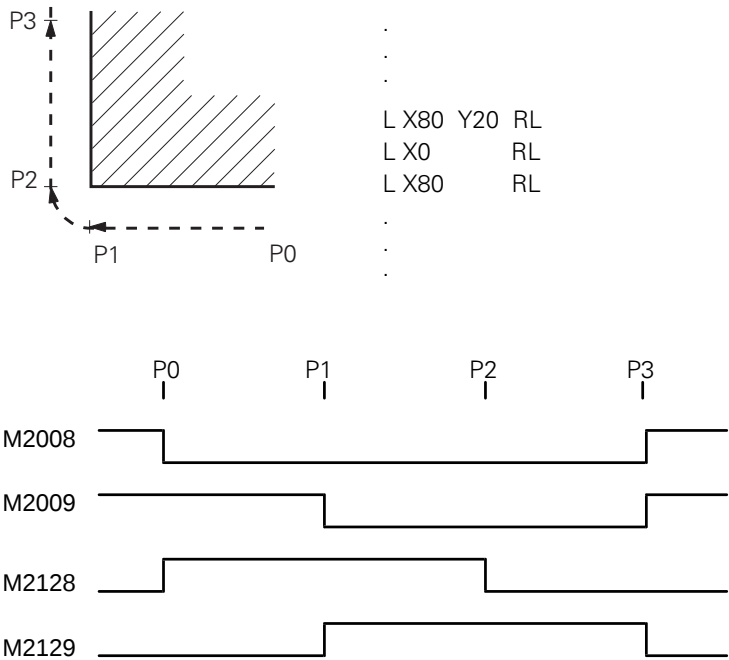
3.6.3 Axes in motion

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."

		Set	Reset
M2128	X axis in motion	NC	NC
M2129	Y axis in motion		
M2130	Z axis in motion		
M2131	4th axis in motion		

Example for markers "Axis in position" and "Axis in motion":



3.6.4 Open control loop

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 were introduced for 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) has been set. After the control loops are opened the markers M2000 to M2003 are reset by the NC.

As soon as the "Axis in position" marker is reset, the control loop must re-closed so that the axes can be moved again.

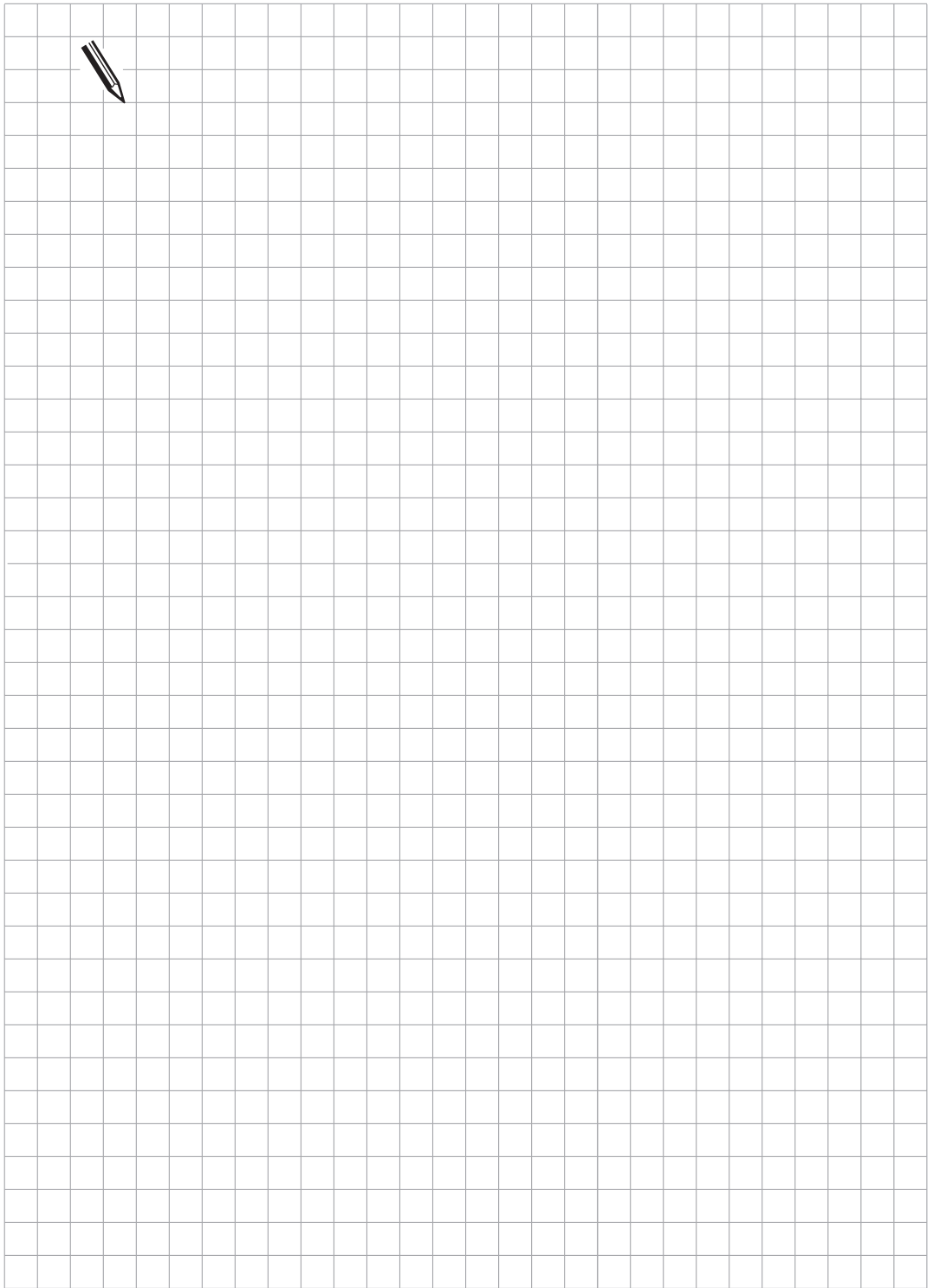
		Set	Reset
M2544	Open control loop X axis	PLC	PLC
M2545	Open control loop Y axis		
M2546	Open control loop Z axis		
M2547	Open control loop 4th axis		
M2492	Await open control loop X axis	PLC	PLC
M2493	Await open control loop Y axis		
M2494	Await open control loop Z axis		
M2495	Await open control loop 4th axis		

3.6.5 Actual - nominal value transfer

If the markers M2552 to M2555 are set, then the actual position value will be transferred to memory as the nominal position value.

Actual value transfer is possible only in the "Manual" and Electronic handwheel" modes of operation.

		Set	Reset
M2552	Actual - nominal value transfer X axis	PLC	PLC
M2553	Actual - nominal value transfer Y axis		
M2554	Actual - nominal value transfer Z axis		
M2555	Actual - nominal value transfer 4th axis		



4 Spindle

The spindle is controlled by the PLC.

The spindle speed (S output) can be either an analog voltage output or a BCD-coded output. The spindle speed is programmed in rpm in the NC program.

The machine parameter MP3010 determines whether the spindle speed is a BCD-coded output or an analog 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

Input value: 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 = Analog output of the spindle speed and gear-change signal,
only when the gear range is changed

4 = Analog output of the spindle speed and gear-change signal
at every TOOL CALL

5 = Analog output of the spindle speed and no gear-change signal

6 = As for Input value 3, but with controlled spindle for orientation

7 = As for Input value 4, but with controlled spindle for orientation

8 = As for Input value 5, but with controlled spindle for orientation

The functions are described in detail in the following sections.

4.1 Analog output of the spindle speed

For analog output of the spindle speed (S analog) spindle speeds can be programmed from 0 to 99 999.999 rpm. The programmed spindle speed is available as an analog 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 analog the value 991 must be entered in the machine parameter MP3020!

If the control is operating with S analog, 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, Word	Function	Set	Reset
M2042	Analog 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-analog 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. the polarity of the analog voltage is inverted.

The S-analog 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 analog 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-analog voltage will then go to 0 V.

MP3130 Polarity of the S-analog voltage

Input value: 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

		Set	Reset
M2489	Change direction of spindle rotation	PLC	PLC
M2485	Status display and output of the analog voltage for M03	PLC	PLC
M2486	Status display and output of the analog voltage for M04	PLC	PLC
M2487	Status display M05 and spindle stop	PLC	PLC
M2608	Inverse display of M03, M04, M05 and S analog = 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-analog 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 producible analog output voltage.

If an S-analog 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-analog 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
Input value: 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

MP3210 S-analog voltage for gear range
Input value: 0 to 9.999 [V]

- MP3210.0 S-analog voltage, gear range 1
- MP3210.1 S-analog voltage, gear range 2
- MP3210.2 S-analog voltage, gear range 3
- MP3210.3 S-analog voltage, gear range 4
- MP3210.4 S-analog voltage, gear range 5
- MP3210.5 S-analog voltage, gear range 6
- MP3210.6 S-analog voltage, gear range 7
- MP3210.7 S-analog voltage, gear range 8

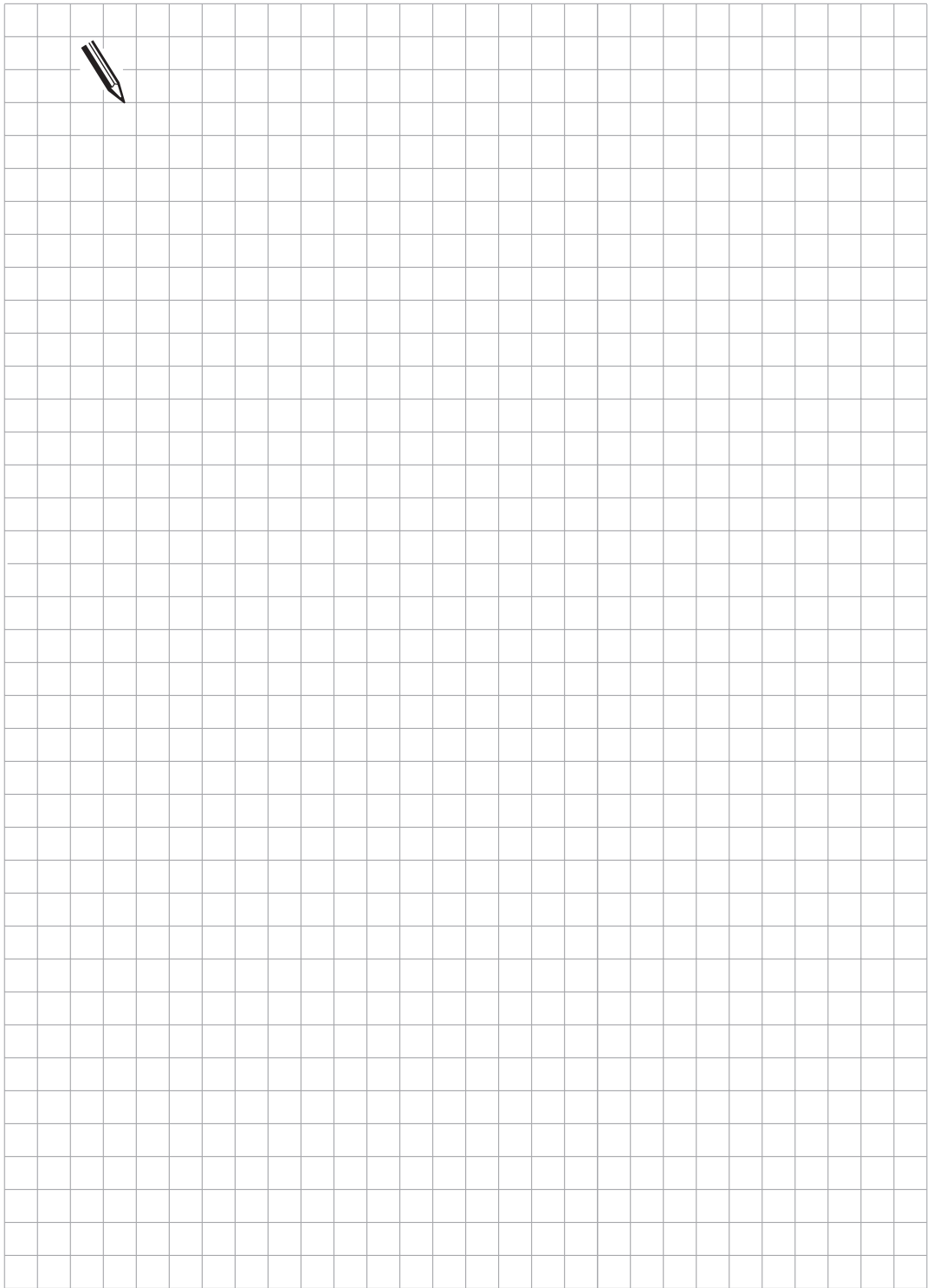
MP3240.0 Maximum S-analog voltage output
Input value: 0 to 9.999 [V]

MP3240.1 Minimum S-analog voltage output
Input value: 0 to 9.999 [V]

MP3120 Zero spindle speed permitted
Input value 0:S = 0 permitted
Input value 1:S = 0 not permitted

MP3410.0 Ramp slope for spindle for M03, M04, M05
Input range: 0 to 1.999 [V/ms]

		Set	Reset
M2004	S-analog voltage not on ramp	NC	NC
M2005	S-analog voltage = 0 V	NC	NC
M2020	Actual spindle speed below minimum rpm from MP3240.1	NC	NC



4.1.3 Spindle override

The spindle speed can be altered within certain limits by the spindle override potentiometer. These limits are fixed by machine parameter MP3310. However, the upper spindle speed limit from MP3240 cannot be exceeded by spindle override.

The % factor which is adjusted by spindle override is kept in Word W492 and W764. However, W764 can 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 nonlinear characteristic curve. In the nonlinear curve, 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 curve, and between 0 and 15 000 for the nonlinear curve.

MP3310.0 Limit with spindle override (maximum)

Input value: 0 to 150 [%]

MP3310.1 Limit with spindle override (minimum)

Input value: 0 to 150 [%]

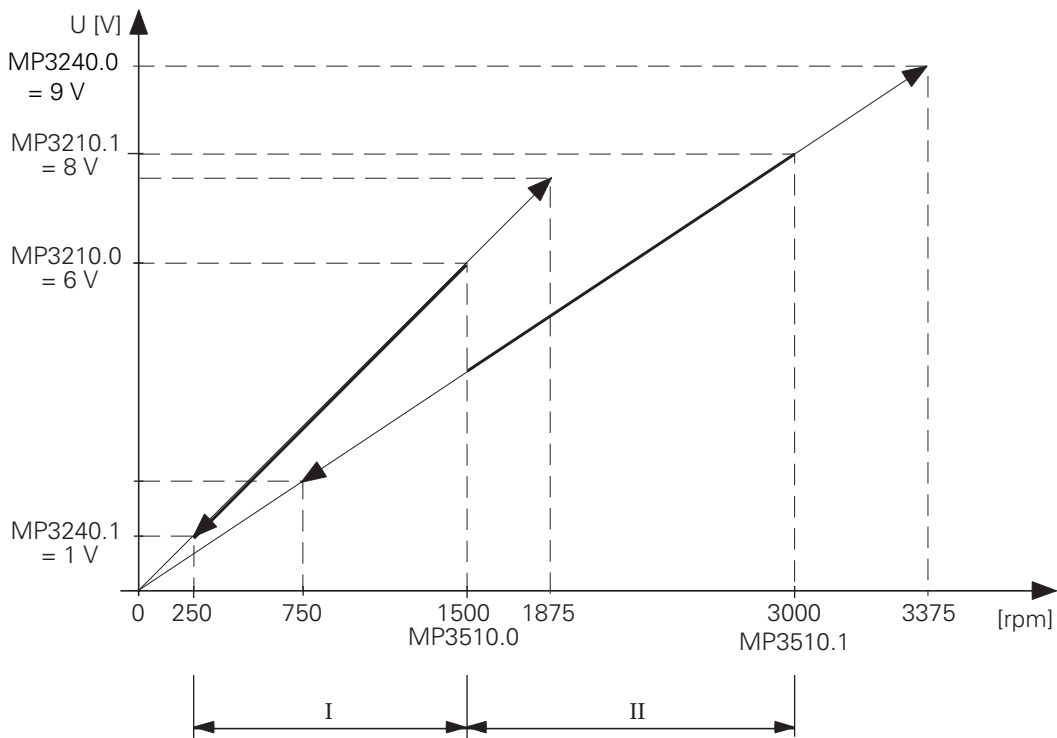
Word	Function	Set	Reset
W492	% factor – spindle override (NC → PLC)	NC	NC
W764	% factor – spindle override (PLC → NC)	NC; PLC	NC; PLC

MP7620 Bit 3 Spindle override in 1% steps or according to a non linear characteristic curve

+0 = 1 % steps

+8 = nonlinear curve

The following diagram illustrates the relationships:



Gear range I:	1 500 rpm at 6 V	(MP3510.0; MP3210.0)
Gear range II:	3 000 rpm at 8 V	(MP3510.1; MP3210.1)
Upper spindle override limit:	125 %	(MP3310.0)
Lower spindle override limit:	50 %	(MP3310.1)
Maximal permissible output voltage:	9 V	(MP3240.0)
Minimal permissible output voltage:	1 V	(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 strobe 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 the PLC setting Markers M2104 – M2106 and Marker M2814. 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 acknowledgment 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.

		Set	Reset
M2043	Gear-code change signal	NC	NC
M2480	Acknowledgment "Gear change completed"	PLC	PLC
M2104	1st bit gear code for S analog (lsb)	PLC	PLC
M2105	2nd bit gear code for S analog	PLC	PLC
M2106	3rd bit gear code for S analog (msb)	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 analog	NC	

An alternating S-analog 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.

MP3240.2 Jog voltage for gear change

Input value 0 to 9.999 [V]

		Set	Reset
M2490	Spindle rotation left (for gear change)	PLC	PLC
M2491	Spindle rotation right (for gear change)	PLC	PLC

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 appears:

Wrong spindle speed

		Set	Reset
M2092	Illegal spindle speed	NC	NC

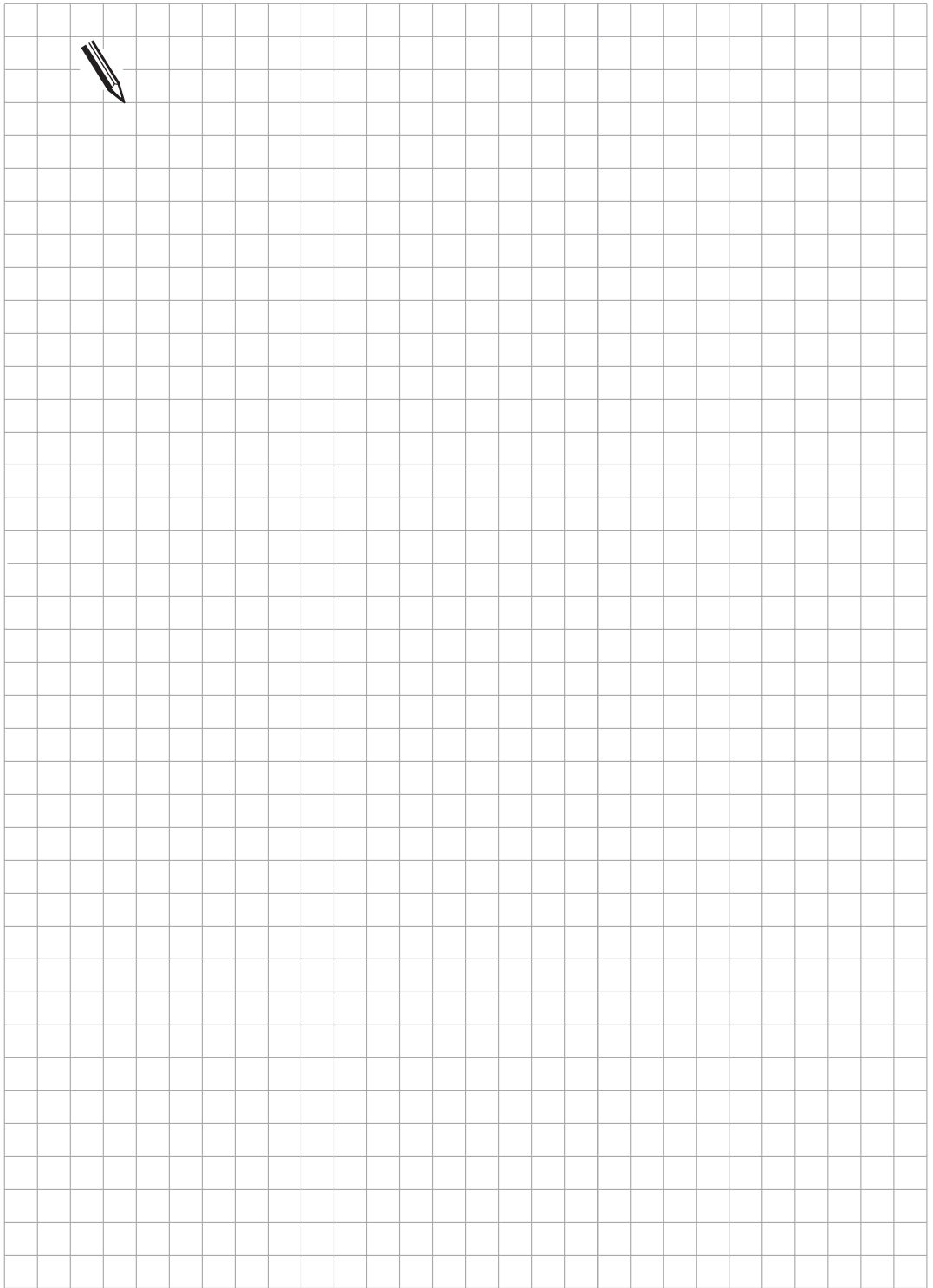
PLC example for gear change and jog-voltage output:

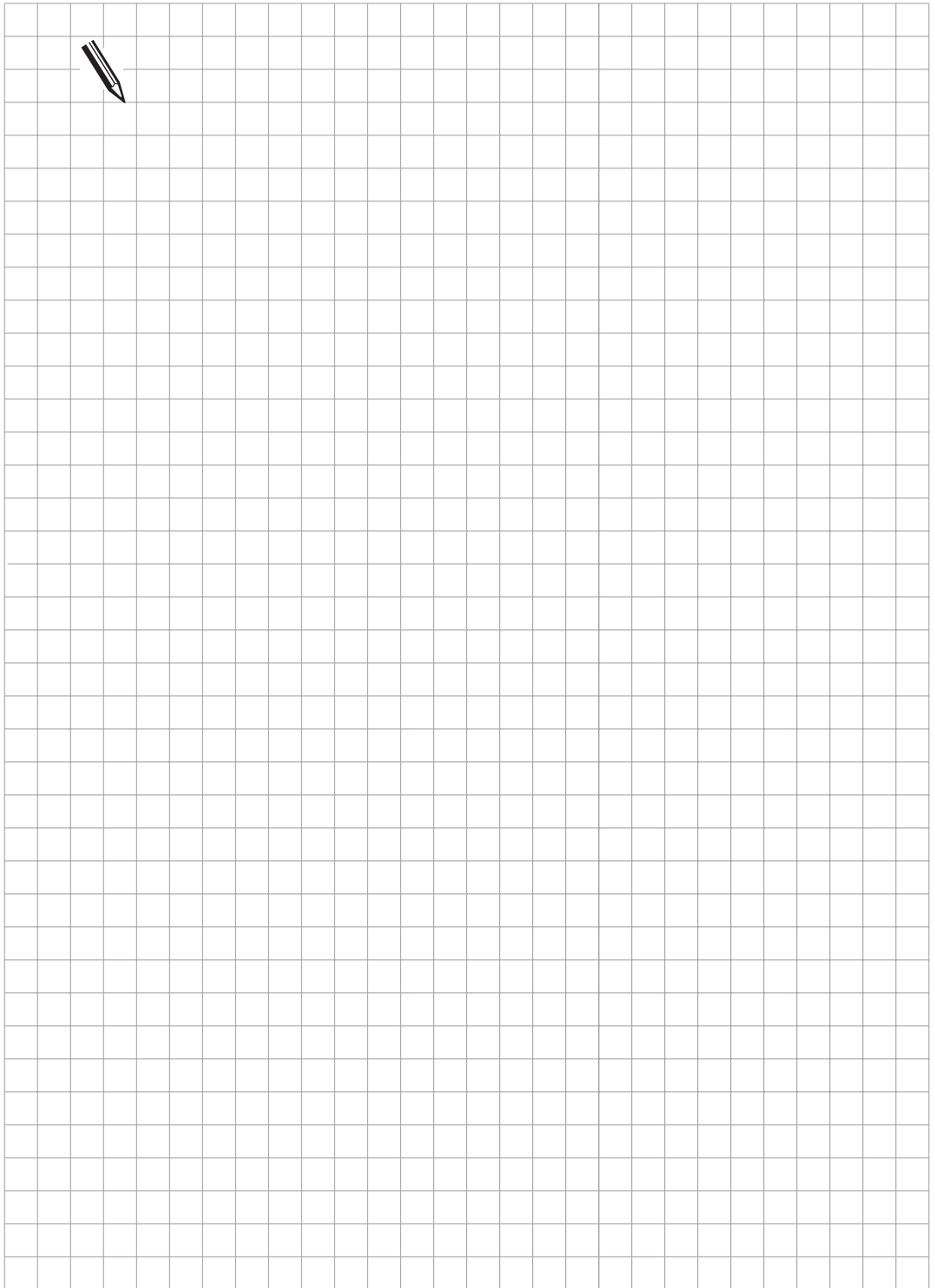
```

.      .      .
.      .      .
.      .      .
L      M2043    ;Gear-code change signal for S analog
CMT    50      ;Module call (output gear code )
.      .      .
.      .      .
.      .      .
L      M2043    ;Gear-code change signal for S analog
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 acknowledgment signal
S      M2480    ;Acknowledgment - gear changed
.      .      .
.      .      .
EM     ;End main program

LBL50
L      M2043    ;Module start
L      W256     ;Gear-code for S analog
B=     015      ;Outputs 15, 16 and 17 for gear change are
                ;activated
EM     ;Module end

```





4.2 Coded output of spindle speed

For coded (binary) output of the spindle speed an S code is entered by the NC in Word W258. 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 acknowledgment that the new spindle speed has been activated. On acknowledgment the change marker M2044 is reset. The S code is not altered before 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 Input 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 input value of 20802 for MP3020. An input 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.

		Set	Reset
M2044	S-code change signal	NC	NC
M2481	S-code acknowledgment	PLC	PLC
W258	S-code	NC	NC
W1008	S-code for minimum spindle rpm	NC	NC
MP3020	Definition of the spindle speed range		
	Input value: 0 to 99 999		

S-code table

S code	rpm
S 00	0
S 01	0.122
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-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 spindle (S axis) is necessary. The orientation is carried out by a miscellaneous function M19 or another M function and must be initiated by the PLC program. MP3010 determines whether the control operates with or without spindle orientation.

Spindle-orientation sequence

In normal operation the spindle is not in the position control loop. If the miscellaneous function M19 is activated, marker M2712 (Strobe - spindle orientation) must be set by the PLC program and the spindle must be switched off. The spindle is driven down the ramp from MP3410.1 to the spindle speed from MP3520.1. An orientation does not begin until marker M2499 has been reset. When this spindle speed has been reached, the spindle is under the control of the position control loop and oriented to the set position of D592, along the ramp of MP3410.1.

During a spindle orientation from a stop position the shortest path is followed, provided the start position of the spindle is in the positioning window (MP3430). If the start position is outside the positioning window, then the positioning is carried out according to markers M2656 and M03 or M04.

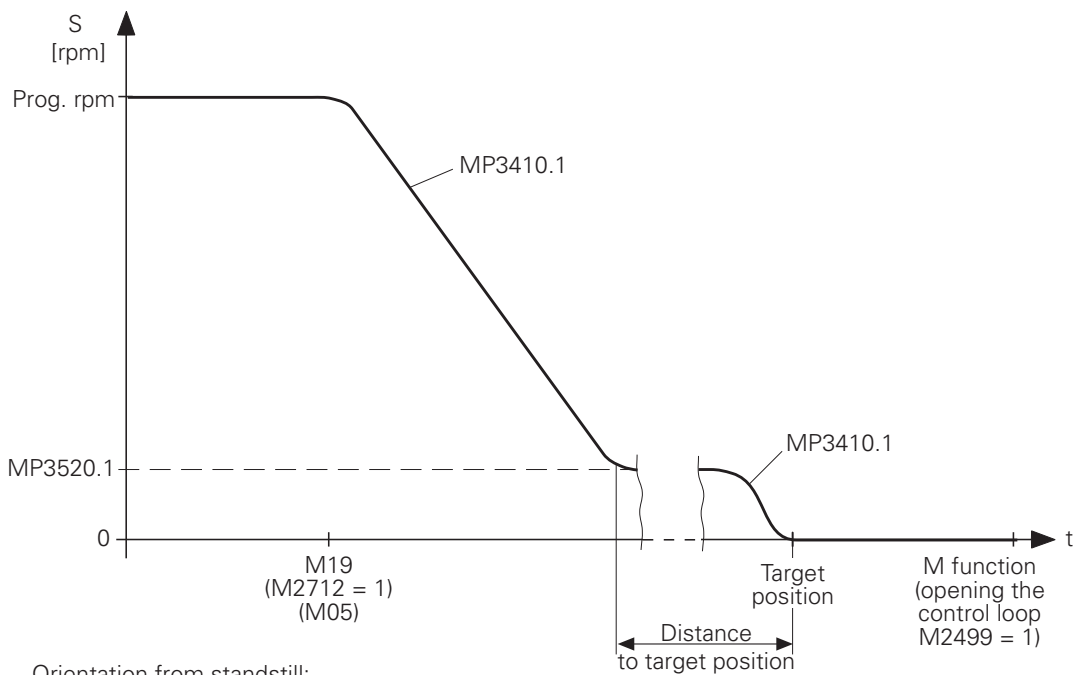
If the spindle does not have to remain in the position control loop after reaching the set position, then the marker M2499 must be set. The spindle is only freed when this marker has been set by the PLC. If M2499 is set continuously the control loop will be opened immediately after every orientation, when the positioning window has been reached.

MP3410.1 Ramp slope of the analog voltage for spindle orientation
Input value: 0 to 1.999 [V/ms]

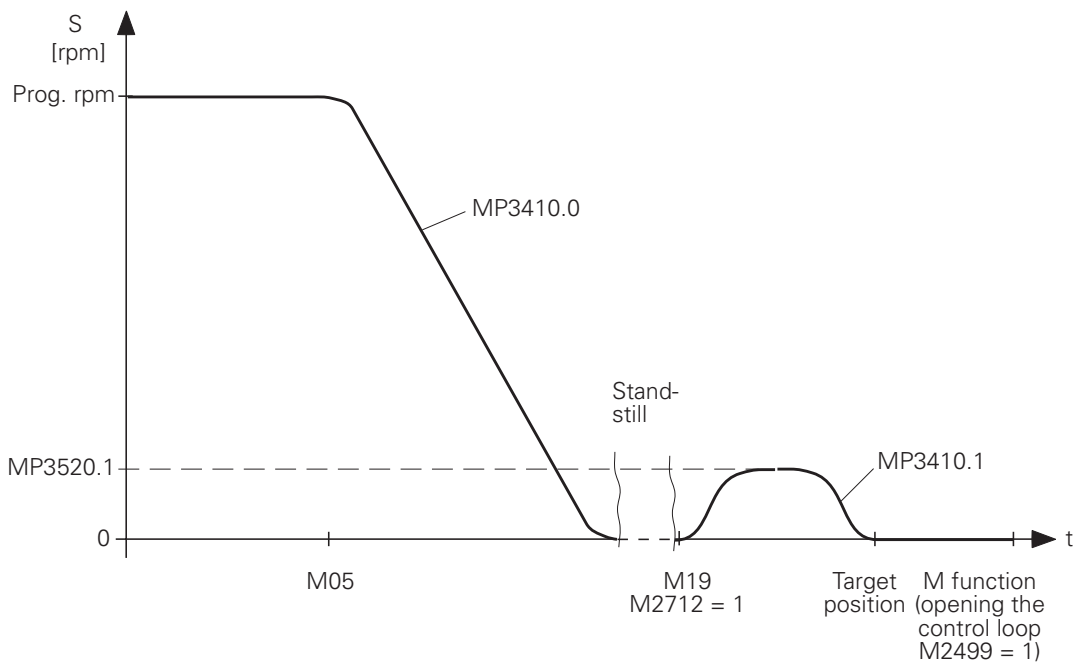
MP3520.1 Spindle speed for spindle orientation
Input value: 0 to 99 999.999 [rpm]

		Set	Reset
M2656	Spindle orientation from stop 0 = Orientation with M03 1 = Orientation with M04	PLC	PLC
M2499	Open spindle control loop	PLC	PLC
M2712	Activate PLC positioning for spindle orientation	PLC	NC
D592	Position spindle orientation (Strobe M2712)		

Orientation from rotation:



Orientation from standstill:



The nominal position can be defined in a machine parameter MP4210.xx , and is referred to the reference point. It must be copied into Doubleword D592 with a PLC command. The nominal position can also be taken from the cycle "Orientation" (CYCL DEF 13). In this case the MSB of D592 must be = 1 and the other Bits set to 0. This can be carried out by the PLC program.

The orientation can be performed to an accuracy of 0.1 degree. This value results from the line count of the rotary encoder which is incorporated (ROD 426.xx8 with 1024 lines) and the quadruple evaluation in the TNC. A positioning window is defined by MP3420. If the spindle is within the positioning window after orientation, marker M2007 will be set.

In order to be able to correct for any misalignment on assembly of the rotary encoder the machine parameter MP3430 was introduced. The deviation between the nominal and actual position of the reference mark which is entered here is taken into account during orientation.

The reference mark is instantly evaluated by the NC when the spindle is turned on for the first time.

MP3420 Positioning window for spindle

Input range: 0 to 65 535 [increments]

1 increment is equivalent to about 0.088 degree

$$= \frac{360 \text{ degrees}}{1024 \text{ lines} \cdot \text{quadruple-evaluation}}$$

MP3430 Reference mark deviation from the desired position (Spindle preset)

Input value: 0 to 360 [degrees]

		Set	Reset
M2007	Spindle in position	NC	NC

In order to control the orientation process precisely, a k_v factor can be entered individually for each gear range via MP3440.

MP3440 k_v factor for Spindle orientation

Input range: 0.1 to 10 [1/min]

MP3440.0	k_v factor gear range 1
MP3440.1	k_v factor gear range 2
MP3440.2	k_v factor gear range 3
MP3440.3	k_v factor gear range 4
MP3440.4	k_v factor gear range 5
MP3440.5	k_v factor gear range 6
MP3440.6	k_v factor gear range 7
MP3440.7	k_v factor gear range 8

After the start of spindle orientation the spindle position is displayed in the status window. The displayed value remains, as long as the spindle is in the closed position control loop. The display is modulo 360 degrees.

A spindle orientation can also be performed by an initiator. For this, marker M2501 must be set by the PLC. This marker activates a spindle rotation with a spindle speed defined by machine parameter MP3520.0 and a direction of rotation according to marker M2656. If the marker M2501 is reset by the PLC (e.g.. via initiator), then the spindle will stop. The positional value will be displayed in the status window. "Jog mode" for the spindle can be achieved in this way.

MP3520.0 Spindle speed activated by marker M2501

Input value: 0 to 99 999.999 [rpm]

		Set	Reset
M2501	Activates spindle speed MP3520.0 and direction of rotation from marker M2656	PLC	PLC

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       ;Acknowledgment 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

Acknowledgment of the orientation

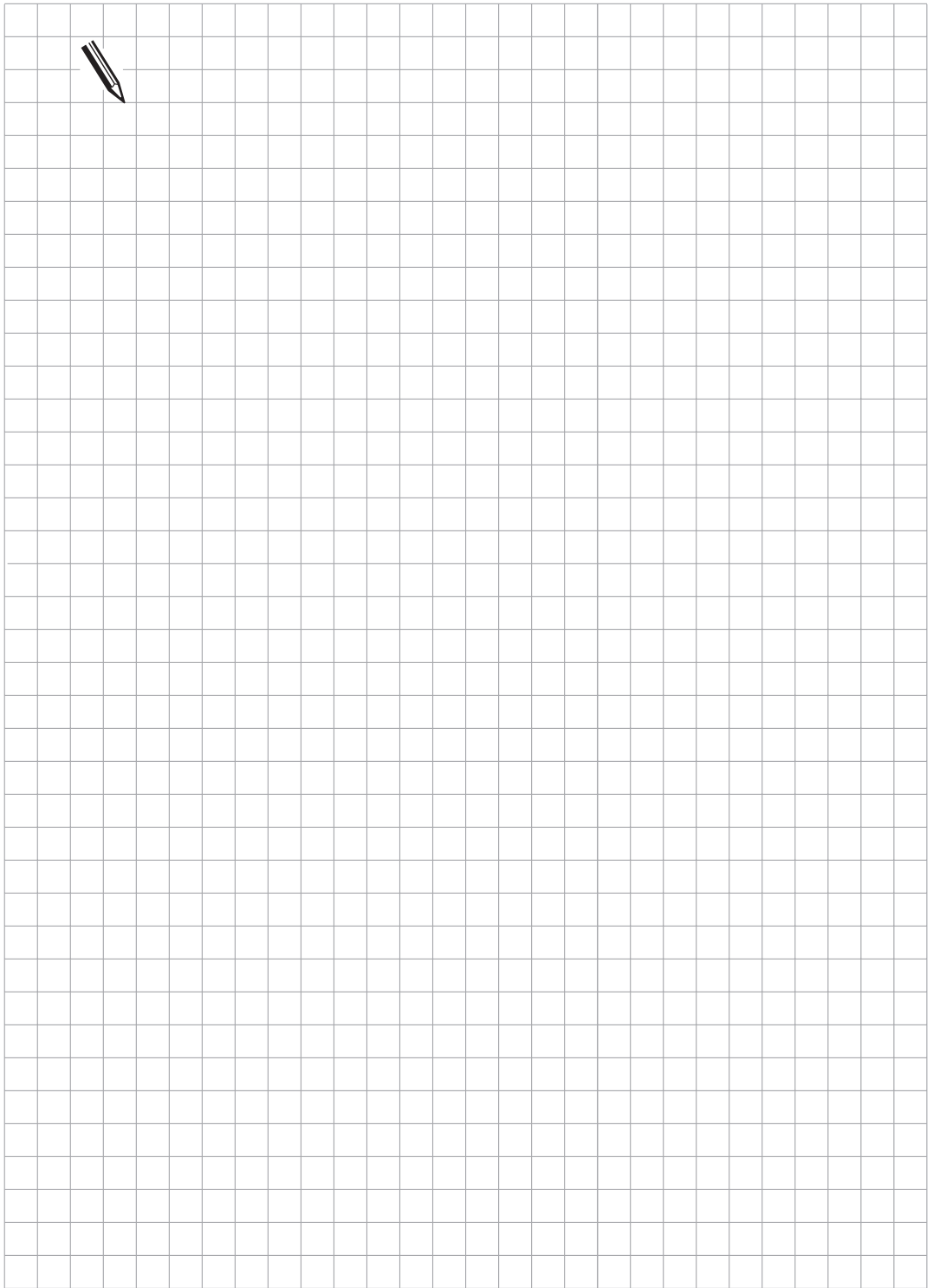
L	M1919	;Decoded M function 19
O	M1920	;Decoded M function 20
A	M2045	;M-code change signal
A	M2007	;Spindle in position
AN	M2712	;Strobe for spindle orientation
S	M2482	;Acknowledgment of M code
.	.	.
.	.	.
.	.	.
EM		;Main program - end

LBL 180; Transfer positional value from the spindle-orientation cycle

L	K+O	;Load the constant value ZERO
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
- tapping without floating tap holder.

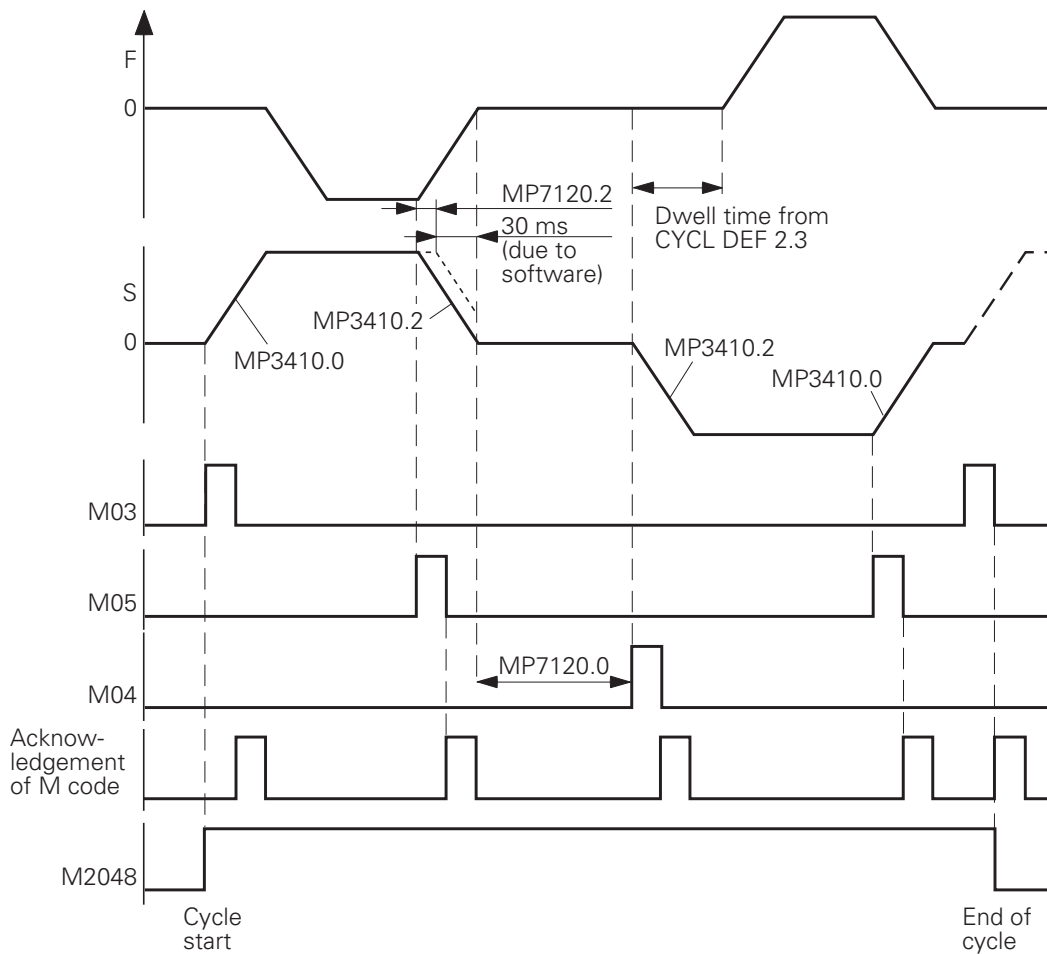
Tapping with floating tap holder is possible for both analog and coded spindle-speed output.

Tapping without floating tap holder is only possible for analog 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 behavior of the machine by machine parameters.

4.4.1 Tapping with floating tap holder for analog 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 acknowledgment (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. The M functions that are output by the NC 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 $s = 1000 \text{ [rpm]} = 1.8 \text{ [V]}$, the thread ramp is 0.05 [V/ms] , and the feed ramp is steeper than 0.05 [V/ms] then:

$$1.8 \text{ [V]} : 0.05 \text{ [V/ms]} = 36 \text{ ms}$$

The spindle is decelerated 36 ms before reaching the total hole depth. The spindle deceleration can be delayed by the spindle slow-down time.

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 for tapping

Input value: 0 to 1.999 [V/ms]

MP7120.0 Dwell time for change of direction of spindle rotation

Input value: 0 to 65.535 [s]

MP7120.2 Spindle run-on time after reaching total hole depth

Input value: 0 to 65.535 [s]

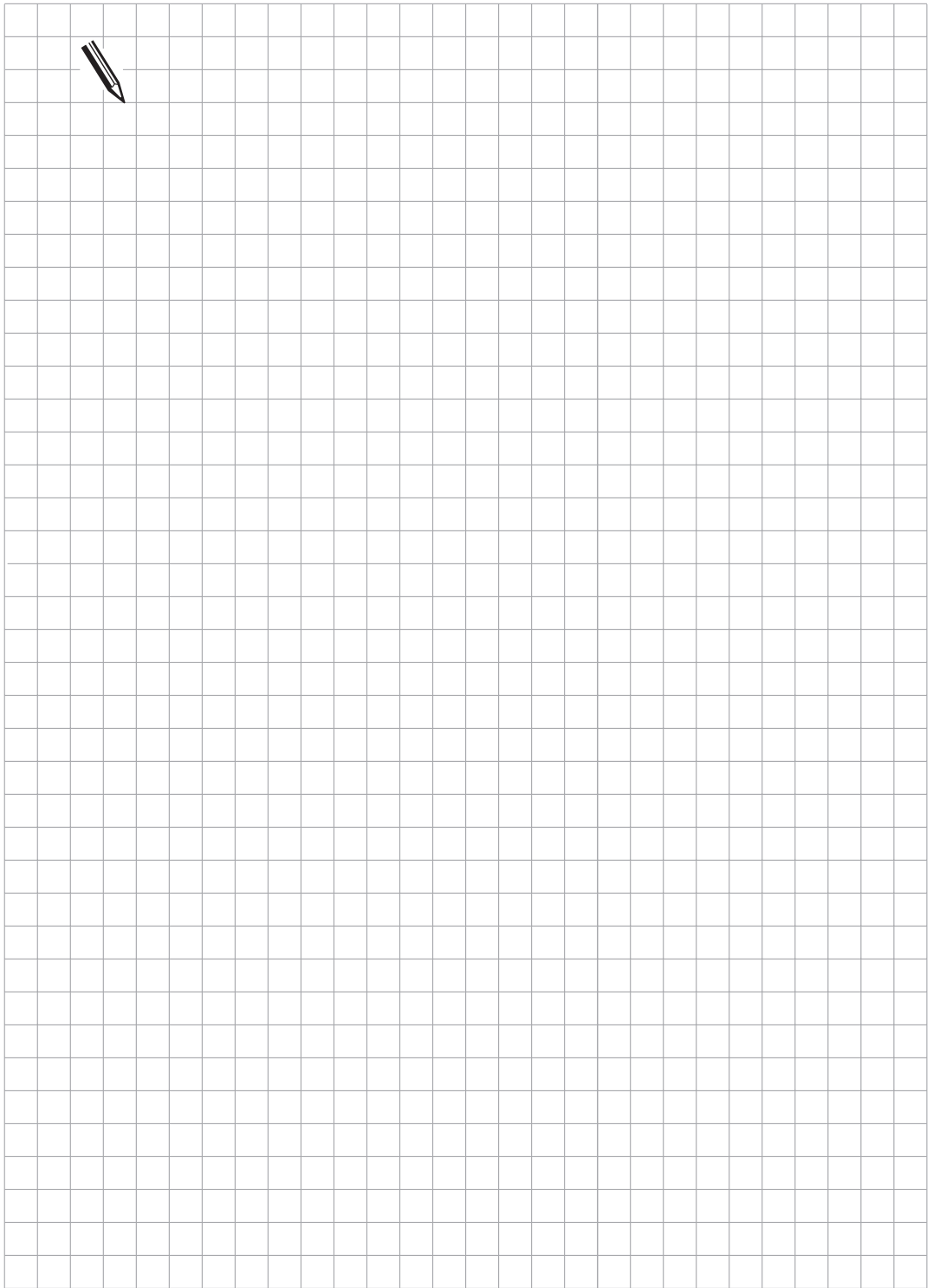
MP7110.0 Minimum for feed override when tapping

Input value: 0 to 150 [%]

MP7110.1 Maximum for feed override when tapping

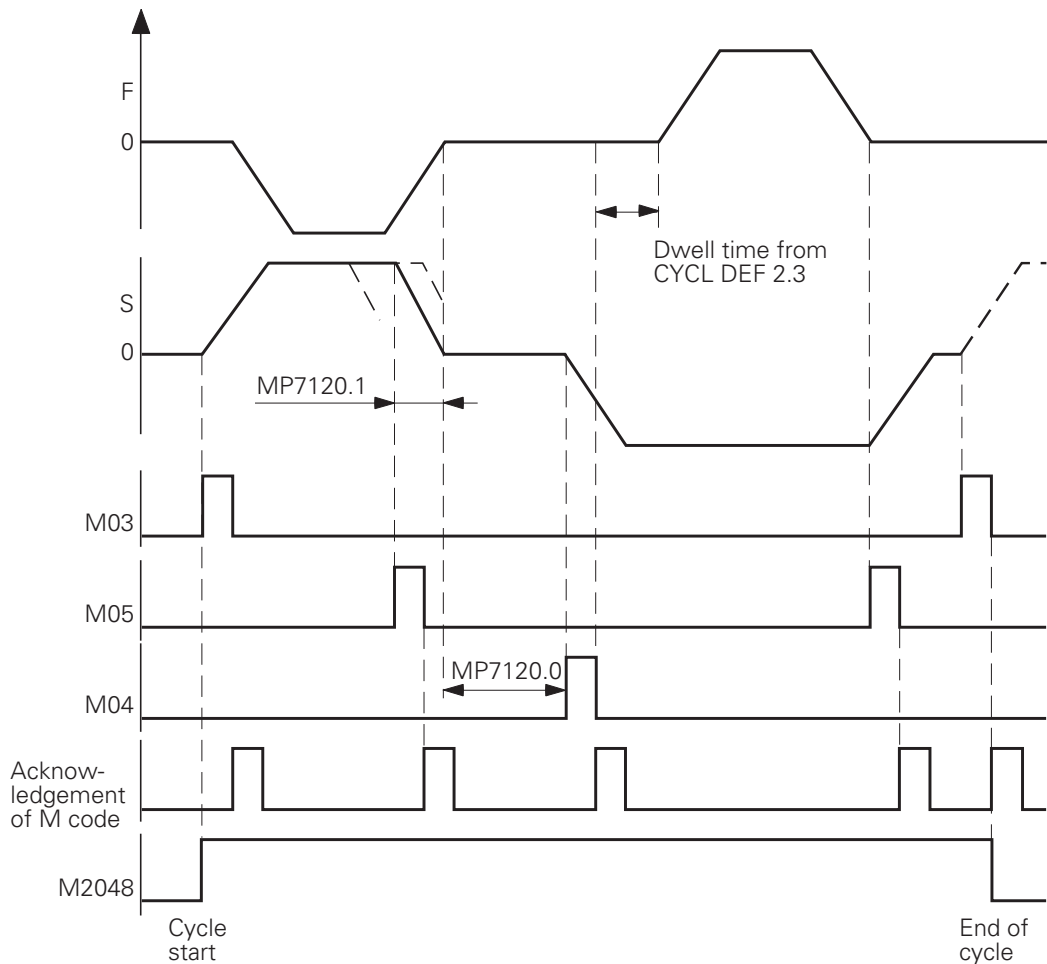
Input value: 0 to 150 [%]

		Set	Reset
M2048	Tapping cycle called	NC	NC



4.4.2 Tapping with floating tap holder and coded spindle-speed output

The following diagram shows the sequence of events in the cycle:



Since, when using the spindle speed output in code, 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. M05 is automatically output by the NC and must be acknowledged.

The machine parameter MP7120.0 (dwell time for change of direction of rotation) and the programmable dwell time are just as effective as for analog spindle-speed output.

MP7120.1 Advanced switch time of spindle for "tapping" cycle (only active with BCD output of spindle rpm)

Input value: 0 to 65.535 [s]

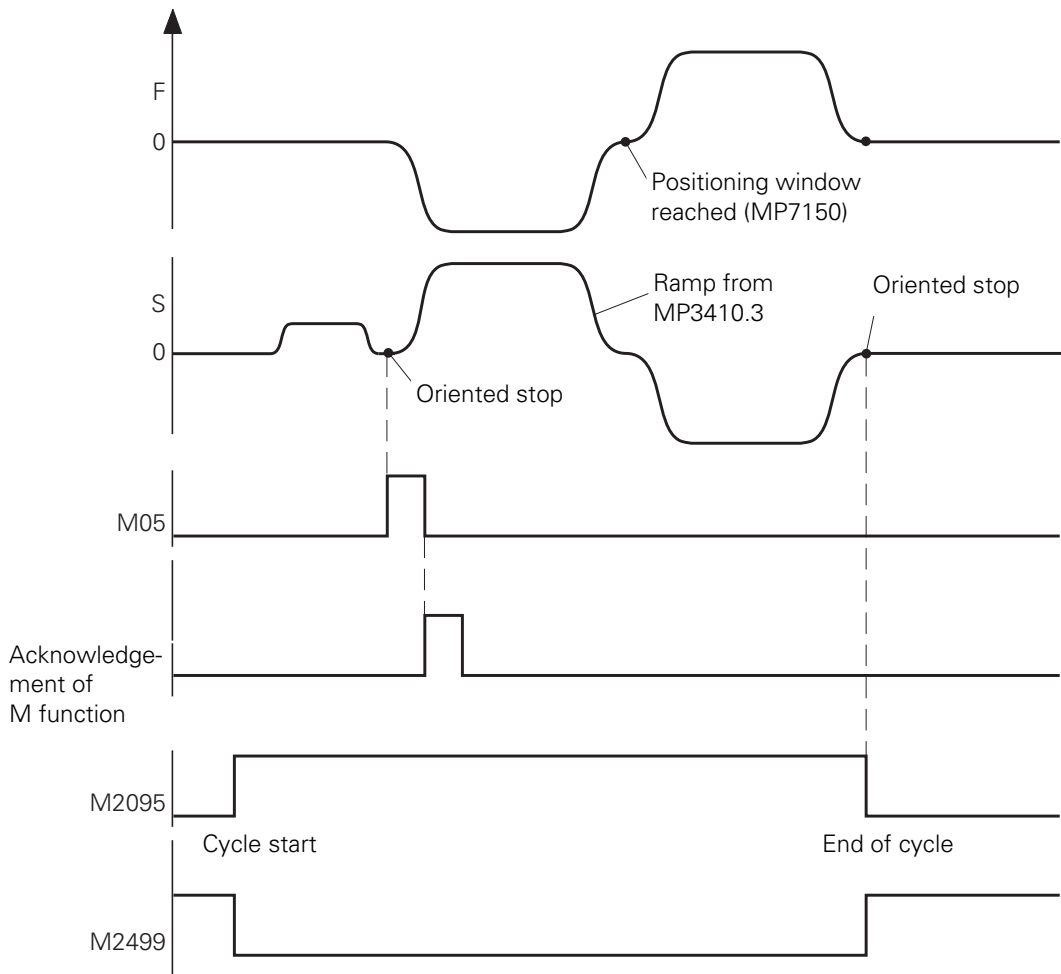
4.4.3 Rigid tapping

Tapping without a floating tap holder is defined in the NC program by "CYCL DEF 17 RIGID TAPPING" and called by CYCL CALL . Here too the dynamic behavior is determined by machine parameters.

Before tapping, the axes (e.g. Z and S) are synchronized by a spindle orientation, i.e. each Z position is assigned to a particular spindle position. This means, that the same thread can be cut more than once. The fixed assignment of the spindle position depends on the thread pitch which is entered in the cycle. Because of the fixed thread pitch the correlated Z-dependent spindle position can not be altered by the operator. The thread pitch can be positive or negative.

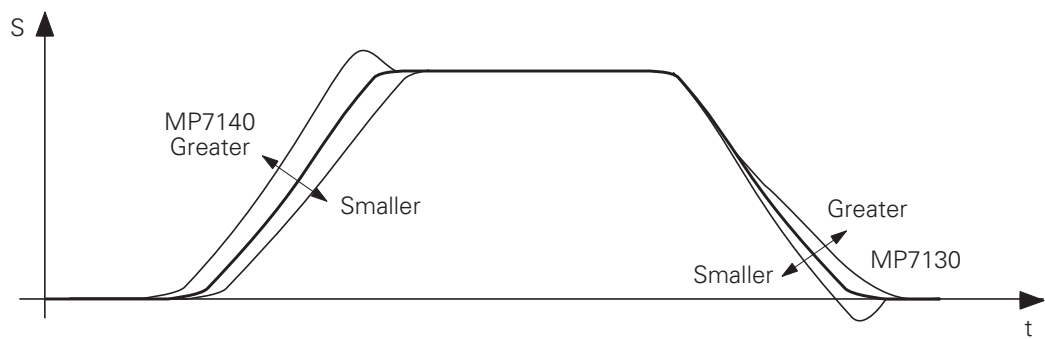
Marker M2095 is set at the start of the cycle. With this marker the PLC program can switch on the spindle, if required, with M03 or M04. The marker M2499 (Close control loop - S-axis) must be set to zero in acknowledgment. Only then can the orientation take place, and spindle and tool axis move synchronously until the cycle has been completely processed. Before orientation the NC outputs an M05. An acknowledgment is necessary.

After this, marker M2095 is reset by the NC and the control loop is opened. The spindle position is displayed during tapping. The ramps for the drilling process are stored in MP3410.3.



Machine parameters MP7130 and MP7140 were introduced so that the spindle ramp for tapping could be better adjusted to the dynamic characteristics of the machine. Machine parameter MP7150 fixes the positioning window for the tool axis. The input value must be less than or equal to MP1030.2.

A spindle overshoot when accelerating and running into position must be avoided at all costs.



Marker	Function	Set	Reset
M2095	Rigid tapping is active	NC	NC
M2499	Open control loop S axis	PLC	PLC

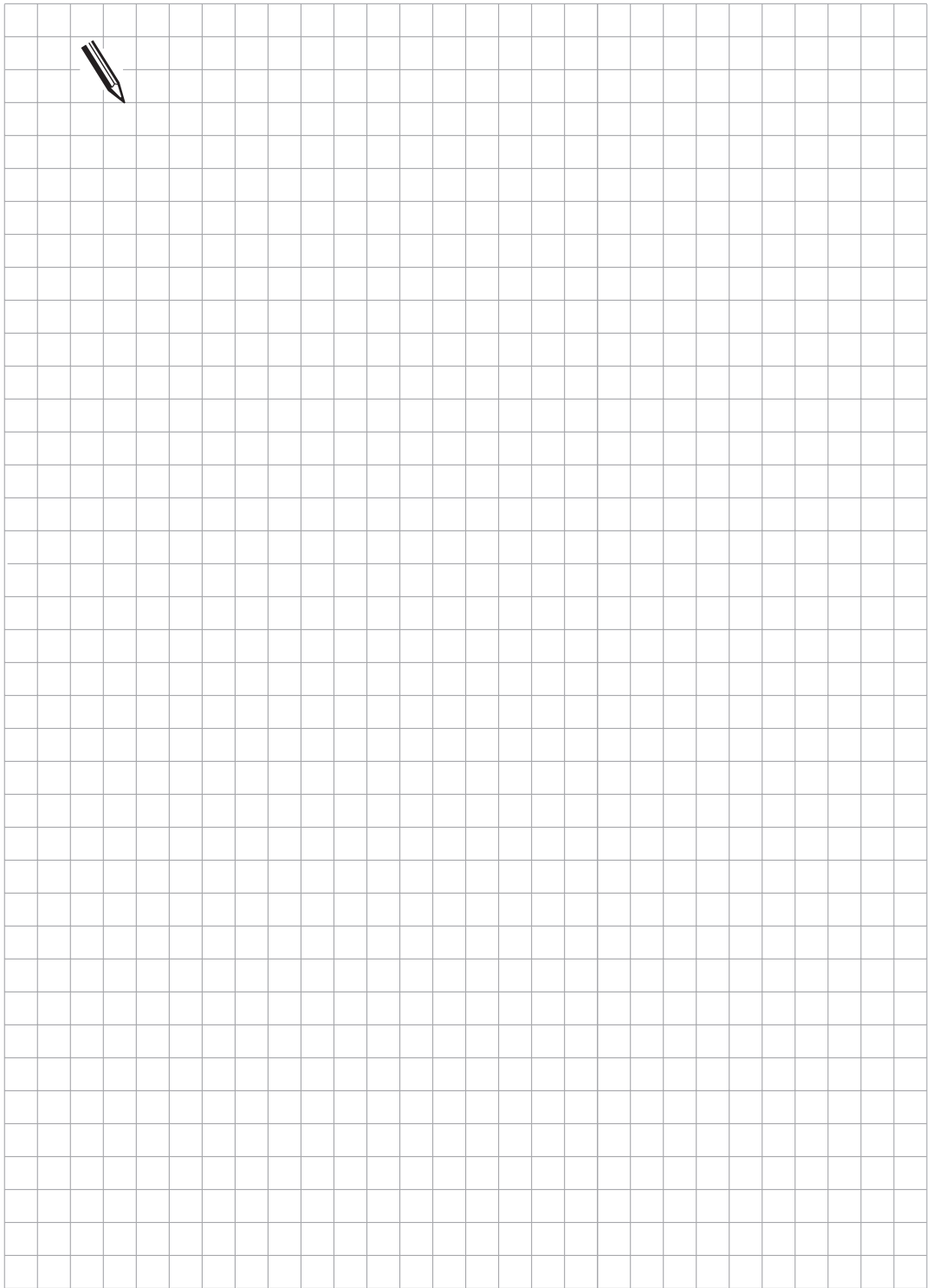
MP3410.3 Accelerations ramp for rigid tapping
Input value: 0 to 1.999 [V/ms]

MP7130 Spindle run-in characteristic
Input value: 0.001 to 10 [°/min] (matching MP1520)

MP7140 Transient response of spindle Spindle, t on acceleration
Input value: 0.001 to 1 (matching MP1530)

MP7150 Positioning window for tool axis
Input value: 0.001 to 2 [mm]

MP7160 Suppression of spindle orientation at beginning of Cycle 17
Input: 0 or 1
0 = Spindle orientation is performed
1 = Spindle orientation is not performed



5 EMERGENCY STOP Routine

A PLC input (X42/4) and a PLC output (X21/34; in the PL400, X3/10; and in the PL 410, 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 off, a flashing error message appears on the VDU screen and the PLC program is stopped. This error message can not be canceled. After removing the fault, the switch-on routine must be gone through again.

If the input "control is ready" 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 canceled 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" will be displayed.

Marker M2827 makes it possible to suppress the external EMERGENCY STOP. If marker M2827 is set, an external EMERGENCY STOP is not sent to the NC, instead all control loops are opened and the nominal values are immediately set to 0 V.

		Set	Reset
M2190	Non-flashing error message is displayed	NC	NC
M2191	"External EMERGENCY STOP" message is displayed		
M2827	Suppress EMERGENCY STOP, open the control loop, NC stop	PLC	PLC

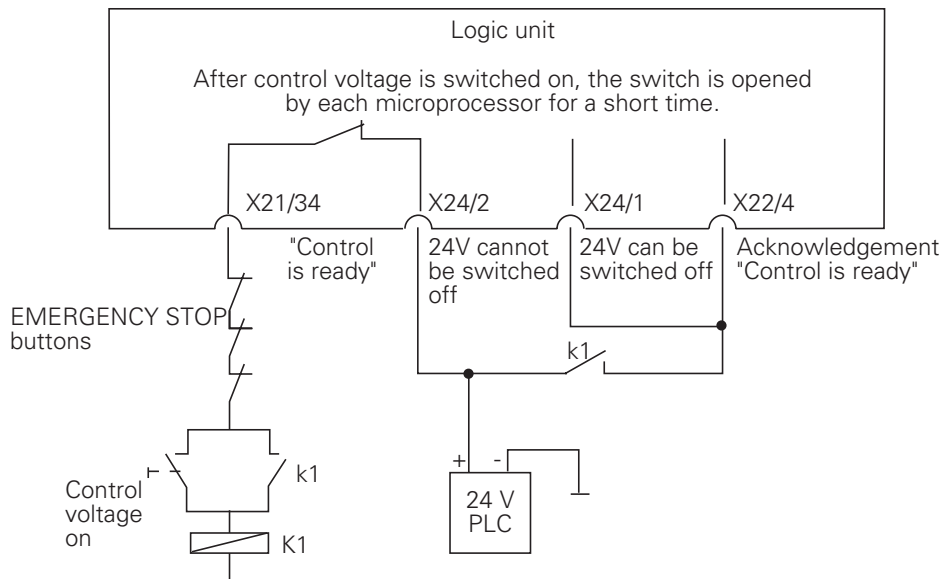
5.1 Connection diagram

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.

HEIDENHAIN recommends the following wiring:

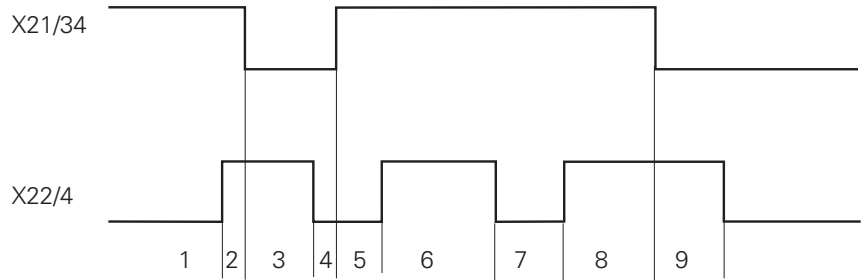


This diagram represents a proposal for circuitry. The machine tool builder is responsible for complying with applicable safety regulations.



5.2 Flow diagram

The external electronics must meet the specified conditions. In particular, the acknowledgment for "Control is ready" must reach the TNC 360 after a maximum of 146 ms.



	VDU display
1. Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
2. Recognize the control voltage on X22/4 and switch-off of the "Control is ready" signal on X21/34 ($t < 70$ ms).	
3. Maximum time during which the acknowledgment of "control is ready" on X22/4 must go to 0 ($t < 146$ ms). If exceeded	EMERGENCY STOP DEFECTIVE
4. Recognize the acknowledgment and set X21/34 ($t < 20$ ms).	
5. Wait for control voltage.	RELAY EXT. DC VOLTAGE MISSING
6. Normal control operation. Output and acknowledgment "control is ready" are high.	
7. Machine control voltage is switched off externally	EMERGENCY STOP
8. After switching on the control voltage again, the error message can be canceled, followed by normal control operation.	
9. The control switches off the output "control is ready" (X21/34) after recognizing a fault.	Flashing error-message

6 Display and Operation

Machine parameters and PLC markers can be used to influence the control behavior 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 section "Display and operation".

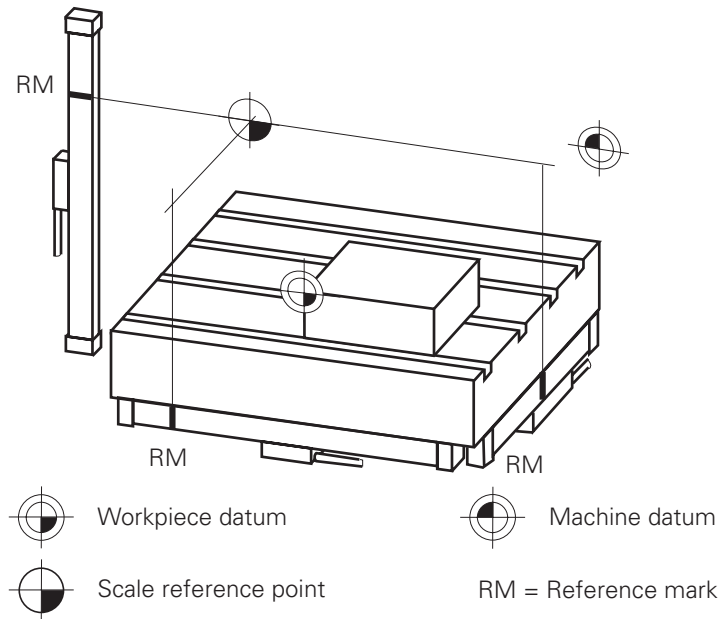
6.1 Machine datum

In the operating mode "Manual operation", 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. On the TNC 360 the machine datum and scale datum point (= Reference point) are identical. All REF-referenced displays and positioning movements (PLC positioning or M91) are referenced to the machine datum.

With MP950.x you can define an additional 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 active only in the blocks in which they are programmed.

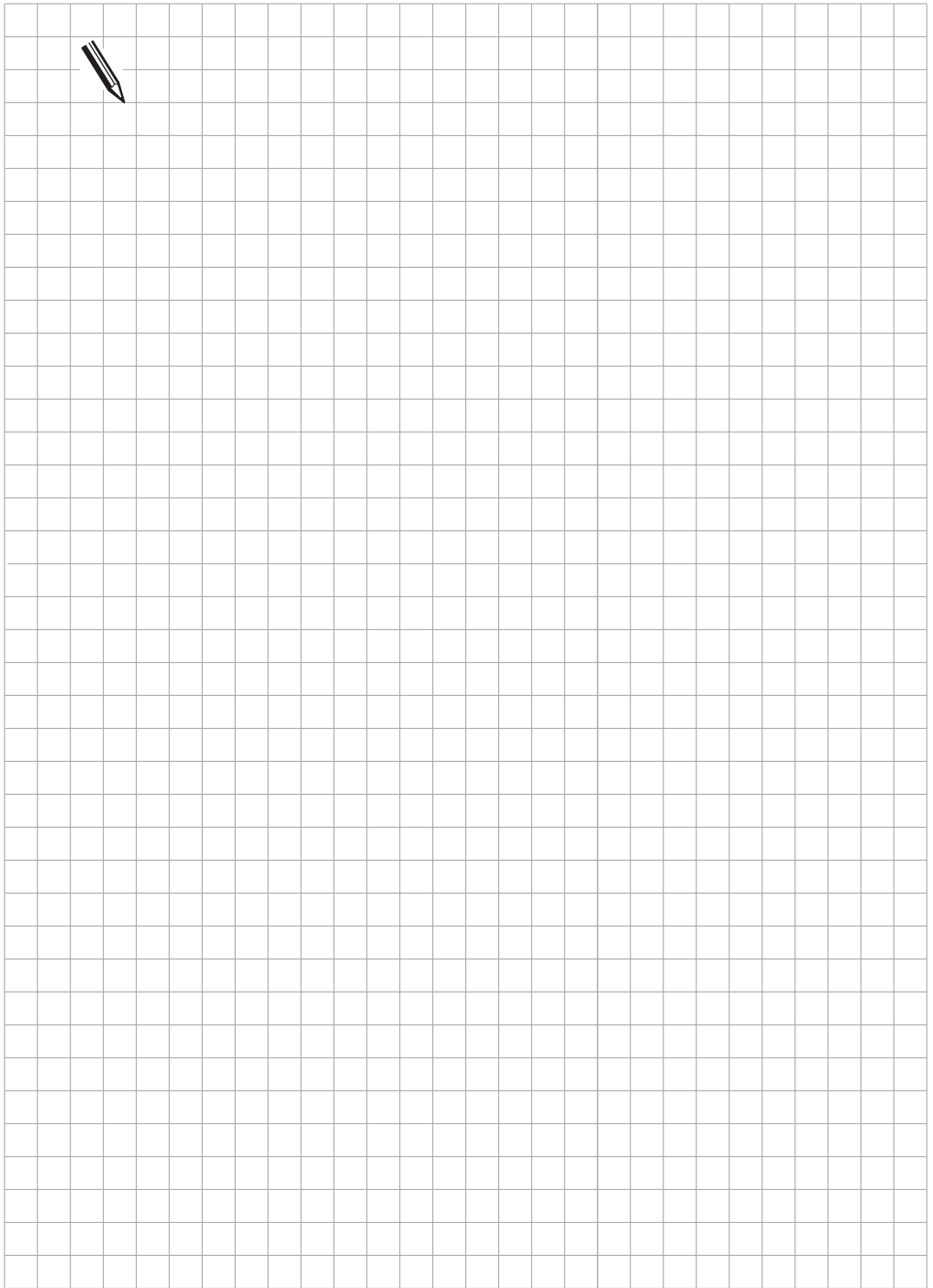


MP950**Datum point for positioning blocks with M92**

Input range: -30 000.000 to +30 000.000 [mm] or [°]

Values referenced to the machine datum

MP950.0	X axis
MP950.1	Y axis
MP950.2	Z axis
MP950.3	4th axis

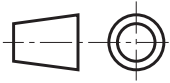


6.2 Graphic simulation

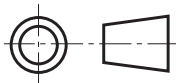
To check for errors, the execution of part programs can be graphically simulated on the basis of a programmed blank form. It is possible to choose between three different graphics display modes. The graphics display on the VDU screen can be altered by machine parameters.

The following alterations are possible:

The 3-plane display mode can be produced according to either European or American convention.

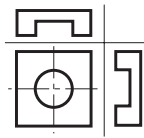
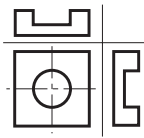


Preferred European

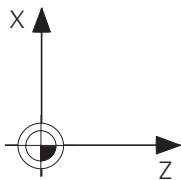


Preferred American

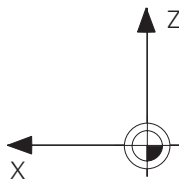
Example:



The coordinate system can be rotated by $+90^\circ$ in the machining plane. This is useful when, for instance, the Y axis is fixed as the tool axis.



No rotation



Rotation by $+90^\circ$

MP 7310

Graphics display

Input range: 0 to 3

Bit 0	Changeover of 3-plane display	+ 0 = European convention + 1 = American convention
Bit 1	Rotation of the coordinate system in the machining plane by $+90^\circ$	+ 0 = no rotation + 2 = coordinate system rotated by $+90^\circ$

6.3 Status display

6.3.1 Position display

The display step for the axis positions can be selected for specific axes by machine parameter MP7290. Regardless of this selection the programming and positioning of the axis position is always carried out to 0.001 mm or 0.001° precisely.

The input resolution of the TNC 360 is 0.001 mm or 0.001°.

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**
Input value: 0 or 1

- 0 = tool length ignored
- 1 = tool length taken into account

MP7290 **Position display step**
Input values: 0 to 4

- 0 = 0.001 mm or 0.001°
- 1 = 0.005 mm or 0.005°
- 2 = 0.010 mm or 0.010°
- 3 = 0.050 mm or 0.050°
- 4 = 0.100 mm or 0.100°

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		

6.3.2 Display mode and traverse direction for rotary axes

Machine parameter MP7470 determines the traverse direction for absolute positioning and the display mode for rotary axes.

- Display from +0 to +359.999°: After a full turn the display value jumps back to 0°. In this operating mode it is not possible to set software limit switches.

During absolute programmed movements of rotary axes (including PLC positioning) the table now takes the shortest path to the target position, provided that Bit 1 of MP 7470 is set. For incrementally programmed movements, however, the table moves according to the sign of the programmed value.

- Display from –30 000° to +30 000°: In this operating mode software limit switches may be entered. Absolute positional values from – 30 000° to + 30 000° can be programmed (166 revolutions).

MP7470

Display mode for rotary axis

Input values: 0 to 3

- Bit 0:
- +0 = 0 to + 359.999° (no software limit switches)
 - +1 = –30 000° to +30 000° (software limit switches active)
- Bit 1:
- +0 = Absolute positioning of rotary axes takes the programmed direction to the target position
 - +2 = Absolute positioning of rotary axes takes the shortest direction to the target position

6.3.3 Feed rate setting, 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 %.

With the feed rate potentiometer the feed rate can be changed to 0% to 150% of the programmed feed rate.

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. W766 must be overwritten in every PLC scan, since otherwise the normal feed rate would become effective again. The value range is 0 to 150 in 1 % increments, for nonlinear curves it is 0 to 15 000, corresponding to 0 to 150%.

To adjust the override values for auxiliary axes (PLC axes) see PLC module 9124.

With the Doubleword D596 the PLC can limit the maximum possible feed rate by entering a feed rate value. The new feed rate becomes effective immediately.

In order to ensure compatibility with previous PLC programs, the Doubleword D596 is given the default value 300 000 mm/min when the control is switched on or after an interruption of the PLC scan, so that the maximum feed rate from the machine parameter becomes effective. With "Tapping" (Cycle 2) and "Rigid tapping" (Cycle 17), the Doubleword D596 has no effect.

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)

Input value: 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)

Input: 0 to 7

- Bit 0 Feed rate override if rapid traverse key is pressed in a "Program run" mode
+0 = Override not effective
+1 = Override effective
- Bit 2 Feed rate override if rapid traverse key and machine axis direction key are pressed in the "Manual" mode
+0 = Override not effective
+4 = Override effective
- Bit 3 Spindle and feed rate override in 1% increments or according to a nonlinear curve
+0 = 1% increments
+8 = Nonlinear curve

W494

% factor – feed rate override (NC → PLC)

W766

% factor – feed rate override (PLC → NC)

D596

Feed rate from PLC for feed rate limiting

6.3.4 Display of M functions

The miscellaneous functions for control of the spindle (M03, M04, M05) and the coolant (M08, M09) are displayed in the status display. The display of these M functions is controlled by the PLC, i.e. the machine tool builder must take this into account when creating the PLC program. The markers M2485 and M2486 also change the polarity of the analog voltage for the spindle. M2608 switches off the analog output for the spindle. The programmed spindle speed is, however, still displayed (see section "Spindle").

Marker	Function	Set	Reset
M2485	Status display and sign of S analog for M03	PLC	
M2486	Status display and sign of S analog for M04		
M2487	Status display for M05 and spindle stop		
M2508	Status display M08 or M09 0 = Status display M09 1 = Status display M08		
M2608	Status display M03, M04, M05 inverse and S analog output = 0V		
M2609	Status display M08, M09 inverse		

6.3.5 Display of run duration

The TNC has three timers that measure the duration in seconds of the following states:

- Control (dialog "Control on")
- Program run (dialog "Program run")
- Spindle (dialog "Spindle on")

Control on

This timer starts with the first PLC run after the control reset.

Program run

This timer measures the duration of program run (see User's Manual TNC 360).

Spindle

The spindle operating time starts when the Markers M2485/M2486 are set and stops when they are reset.

Display of run duration

The three timers can be called by entering the code number 85 7282 (see User's Manual TNC 360) and can then be reset, depending on MP 7235.

Resetting the run duration timers

All timers are kept in the battery-buffered RAM and are reset after the error message "Operation parameters erased."

The machine parameter MP 7235 allows a specific resetting of the run duration timers.

MP7235 Resetting the timing

Input value: 0 to 7

Bit 0	Run time – Control on +0 cannot be reset +1 can be reset
Bit 1	Run time – Program run +0 cannot be reset +2 can be reset
Bit 2	Run time – Spindle +0 cannot be reset +4 can be reset

It is not possible to reset a timer during a program run.

6.3.6 Control is operational

If the control is operational, 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, a "*" 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."

		Set	Reset
M2183	Program interruption (display "Control is ready" flashes)	NC	NC
M2184	Control is ready ("Control is ready" display goes on or flashes)		

6.3.7 Cancel status display

A machine parameter can be used to decide whether the status display should be reset and the contents of the Q parameters and the tool data erased with the M functions M02 and M30, as well as with NC block "END PGM" and the selection of a program. 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, Q parameters, and tool data

Input value: 0 to 7

- 0 = Cancel status display, Q parameters and tool data when a program is selected
- 1 = Cancel the status display, Q parameters and tool data with M02, M30, END PGM and when a program is selected.
- 2 = Cancel the status display and tool data when a program is selected
- 3 = Cancel the status display and tool data with M02, M30, END PGM and when a program is selected
- 4 = Cancel the status display and Q parameters when a program is selected
- 5 = Cancel the status display and Q parameters with M02, M30, END PGM and when a program is selected
- 6 = Cancel the status display when a program is selected
- 7 = Cancel the status display with M02, M30, END PGM and when a program is selected

6.3.8 Expanded spindle display

A moving bar diagram with the dialog "Spindle power" can be added beneath the status display. The selected gear range can be shown in the status display next to the spindle speed (e.g.: S1000/2).

With machine parameter MP7274 the display of the current gear range can be selected through the bit.

Bit 1 determines whether information necessary for spindle power display is measured directly through the analog input at connector X8 (0 to 4.999 V) or through the PLC Word W600.

In machine parameter MP7275 the reference value of the spindle power is entered at 100%. This results from the maximum value to be acquired and the maximum percentage factor to be shown.

$$\text{MP7275} = \frac{\text{Maximum value to be acquired}}{\text{Maximum percentage factor to be displayed}} \times 100$$

MP7274 Expanded spindle display

Input values: 0 to 3

Bit 0	Display of current gear range	+0 = No display +1 = Display
Bit 1	Display of spindle power	+0 = Acquisition via analog input (Value range 0 to 4.999 [V]) +2 = Acquisition via PLC Word W600 (Value range 0 to 65.535)

MP7275 Reference value of displayed spindle power

for acquisition via analog input

Input range: 0 to 4.999 [V]

for acquisition via PLC Word W600

Input range: 0 to 43 690

MP7275 Value for spindle power display

Example 1: Acquisition via analog input

A voltage of 0 to 4.999 volts is acquired via analog input. The spindle power should be displayed between 0 and 150 %.



Example 2: Acquisition via PLC Word W600

The information needed to display the spindle power is acquired through the 8 PLC inputs I100 to I107. Values between 0 and 127 can be acquired. You wish to display the spindle power between 0 and 150 %. The reference value at 100% results as follows:

$$\frac{127}{150} \times 100 = 84.66$$

The information is to be evaluated and displayed through the word W600:

Machine parameters:

MP7274 = 2

MP7275 = 85

PLC Program:

```
.  
.   
.   
LW I100 ;Read in the status of the PLC inputs I100 to I107.  
A K+255 ;Evaluate only the lower 8 bits (PLC inputs)  
= W600 ;and transfer for the spindle power display.
```

6.3.9 Generating NC blocks with the actual-position-capture key

MP7225 was introduced to permit automatic generation of NC blocks in the "Programming and editing" mode of operation. In plain language dialog (not in ISO programming), the actual-position-capture key can generate a positioning block with up to 3 axes, without feed rate, radius compensation or M functions. This positioning block is inserted behind the block currently selected in the "Programming and editing" mode. The actual positions are transferred to memory as nominal position values. The axis is selected with the dialog "AXIS SELECTION" in the MOD mode of operation. Here up to 3 of 4 axes can be selected by pressing the desired axis keys.

The PLC marker M2829 can also generate a block. The marker is reset by the NC.

MP 7225	Generating blocks with the "ACTUAL POSITION CAPTURE" key		
	Input: 0 to 3		
	0 = no block generation possible		
	1 = block generation with the "ACTUAL POSITION CAPTURE" key		
	2 = block generation with M2829		
M2829	NC block generation	Set PLC	Reset NC

6.4 PLC text window

The PLC can display a text above the status line for the circle center. The text length is 16 characters or 32 small characters.

Word W602 causes the text to be displayed.

W 602 A text is displayed in the PLC text window

Bit 0 to 7:	Number of PLC error text/dialog (0 to 199)
Bit 8 to 11:	0000 = Erase display 0001 = Normal display 1111 = Inverse display
Bit 12 to 13:	00 = Erase display 01 = Display with small characters 10 = Display with normal characters
Bit 14 to 15:	reserved

The text display for PLC Trace (PCTR) was placed up next to * for "Control in operation" if the actual value display is small.

6.5 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 canceled 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 machine tool builder determines the conditions under which the PLC will produce error messages. Up to 100 different PLC error messages can be generated.

The dialogs for PLC error messages can be determined by the manufacturer of the machine. Please contact HEIDENHAIN about this. The standard version has dialogs 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 canceled 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).

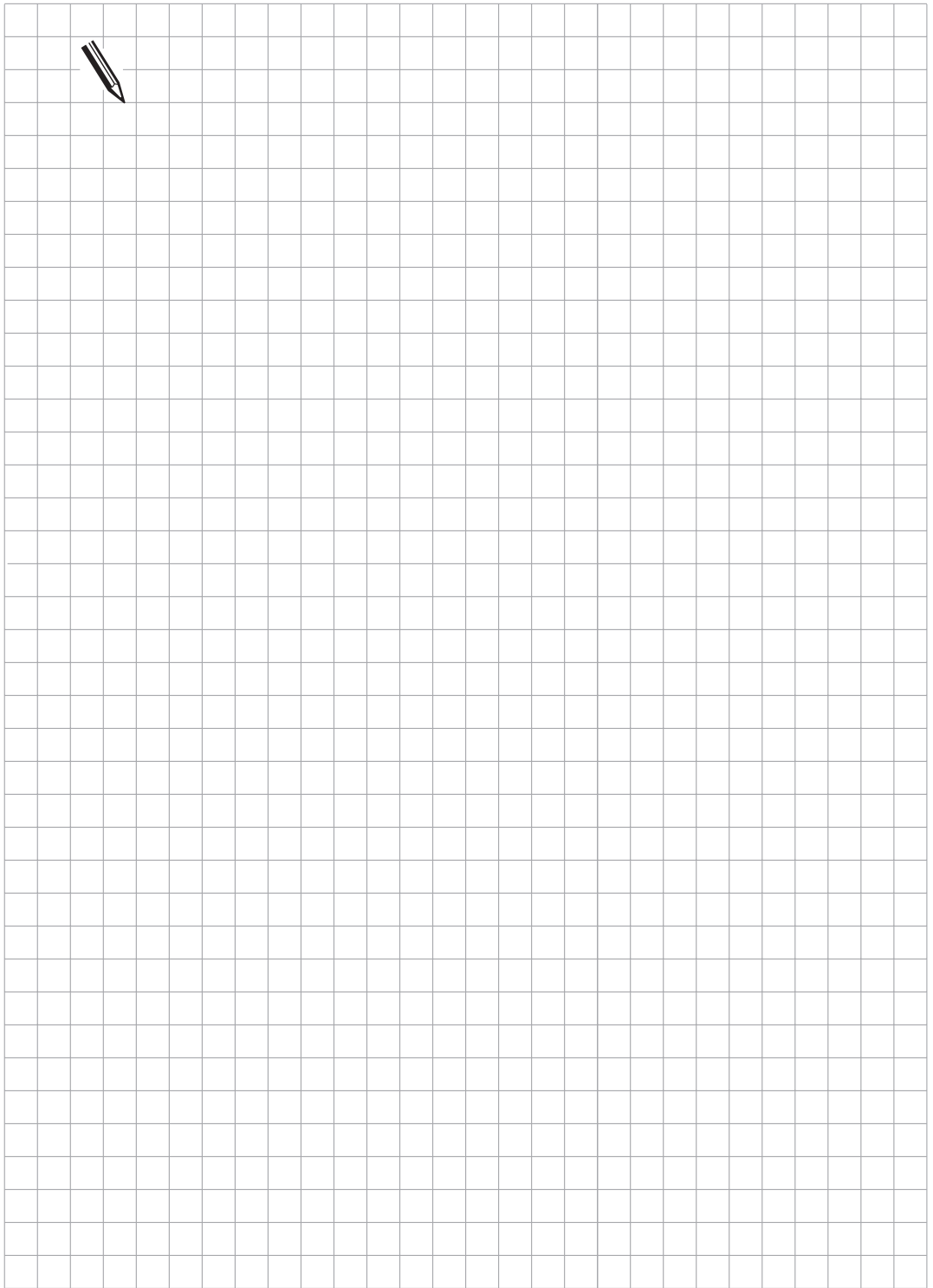
In order to show a PLC error message as a flashing message, the marker M2815 must also be set. If marker M2815 is set, but none of the 100 PLC error messages is activated, then the flashing error message "EMERGENCY STOP PLC" will be displayed.

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	NC	NC
M2924	PLC error message 0	PLC	NC;PLC
M2925	PLC error message 1		
M2926	PLC error message 2		
M2927	PLC error message 3		
M2928	PLC error message 4		
M2929	PLC error message 5		
M2930	PLC error message 6		
M2931	PLC error message 7		
M2932	PLC error message 8		
M2933	PLC error message 9		
M2934	PLC error message 10		
M2935	PLC error message 11		
M2936	PLC error message 12		
M2937	PLC error message 13		
M2938	PLC error message 14		
M2939	PLC error message 15		
M2940	PLC error message 16		
M2941	PLC error message 17		
M2942	PLC error message 18		
M2943	PLC error message 19		
M2944	PLC error message 20		
M2945	PLC error message 21		
M2946	PLC error message 22		
M2947	PLC error message 23		
M2948	PLC error message 24		
M2949	PLC error message 25		
M2950	PLC error message 26		
M2951	PLC error message 27		

M2952	PLC error message 28
M2953	PLC error message 29
M2954	PLC error message 30
M2955	PLC error message 31
M2956	PLC error message 32
M2957	PLC error message 33
M2958	PLC error message 34
M2959	PLC error message 35
M2960	PLC error message 36
M2961	PLC error message 37
M2962	PLC error message 38
M2963	PLC error message 39
M2964	PLC error message 40
M2965	PLC error message 41
M2966	PLC error message 42
M2967	PLC error message 43
M2968	PLC error message 44
M2969	PLC error message 45
M2970	PLC error message 46
M2971	PLC error message 47
M2972	PLC error message 48
M2973	PLC error message 49
M2974	PLC error message 50
M2975	PLC error message 51
M2976	PLC error message 52
M2977	PLC error message 53
M2978	PLC error message 54
M2979	PLC error message 55
M2980	PLC error message 56
M2981	PLC error message 57
M2982	PLC error message 58
M2983	PLC error message 59
M2984	PLC error message 60
M2985	PLC error message 61
M2986	PLC error message 62
M2987	PLC error message 63
M2988	PLC error message 64
M2989	PLC error message 65
M2990	PLC error message 66
M2991	PLC error message 67
M2992	PLC error message 68
M2993	PLC error message 69
M2994	PLC error message 70
M2995	PLC error message 71
M2996	PLC error message 72
M2997	PLC error message 73
M2998	PLC error message 74
M2999	PLC error message 75
M3000	PLC error message 76
M3001	PLC error message 77
M3002	PLC error message 78
M3003	PLC error message 79
M3004	PLC error message 80
M3005	PLC error message 81

M3006	PLC error message 82
M3007	PLC error message 83
M3008	PLC error message 84
M3009	PLC error message 85
M3010	PLC error message 86
M3011	PLC error message 87
M3012	PLC error message 88
M3013	PLC error message 89
M3014	PLC error message 90
M3015	PLC error message 91
M3016	PLC error message 92
M3017	PLC error message 93
M3018	PLC error message 94
M3019	PLC error message 95
M3020	PLC error message 96
M3021	PLC error message 97
M3022	PLC error message 98
M3023	PLC error message 99



6.6 Cycles

The HEIDENHAIN contouring controls permit calling HEIDENHAIN standard cycles within the NC program (e.g.. pecking, tapping, pocket milling etc.).

In addition, the machine tool builder can program manufacturer cycles and store them in the control (see chapter "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 "Spindle."

6.6.1 Cycle inhibit

Machine parameter MP7245 can be used to selectively inhibit the HEIDENHAIN standard cycles.

MP7245.0 Inhibit the HEIDENHAIN Cycles 1 to 15		
Input range: 0 to 65 534		
Bit 1	Cycle 1	+ 0 = enable + 2 = inhibit
Bit 2	Cycle 2	+ 0 = enable + 4 = inhibit
Bit 3	Cycle 3	+ 0 = enable + 8 = inhibit
Bit 4	Cycle 4	+ 0 = enable + 16 = inhibit
Bit 5	Cycle 5	+ 0 = enable + 32 = inhibit
Bit 6	Cycle 6	+ 0 = enable + 64 = inhibit
Bit 7	Cycle 7	+ 0 = enable + 128 = inhibit
Bit 8	Cycle 8	+ 0 = enable + 256 = inhibit
Bit 9	Cycle 9	+ 0 = enable + 512 = inhibit
Bit 10	Cycle 10	+ 0 = enable + 1024 = inhibit
Bit 11	Cycle 11	+ 0 = enable + 2048 = inhibit
Bit 12	Cycle 12	+ 0 = enable + 4096 = inhibit
Bit 13	Cycle 13	+ 0 = enable + 8192 = inhibit
Bit 14	Cycle 14	+ 0 = enable + 16 384 = inhibit
Bit 15	Cycle 15	+ 0 = enable + 32 768 = inhibit

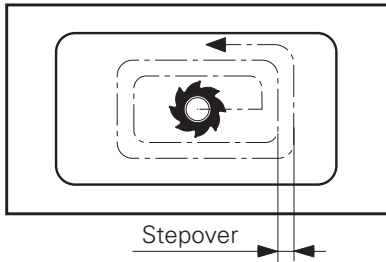
MP7245.1**Inhibit the HEIDENHAIN standard cycles 16 to 31**

Input range: 0 to 65 535

Bit 0	Cycle 16	+ 0 = enable + 1 = inhibit
Bit 1	Cycle 17	+ 0 = enable + 2 = inhibit
Bit 2	Cycle 18	+ 0 = enable + 4 = inhibit
Bit 3	Cycle 19	+ 0 = enable + 8 = inhibit
Bit 4	Cycle 20	+ 0 = enable + 16 = inhibit
Bit 5	Cycle 21	+ 0 = enable + 32 = inhibit
Bit 6	Cycle 22	+ 0 = enable + 64 = inhibit
Bit 7	Cycle 23	+ 0 = enable + 128 = inhibit
Bit 8	Cycle 24	+ 0 = enable + 256 = inhibit
Bit 9	Cycle 25	+ 0 = enable + 512 = inhibit
Bit 10	Cycle 26	+ 0 = enable + 1024 = inhibit
Bit 11	Cycle 27	+ 0 = enable + 2048 = inhibit
Bit 12	Cycle 28	+ 0 = enable + 4096 = inhibit
Bit 13	Cycle 29	+ 0 = enable + 8192 = inhibit
Bit 14	Cycle 30	+ 0 = enable + 16 384 = inhibit
Bit 15	Cycle 31	+ 0 = enable + 32 768 = inhibit

6.6.2 Pocket milling

The overlap factor for clearing out a rectangular or circular pocket (Cycle 4 and Cycle 5) can be altered by machine parameters.



$$\text{Stepover} = (\text{MP7430}) \times \text{cutter radius}$$

MP7430

Overlap factor for pocket milling

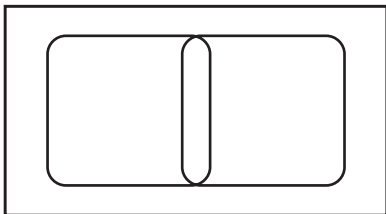
Input range: 0.1 to 1.414

6.6.3 Milling cycles for pockets with free-programmed contours

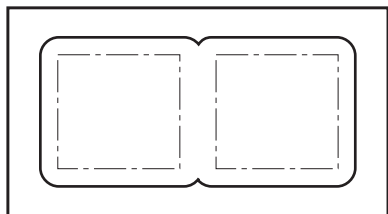
The sequence for milling cycles for pockets with free programmed contours (Cycles 6, 14, 15, 16) can be changed by machine parameters.

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 center 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.

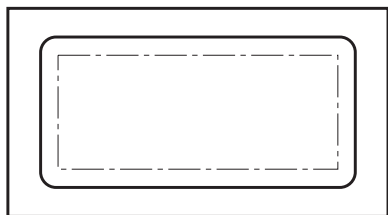


The programmed contours of two pockets have a small overlap.



MP7420 Bit 2 = 0 (entry +0):

The control clears out the pockets separately, since the tool center 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.

Input range: 0 to 15

Bit 0	Direction for channel milling	+ 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 clearing and channel milling	+ 0 =	First channel milling, then clear out pocket
		+ 2 =	First clear out pocket , then channel milling
Bit 2	Merge programmed contours	+ 0 =	Contours merged only if the tool center 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
		+ 8 =	For each peck, first perform channel milling, and then clearing out (depending on Bit 1) before next peck

6.6.4 Scaling factor

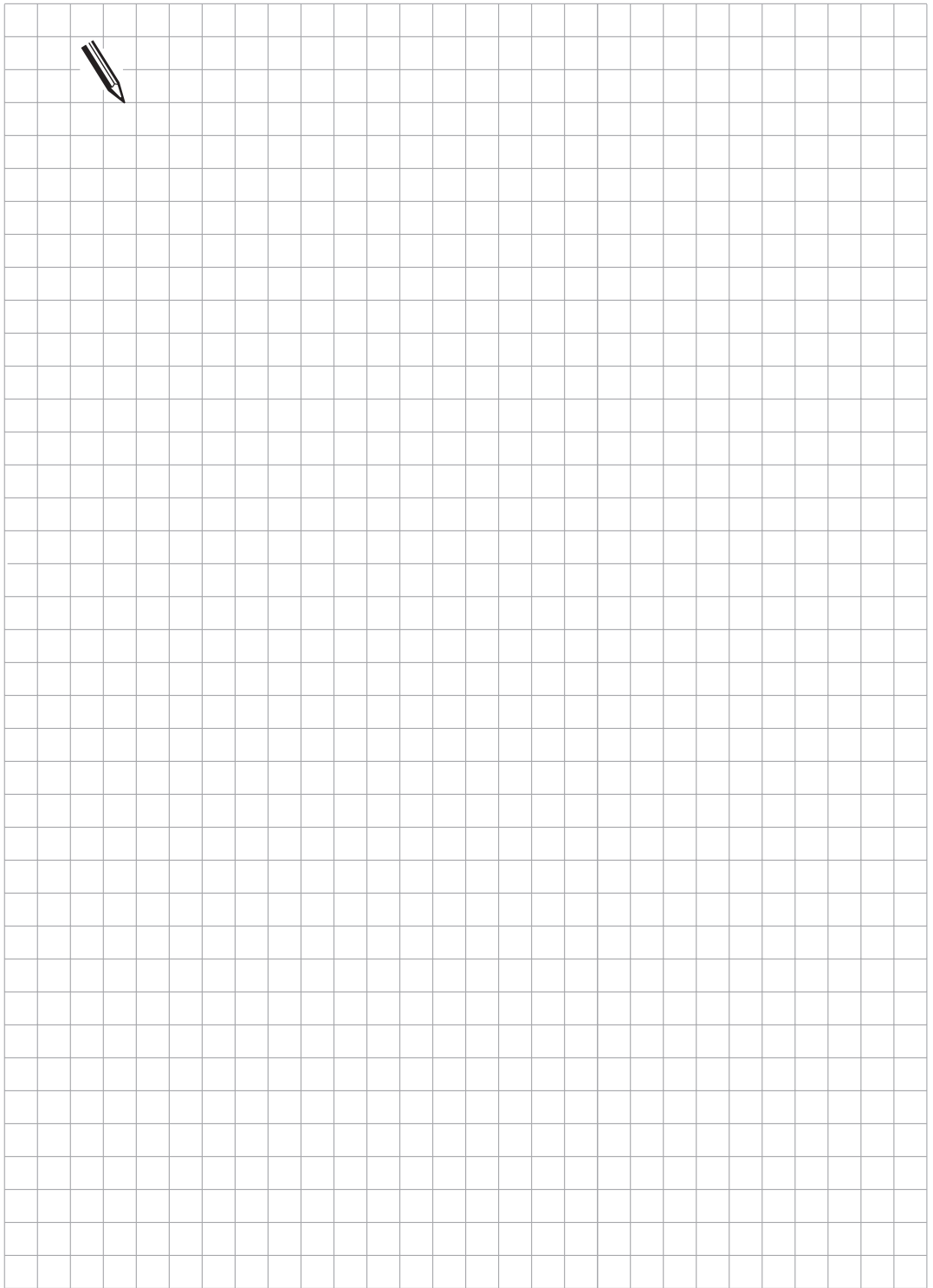
Machine parameters can be used to decide whether the "Scaling factor" cycle only operates in the machining plane or also parallel to the tool axis.

MP7410 "Scaling factor" cycle active in two or three axes

Entry: 0 or 1

0 = Cycle "Scaling factor" operates in all three principle axes

1 = Cycle "Scaling factor" only operates in the machining plane



6.7 File types

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.

- HEIDENHAIN dialog programs
- ISO programs

ISO files are indicated by the file name extension ISO.

6.7.1 File types – disable

MP7222 determines whether the TNC executes HEIDENHAIN dialog or ISO programs.

This selection can be made either through the MOD functions (see User's Manual for the TNC 360) or the corresponding file types are disabled directly. If the ISO programming is activated via MOD functions, marker M2060 is set.



If a file type is disabled all files of that type are erased.

MP7222 **Programming language selection: dialog or ISO**
Input: 0 to 2

- 0 = Selection via MOD function
- 1 = Programming in HEIDENHAIN dialog
- 2 = Programming in ISO

M2060	ISO programming is selected	Set	Reset
		NC	NC

6.7.2 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**
Input range: 0 to 255

6.8 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 dialog appears on the screen. Machine parameter MP7340.x determines which dialog should be displayed.

The dialog is taken from the PLC error file (input 0 to 99) or from the dialog file (100 to 199).

MP7330 Determination of the User Parameters

Input range: 0 to 9999.00 (No. of the desired machine parameter).

MP7330.0	User parameter 0
MP7330.1	User parameter 1
.	.
.	.
MP7330.14	User parameter 14
MP7330.15	User parameter 15

MP7340 Dialogs for User Parameters

Input range: 0 to 199 (line number of the PLC dialog)

MP7340.0	Dialog for User parameter 0
MP7340.1	Dialog for User parameter 1
.	.
.	.
MP7340.14	Dialog for User parameter 14
MP7340.15	Dialog for User parameter 15

6.9 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 nonlinear 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 the control while pressing the MOD key, then enter the code number.
	The entire contents of the RAM memory (MPs, NC programs, PLC programs, all markers, all bytes) are erased.
857 282	Clock displays

6.10 Programming station

Machine parameters can be used to set the control so that it can be utilized without a machine tool as a programming station, without the machine. In this setting only the operating modes "Programming and editing" and "Test run" function. It is possible to select whether the PLC should be active or not in the "programming station" setting.

MP7210	Programming station
	Input values: 0 to 2
Entry:	0 = Control and programming 1 = Programming station, "PLC active" 2 = Programming station, "PLC inactive"

6.11 Dialog language

The HEIDENHAIN contouring controls are available in ten different dialog languages, see chapter "Introduction." The dialog 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	Switching the dialog language Entry: 0 or 1 0 = First dialog language 1 = Basic language - English		
Marker	Function	Set	Reset
M 2041	Basic language - English is selected	NC	NC

6.12 Decimal sign

The decimal sign can be selected by machine parameter.

MP7280	Decimal sign
	Entry: 0 or 1
	0 = Decimal comma 1 = Decimal point

6.13 Memory test

A machine parameter can be used to select if the RAM and the EPROM memory areas should be tested upon switching on the control.

MP7690	Memory test at switch-on Input range: 0 to 3	
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 + 2 = No memory test at switch-on

6.14 End of program run

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.

M2061	END PGM, M02 or M30 has been executed	Set NC	Reset NC
--------------	---------------------------------------	-----------	-------------

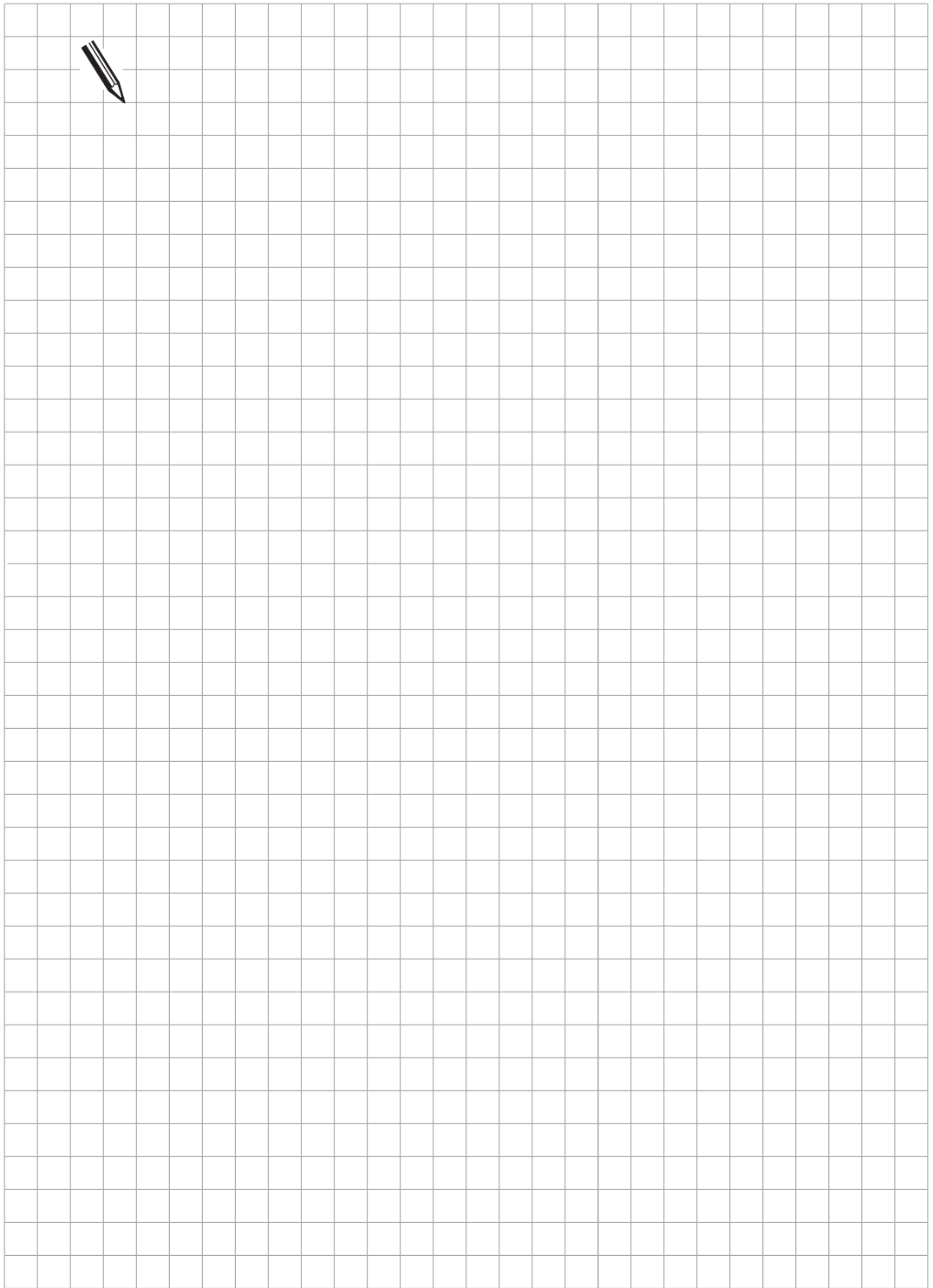
6.15 Overwrite Q parameters overwriting

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.

The Doubleword D528 has a multiple usage. See also sections "PLC positioning" and "Datum shift."

W516	Number of the Q parameter to be overwritten (Q100 to Q107 = 0 to 7)		
D528	Value to be transferred to the Q Parameter		
M2713	Activate the transfer of the value from D528 to the Q Parameter defined in W516	Set PLC	Reset NC



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 assignable.

M function	Effective at: Beginn- End of ing of block block-	
* M 00		●
M 01		●
* M 02		●
* M 03	●	
* M 04	●	
* M 05		●
¹ 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 function	Effective at: Beginn- End of ing of block block	
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 function	Effective at: Beginn- End of ing of block block	
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 signaled 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.

		Set	Reset	
W260	Code for M function	NC		
		Set	Reset	
M2045	Strobe signal for M function	NC	NC	
M2482	Acknowledgment of M function, acknowledgment of		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	

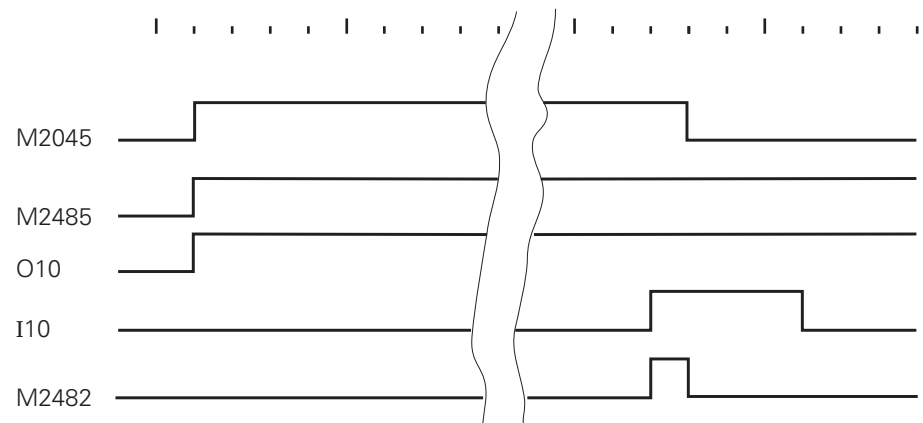
Example:

Evaluation of the miscellaneous function M03 in the PLC.

PLC output: O10 = Spindle ON/OFF

PLC input: I10 = Acknowledgment of M function

```
100      LN   M1
101      S    M1          ;Generate one
.
.
199      L    M2045       ;Strobe signal for M function
200      RN   M2482       ;Reset acknowledgment 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, algebraic sign S analog
1173     R    M2486       ;Reset M04
1174     R    M2487       ;Reset M05
1175     S    O10         ;Spindle ON
1176     L    I10         ;Input for acknowledgment of M function?
1177     EM
```



7.1 Program halt on 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). In such applications the program should be executed continuously and not wait for the acknowledgment 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 for milling machines and boring machines.

7.2 Program halt on 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 halt 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 M function 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 canceled 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.

MP7440 Output of M functions
 Input range: 0 to 7

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 + 2 = modal cycle call M89 at end of block
Bit 2	Program halt on M functions	+ 0 = Program halt until acknowledgment of M function + 4 = No program halt do not wait for acknowledgment

8 Key Simulation

The HEIDENHAIN contouring control is operated through the keys on the TNC keyboard (TE 355) and the manufacturer's own machine control panel. The two control panels are joined to connectors X23 and X27 on the logic unit by a connecting cable (see Chapter "Mounting 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 X27. These PLC inputs and outputs must be evaluated by the PLC and the appropriate information passed to the NC.

8.1 TNC keyboard

The key code from the TNC keyboard is directly evaluated by the NC.

All keys on the TNC keyboard can be inhibited by the PLC with M2855 to M2923. If an inhibited key is pressed, the NC sets the marker M2182 and stores the corresponding key code in Word W274. The PLC must reset the marker M2182 after evaluating this information.

The keys on the TNC keyboard 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. For a key simulation the operating mode can be taken from W272.

		Set	Reset
W272	Operating mode 0 = Programming and editing 1 = Manual operation 2 = Electronic handwheel 3 = Positioning with manual entry 4 = Program run/single block 5 = Program run/full sequence 6 = Test run 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 simulation of TNC keys Activate with M2813	PLC	PLC
		S	R
M2182	Inhibited key was operated	NC	PLC
M2813	Activate the key from W516	PLC	NC

Marker	Function	Key code	Set	Reset
M2855	 inhibit	59	PLC	PLC
M2856	 inhibit	60	PLC	PLC
M2857	 inhibit	61	PLC	PLC
M2858	 inhibit	62	PLC	PLC
M2859	 inhibit	63	PLC	PLC
M2860	 inhibit	64	PLC	PLC
M2861	 inhibit	65	PLC	PLC
M2862	 inhibit	66	PLC	PLC
M2863	 inhibit	67	PLC	PLC
M2864	 inhibit	68	PLC	PLC
M2865	 inhibit	69	PLC	PLC
M2867	 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	 inhibit	78	PLC	PLC
M2880	 inhibit	84	PLC	PLC
M2881	 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	 inhibit	91	PLC	PLC
M2888	 inhibit	92	PLC	PLC
M2889	 inhibit	93	PLC	PLC
M2890	 inhibit	94	PLC	PLC

Marker	Function	Key code	Set	Reset
M2891	 inhibit	95	PLC	PLC
M2892	 inhibit	96	PLC	PLC
M2893	 inhibit	97	PLC	PLC
M2894	 inhibit	98	PLC	PLC
M2895	 inhibit	99	PLC	PLC
M2896	 inhibit	100	PLC	PLC
M2897	 inhibit	101	PLC	PLC
M2898	 inhibit	102	PLC	PLC
M2899	 inhibit	103	PLC	PLC
M2900	 inhibit	104	PLC	PLC
M2901	 inhibit	105	PLC	PLC
M2902	 inhibit	106	PLC	PLC
M2903	 inhibit	107	PLC	PLC
M2904	 inhibit	108	PLC	PLC
M2905	 inhibit	109	PLC	PLC
M2906	 inhibit	110	PLC	PLC

Marker	Function	Key code	Set	Reset
M2907	 inhibit	111	PLC	PLC
M2908	 inhibit	112	PLC	PLC
M2909	 inhibit	113	PLC	PLC
M2910	 inhibit	114	PLC	PLC
M2911	 inhibit	115	PLC	PLC
M2912	 inhibit	116	PLC	PLC
M2913	 inhibit	117	PLC	PLC
M2914	 inhibit	118	PLC	PLC
M2915	 inhibit	119	PLC	PLC
M2920	 inhibit	124	PLC	PLC
M2921	 inhibit	125	PLC	PLC
M2922	 inhibit	126	PLC	PLC
M2923	 inhibit	127	PLC	PLC
M2916	 inhibit	120	PLC	PLC
M2917	 inhibit	121	PLC	PLC

Marker	Function	Key code	Set	Reset
M2918	 inhibit	122	PLC	PLC
M2919	 inhibit	123	PLC	PLC

Example:

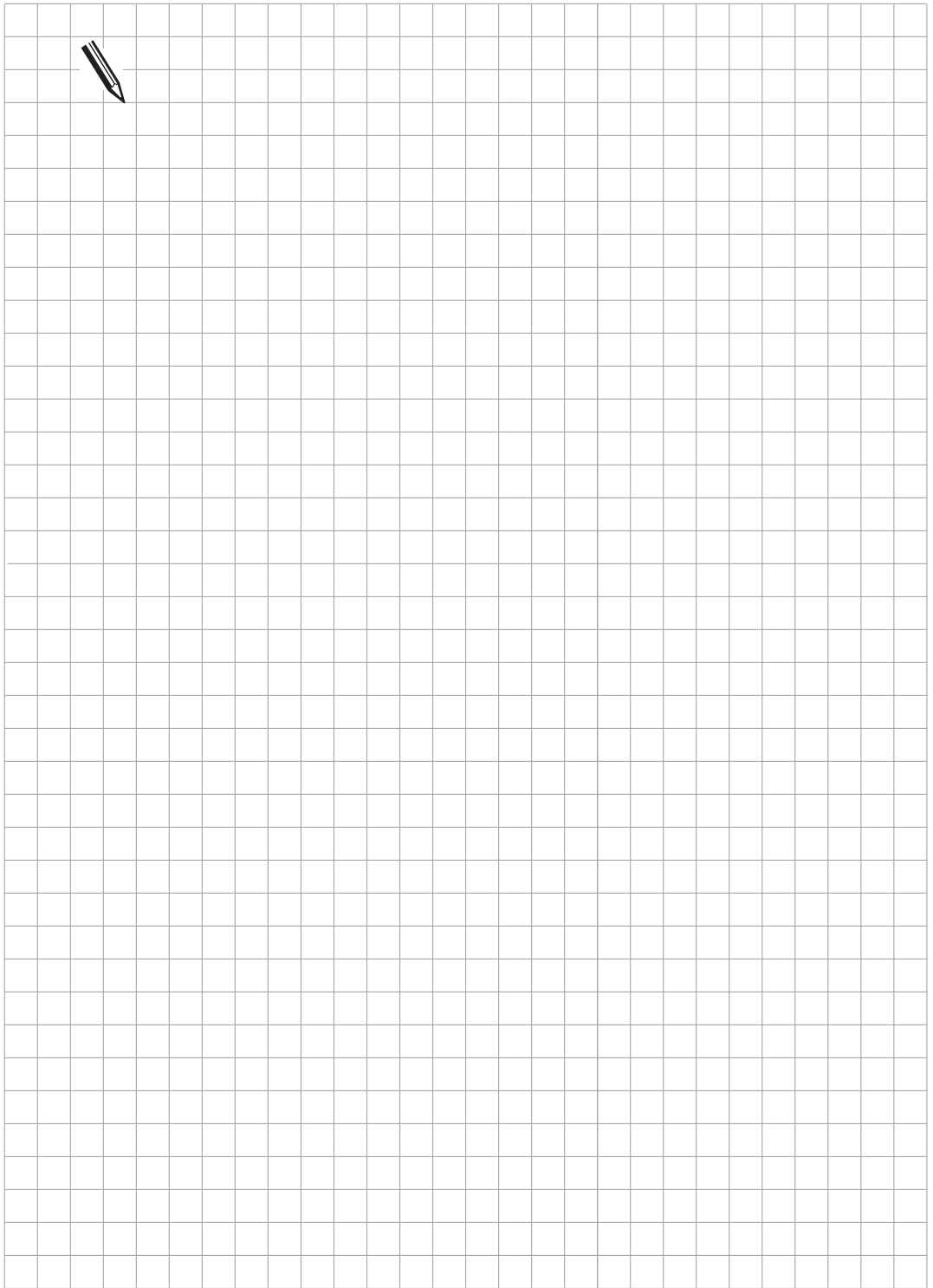
If the "Position transfer" key is pressed in the operating mode "Positioning with manual entry", a linear NC block with all three principal coordinates (X, Y, Z) is to be generated.

```

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
1130         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	
1140	LBL 133	;Y
1141	L K108	;Key code for Y
1142	= W 102	
1143	CM 136	;Simulate key
1144	EM	
1145	LBL 134	;Z
1146	L K107	;Key code for Z
1147	= W102	
1148	CM 136	;Simulate key
1149	EM	
1150	LBL 135	;"END BLOCK"
1151	L K119	;Key code for "END BLOCK"
1152	= W102	
1153	CM 136	;Simulate key
1154	EM	
1155	LBL 136	;Key simulation
1156	L M2813	;Strobe key transfer from W516
1157	JPT 137	;Still set, then wait
1158	L B200	;Case byte
1159	+ K+1	
1160	= B200	;Increment case byte
1161	L W102	;Buffered key code
1162	= W516	;To NC
1163	LN M2813	
1164	S M2813	;Set strobe (activate simulation)
1165	EM	
1166	LBL 137	;Return marker
1167	EM	
1168	LBL 141	;End key simulation
1169	L M2813	;Simulation performed?
1170	JPT 137	;No, then wait
1171	L K+0	
1172	= B200	;Reset step counter
1173	L M10	
1174	R M10	;Reset marker "Key simulation active"
1175	R M2182	;Reset marker "Disabled key operated"
1176	EM	



8.2 Machine control panel

A manufacturer's specific machine control panel can be connected to the HEIDENHAIN contouring controls. See Chapter "Mounting and electrical installation".

24 PLC inputs (I 128 to I 151) and 8 PLC outputs (O 0 to O 7) are available on the female connector X27 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 - Memory function for axis-direction keys, memory function for

Input: 0 or 1

+ 0 = Not stored

+ 1 = Stored

Marker	Function	Error message	Set	Reset
M2448 M2464	NC start (edge evaluation) Complement NC start	1A	PLC	PLC
M2449 M2465	Rapid traverse Complement – rapid traverse	1B		
M2488	NC stop ("0" signifies stop)	–		
M2450 M2466	Memory function for axis-direction keys Complement – memory function for axis-direction keys	1C		
M2451 M2467	Feed release for all axes Complement – feed release	1D		
M2456 M2472	Manual traverse X+ Complement - manual traverse X+	1I		
M2457 M2473	Manual traverse X- Complement - manual traverse X-	1J		
M2458 M2474	Manual traverse Y+ Complement - manual traverse Y+	1K		
M2459 M2475	Manual traverse Y- Complement - manual traverse Y-	1L		

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-			

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      .
72      L I128          ;First contact NC start key
73      = M2448         ;NC start
74      LN I129         ;Second contact NC start key
75      = M2464         ;Complement – NC start

100     .
101     L I130          ;Axis-direction key X+
102     = M2456         ;Manual traverse X+
103     LN I130         ;Complement - manual traverse X+
104     = M2472
105     .

```

9 Touch Probe

The 3-D TS Touch Probes by HEIDENHAIN are available in two versions:

- 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).

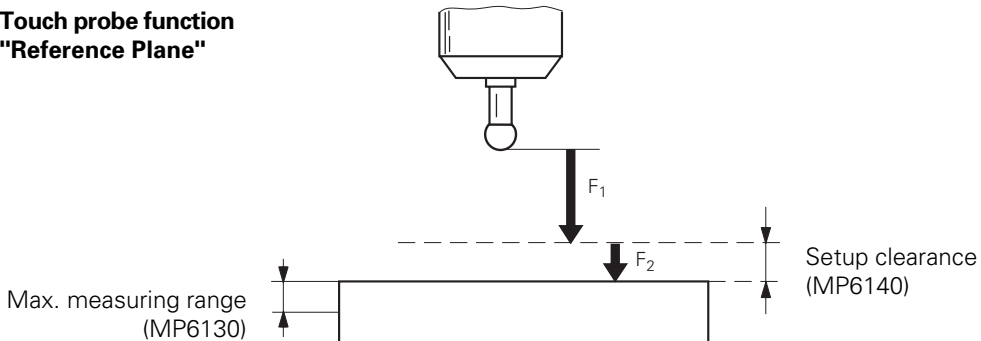
The chapter "Mounting and electrical installation" contains instructions for connecting the touch probes.

9.1 Interfacing the touch probe

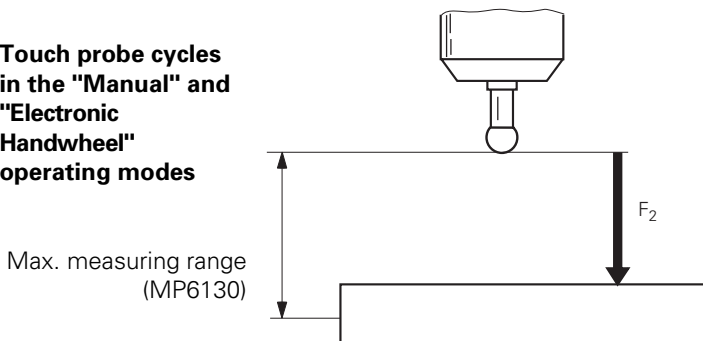
Machine parameter MP6010 is used to select either touch probe TS 120 or TS 511. The machine tool manufacturer must ensure that the spindle is interlocked when the touch probe is in use.

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 TNC 360 User's Manual). The touch probe is interfaced to the measuring conditions using machine parameters MP6120 to MP6150.

Touch probe function "Reference Plane"



Touch probe cycles in the "Manual" and "Electronic Handwheel" operating modes



F_1 = Rapid traverse during probing cycle (MP6150)

F_2 = Probing feed rate (MP6120)

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 after 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 M2072. This information can be processed further in the PLC program. The probing function is controlled entirely from the NC.

In all modes when the stylus is deflected and marker M2502 is set, the controller stops the machine. (This only applies for TS 120!). 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.

The machine parameter MP7411 determines whether, during execution of the TOUCH PROBE block the current compensation data for touch probe length and radius is taken from the tool file or from the probe calibration process. MP7411 is also effective for the tool length compensation in the digitizing cycles.

MP6010	Selection of the touch probe system Input: 0 or 1 0 = TS 120 1 = TS 511
MP6120	Probing feed rate Input range: 80 to 30 000 [mm/min]
MP6130	Maximum measuring range Input range: 0.001 to 99 999.999 [mm]
MP6140	Setup clearance above measuring point Input range: 0.001 to 99 999.999 [mm]
MP6150	Rapid traverse in probing cycle Input range: 80 to 30 000 [mm/min]
MP7411	Compensation data for TOUCH PROBE block Input: 0 or 1 0 = Compensation value from the calibration process 1 = Compensation value from the tool file

		Set	Reset
M2502	NC STOP in all operating modes if stylus is deflected	PLC	PLC
M2503	Enable marker for probing functions	NC	PLC
M2022	Touch probe not ready (no standby signal at connector X12)	NC	NC
M2023	Stylus deflected before start of probing cycle	NC	NC
M2025	Stylus deflected (probing sequence is executed)	NC	PLC
M2026	Probe operation ended or interrupted	NC	NC
M2027	Battery voltage too low (battery warning on connector X12); evaluated only during the probe operation	NC	NC

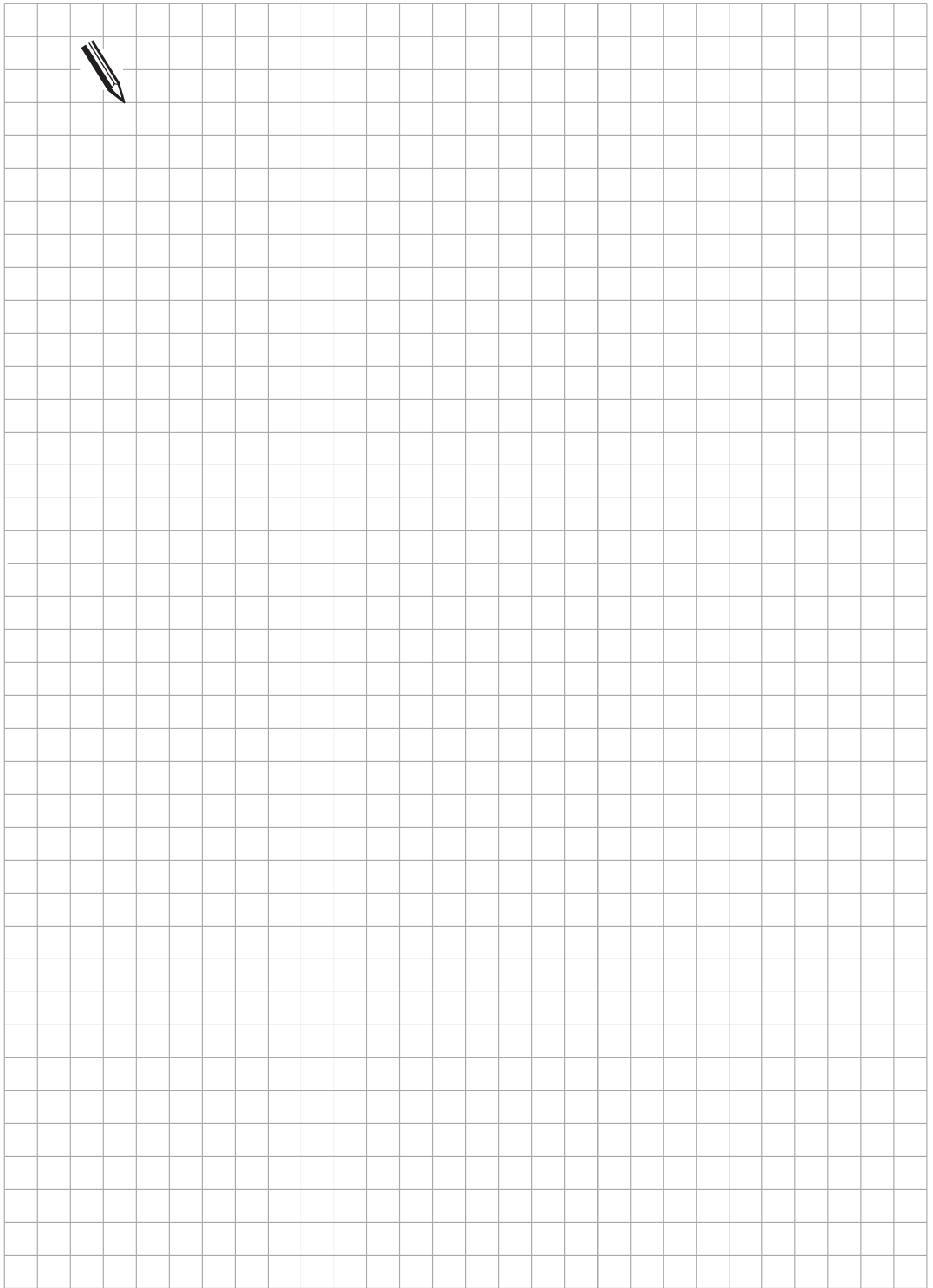
Example:

Enable of probing function in the PLC.

```

.
.
.
489   LN      I9      ;I9 = 0 => TS not in holder
490   =      M2502    ;stop if stylus deflected
491   ;
492   L      M2503
493   AN     I9
494   R      M2503    ;acknowledge probing cycle
495   ;
496   ;
497   ;safety function
498   ;M03/M04 deactivate if TS not in holder
499   ;
500   L      M2485    ;M03 activated?
501   O      M2486    ;M04 activated?
502   AN     I9      ;touch probe not in holder
503   S      M2487    ;display M05
504   R      M2485    ;deactivate M03
505   R      M2486    ;deactivate M04
506   R      M922     ;clear buffered marker M03
507   R      M923     ;clear buffered marker M04
508   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 360 using software types 259 90x03 (TNC 360) and 260 02x 04 (TNC 360 C) See also Chapter "Introduction", Section "Software".
- 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 Chapter "Introduction", Sections "EPROM locations", and in Chapter "Mounting and electrical installation", Section "Hardware components"). 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 "Servo lag 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 (servo lag 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, P.P.INT from the scanning cycles "Meander" and "Contour Lines":

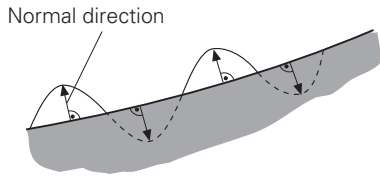
$$F \text{ [mm/min]} = \text{PP.INT [mm]} \times \text{oscillations [1/s]} \times 60 \text{ [s/min]}$$

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. If the input value is too small, the touch probe system may get caught up in a repeated attempt to get out of an inside corner.

Machine parameter MP6260 defines an M function that stores technology data such as tool dimensions, spindle speed, feed rate and certain M functions (e.g. M3, M13). A small program is generated containing these technology data and the M functions determined by MP6250. If the program with the digitized data is run in blockwise mode this program is run immediately after the generated program, the technology data is taken over.

Machine parameter MP6260 determines whether an M90 is added to each NC block (see also "Constant feed rate in corners with M90" in Section "Servo positioning of the NC axes").

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 per second in normal direction.

Input range 0 to 65.535 [1/s]

MP6230 Feed rate in normal direction

Input range: 0 to 10 000 [mm/min]

MP6240 Maximum deflection of stylus

Input range: 0 to 10.000 [mm]

MP 6254 Numerical value for M function for transferring modal values

Input value: 0 = no take-over
1 to 89 = take-over with M (1 ... 89)

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 of trailing zeros

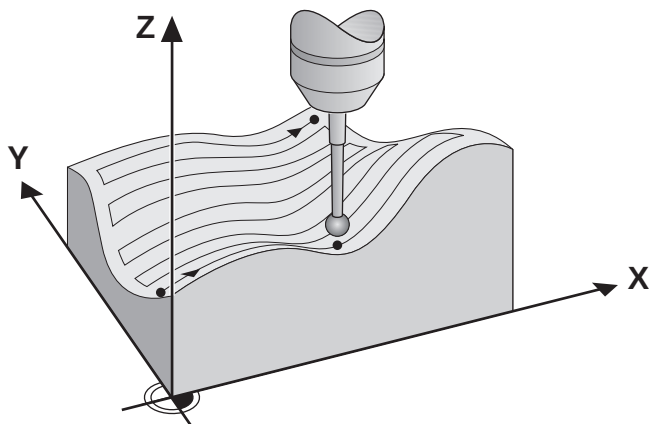
Input value: 0 = output 0.001 mm steps (1 µm)
1 = output 0.01 mm steps (10 µm)

9.2.1 Scanning cycles

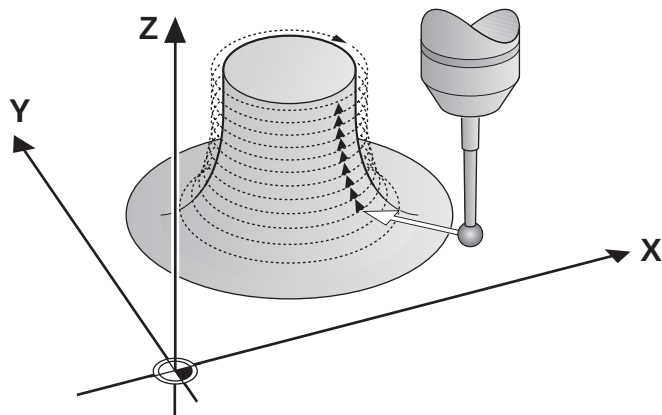
Direct access to the position control loop of the TNC controller allows the rapid recording of measured values (3 to 5 values per second). With a programmed probe point interval of 1 mm, this produces a scanning feed rate of 180 to 300 mm/min. Three scanning cycles are used for digitizing: "Range", "Meander" and "Contour Lines".

The "Range" cycle defines the cuboid scanning range and the file where the digitized data are stored.

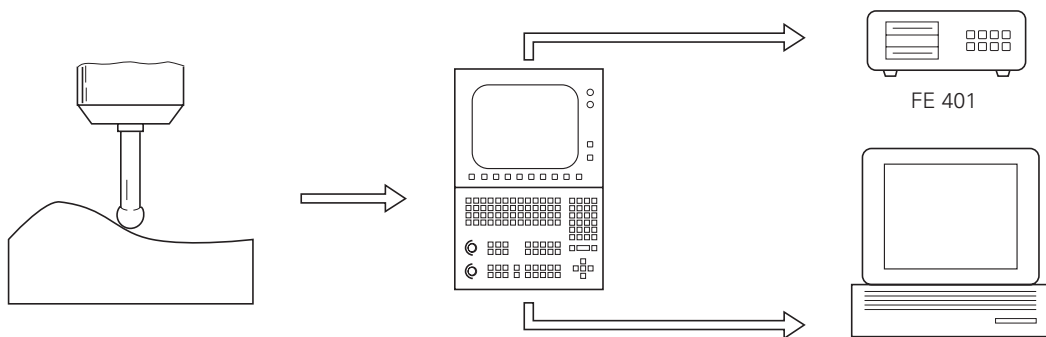
The "Meander" cycle digitizes a 3-D form meander-wise (line by line) in the pre-defined range.



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.



The acquired data can be stored in the controller's program memory, on the FE 401 floppy disk unit or in a PC.



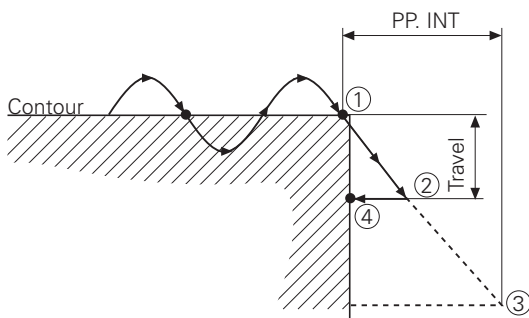
Further particulars about scanning cycles will be found in the TNC 360 User's Manual.

9.2.2 Response of the scanning 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.

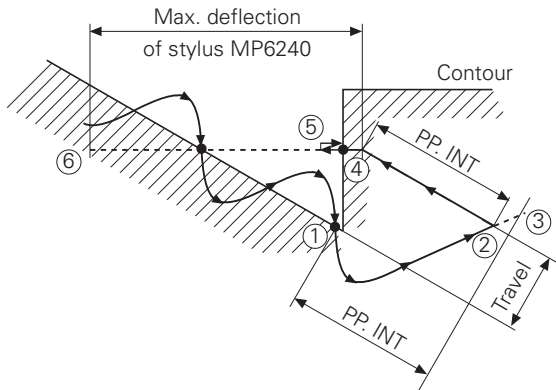
The travel is also responsible for geometrical accuracy at the corners. The smaller the stroke, the greater the accuracy of corner resolution. If too small a stroke is defined however, it may affect clearance at acute inside corners (minimum travel = 0.1 mm).

Outside Corners



On outside corners, having probed the last point 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 ③ 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 paraxially 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 scanning sequence

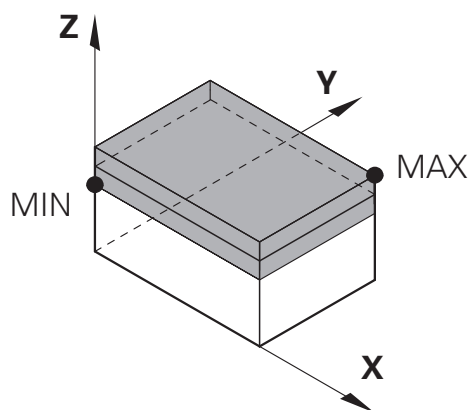
The following preparations should be made before optimizing machine and controller.

- 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"

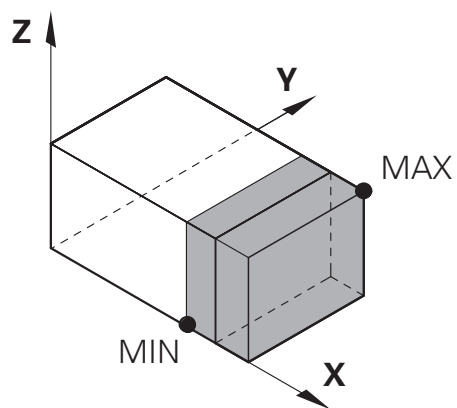
MP6210	=	5 [1/s]	Oscillations in normal direction
MP6230	=	30 [mm/min]	Feed rate in normal direction
MP6240	=	5 [mm]	Maximum stylus deflection
- The interface must be configured according to the type of desired memory for the digitized data (FE 401 or PC with "Block Transfer" program). See TNC 360 User's 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 1 MM	
5	TCH PROBE 5.0 RANGE	Define the range
6	TCH PROBE 5.1 PGM NAME:2	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 stroke (for
12	TCH PROBE 6.2 TRAVEL:0.5 L.SPAC:1 P.P.INT:1	clearing steep edges), line
13	END PGM 1 MM	spacing and probe point
		interval

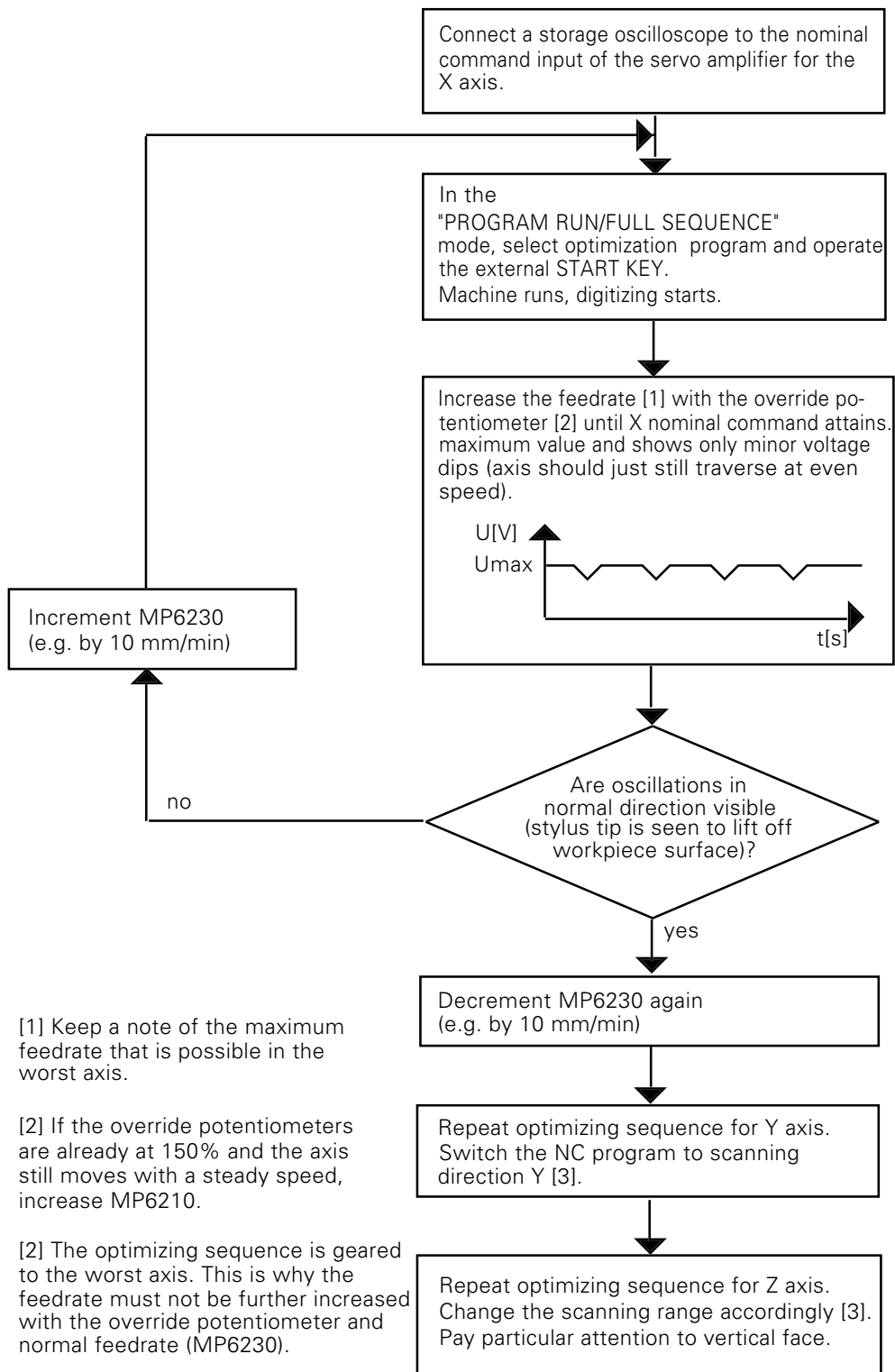
Optimize the X and Y axes by defining the range such that only the level surface of the component is scanned.



Optimize the Z axis by defining the range such that mainly the vertical face is scanned.



Procedure for Optimizing Machine Parameters

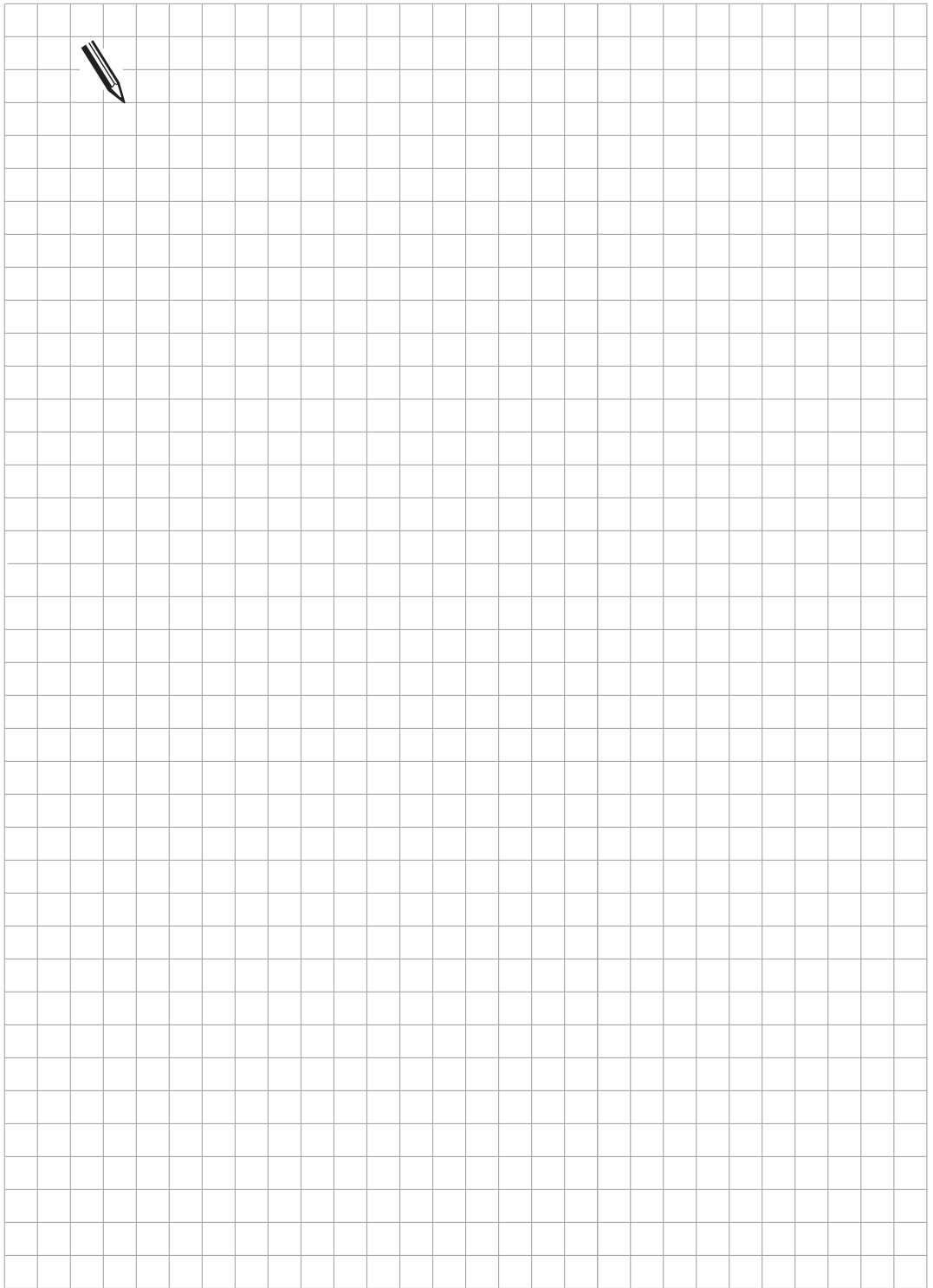


Calculation of Possible Oscillations in Normal Direction

$$\text{MP 6210 [1/s]} = \frac{\text{optimised F } ^{1)} \text{ [mm/min]}}{\text{PP.INT } ^{2)} \text{ [mm]} \times 60 \text{ [s/min]}}$$

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).



10 Electronic Handwheel

Either a

- an panel-mounted handwheel (HR 130)
 - an portable handwheel (HR 330 or HR 331)
 - or up to three single-axis integral handwheels (HR 150) via handwheel adapter (HRA 110)
- can be connected to HEIDENHAIN contouring controls (see also Chapter "Mounting and electrical installation "). The operation of the electronic handwheel is described in the TNC 360 User's Manual.

MP7640 determines which handwheel is connected. If a value greater than zero is entered but no handwheel is connected, the error message "HANDWHEEL?" will appear.

The counting direction for the measuring system signals of the handwheel are entered in machine parameter MP7650.

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. Setting the Marker M2826 inhibits traverse with the handwheel, i.e. handwheel pulses are suppressed.

For HR 130, HR 330, and HR 332 one interpolation factor can be selected in the "Handwheel" operating mode. For HR 150 with HRA 110 up to three can be selected. This interpolation factor determines the traverse distance per handwheel revolution. 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. For the HR 130, HR 330, and HR 332 handwheels, the NC goes by the smallest value which was entered, i.e. according to the slowest axis.

Machine parameter MP7641 bit 0 defines whether the interpolation factor can be entered through the TNC keyboard (see User's Manual) or through the PLC module 9036 (see Chapter 7, section "PLC Modules"). With machine parameter MP7641 bit 2 the interpolation factor for the handwheel can be defined both from the keyboard and from the PLC module 9036. If bit 2 = 0, the machine parameters function as before (entry from the keyboard or from PLC module 9036).

If in machine parameter MP7641 bit 3 is set, the PLC can poll the keys (except for the axis keys) in any mode of operation. This function takes effect before the REF run. The axis keys and the handwheel pulses do not become effective until handwheel mode is active. After an error the handwheel is not driven until the handwheel key is pressed.

In the "Programming and editing" mode of operation the axis can be moved by the handwheel if in machine parameter MP7641 bit 1 = 1. The interpolation factor or the handwheel axis can be changed only in the "Electronic handwheel" mode, regardless of the handwheel model. Once a handwheel axis has been selected, it remains active even after a switch of operating modes. Simultaneous use of the handwheel interface and the RS-232-C interface causes a conflict in baud rates (38 400 and 19 200 baud) resulting in the error message BAUD RATE NOT POSSIBLE.

For the HR 150 with the HRA 110 handwheel adapter the control sets the interpolation factor individually for up to three axes.

Interpolation factor	Traverse distance per handwheel revolution [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.

MP7640 Handwheel
Input: 0 to 6

- 0 = Machine without electronic handwheel
- 1 = Portable handwheel HR 330
- 2 = Panel-mounted handwheel HR 130
- 3 = Portable handwheel HR 330 (evaluation of keys via PLC)
- 4 = Portable handwheel HR 332 (evaluation of keys via PLC)
- 5 = Up to three HR 150 via HRA 110
- 6 = Reserved

MP7641 Entry of interpolation factor
Input: 0 to 15

- Bit 0 +0 = Entry of interpolation factor via TNC keyboard
+1 = Entry of interpolation factor via PLC module 9036
- Bit 1 +0 =
+2 = Handwheel active in the Programming and editing mode
- Bit 2 +0 =
+4 = Entry of interpolation factor via PLC module **and** TNC keyboard
- Bit 3 +0 =
+8 = Key polling (except axis keys) from the PLC in every operating mode

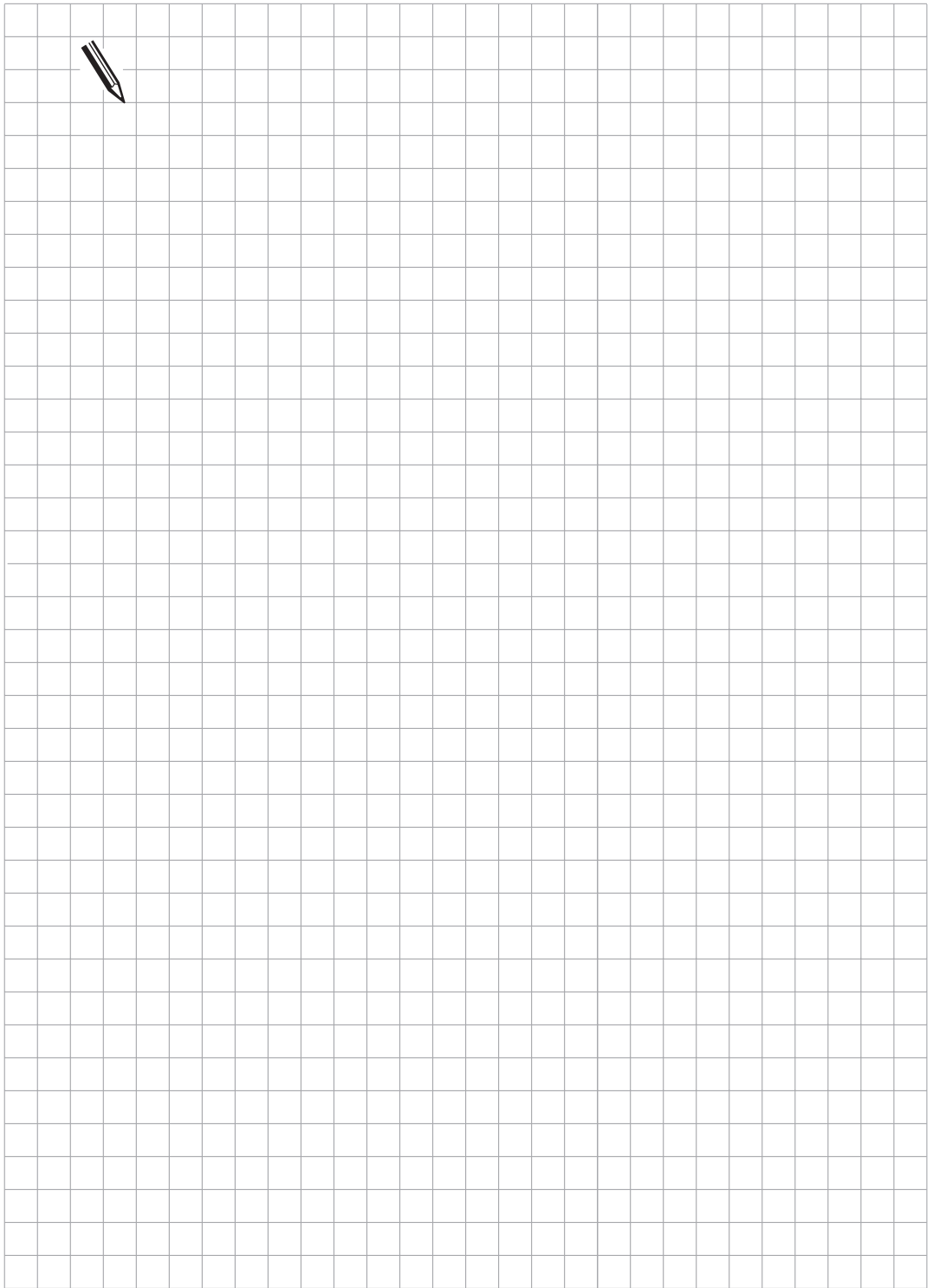
MP7650 Count direction for handwheel
Input: 0 or 1

- 0 = positive count direction
- 1 = negative count direction

MP7660 Threshold sensitivity for electronic handwheel
Input range: 0 to 65 535 [increments]

MP7670 Minimum interpolation factor for handwheel
Input range: 0 to 10

Marker	Function	Set	Reset
M2826	Suppress handwheel pulses	PLC	PLC



10.1 Panel-mounted handwheel HR 130

MP7640 = 2

10.2 Portable handwheel HR 330

MP7640 = 1 with evaluation of all keys through the **NC**

MP7640 = 3 with evaluation of the +, – and rapid traverse keys in the **PLC**

Assignment of keys to the PLC inputs

Key	PLC input
+	I160
–	I161
"Rapid traverse"	I162

10.3 Portable handwheel HR 332

MP7640 = 4 with evaluation of all keys in the **PLC**

MP7645.0 determines whether all 12 keys and their LEDs are accessible via PLC or whether the axis selection keys and their LEDs are excepted.

MP7645.0 Assignment of handwheel keys for HR 332
Input: 0 or 1

- 0 = All keys except axis selection keys and their LEDs are accessible via PLC
- 1 = All 12 keys and LEDs are accessible via PLC

10.3.1 Assignment of keys and LEDs to the PLC inputs and outputs

The 12 keys of the HR 332 handwheel are assigned to various PLC inputs and the 12 indicator LEDs to various PLC outputs according to the input value of the machine parameter MP7645.0.

When a key is pressed, the corresponding PLC input is set. When a PLC output is set, the corresponding LED lights up.

The PLC inputs I160 and I175 generate signal edges. The positive edge is assigned to markers M1660 to M1675, the negative edge is assigned to markers M1860 to M1875.

If MP7645.0 = 0 the keys X, Y, Z, IV and their LEDs are assigned to the NC. The remaining keys are assigned to the 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

If MP7645.0 = 1 all twelve keys are assigned to the PLC inputs I160 to I171 and all LEDs to the 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

In the standard version of the HR 332 the keys assigned the following functions:

X	IV	
Y	V	
Z	~	~
	-	+

The key without symbols can be assigned any function by the PLC program.

10.3.2 PLC example

In the following example the currently selected axis is evaluated so that this axis can be moved with the +/- keys. This example is for the X axis (MP7640=4, MP7641=0 and MP7645.0 = 0).

Key on HR 332	PLC input or output
+ Key	I170
– Key	I169
+ LED	O105
– LED	O106

;Main program

LBL 1

PS K+9 Interrogate the selected handwheel axis with 9035

CM 9035

PL B0

L K+0 Erase the 8 Markers for "manual"

B= M2456 traverse of the axes X to IV

L K\$FF Set the 8 complement markers

B= M2472

A MO Erase the PLC outputs

ON MO O100 to O106.

R O100 All LEDs except the

R O101 LED of the selected

R O102 axis are switched off.

R O103

R O104

R O105

R O106

Case B0 Evaluate the selected handwheel axis

CM 10 X axis

CM 11 Y axis

CM 12 Z axis

CM 13 IV axis

ENDC

EM

LBL 10 X axis

LN M2457 If X– is not active

A I170 and the + key is pressed,

S M2456 the axis moves in X+

R M2472 direction.

S O105 LED on the + key lights up.

LN M2456 If X+ is not active

A I169 and the – key is pressed,

S M2457 the axis moves in X–

R M2473 direction.

S O106 LED on the – key lights up.

EM

LBL 11	Y axis
.	
EM	
LBL 12	Z axis
.	
EM	
LBL 13	IV axis
.	
EM	

10.4 Panel-mounted handwheels HR 150 with handwheel adapter HRA 110

MP7640 = 5

MP7641 determines whether the interpolation factor can be entered directly through the TNC keyboard or through the PLC module 9036 (see also Chapter 7, Section "PLC Modules").

If the switch for the selection of the interpolation factor is used, the inputs I160 to I167 are to be evaluated in binary values before using the PLC Modules 9036. The third handwheel can be assigned to any axis. MP7645.2 determines whether the axis for the third handwheel is selected through the axis selection key (see MP7645.0) or via the machine parameter MP7645.1.

MP7641 **Entry of interpolation factor for HR 150 with HRA 110**
Input: 0 or 1

0 = via TNC keyboard
1 = via PLC module 9036

MP7645 **Assignment of the third handwheel**

MP7645.0 Assignment of third handwheel via axis selection switch
Input: 0 to 2

	Switch setting	3rd handwheel
0:	1 (Left-most setting)	Z axis
	2	IV axis
1:	1 (Left-most setting)	X axis
	2	Y axis
	3	Z axis
	4	IV axis
2:	3	Z axis
	4	IV axis

If no axis selection switch is connected (see also MP 7645.2), the third handwheel is permanently assigned to an axis as entered in MP 7645.1.

MP7645.1 Assignment of 3rd handwheel via machine parameter

Input: 0 to 8

- 0: Simulation of switch setting 1 (Left-most setting)
The third handwheel is assigned from MP7645.0
(Input value 0 or 1).
- 1: X axis
- 2: Y axis
- 4: Z axis
- 8: IV axis

MP7645.2: Assignment of 3rd handwheel via axis selection switch or MP7645.1

Input: 0 or 1

- 0: Assignment via axis selection switch
according to MP7645.0
- 1: Assignment via MP7645.1.
The axis selection switch transmits data to the PLC inputs I168 to I175.
The axis selection switch can therefore also serve other functions.

10.4.1 Assignment of the switch setting to the PLC inputs

The following tables show the assignments of the switch settings of S1 and S2 to the PLC inputs I160 to I175.

Both switches function with 0V logic. If, for example, the switch S1 is at setting 3, then input I162 is logically 0 and inputs I160, I161 and I163 to I167 are logically 1.

Step switch S1

Switch for the selection of the interpolation factor

Switch setting	PLC input
1 (Left-most setting)	I160
2	I161
3	I162
4	I163
5	I164
6	I165
7	I166
8 (Right-most setting)	I167

Step switch S2

Step switch for axis selection

Switch setting	PLC input
1 (Left-most setting)	I168
2	I169
3	I170
4	I171
5	I172
6	I173
7	I174
8 (Right-most setting)	I175

10.4.2 PLC Example

In the following example the settings of switch S1 are assigned for the selection of the interpolation factor through evaluation of the inputs I160 to I167 from the smallest permissible interpolation factor from MP7670 to the largest interpolation factor = 10. They are then transferred from the PLC to the NC (MP7640 = 5 and MP7641 = 1).

In this PLC example the minimum interpolation factor is read and processed with the PLC module 9032 from MP7670.

LB	I160	Assign the inputs I160 to I167
=	B1	to Byte 1 (0V logic, negative logic)
==	K0	If all inputs are 0, then
JPT	103	error message
LN	B1	Invert Byte 1 (positive logic)
=	B2	
==	K0	If all Bits are 0, then
JPT	103	error message
LBL	100	Loop for bit test of the "inverted inputs"
L	B2	I160 to I167 in Byte 2
BT	B3	
JPT	101	Jump if Bit is set, i.e. input Ixxx = 0V
L	B3	prepare next Bit for interrogation, i.e.
+	K+1	read input, if input Ixxx was not 0V.
=	B3	
JP	100	
LBL	101	
PS	K+7670	Read the smallest interpolation factor
PS	K+0	from MP7670 with PLC module 9032
CM	9032	

PL	B4	Result is in Byte 4
L	B3	Current switch setting (0 to 7) plus
+	B4	minimum interpolation factor from MP7670
=	B5	is new interpolation factor
<=	K+10	If the new factor is greater than 10,
JPT	102	
L	K+10	... then the new interpolation factor is set
=	B5	to equal 10.
LBL	102	
PS	K+4	Transfer new interpolation factor
PS	B5	to all axis interpolation (PLC ->
CM	9036	NC) factors.
PL	B6	Read error code
L	B6	If transfer is faulty, then
>	K0	error message
JPT	103	
EM		
.		
LBL	103	Error-message subprogram
.		

Example:

Switch setting: 4

Minimum interpolation factor: MP 7670 = 5

In this PLC example are the following events:

Inputs: I160 to I162 and I164 to I167 are logically 1

Input: I163 is logically 0

Byte 1: 11110111

Byte 2: 00001000

Byte 3: 3

Byte 4: 5

Byte 5: 8

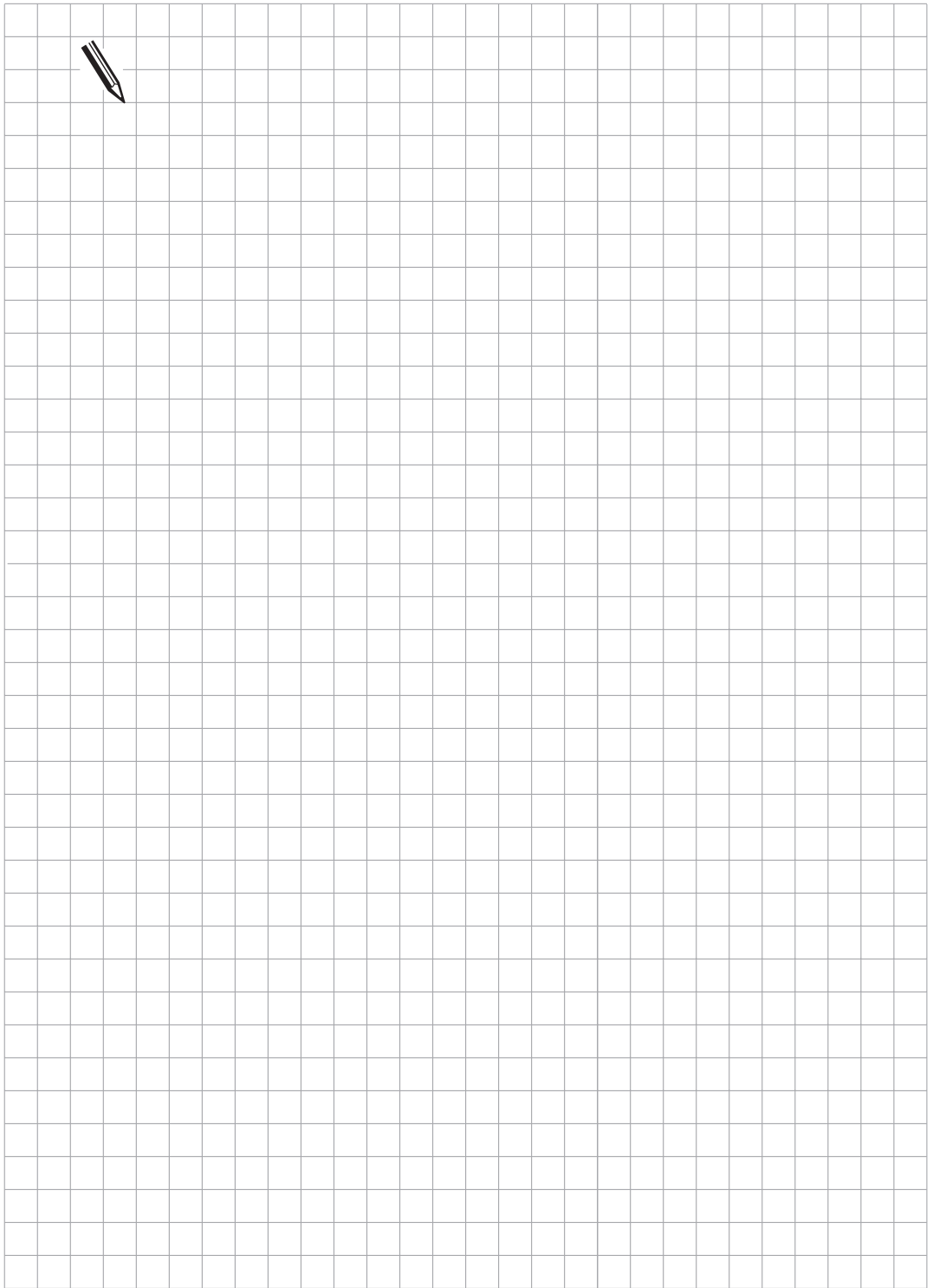
In the above example, the switch S2 is configured as follows:

Switch setting	Interpolation factor	
1 (Left-most setting)	5	
2	6	
3	7	
4	8	Present switch setting
5	9	
6	10	
7	10	
8 (Right-most setting)	10	

11 Analog Inputs

The function of the TNC 360's analog input for displaying spindle power is described in Section "Expanded spindle display" (see 6.3.8).

The analog input is located at connector X8 (Pin 2 analog input, Pin 10 0V for analog input).



12 Incremental Jog Positioning

The "incremental jog" function for the electronic handwheel operating mode can be activated through the PLC program. This function permits the user to enter a jog increment (0.001 to 50 mm) by which an axis moves whenever a machine axis direction button (e.g. X+) is pressed.

The incremental jog positioning function is activated through the PLC marker M2498. In the electrical handwheel mode the dialogue "jog increment" is then displayed to prompt the user to enter the jog increment.

Incremental jog positioning is normally executed through axis direction keys. In order to cause only a single movement when the button is pressed, the PLC input signal for the axis direction button must be evaluated for one edge (e.g. positive edge). See also Chapter "PLC programming", Section "Edge evaluation of the PLC inputs".

This inhibits the function for manual traverse in the electronic handwheel mode when the axis direction button is pressed for a long time. Each time the button is pressed, the axis moves only once and only by the entered jog increment.

In the PLC program the start marker and the corresponding complement marker must be taken into account. If this is not the case the error message "PLC ERROR xx" appears.

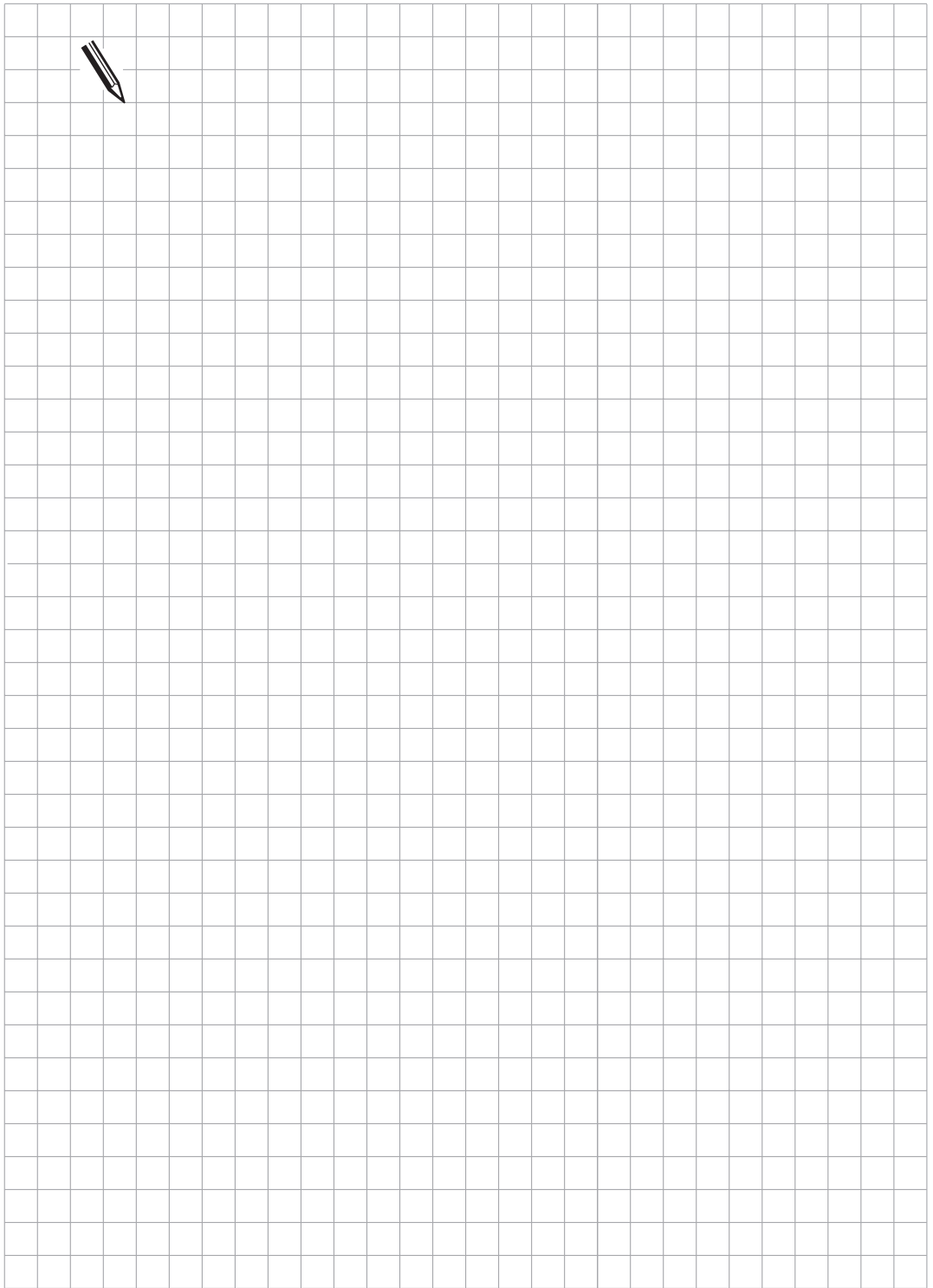
Marker	Function and input	Error message	Set	Reset
M2052	Operating mode: Electronic handwheel or Word W 272		NC	NC
M2497	Activate the edge evaluation for PLC inputs		PLC	PLC
M2498	Release jog positioning		PLC	PLC
M2512	Start jog positioning X+	2A	PLC	PLC
M2528	Complement jog positioning X+			
M2513	Start jog positioning X–	2B		
M2529	Complement jog positioning X–			
M2514	Start jog positioning Y+	2C		
M2530	Complement jog positioning Y+			
M2515	Start jog positioning Y–	2D		
M2531	Complement jog positioning Y–			
M2516	Start jog positioning Z+	2E		
M2533	Complement jog positioning Z+			
M2517	Start jog positioning Z–	2F		
M2534	Complement jog positioning Z–			
M2518	Start jog positioning 4th axis +	2G		
M2535	Complement jog positioning 4th axis +			
M2519	Start jog positioning 4–	2H		
M2536	Complement jog positioning 4–			

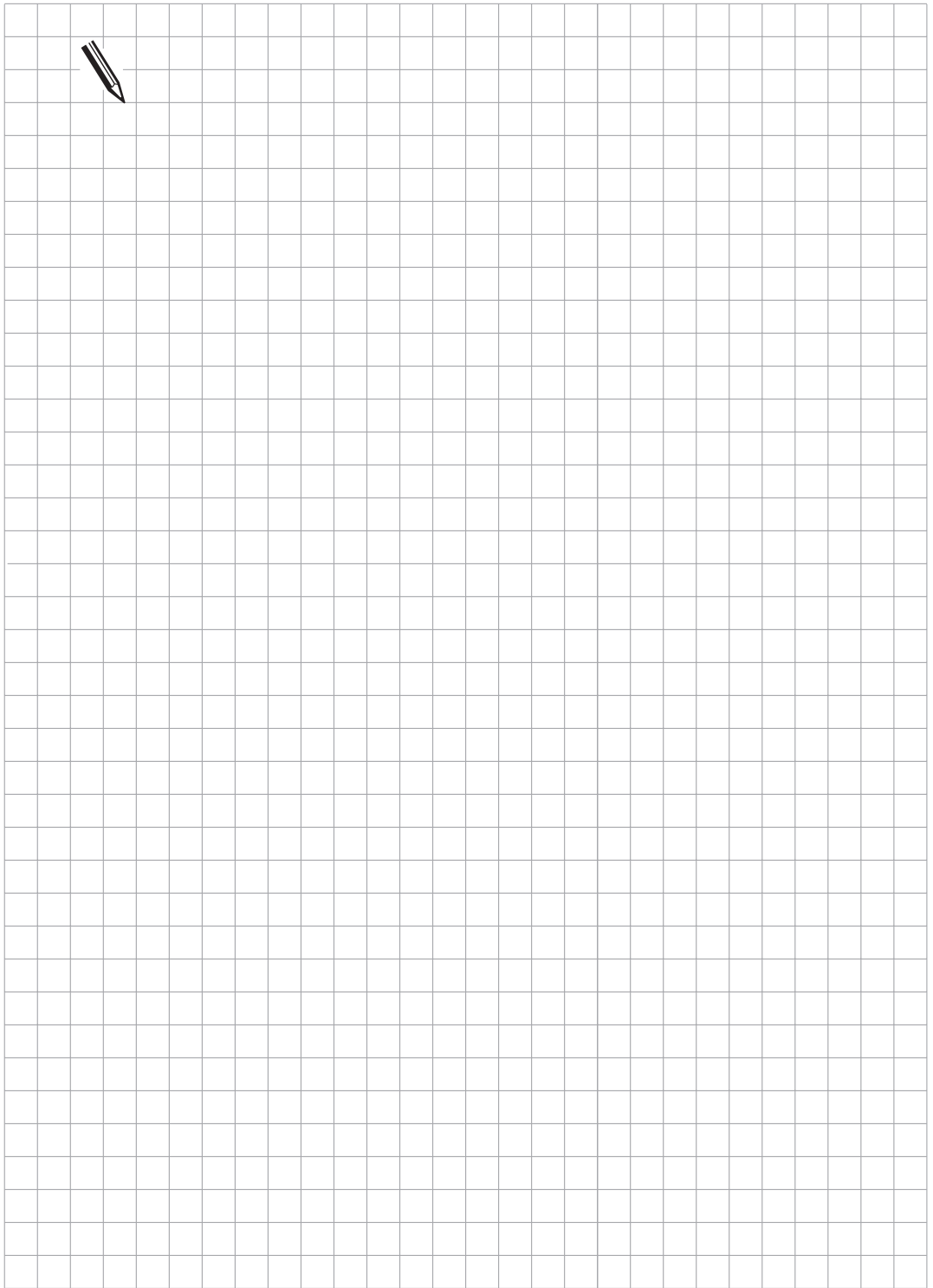
PLC example:

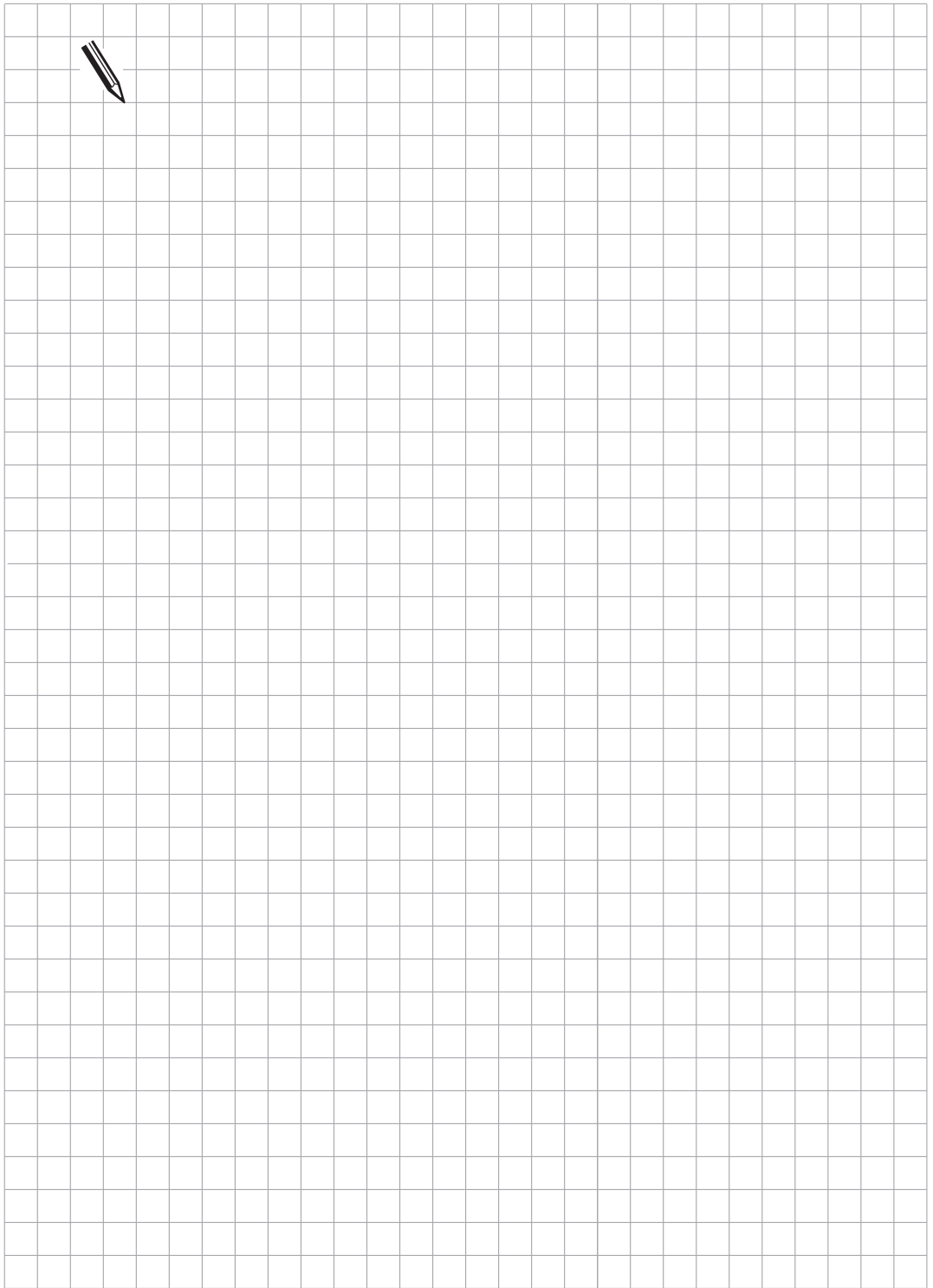
Axis direction button X+ with contact I138

Axis direction button X– with contact I133

L	M2052	Interrogate operating mode
CMT	10	Electronic handwheel
.		
.		
EM		End of main program
.		
.		
LBL	10	
L	M2052	Incremental jog positioning
S	M2497	Activate edge evaluation of the PLC inputs
S	M2498	Release incremental jog positioning
R	M2512	Erase marker for incremental jog positioning in axis X
R	M2513	
S	M2528	
S	M2529	
L	M1638	Incremental jog positioning in axis X+
AN	I133	
S	M2512	
R	M2528	
L	M1633	Incremental jog positioning in axis X–
AN	I138	
S	M2513	
R	M2529	
EM		
.		
.		
.		







13 Hirth Coupling

Many machine tool use Hirth couplings to clamp rotary axes and swivel heads. A Hirth coupling is a pair of more or less finely toothed plates that form a rigid rotary connection when pressed together.

The Hirth functions are realized through the PLC program.

A repositioning is performed as PLC positioning or an error message is displayed, depending on whether the axis with Hirth coupling is positioned automatically or manually. The default values for grid, positioning accuracy and further parameters for positioning before Hirth clamping are performed through free machine parameters by transferring the input values to the PLC.

13.1 Positioning in the manual operating mode

When axis 4 is selected and positioned with the electronic handwheel, the "Axis in position" marker (M2011) is reset. This can be used for de-clamping the Hirth coupling. The same applies for traverse via axis direction button. When the "Axis is position" marker is set, the nominal position is checked for its agreement to the Hirth coupling grid. The PLC takes the shortest path to reposition the axis, i.e. the axis is moved to the nearest grid point.

13.2 Positioning in controlled operation

When the NC is started, the "Axis in position" marker is also reset. This starts an internal examination of the target position. If the target position is not in the Hirth grid, an error message is released. After the error message is erased and the NC is restarted, the axis is moved from the target position to the nearest Hirth grid point.

13.3 Program example

```
.
.
.
LBL    14          ;HIRTH_AXES
          ;INITIALIZE HIRTH
L      M2180       ;1ST PLC CYCLE AFTER POWER ON
O      M2185       ;1ST PLC CYCLE AFTER PLC_PGM COMPILE
S      M528        ;INIT_ACTIVE
L      M528        ;1ST PLC CYCLE DELAY DUE TO TNC 360
AN     M2180       ;1ST PLC CYCLE AFTER POWER ON
AN     M2185       ;1ST PLC CYCLE AFTER PLC_PGM COMPILE
JPT    26          ;INIT_HIRTH
          ;---READ COORDINATES ---
PS     K128
PS     K2          ;REFERENCE_VALUE
CM 9040          ;READ COORDINATE VALUES
L      M3171       ;NP_M3171_MODULE_ERR
S      M2955       ;PN_MODULEEERRCYCL
S      M2815       ;PN_M2815_PLC_EMERGENCY_STOP
          ;--- HIRTH MODULES FOR THE AXES---
L      M595        ;G_HIRTH_ACTIVE_4
```

```

CMT 47 ;HIRTH_4_AXIS
;---CLEAR ERROR MESSAGE---
LN M527 ;HIRTH_ERR_4
AN M527 ;HIRTH_ERR_4
R M2933 ;PN_ERR_HIRTH_NOTINPOS
EM ; END OF HIRTH MODULE

LBL 47 ;HIRTH_4_AXIS
;---CALCULATE LEFT AND RIGHT GRID POINT ---
L D140 ;REFPOS_4TH_AXIS
/ W232 ;GRID_4
PSW
X W232 ;GRID_4
= D152 ;RIGHT_GRID
PLW
+ K1 ;ONE
X W232 ;GRID_4
= D156 ;RIGHT_GRID
;---CALCULATE DIST (DIFFERENCE FROM LEFT AND RIGHT GRID) ---
L D140 ;REFPOS_4_AXIS
- D152 ;RIGHT_GRID
PSW
CM 29 ;GENERATE ABSOLUTE VALUE
PL D160 ;DIF_LEFT
L D140 ;REFPOS_4_AXIS
- D156 ;RIGHT_GRID
PSW
CM 29 ; GENERATE ABSOLUTE VALUE
PL D164 ;DIF_RIGHT
;---COMPARE DISTANCE LEFT > < RIGHT ? ---
L D164 ;LOAD DIF.-RIGHT
< D160 ;LOAD DIF.-LEFT
CMT 49 ;RIGHT GRID IS NEARER
CMF 50 ;RIGHT GRID IS NEARER
;---HIRTH GRID CONTACTED---
L M2 ;MP_EDITOR_ACTIVE
A M2011 ;NP_M2011_4_INPOS
= M597 ;G_HIRTH_4A_INPOS
;---HIRTH-ERROR 4-AXIS---
LN M600 ;MANAL MODE OF OPERATION
AN M2 ;MP_EDITOR_ACTIVE
A M2011 ;NP_M2011_4_INPOS
S M527 ;HIRTH_ERR_4
;---ERASE ERROR WITH NC START---
L I132 ;I_NC_START
R M527 ;HIRTH_ERR_4
;---DISPLAY ERROR---
L M527 ;HIRTH_ERR_4
S M2933 ;PN_ERR_HIRTH_NOTINPOS
;---RELEASE FOR POSITIONING TO THE GRID---
LN M2011 ;NP_M2011_4_INPOS
S M530 ;POSRELEASE_4A
;---MOVE HIRTH-ACHSE TO THE GRID---
L M600 ;MANUAL OPERATING MODE

```

```

AN      M2          ;MP_EDITOR_ACTIVE
A       M2011       ;NP_M2011_4_INPOS
AN      M2707       ;PN_M2707_START_4_POS
AN      M527        ;HIRTH_ERR_4
A       M530        ;POSRELEASE_4A
CMT     51          ;START_4_PLC_POS
;---INHIBIT FOR POSITIONING TO THE GRID
L       M597        ;G_HIRTH_4A_INPOS
R       M530        ;POSRELEASE_4A
EM

LBL 49          ;R_4A_GRID
L       D164        ;DIF_RIGHT
<       W236        ;POS_WINDOW_4
=       M2          ;MP_EDITOR_ACTIVE
L       D156        ;RIGHT_GRID
=       D148        ;PLC_POS
EM

LBL 50          ;L_4A_GRID
L       D160        ;DIF_LEFT
<       W236        ;POS_WINDOW_4
=       M2          ;MP_EDITOR_ACTIVE
L       D152        ;RIGHT_GRID
=       D148        ;PLC_POS
EM

LBL 51          ;START_4_PLC_POS
L       D148        ;PLC_POS
=       D540        ;PN_D540_4_PLC_POS
L       W966        ;NP_W966_FEED_4_ACHSE
=       W566        ;PN_W566_POSFEED_4
LN      M2707       ;PN_M2707_START_4_POS
S       M2707       ;PN_M2707_START_4_POS
EM

LBL 26          ;INIT_HIRTH
;---READ MP4230 AXIS ACTIVE---
PS      K4230       ;MP420
PS      K3          ;MP_EDITOR
CM      9032
L       M3171       ;NP_M3171_MODULE_ERR
S       M2956       ;PN_MODULEEERRSUBM
S       M2815       ;PN_M2815_PLC_EMERGENCY_STOP
PLW
<>     K0          ;ZERO
=       M595        ;G_HIRTH_ACTIVE_4
;---READ MP4239 GRID---
PS      K4239       ;MP430
PS      K3          ;MP_EDITOR
CM      9032
L       M3171       ;NP_M3171_MODULE_ERR
S       M2956       ;PN_MODULEEERRSUBM
S       M2815       ;PN_M2815_PLC_NOT_AUS

```

```

PL      W232          ;GRID_4
;---READ MP1030.3 INPOS---
PS      K1030         ;MP1030
PS      K3            ;MP_EDITOR
CM      9032
L        M3171         ;NP_M3171_MODULE_ERR
S        M2956         ;PN_MODULEERRSUBM
S        M2815         ;PN_M2815_PLC_EMERGENCY_STOP
PL      W236          ;POS_WINDOW_4
L        M528          ;INIT_ACTIVE
R        M528          ;INIT_ACTIVE
EM

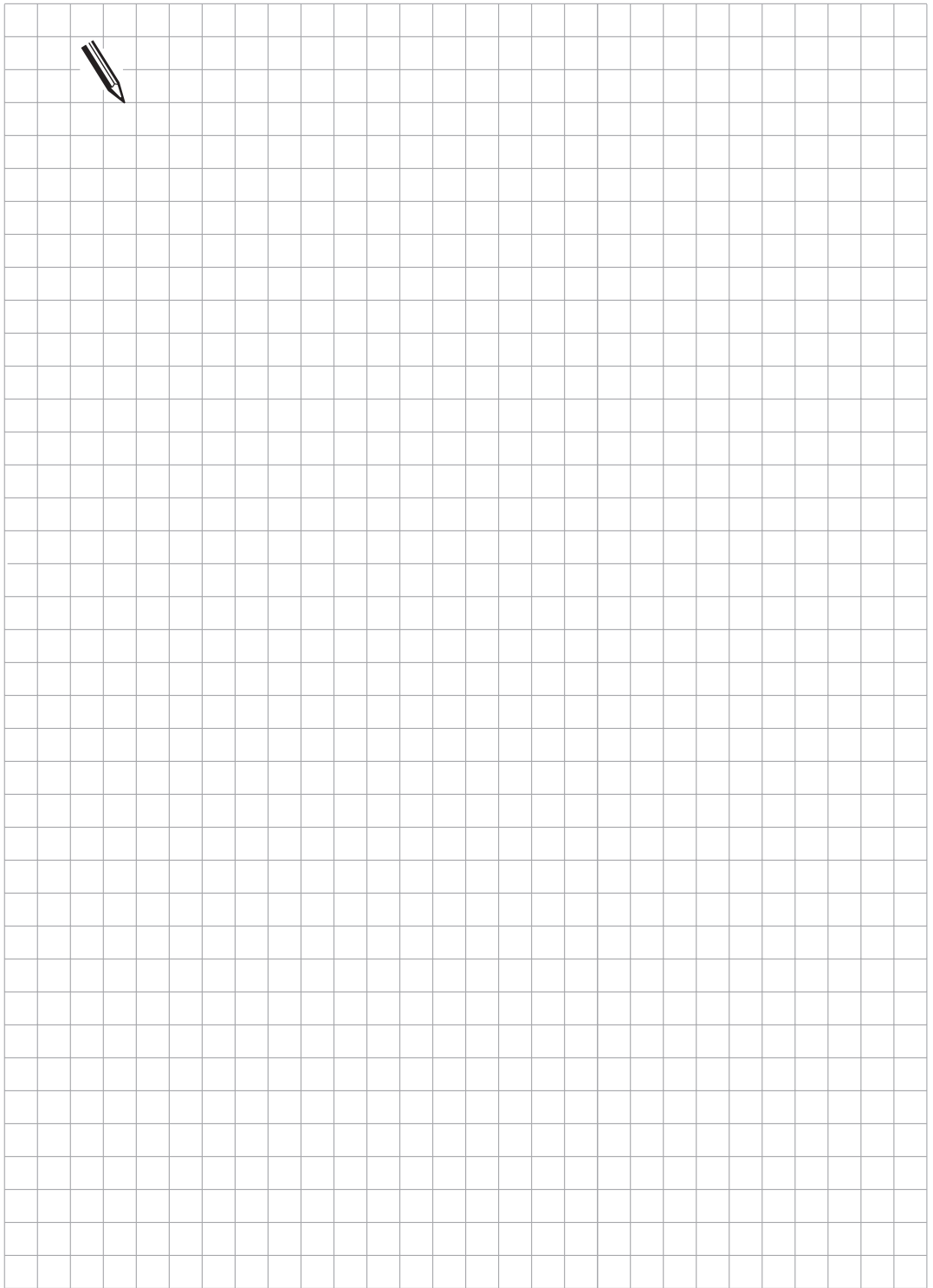
```

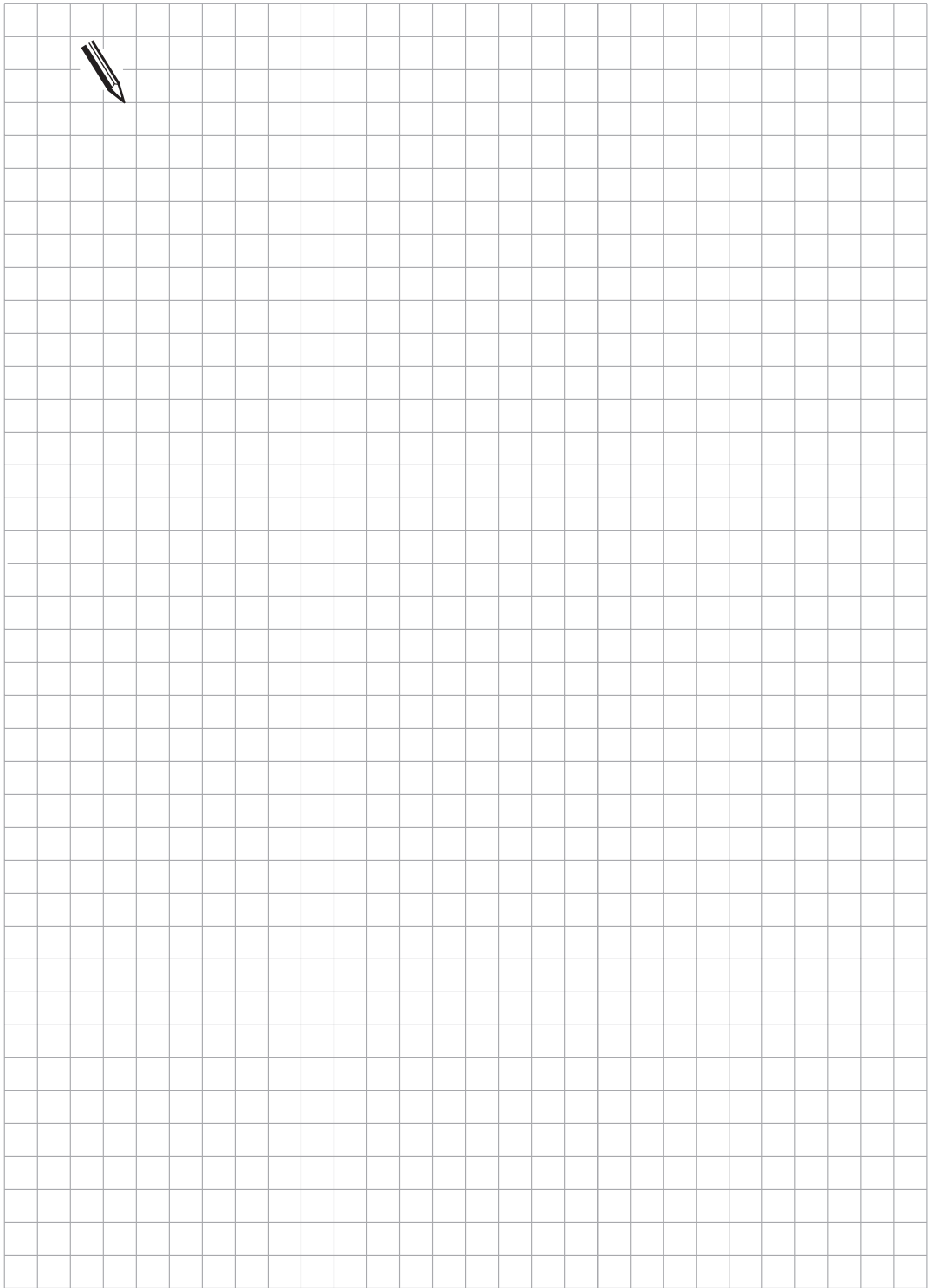
; ABSOLUTE VALUE GENERATION

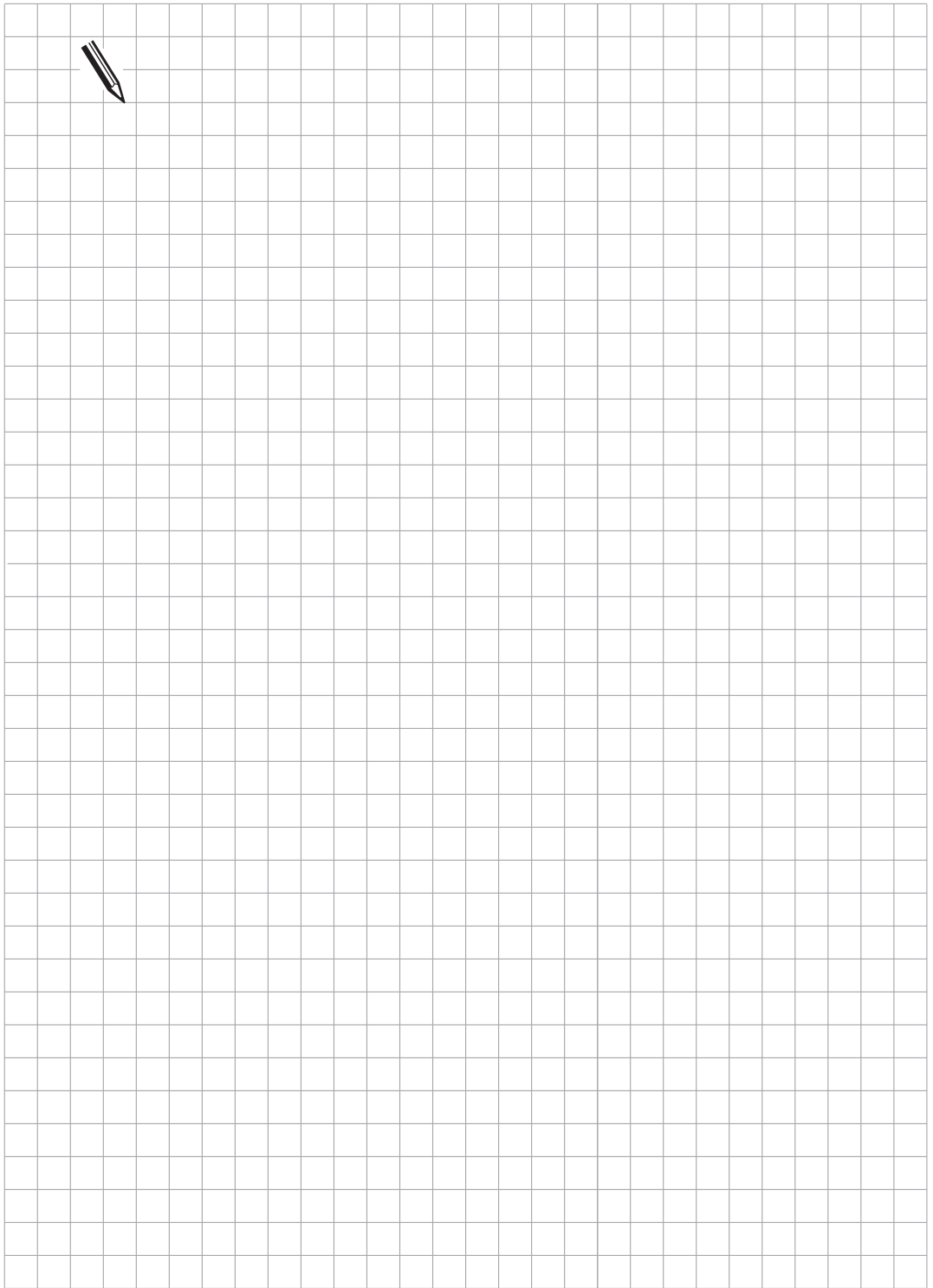
```

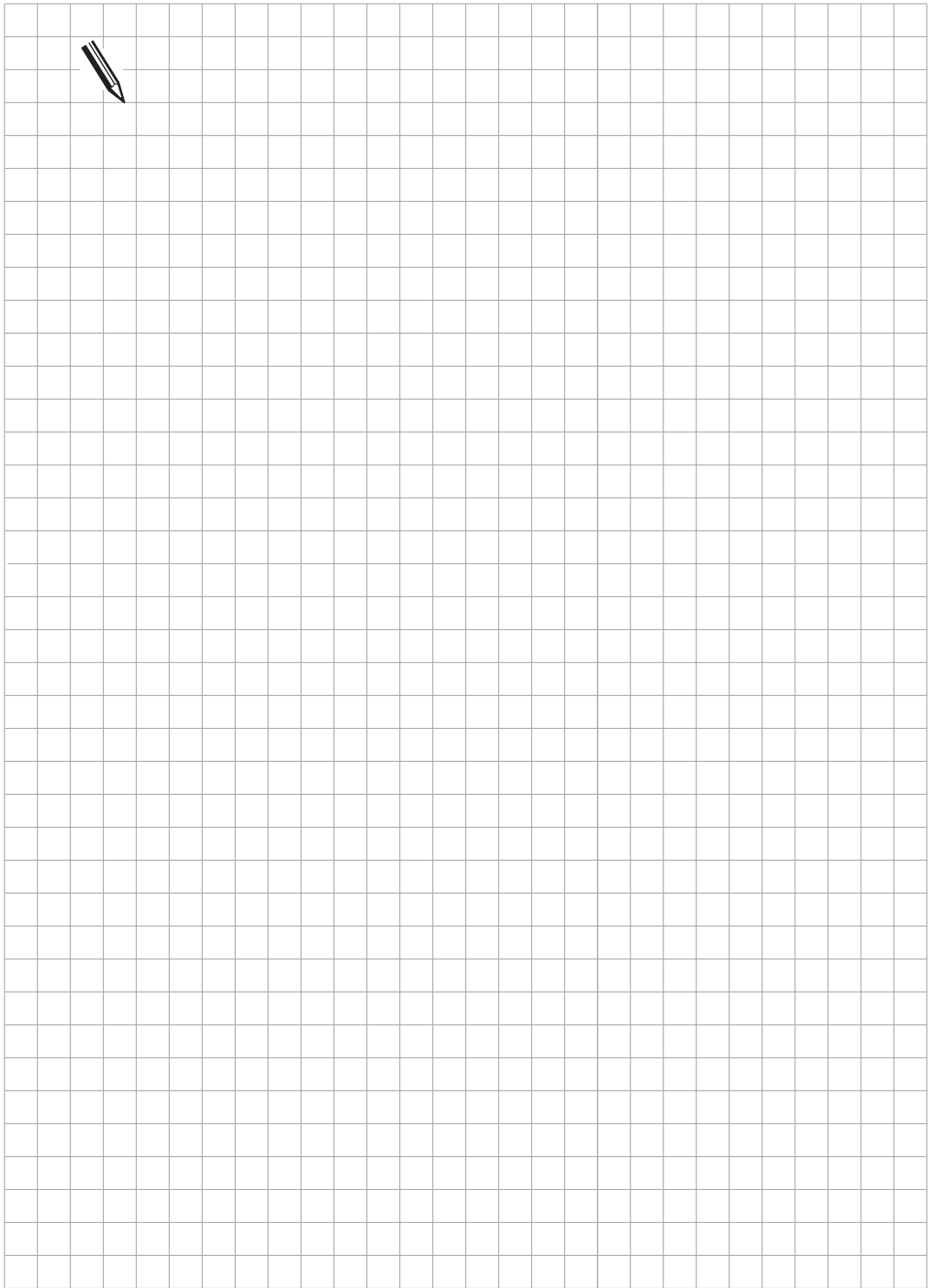
LBL    29          ;ABSOLUTE VALUE GENERATION
PLW                      ;READ DATA FROM D-STACK
PSW                      ;AND WRITE AGAIN
>=      K0             ;IF POSITIVE THEN END OF MODULE
JPT      30            ;LABEL END OF MODULE
PLW                      ;IF NEGATIVE THEN 2ND COMPLEMENT
XON      K0            ;ZERO
+        K1            ;ONE
PSW                      ;STORE DATA ON STACK
EM

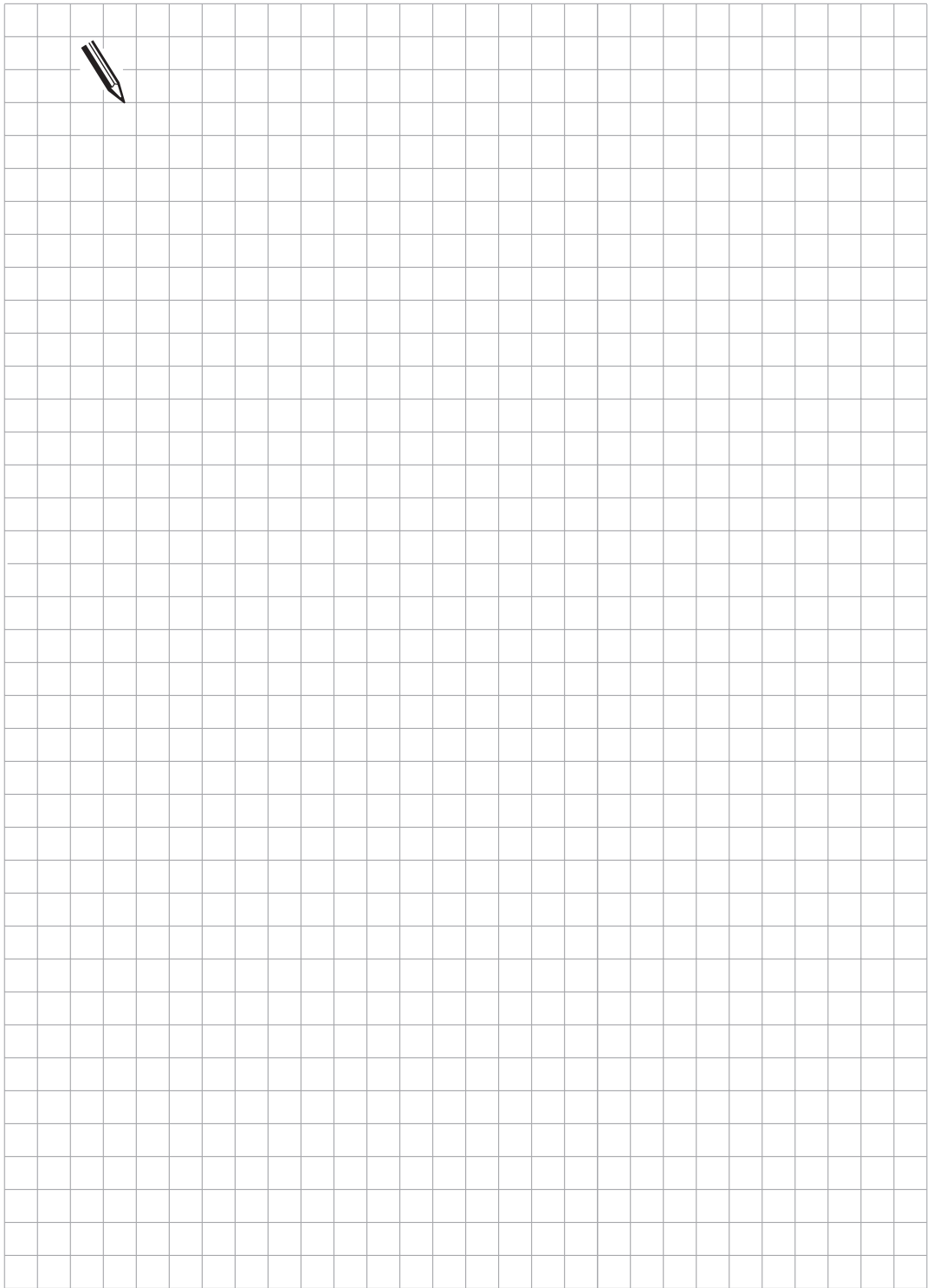
```

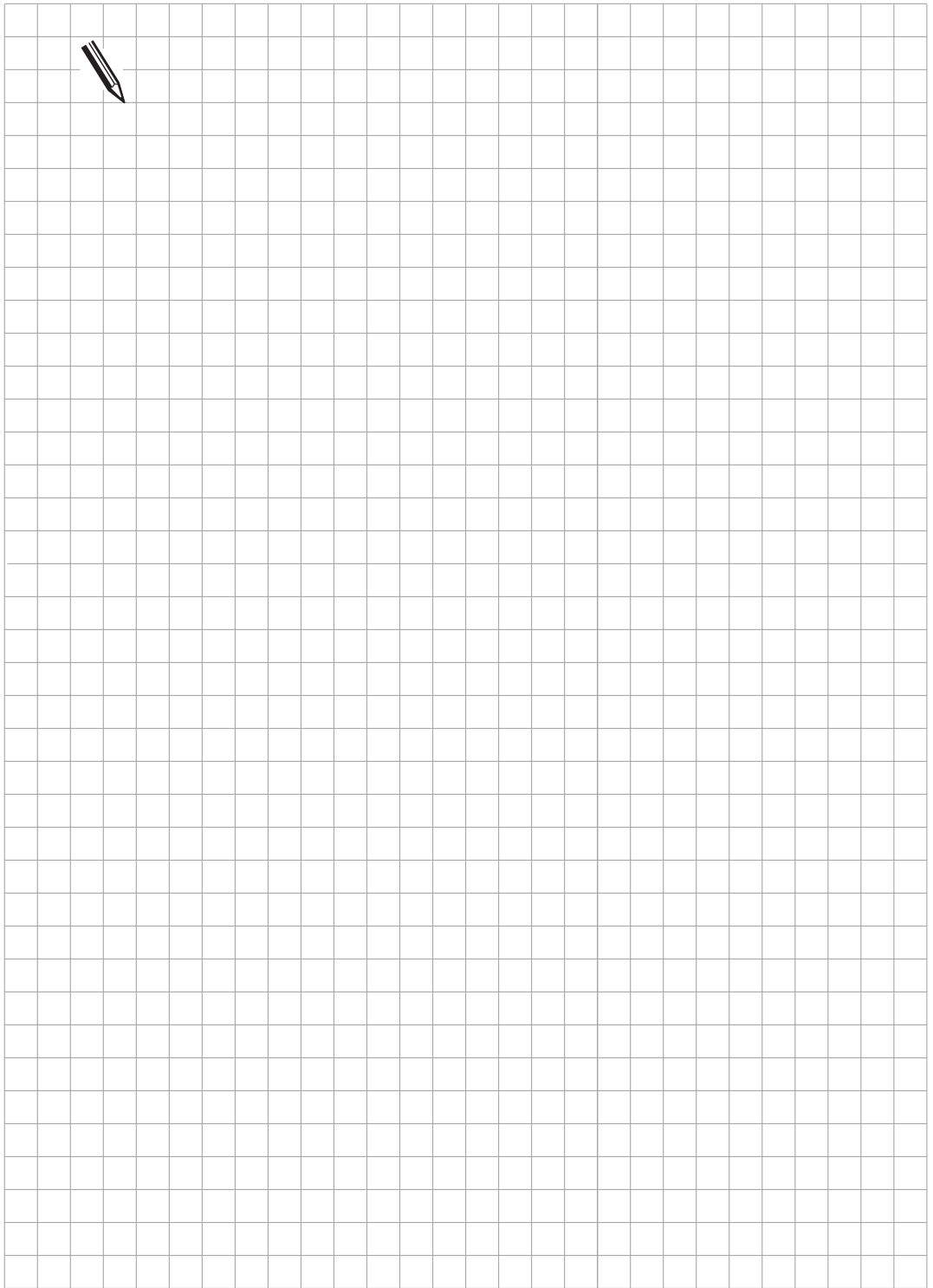


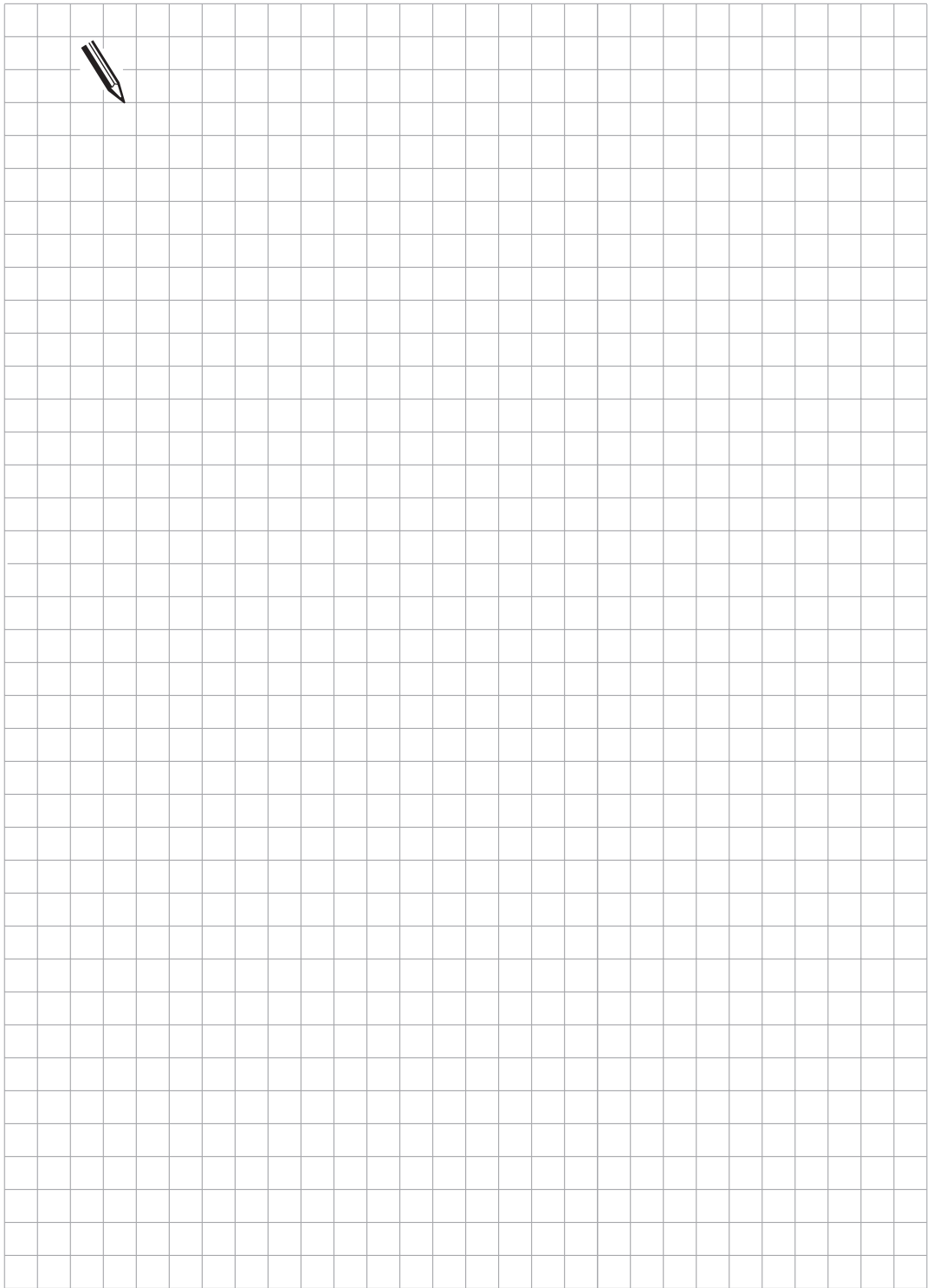












14 Datum Correction

The PLC datum correction function is used to shift the zero or datum point with the PLC program. Datum corrections are required for machines that have swivel heads, e.g. when swivelling the head causes a correction in the X/Z axis.

Each axis (X, Y, Z, 4) is assigned a doubleword (D528 to D540) for the correction value. If a value for the appropriate axis is entered via the PLC program and strobe marker M2716 is set, then a datum correction will be active at the next M/S/T strobe. The correction is computed in the actual position display. The display then indicates the shifted system of coordinates.

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 doublewords, D528 to D540, 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 D540.

If several datum shifts are executed in one axis, the correction values refer to the uncorrected reference system. In datum compensation with the value 0, previous (axis specific) corrections are canceled.

D528	Datum correction for X axis
D532	Datum correction for Y axis
D536	Datum correction for Z axis
D540	Datum correction for 4th axis

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       M2025      ;M21 activated
A       M10        ;Buffered marker for strobe marker 2716
CMT201      ;Deactivate datum correction call
R       M10

L       M1920
O       M1921
A       M2045
AN      M2716
S       M2482      ;Acknowledgment

EM

LBL 200      ;Activate module for datum correction call
L       D896      ;Value from MP4210.32
=       D528      ;Shift X axis
L       K0
=       D532      ;Do not shift Y axis
=       D536      ;Do not shift Z axis
=       D540      ;Do not shift 4th axis
L       M10
S       M2716
EM

LBL 201      ;Cancel module for datum compensation
L       D900      ;Value from MP4210.33
=       D528
L       K0
=       D532      ;Do not shift Y axis
=       D536      ;Do not shift Z axis
=       D540      ;Do not shift 4th axis
L       M10
S       M2716
EM
.
.

```

15 Tool Changer

A tool changer can be operated with the PLC of the HEIDENHAIN contouring control. Tool changers are often controlled by proximity switches.

Tool data are stored in the tool table and tool management (special tool, tool pocket management 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

The operator can edit the tool table (program 0) in the "Programming and editing" mode (see User's Manual). The tool table can be uploaded and downloaded through the data interface. The tool number (T), length (L), radius (R), pocket number (P), and special tool (ST) are all defined in the tool table. Special tools and pocket number are only displayed if MP7266 and MP7480 have been set accordingly. The pocket numbers can be protected from editing or erasure by setting marker M2594.

BEGIN	TOOL	MM	
T0	P0	L+0	R+0
T1	P1	L+10	R+20
T2	P2	L+50	R+30
T3		L+0	R+0
T4	SP4	L+100	R+300
T5		L+0	R+0
T6	P6	L+50	R+15
.	.	.	.
.	.	.	.
.	.	.	.
END	TOOL	MM	

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 (program "0" does not exist). Tool length and tool radius are programmed in the NC program with TOOL DEF (see User's Manual).

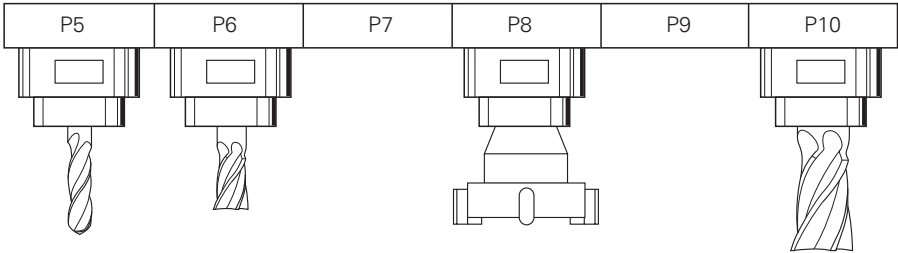
MP7260 Number of tools in the tool table

Input range: 0 to 99
0 = no tool table
1 to 99 = number of tools

		Set	Reset
M2594	Inhibit editing of pocket number in the tool table in the tool table	PLC	PLC

15.1.1 Special tools

Special tools are oversize tools where pockets on either side in the tool magazine have to be left vacant .



The number of pockets to be left vacant on both sides of the tool is defined in machine parameter MP7264. Special tools are identified by an "S" in the tool table. The numbers of the vacant pockets are not displayed. Variable pocket coding can be limited for special tools. If marker M2601 is set, then special tools are returned to their original pocket despite the "variable pocket coding" setting.

MP7264	Number of pockets reserved next to special tool Input range: 0 to 3	Set	Reset
M2601	Special tool to original pocket despite variable pocket coding	PLC	PLC

15.2 Controlling the tool changer

Controlling the tool changer, i.e. positioning the changing arm and carousel and the complete toolchange sequence is performed by the PLC. The NC handles tool pocket assignment and evaluation of the TOOL DEF and TOOL CALL blocks. The NC and PLC communicate by markers and words.

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). 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 tool number and the pocket 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

Input range: 0 to 3

- 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

Reading and writing to the tool file

If the PLC enters the tool number in Word W512, the associated pocket number is stored in W264 through strobe marker M2715. If the PLC enters the tool number in W512 and the pocket number in W514, the strobe marker M2714 of the T code and P code is transferred to the NC.

MP7480.1 Output tool or pocket number with TOOL DEF block

Input range: 0 to 3

- 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

W262	Pocket number when MP7480 = 3	NC	NC
	Tool number when MP7480 = 1 or 2		
W264	Tool number when MP7480 = 3	NC	NC
W512	T code	PLC	PLC
W514	P code	PLC	PLC
		Set	Reset
M2046	Strobe signal T code (P code) with TOOL CALL	NC	NC
M2047	Strobe signal 2nd T code (2nd 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
M2063	PGM 0 is selected	NC	NC
M2597	Binary format for T code and P code in W262 and W264	PLC	PLC

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.

Unlike variable pocket coding, the pocket numbers are displayed in the tool table but are not changed by the NC.

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). 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.

If marker M2612 (suppress pocket exchange in the tool table) is set before a T strobe is output (M2046), the pocket numbers are not exchanged. The PLC only acknowledges the T strobe without having made an exchange in the tool magazine and resets the PLC marker M2612. The new tool number is displayed highlighted (inversely) in the status and the associated tool data (length, radius) are activated. The inverse display of the tool number in the status display remains until a tool has

been physically changed. If the control is switched off and back on again in the above-mentioned condition, the tool last inserted becomes active again.

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 M2401 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 variety of tool types can be called from the machining program. The following definitions are used in the examples that follow:

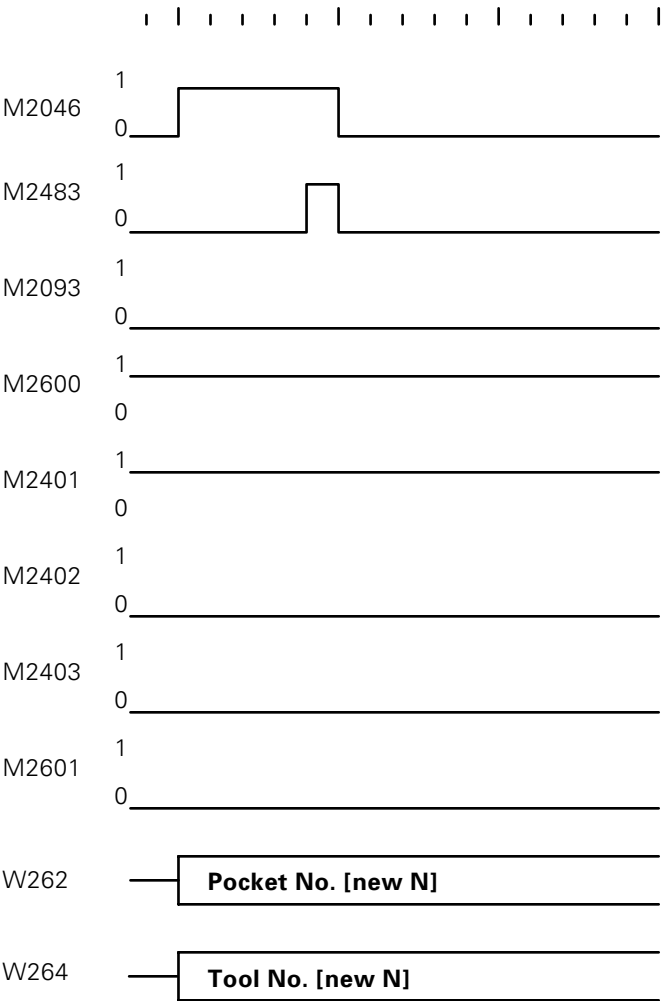
- N** = Tool for which a pocket number is defined in the tool table. (**N**ormal).
- M** = Tool for which no pocket number is defined in the tool table. These tools must be changed by hand. (**M**anual).
- S** = **S**pecial tool (defined in tool table).

Nine different combinations of toolchange sequence are therefore possible. For many toolchange 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 toolchange 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.



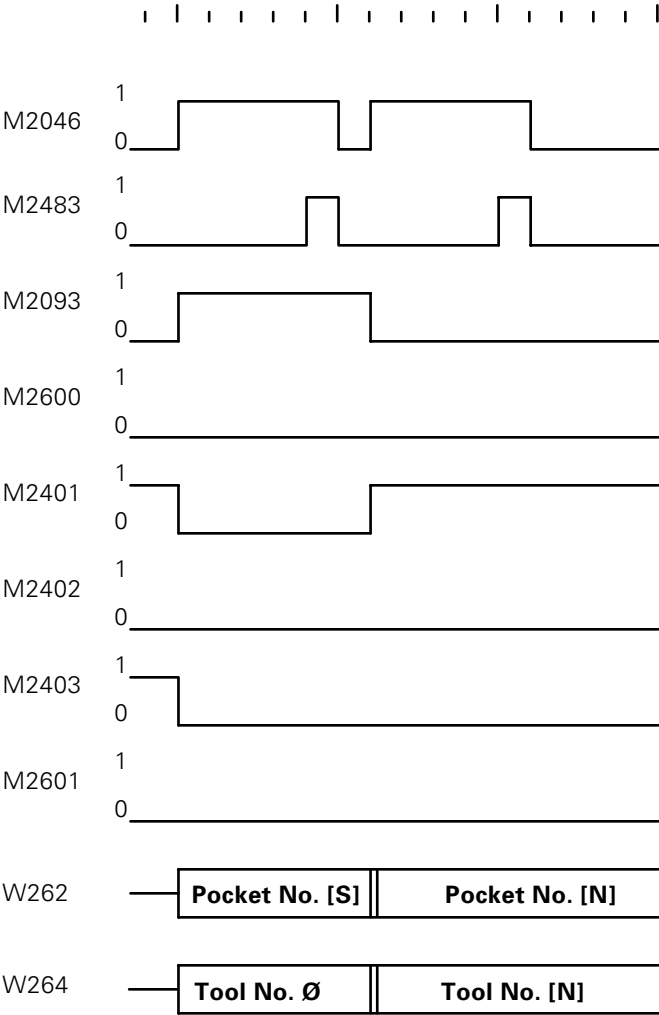
S → N: Normal tool follows Special tool

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.

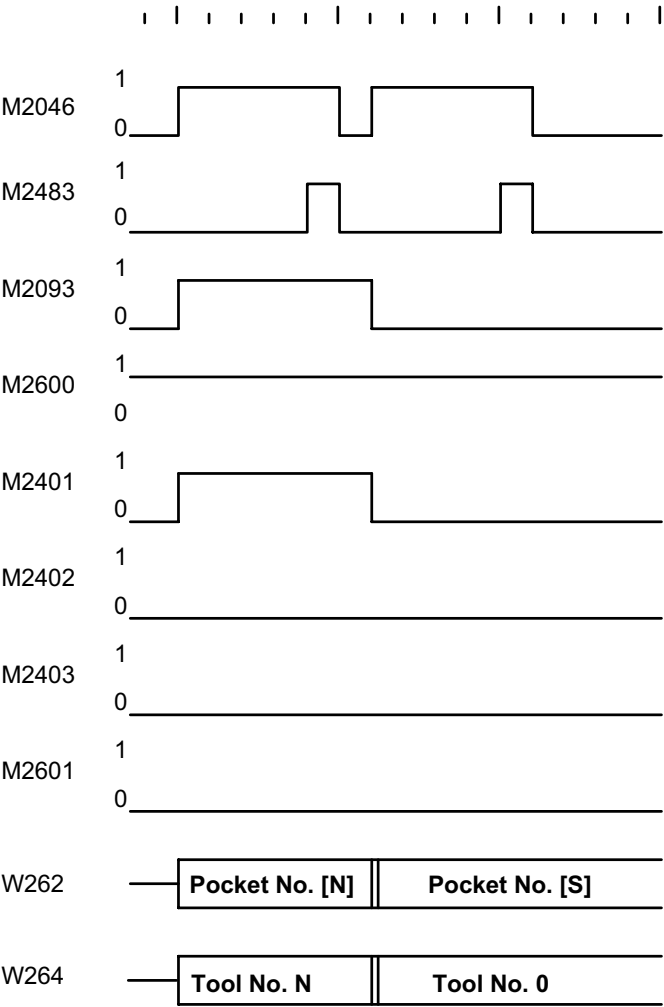
S → N, Single Changing Arm (M2600 = 0):

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):

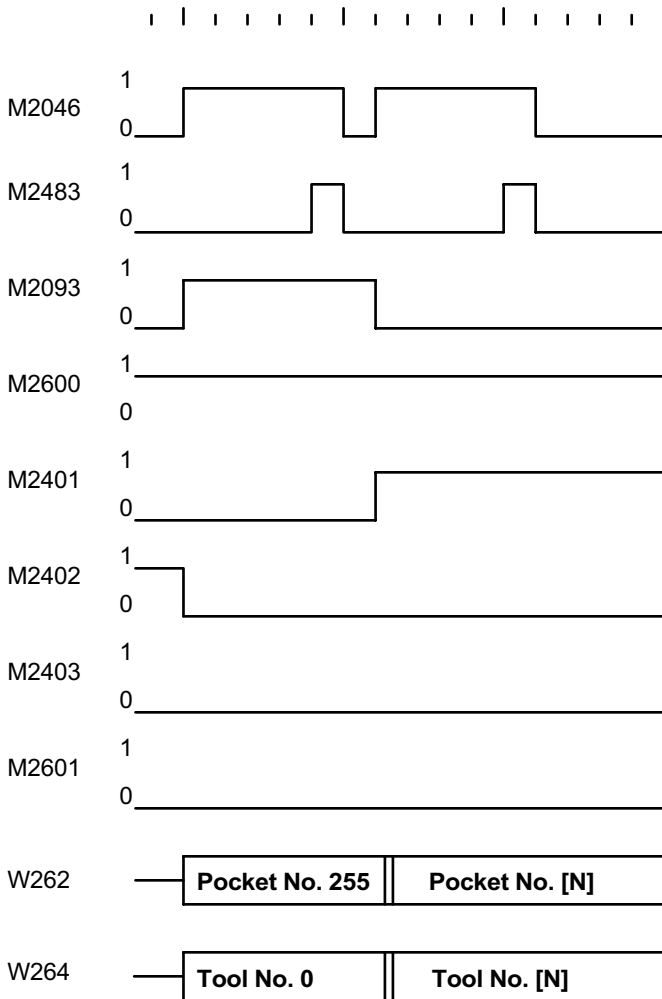
The pocket number and tool number of the new tool are transferred first. After acknowledging with M2483 the pocket number of the old tool and tool number 0 are transferred. Tool number 0 tells the PLC to clear the spindle.



M → N: Normal tool follows Manual tool

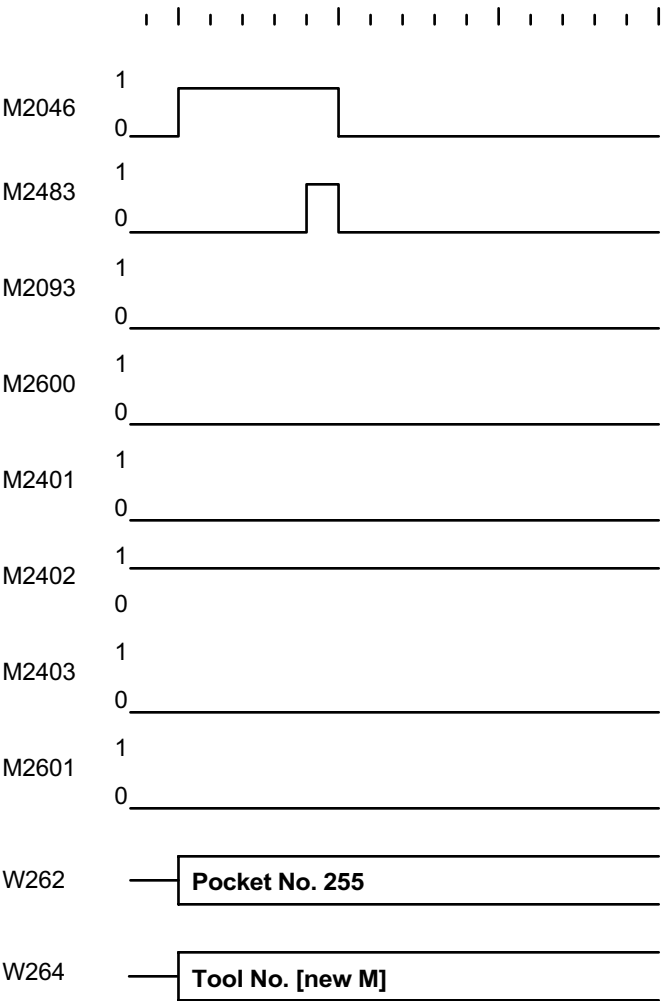
With this toolchange sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

Irrespective of marker 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

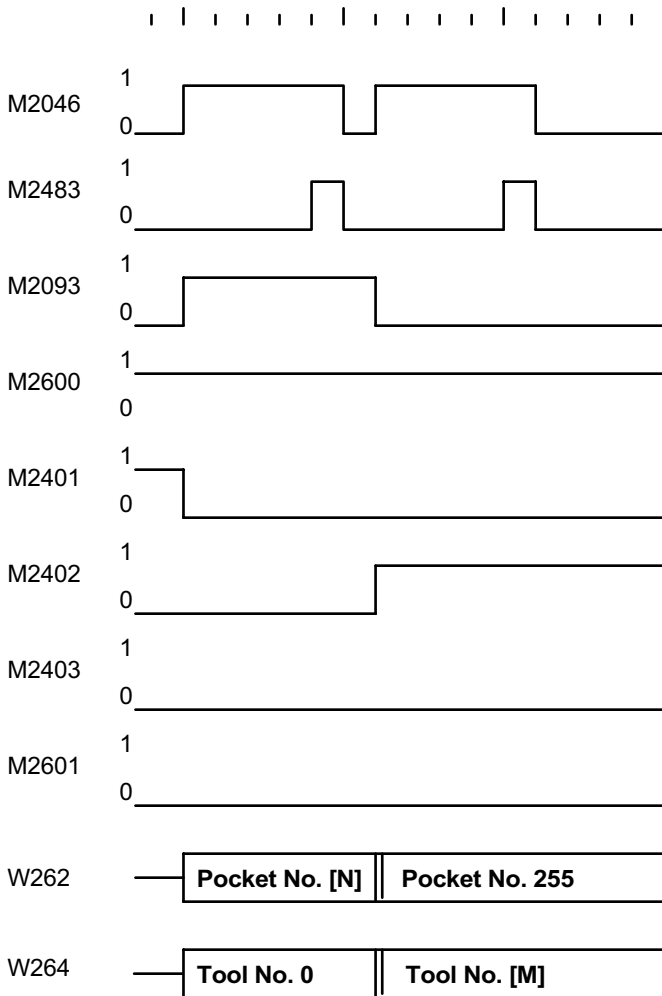
The pocket number 255 tells the PLC that there is no pocket in the tool magazine for the called tool.



N → M: Manual tool follows Normal tool

With this toolchange sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

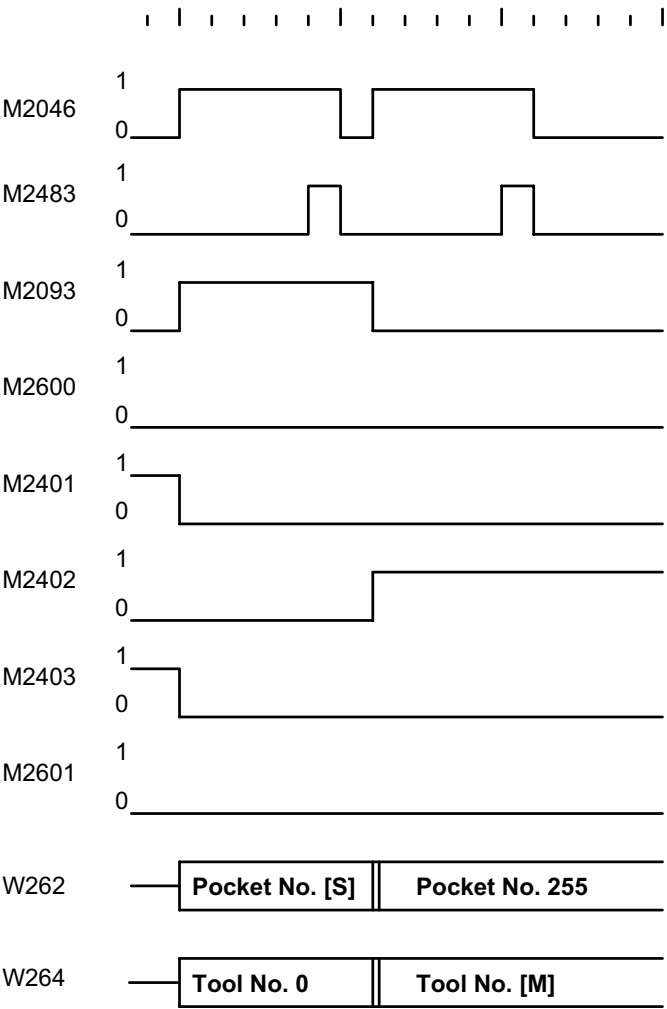
Irrespective of marker M2600, the pocket number of the old tool and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. After acknowledging with M2483, pocket number 255 and the tool number of the called tool are transferred. Pocket number 255 tells the PLC that there is no pocket in the tool magazine for the called tool.



S → M: Manual tool follows Special tool

With this toolchange sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

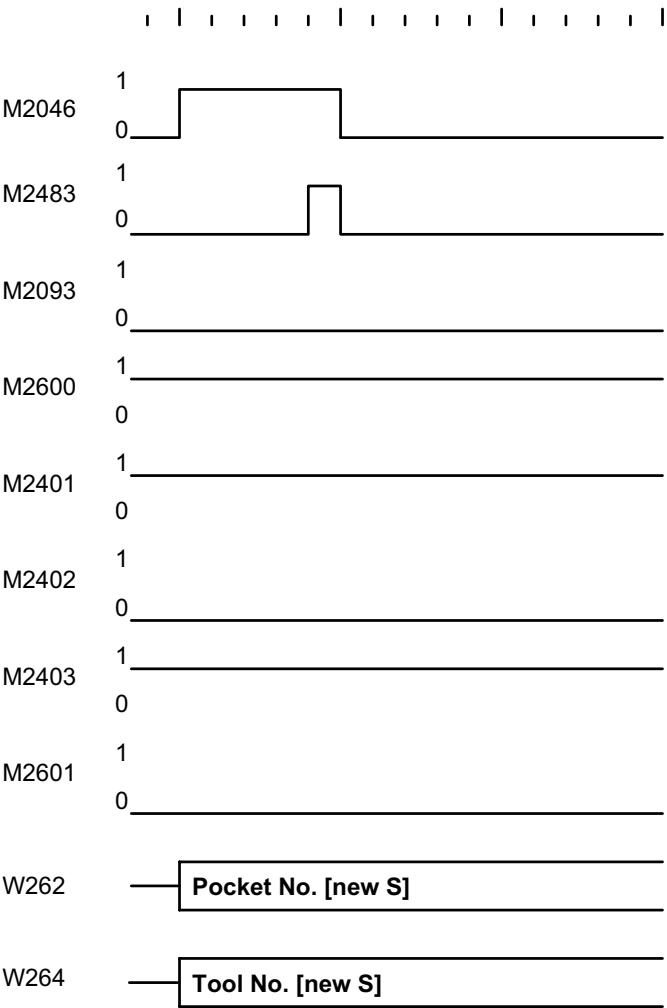
Irrespective of marker M2600, the pocket number of the old tool and tool number 0 are transferred first. Tool number 0 tells the PLC to clear the spindle. After acknowledging with M2483, pocket number 255 and the tool number of the called tool are transferred. Pocket number 255 tells the PLC that there is no pocket in the tool magazine for the called tool.



S → S: Special tool follows Special tool

Marker 2601 can be used to set whether the special tool is to be returned to its original pocket despite variable pocket coding.

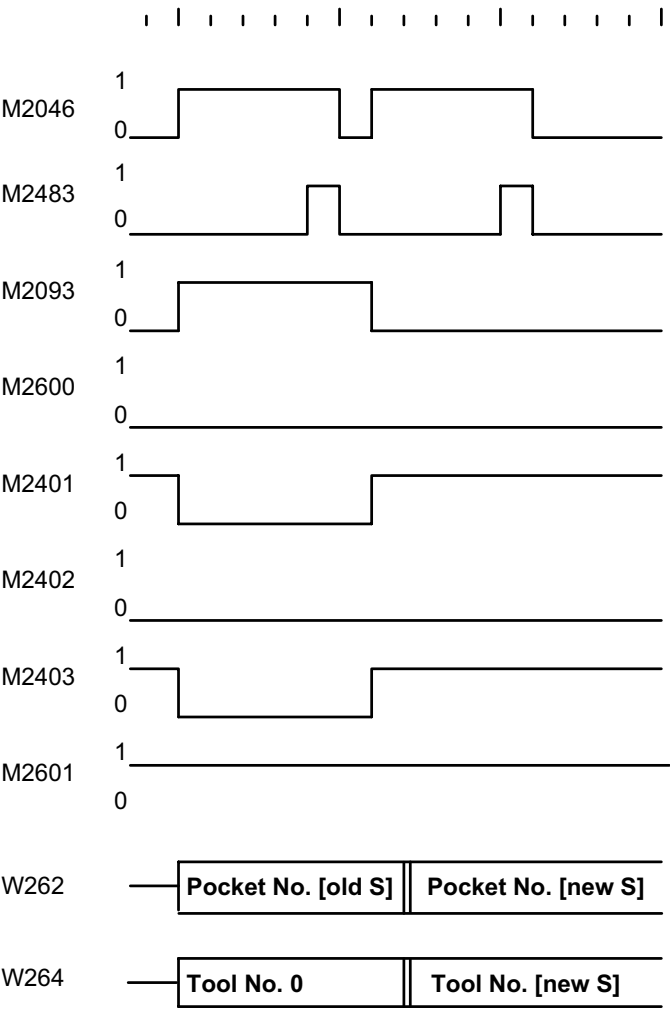
The same logic diagram applies to single and double changing arms (M2600 = 0 and 1) with variable pocket coding for special tools (M2601 = 0).



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 ($M2600$).

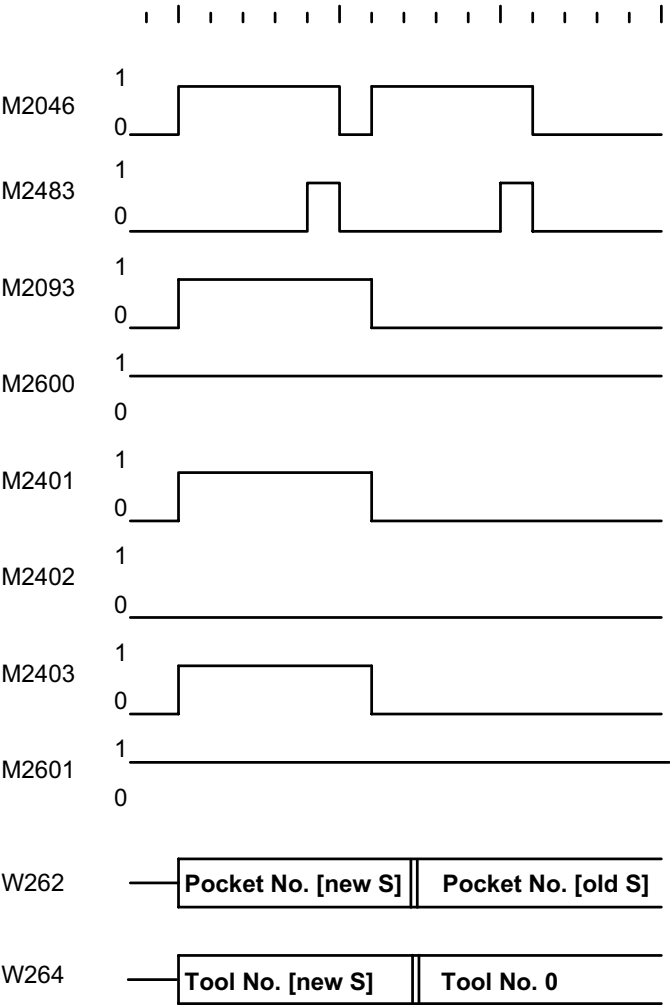
S → S, Single Changing Arm ($M2600 = 0$)

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.



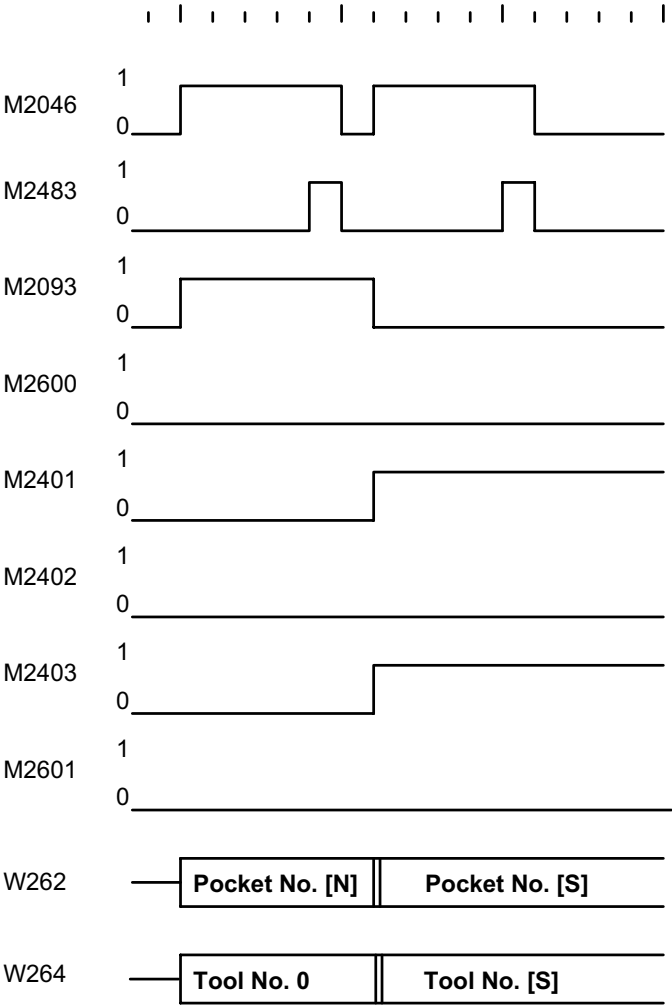
N → S: Special tool follows Normal tool

With this toolchange 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.

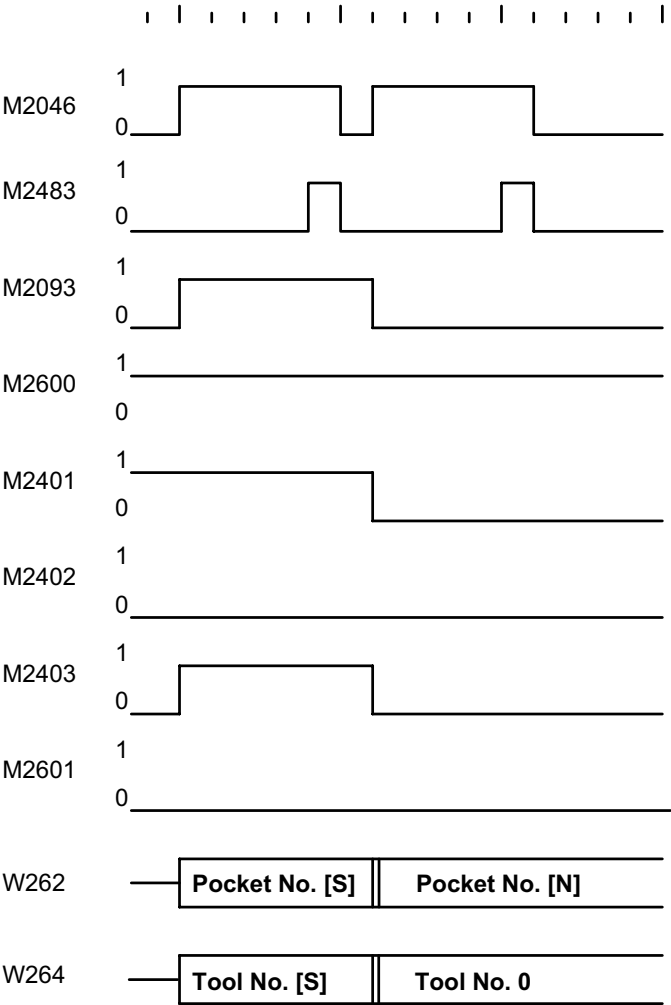
N → S, Single Changing Arm (M2600 = 0)

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)

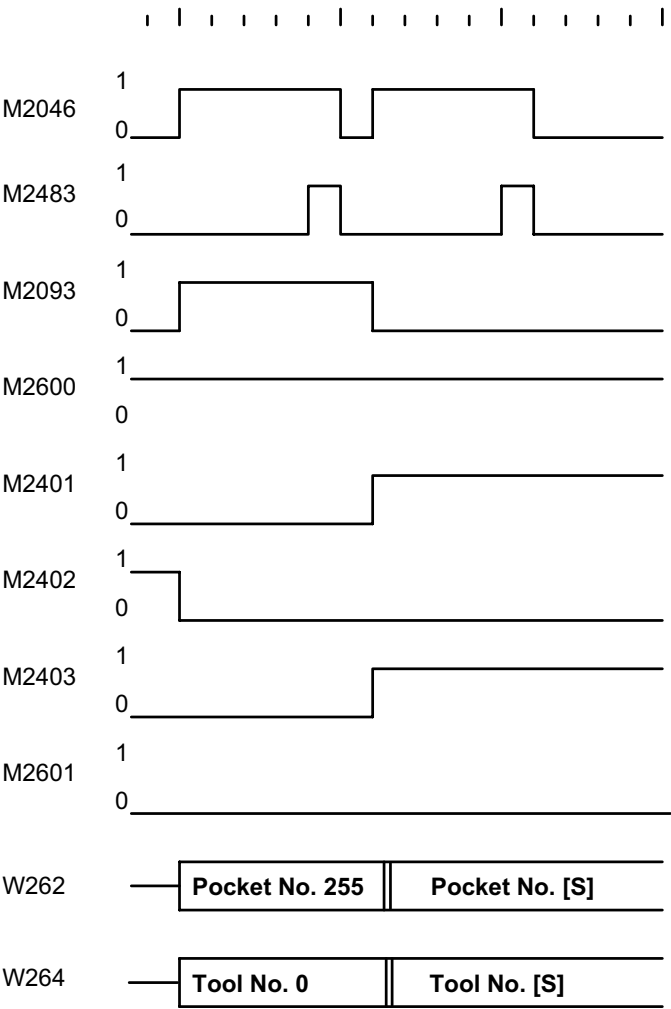
The pocket number and tool number of the new tool are transferred first. After acknowledging with M2483, the pocket number of the old tool and tool number 0 are transferred. Tool number 0 tells the PLC to clear the spindle.



M → S: Special tool follows Manual tool

With this toolchange sequence two pocket numbers (or tool numbers) must be transferred in succession. M2093 indicates that another TOOL CALL strobe (M2046) follows.

Irrespective of marker 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.



MP7261 **Number of pockets in tool magazine** Tool magazine, n
Input range: 0 to 254

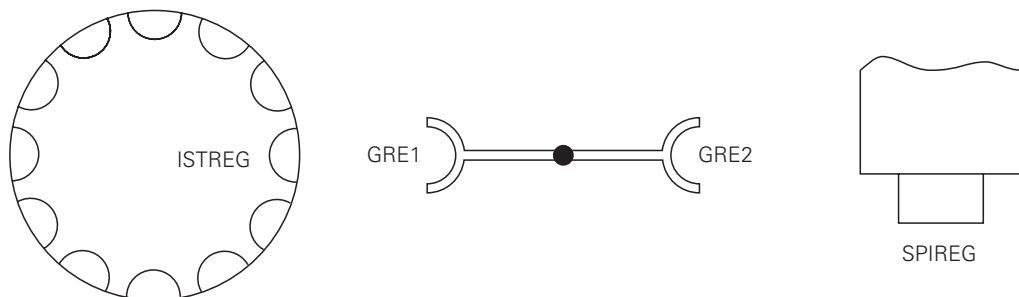
Marker	Function	Set	Reset
M2093	Another T code (P code) follows with TOOL CALL	NC	NC
	0 = Normal tool follows Normal tool (N -> N) or Manual tool follows Manual tool (M -> M) or Special tool follows Special tool (S -> S)		
	1 = Special tool follows Manual tool (M -> S) or Special tool follows Normal tool (N -> 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)		
M2600	Sequence of tool number or pocket number transfer (M2093 = 1) 0 = Number of old tool first, then number of new tool (single changing arm) 1 = Number of new tool first, then number of old tool (double changing arm)	PLC	PLC
M2401	Active tool with pocket number (MP7260/MP7261) (active only when MP7480 = 3 and TOOL CALL)	NC	NC
M2402	Active tool without pocket number (MP7260/MP7261) (active only when MP7480 = 3 and TOOL CALL)	NC	NC
M2403	Active tool = special tool	NC	NC
M2601	Special tool to original pocket despite variable pocket coding	PLC	PLC

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 (set auxiliary markers etc.) the constraints on a PLC program sequence must be remembered .

The example describes a tool changer with the following features:

- Up to 99 tools
- Variable pocket coding (MP7480 = 3)
- 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 toolchange 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 = 3 Output of pocket number and tool number with every TOOL CALL block
 MP7480.1 = 3 Output of pocket number and tool number with every TOOL DEF block

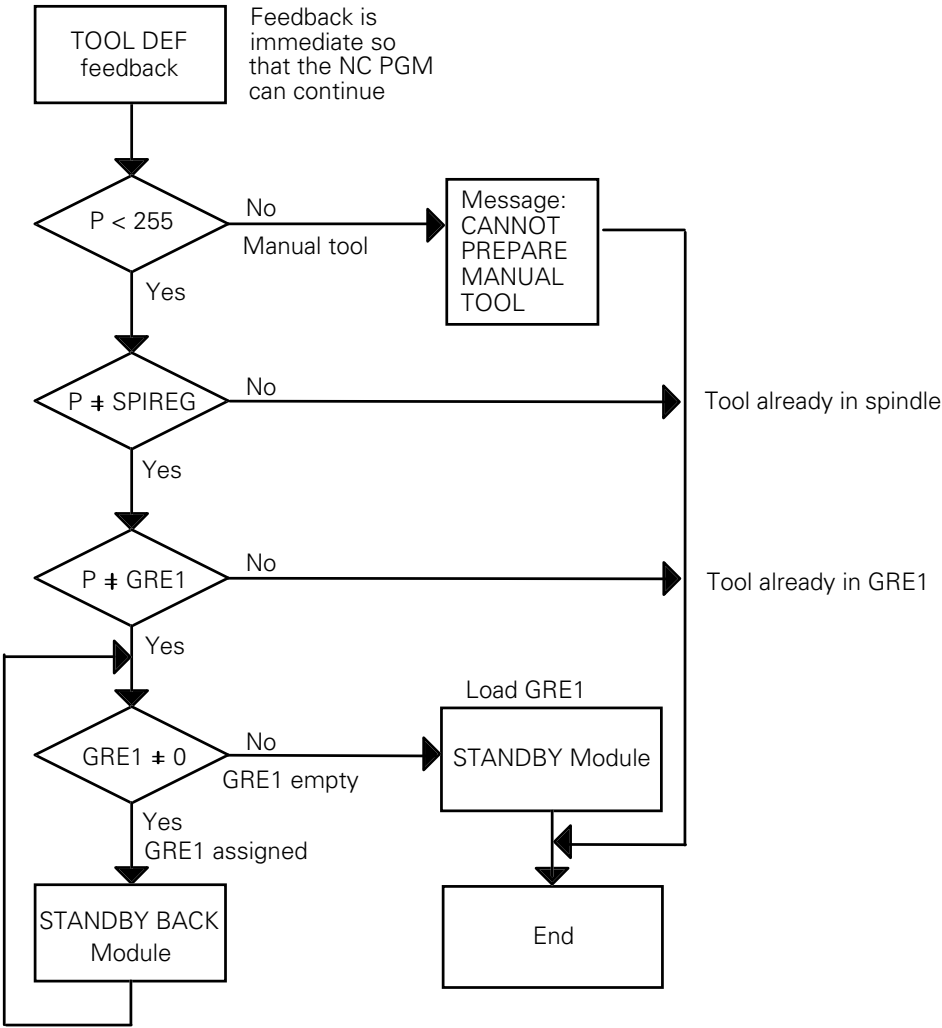
The sequence diagram for this tool changer is subdivided into modules.

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 ROUTE
- COMPARE P-CODE WITH ISTREG
- COMPARE GRE1 WITH ISTREG

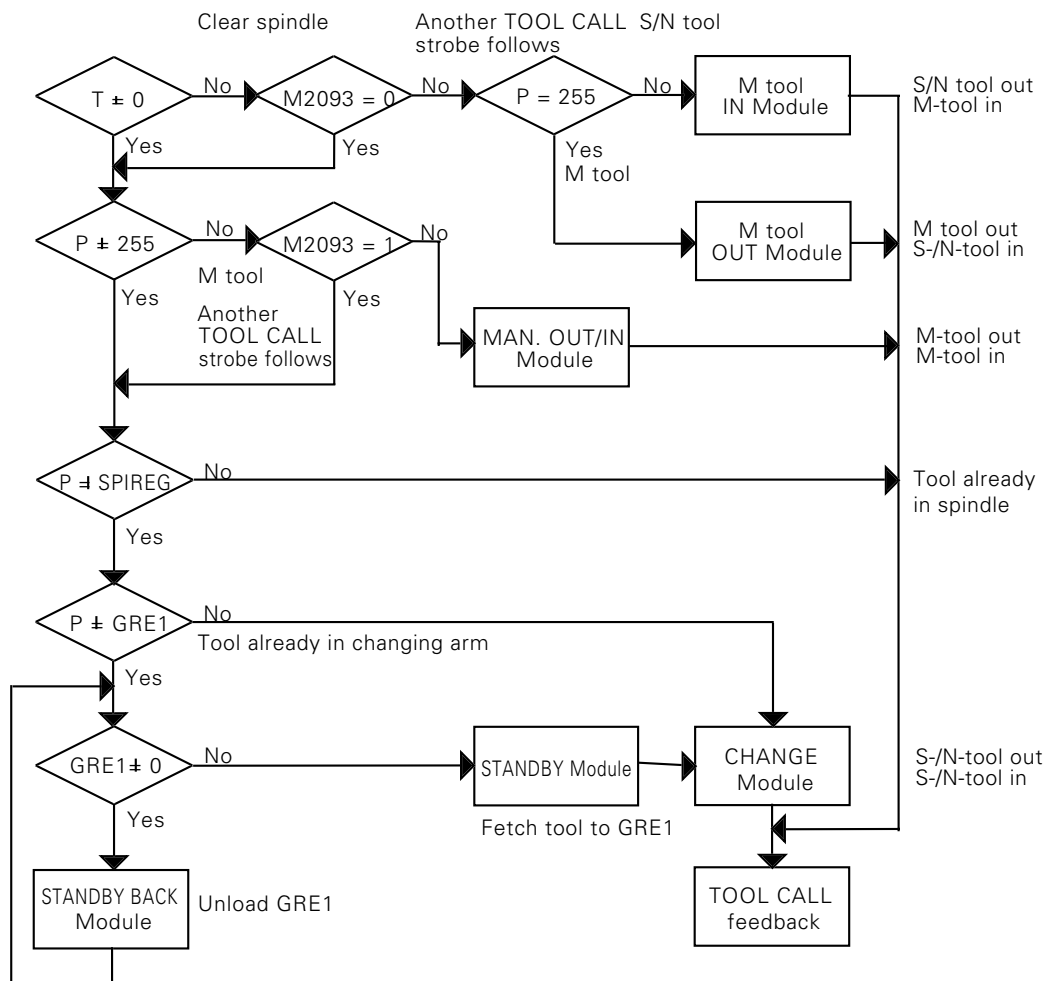
15.3.1 Program module TOOL DEF

Search tool and load in GRE1



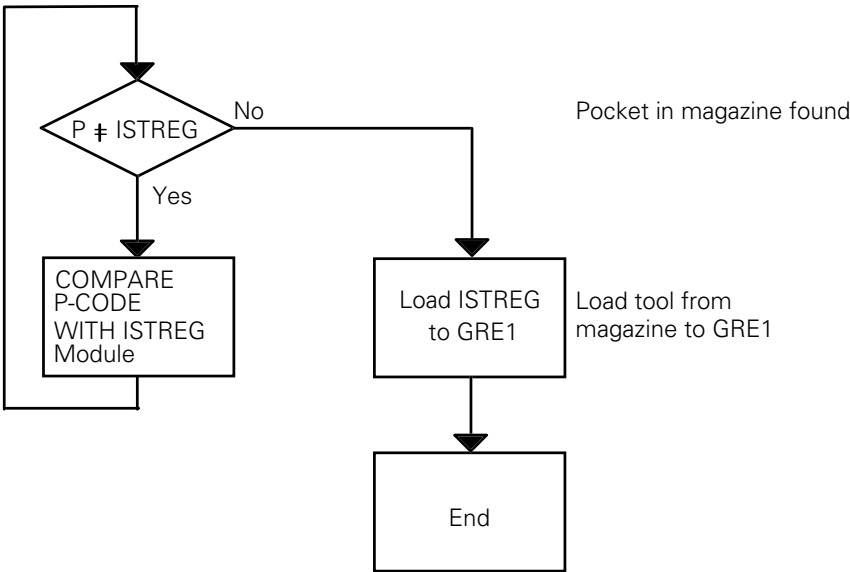
15.3.2 Program module TOOL CALL

Automatic toolchange (main program)



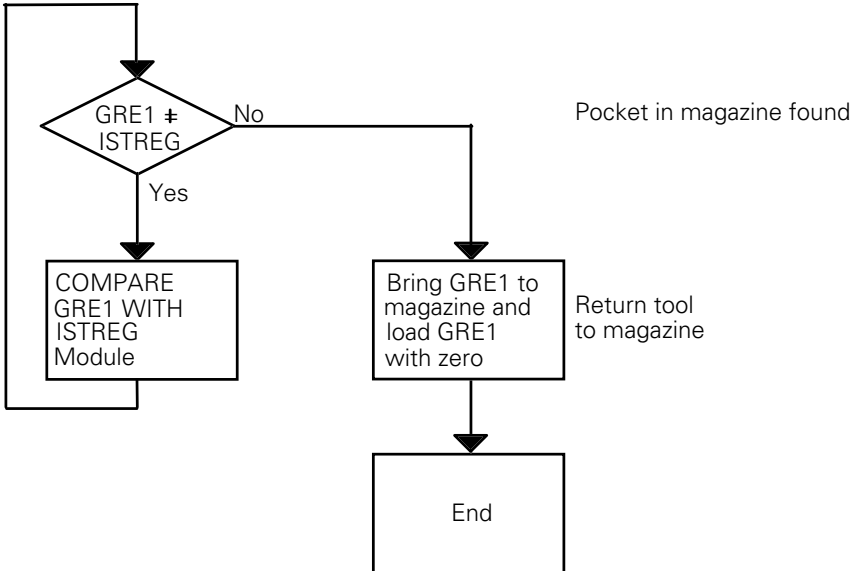
15.3.3 Program module STANDBY

Search tool and load in GRE1



15.3.4 Program module STANDBY BACK

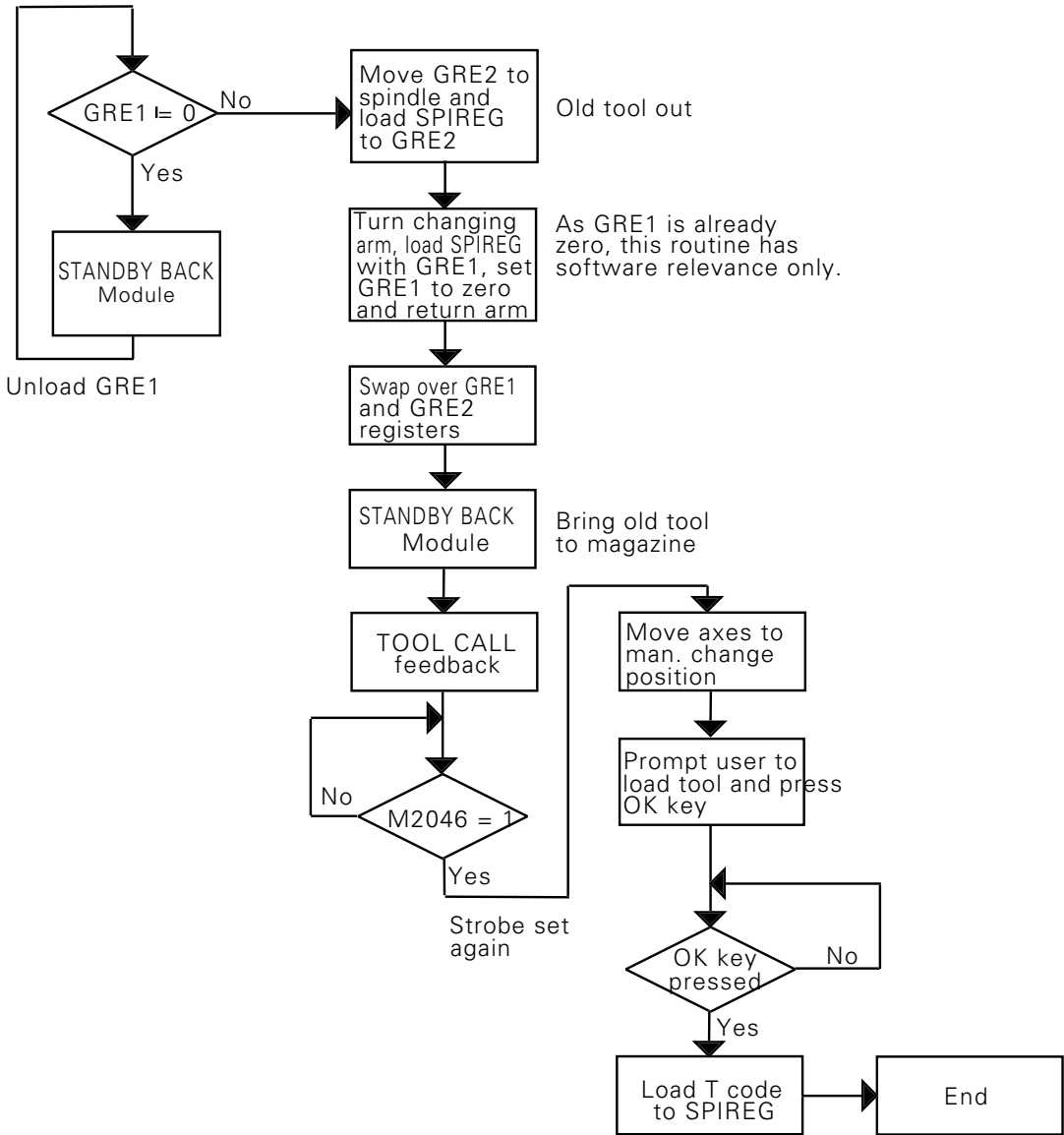
Return tool from GRE1 to magazine



15.3.5 Program module MANUAL TOOL IN

N -> M or S -> M: Manual tool follows Normal or Special tool.

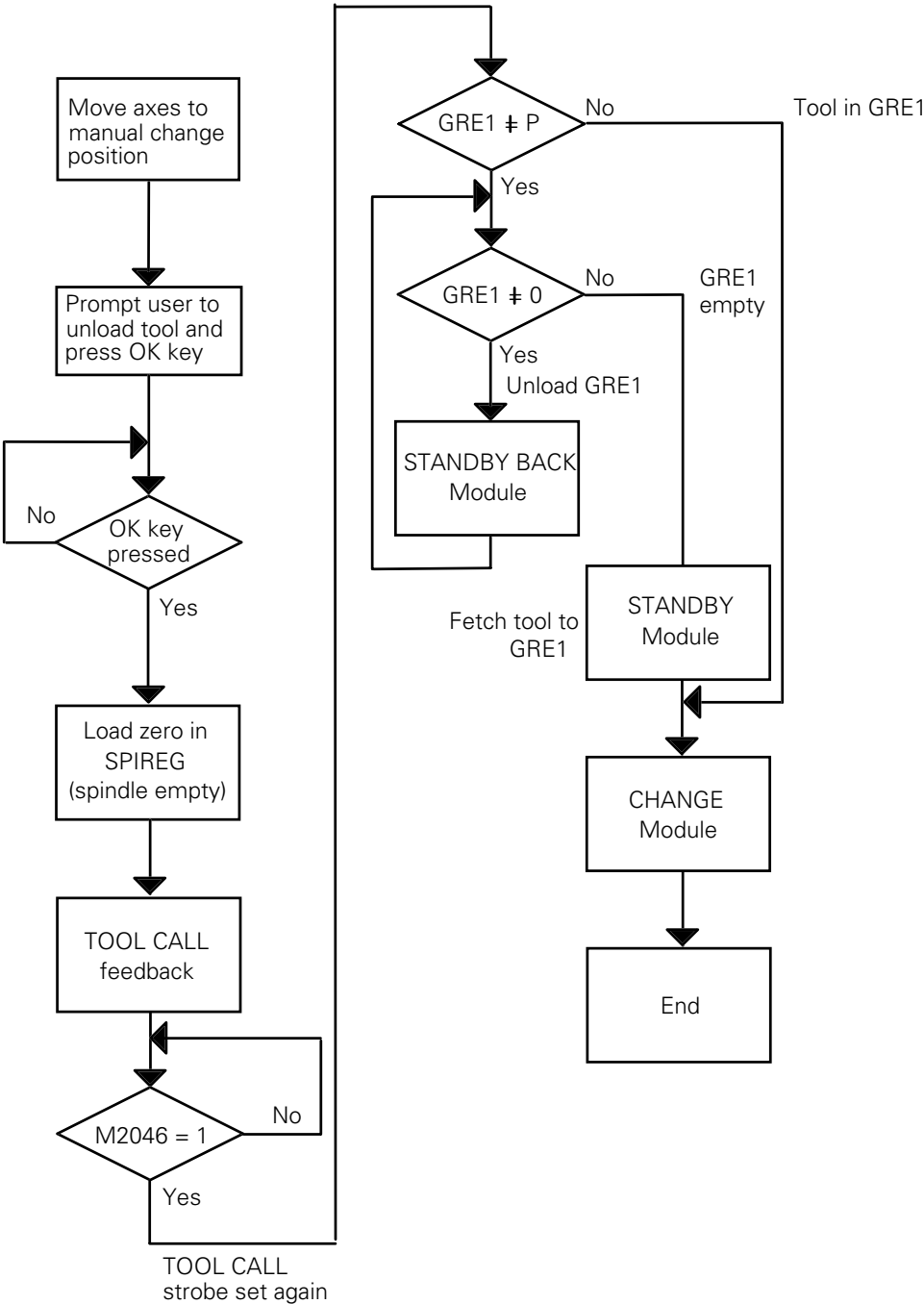
The old tool is returned to the tool magazine and the operator is prompted to load a manual tool (not in tool magazine).



15.3.6 Program module MANUAL TOOL OUT

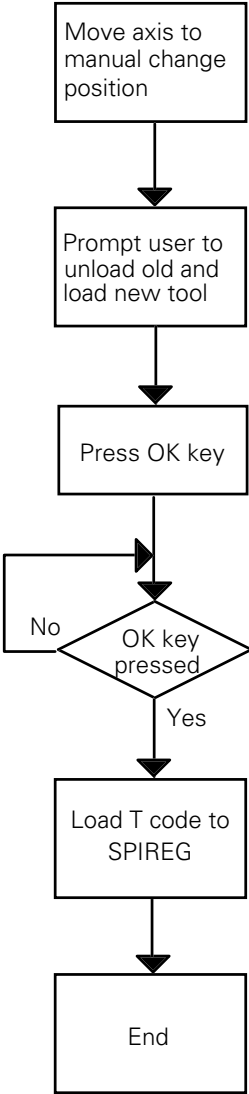
M -> N or M -> S: Normal or Special tool follows Manual tool

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.



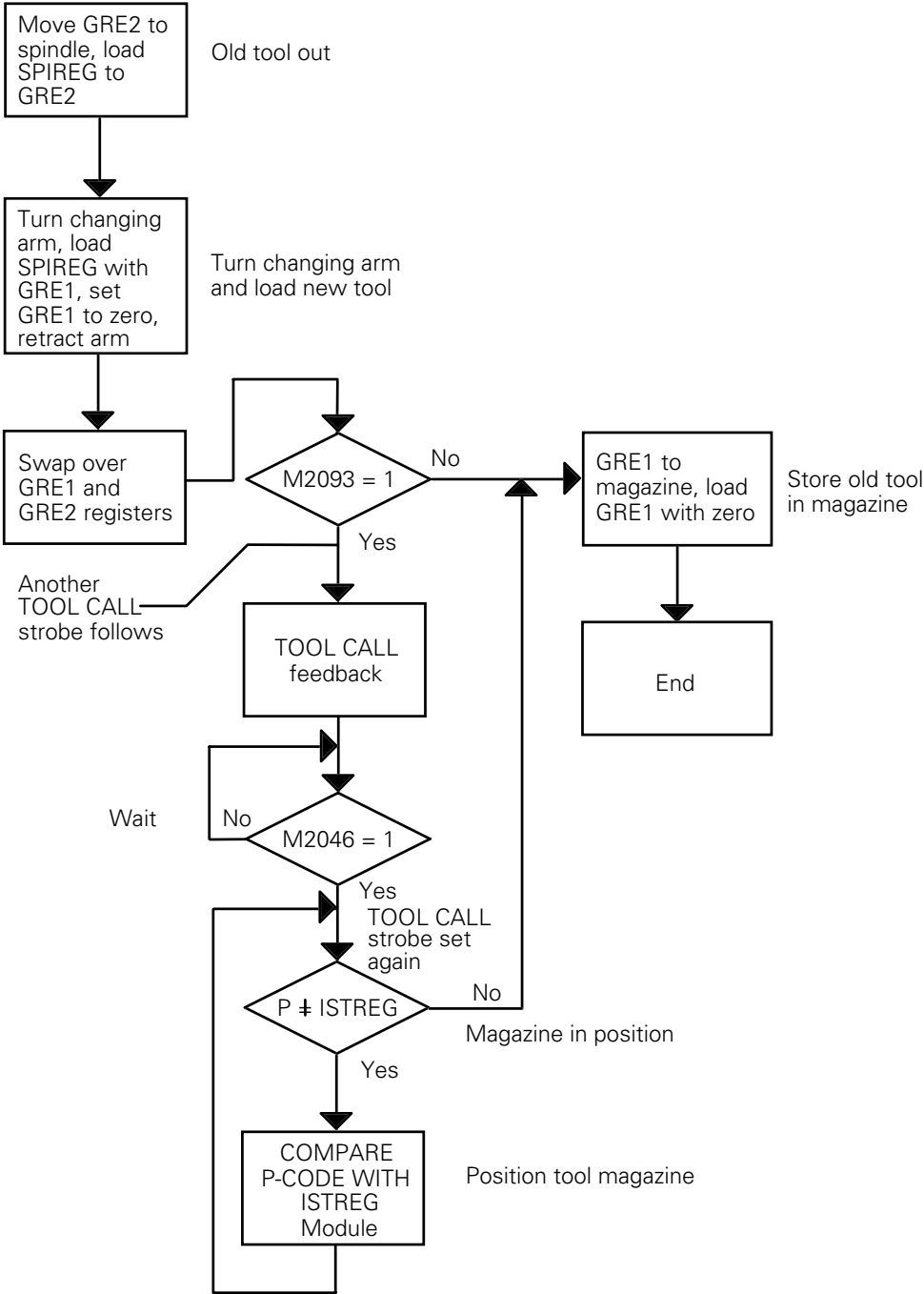
15.3.7 Program module MANUAL TOOL IN/OUT

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.



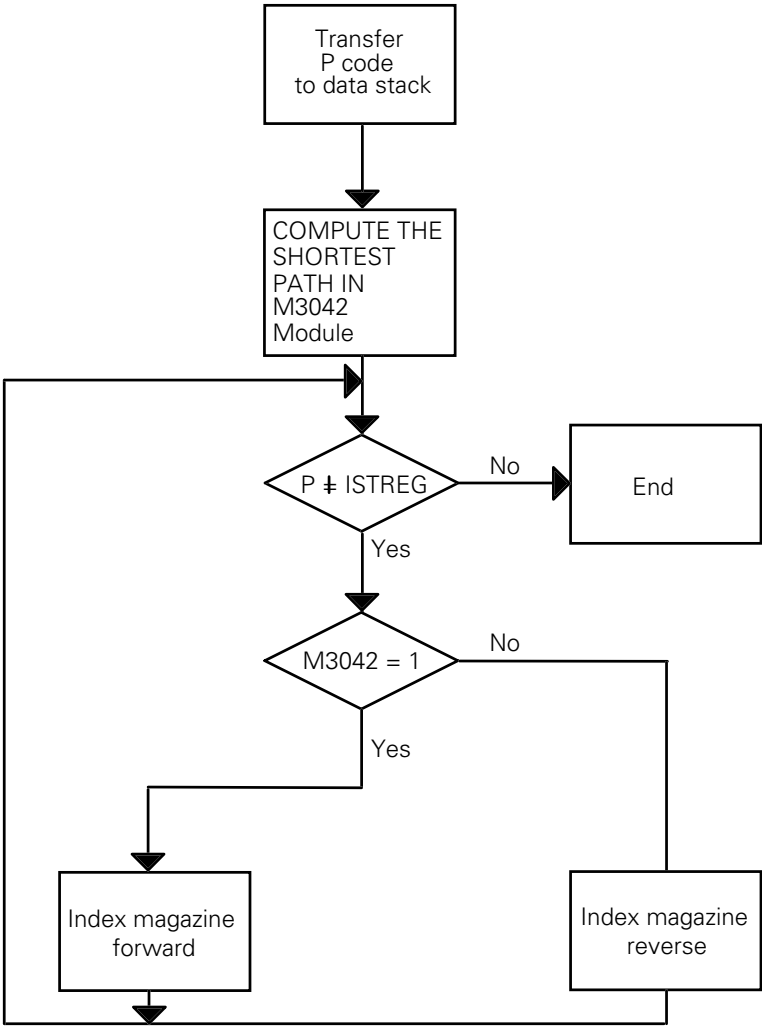
15.3.8 Program module CHANGE

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).



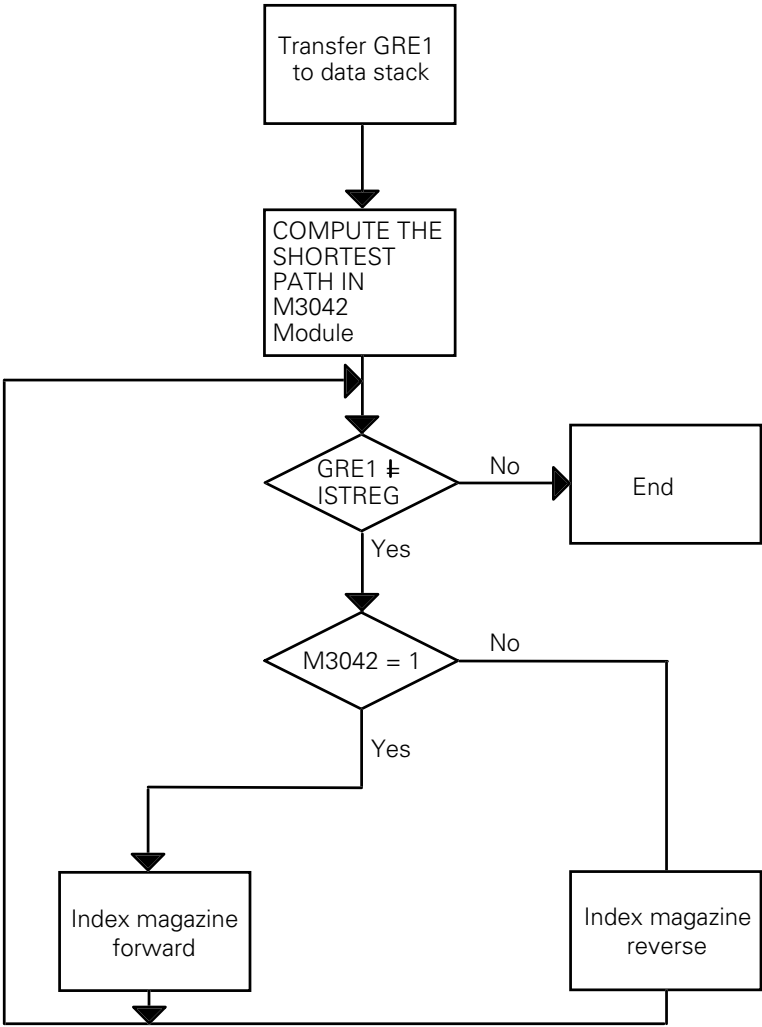
15.3.9 Program module COMPARE P CODE WITH ISTREG

The tool magazine is positioned at the search pocket number by the shortest path.



15.3.10 Program module COMPARE GRE1 WITH ISTREG

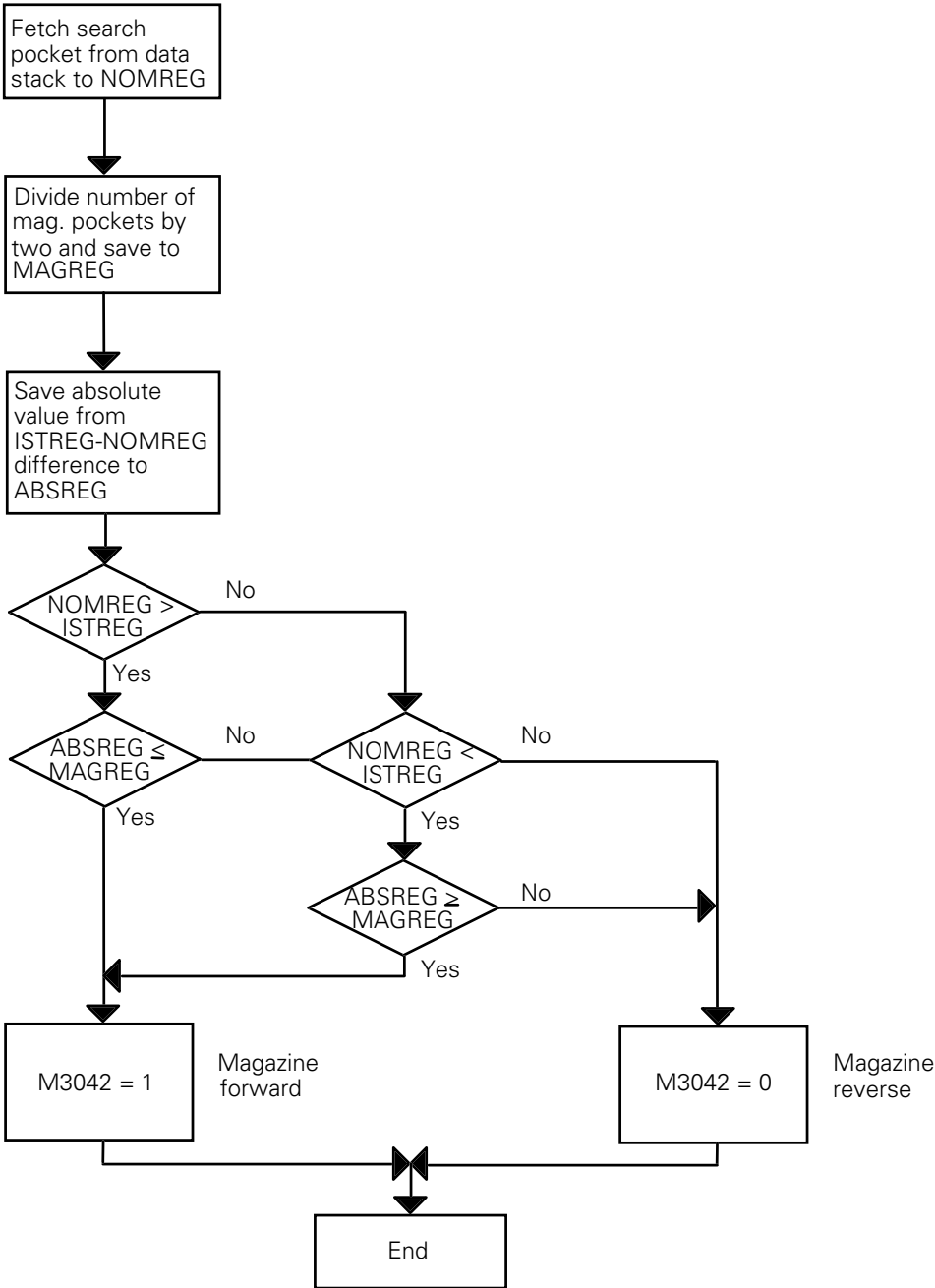
The tool magazine is positioned at the pocket number of the tool in GRE1 by the shortest path.



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
9 51 48	List of machine parameters (see Chapter "Machine parameters")
80 76 67	PLC operation (see Chapter "PLC programming")
53 12 10	Cancel markers M1000 to M2000 and B0 to B127
7 53 68	Automatic offset adjustment (see Section "Servo positioning")
10 52 96	Non-linear axis error compensation (see Section 1.6.4)

16.2 Preparation of the machine

The machine is prepared without a control being connected.

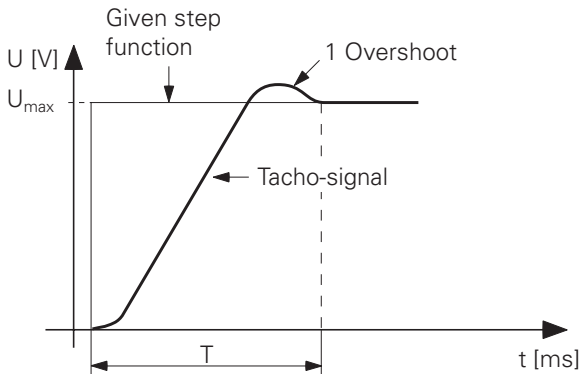
Follow the sequence below to commission the drive amplifiers:

- 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 tachopotentiometer 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:

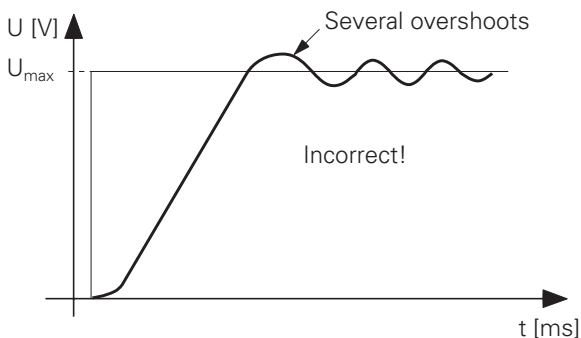
As far as the control is concerned, the actual servo-loop consists of the drive amplifier, motor and axis slide (see Section "Servo positioning"). The servo-loop must be tuned before the position control loop in the control can be optimized.

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 of the tachometer signal can be recorded by an oscillograph. 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.



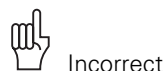
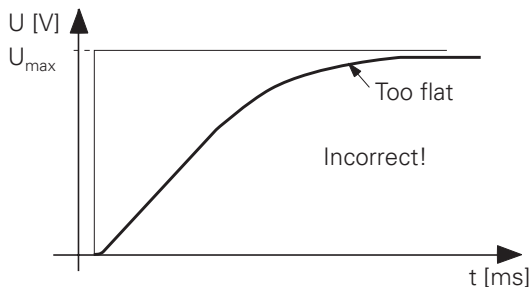
 Correct

The following pictures show **incorrectly adjusted** tacho-signals:



 Incorrect

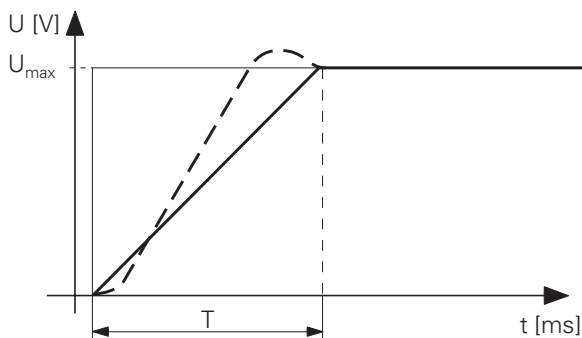
The P component of the subordinate control loop is too high, or the I component too low.



The P component of the subordinate control loop is too low, or the I component too high.

Determination of the 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:

$$a = \frac{v_{\max}}{T \cdot 1.1} \quad \begin{array}{l} v_{\max} = \text{convert velocity at } U_{\max} \text{ in [m/sec]} \\ a = \text{acceleration [sec]} \\ T = \text{Acceleration time [sec]} \end{array}$$

The step response must be investigated for all axes.

The acquired acceleration values are entered in MP1060.0 to MP1060.3.

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 Chapter "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 "Operation 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 your HEIDENHAIN customer service agency

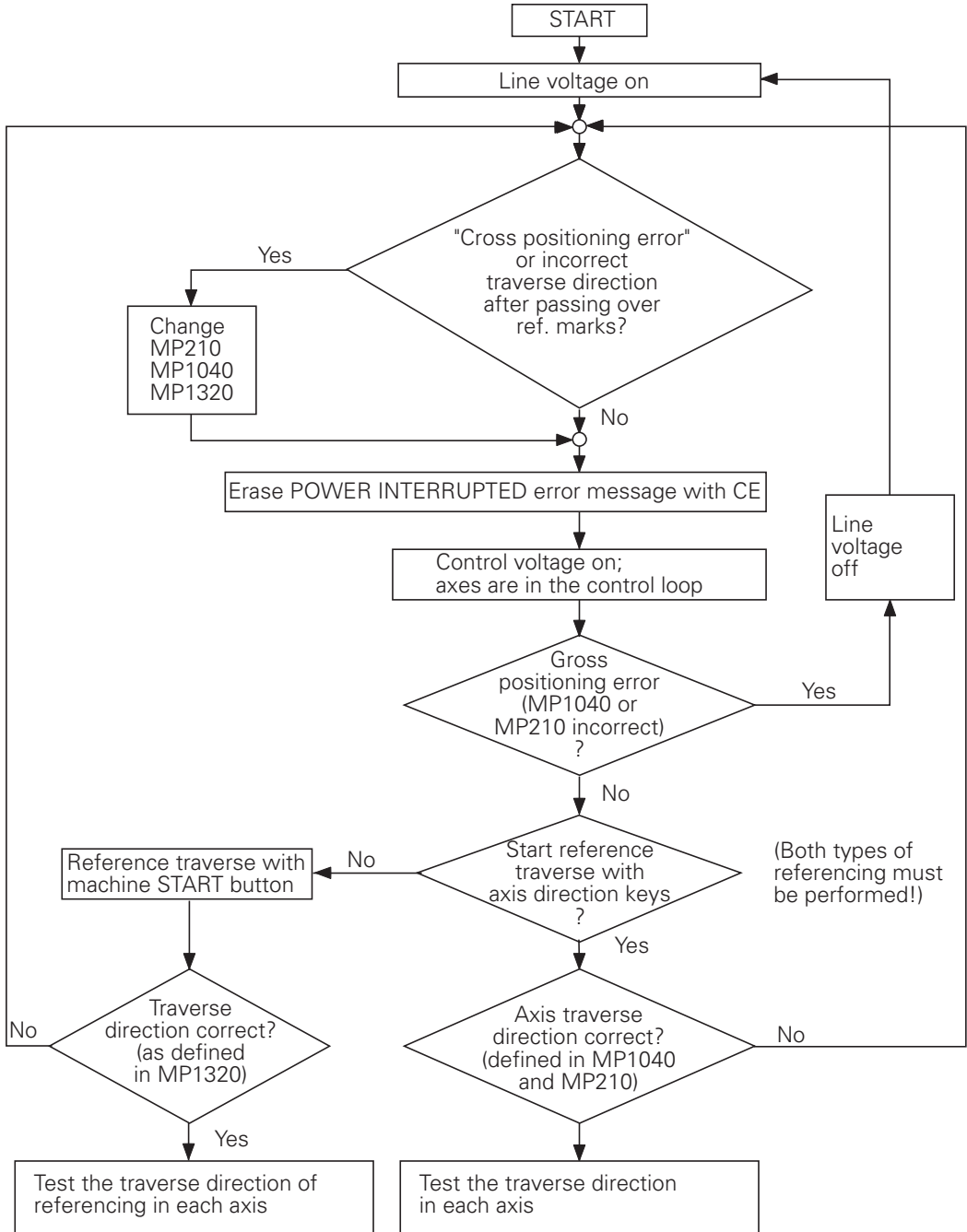
16.3.3 Testing the EMERGENCY STOP routine

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 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

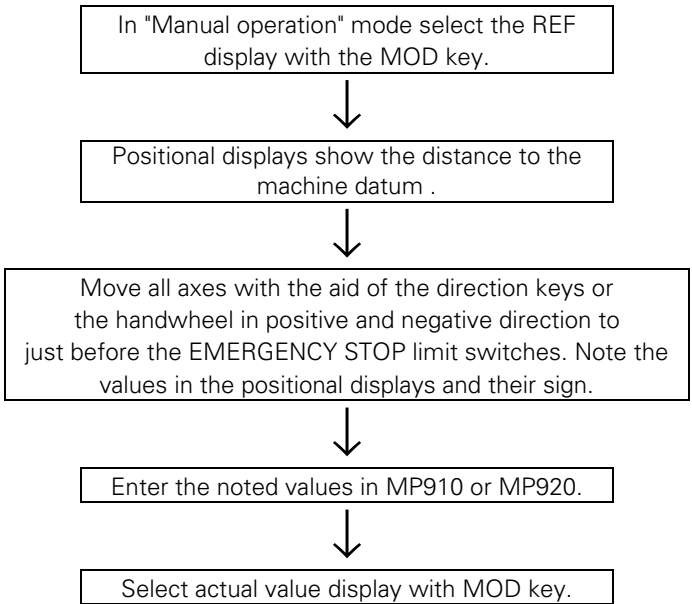
Test the controls for the direction of traverse according to the following diagram (effective machine parameters: MP210 count direction of the measuring system signals, MP1040 polarity of the nominal value voltage, MP1320 direction of traverse on passing over the reference marks).



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:



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	Analog 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	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 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)

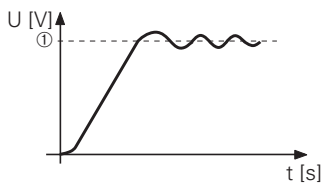
Connect storage oscilloscope to tachometer of the servo-amplifier of X axis.

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
```

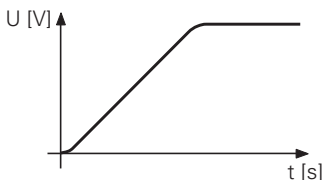
Press machine START button in the "PROGRAM RUN/FULL SEQUENCE" operating mode. Machine runs.
Note: Set feed override to 100%.

MP1810: Increase entry value until position loop oscillates or until an overshoot occurs after the acceleration ramp.



① Machining feed

MP1810 (k_v factor): Reduce entry value until no oscillations are detectable.



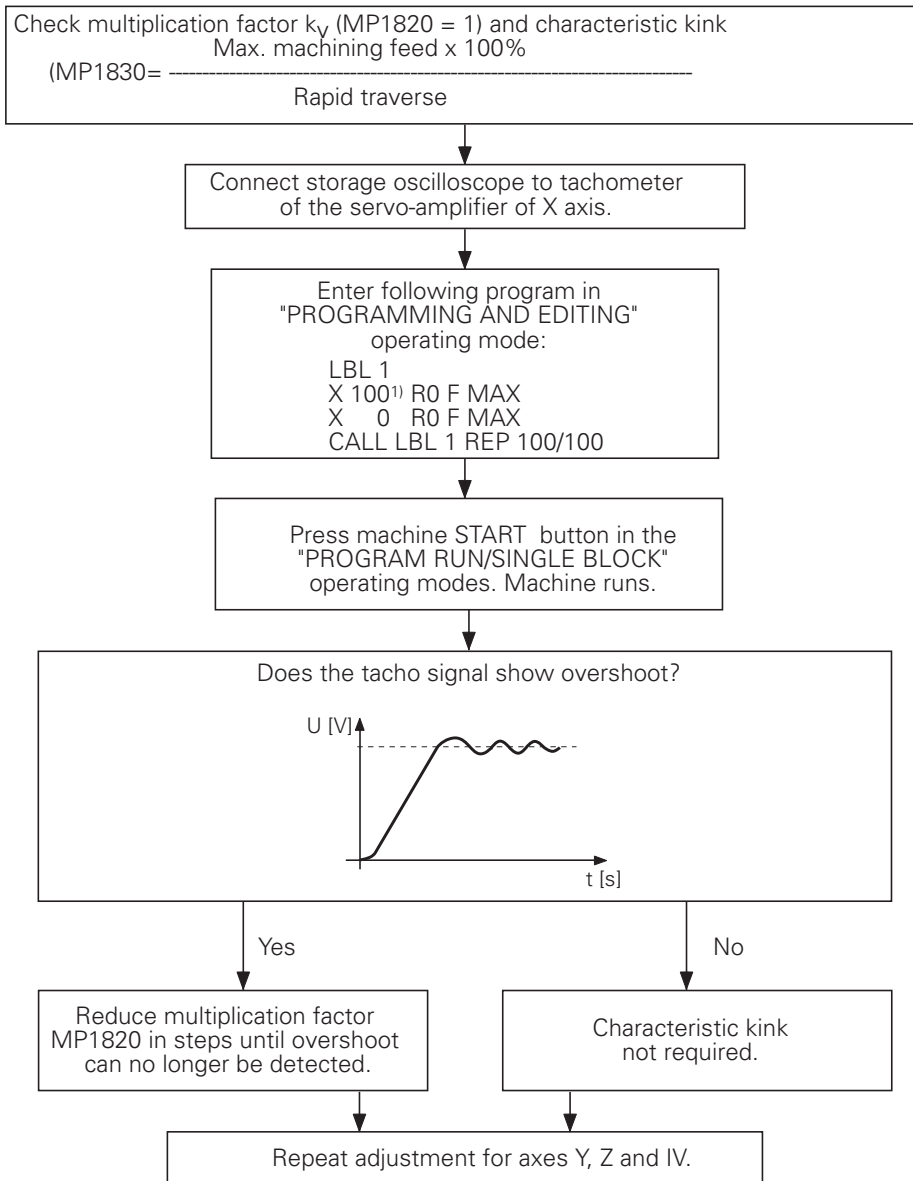
Repeat adjustment for axes Y, Z, and IV.

¹⁾ Program the traverse paths for the axis concerned as large as possible.

²⁾ Enter the max. feed rate for machining.

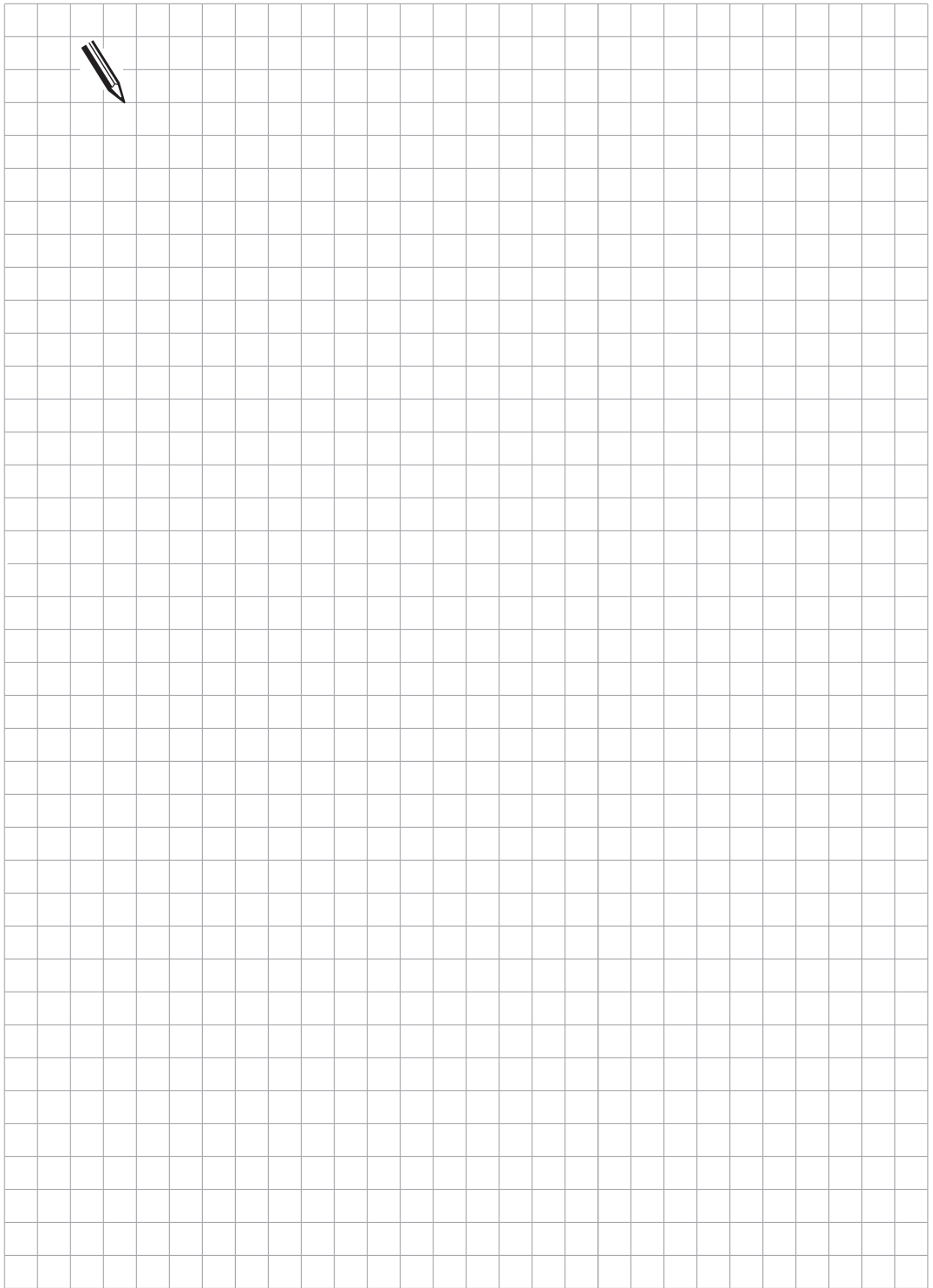
For axes which are mutually interpolated the k_v factor must be the same.
In this case the worst axis determines the entry value.

Optimize k_v factor for the rapid traverse rate



¹⁾ Program the traverse paths for the axis concerned as large as possible.

The worst axis determines the entry value for MP 1820.



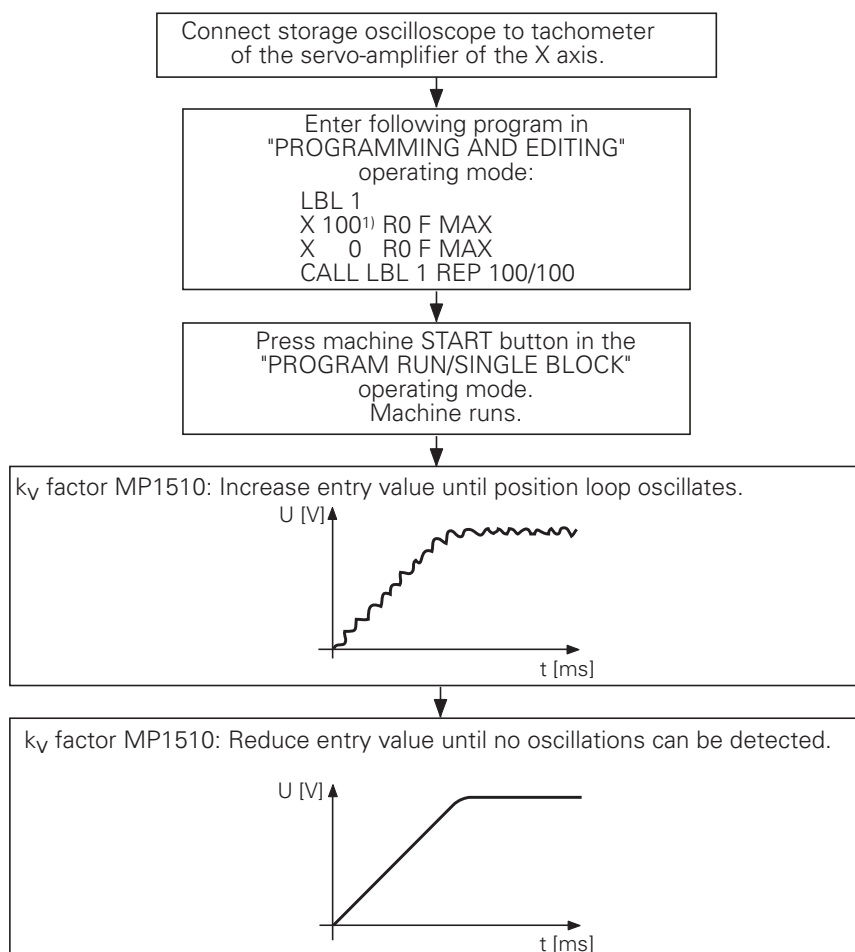
16.3.7 Optimizing the control with feed forward control (MP1390 = 0)

The following provisional values can be entered for the machine parameters which determine the control characteristics:

Machine parameter	Function	Provisional entry value
MP1050	Analog 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]
MP1530	Transient response	0.1

These values can usually be further optimized.

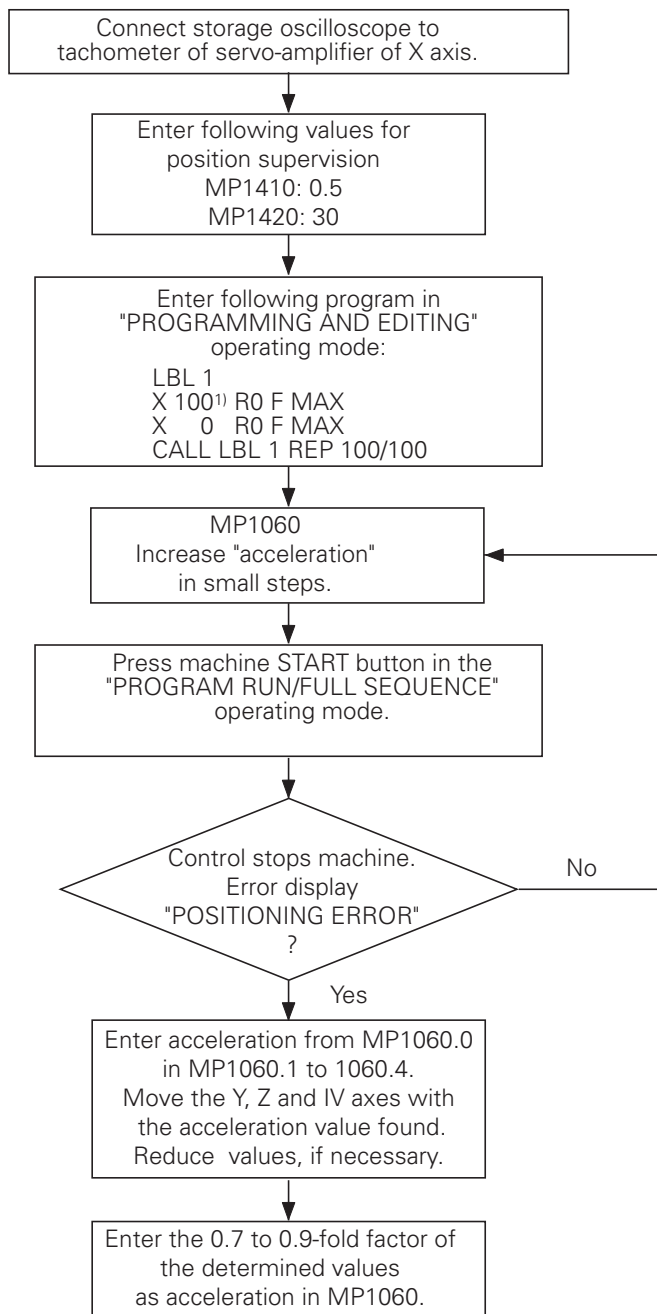
Optimize k_v factor



¹⁾ Program the traverse paths for the axis concerned as large as possible.

Optimizing acceleration

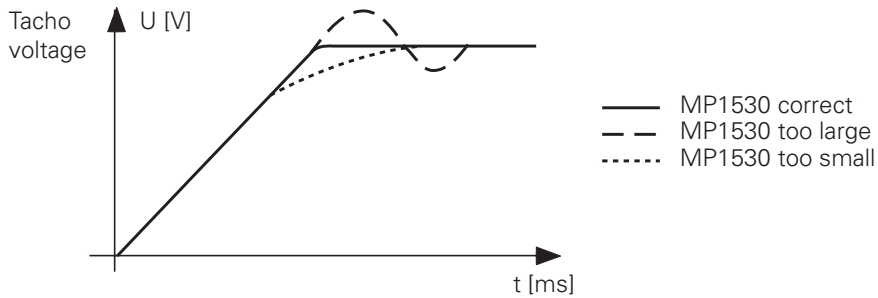
If the maximum acceleration of the servo loop cannot be determined with the battery supply, the acceleration can be optimized as follows:



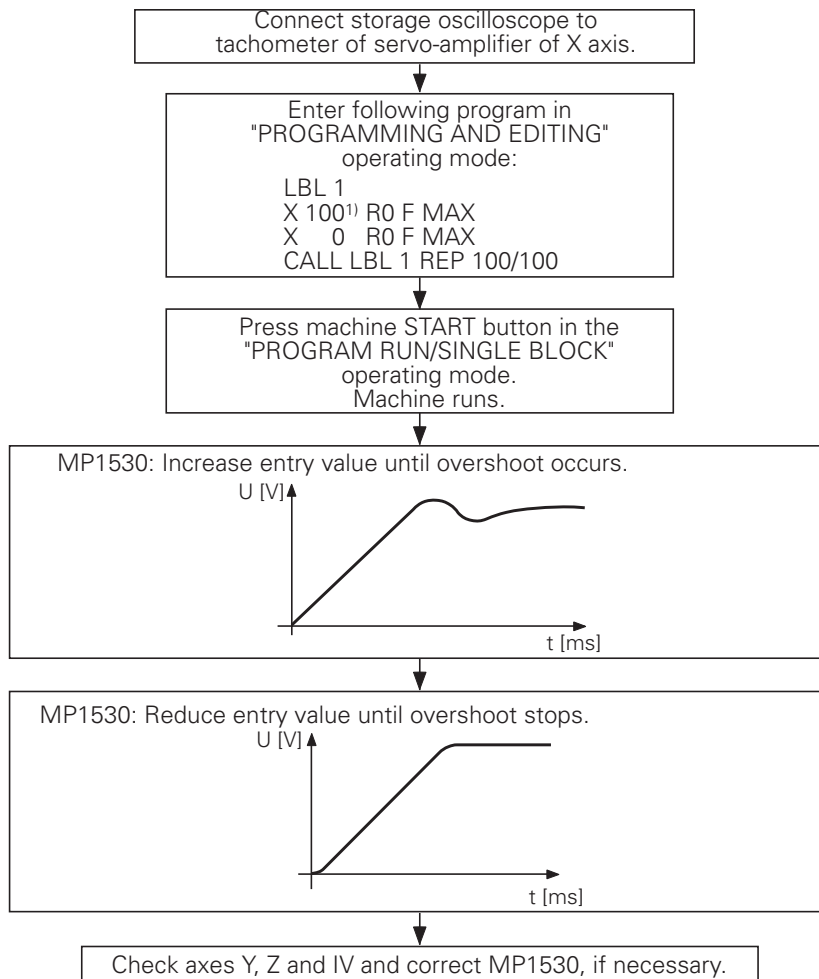
¹⁾ Program the traverse paths for the axis concerned as large as possible.

Optimizing transient response

MP1530 affects the form of the acceleration ramp in all axes .



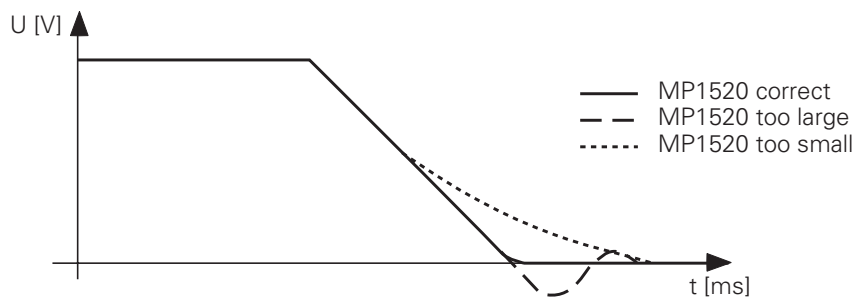
Since MP1530 is effective for all axes, the worst axis determines the entry value. No axis should show an overshoot.



¹⁾ Program the traverse paths for the axis concerned as large as possible.

Optimize the position approach

For all axes, MP1520 influences the slope of the run-in characteristic when close to the nominal position



Since MP1520 is effective for all axes, the worst axis determines the entry value. No axis should show an overshoot of the nominal position.

If MP1520 is made too small, the approach times to the nominal position will be significantly longer. So MP1520 must be carefully optimized!

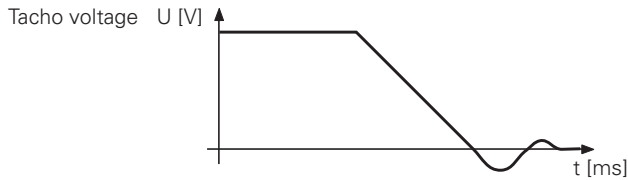
Connect storage oscilloscope to tachometer of servo-amplifier of X axis.

Enter following program in "PROGRAMMING AND EDITING" operating mode:

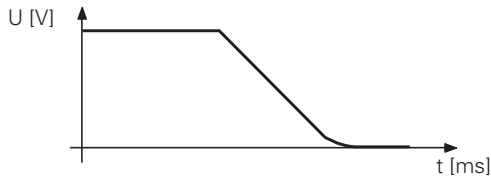
```
LBL 1  
X 1001) R0 F MAX  
X 0 R0 F MAX  
CALL LBL 1 REP 100/100
```

Press machine START button in the "PROGRAM RUN/FULL SEQUENCE" operating mode.
Machine runs.

MP1520: Increase entry value until the position is clearly traversed when approaching. The overshoot can also be observed in the actual value display.



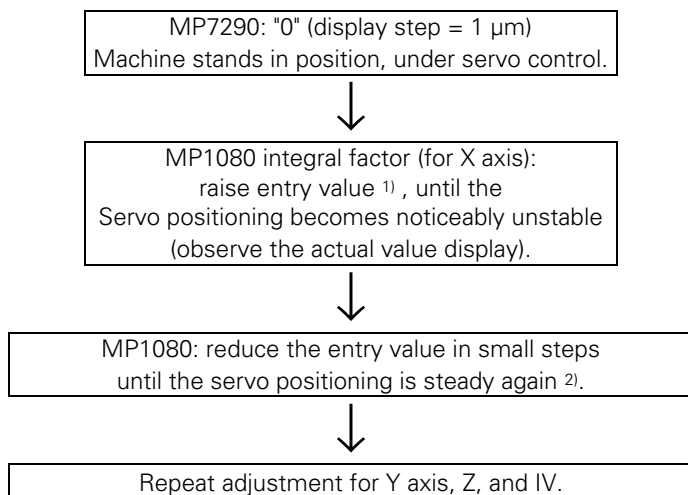
MP1520: Reduce the entry value until the position is no longer traversed.



Check axes Y, Z and IV and correct MP1530, if necessary.

¹⁾ Program the traverse paths for the axis concerned as large as possible.

16.3.8 Optimizing the integral factor



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

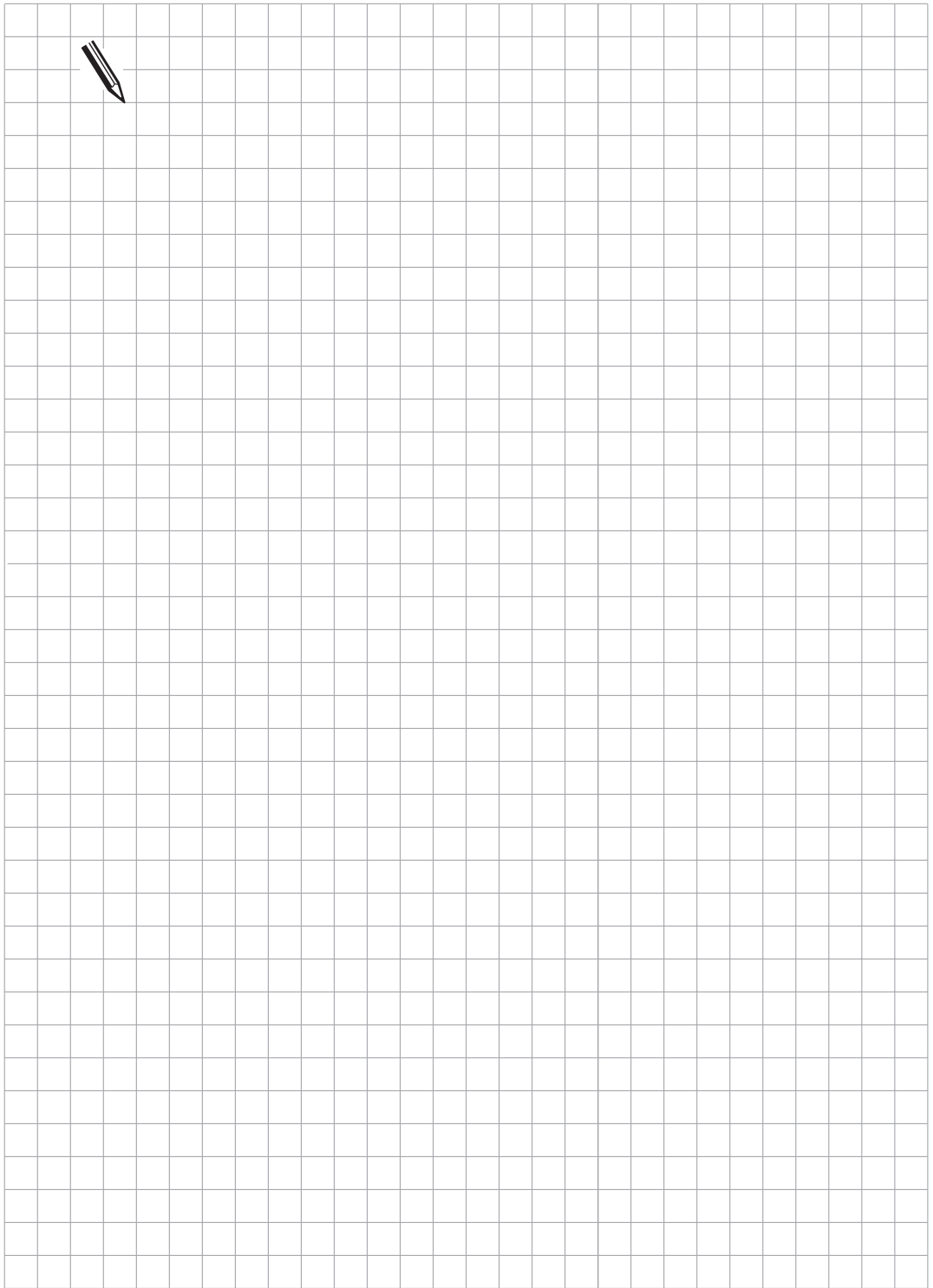
The rough offset adjustment has already been carried out on the servo amplifier. A fine offset adjustment can be performed with the offset adjustment possibility described in Section "Servo positioning".

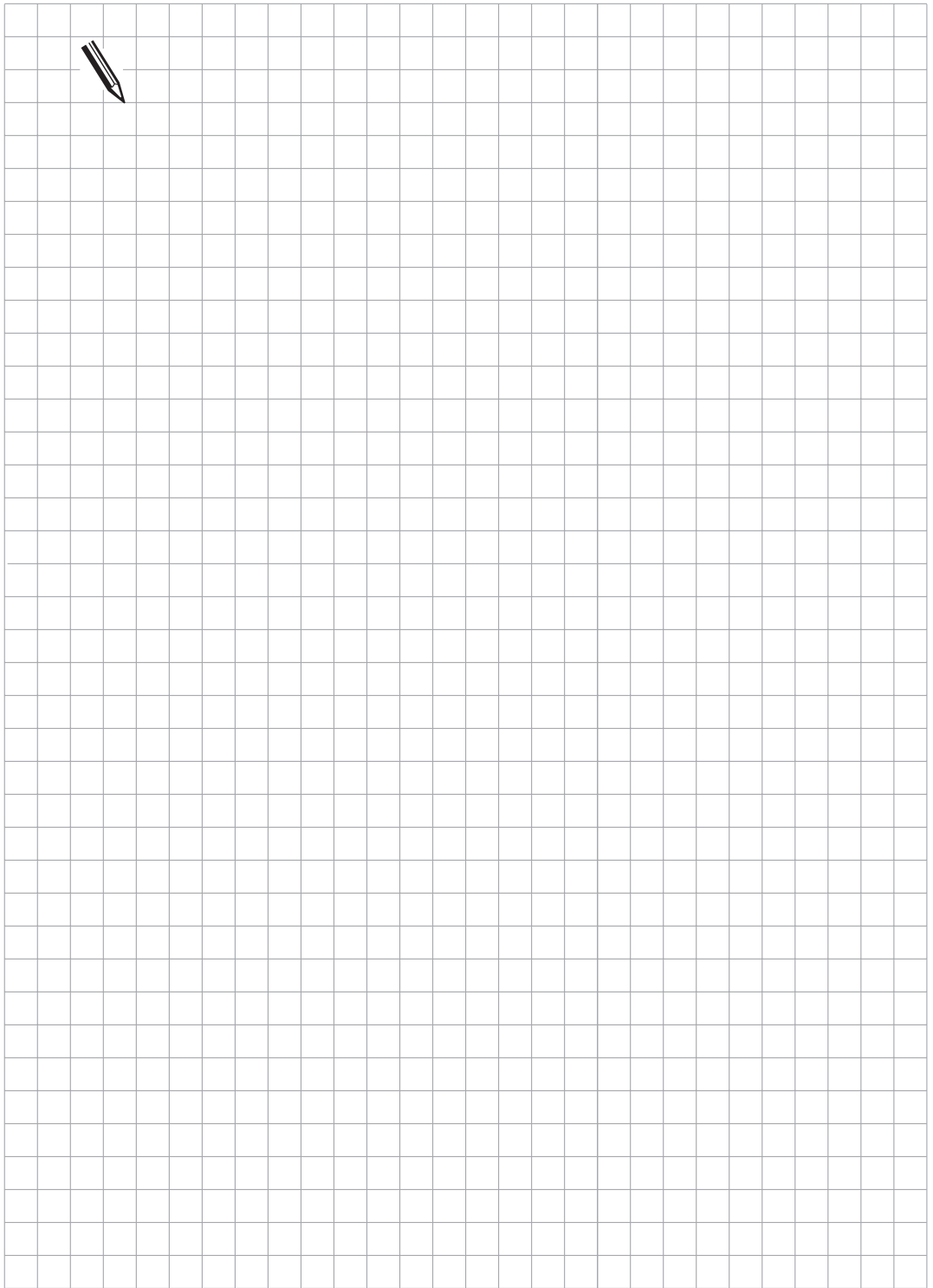
16.3.10 Adjustment of the monitoring functions

The following entry values are recommended for the monitoring functions (see Section "Servo positioning"):

Machine parameter	Function	Entry value
MP1710	Position monitoring	$1.2 \times \text{lag}$ in rapid traverse
MP1720	Position monitoring (EMERGENCY STOP)	$1.4 \times \text{lag}$ 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 Point-to-Point and Straight-Cut Control

The LE 360 logic unit is used as a control for simple machine tools with common drive.

All functions for which two or more axes must be interpolated are inhibited with machine parameter MP7215, while certain other machine parameters are without function.

MP7215	Contouring control or point-to-point and straight-cut control		
	Input: 0 to 3		
	Input value 0:	Contouring control	
	Input value 1:	Point-to-point and straight-cut control	
		Contouring functions inhibited	
		Basic rotation inhibited	
		ISO inhibited	
		MP 1070 without function	
		MP 7220 without function	
		MP 7222 without function	
		MP 7420 without function	
		MP 7430 without function	
		MP 7460 without function	
	Input value 3:	Position display inhibited, otherwise same as input value 1 (only nominal, REF, lag, distance-to-go display possible)	
		Set	Reset
M2040	Control is point-to-point and straight-cut control	NC	NC

Machines with common drive

On machines with common drive is necessary to set the analog voltages for the axis drives to a common output through machine parameter MP120 (see Section 1.2.1).

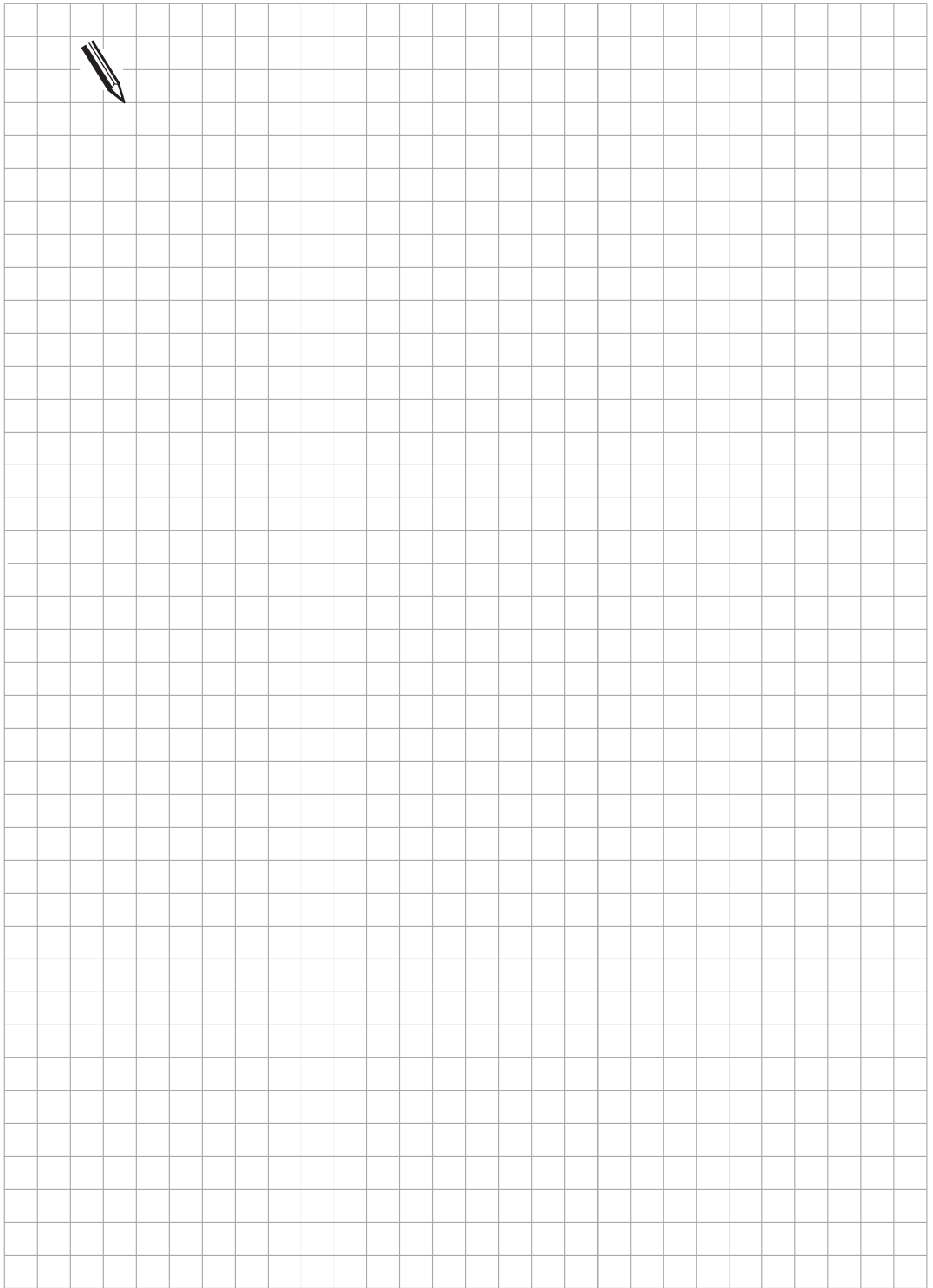
Example for MP120:

Input for all axes: 0 $\triangleq$ Output 1

X axis MP 120.0 = 0
Y axis MP 120.1 = 0
Z axis MP 120.2 = 0

This means that the analog voltage for the X, Y and Z axis is sent to output 1.

After an axis is positioned, its control loop must be opened through the PLC program and its axis must be clamped (see Section 3.6.4).



Machine parameters – Contents

1	What is a Machine Parameter?	5-2
1.1	User parameters	5-2
2	Input/Output of Machine Parameters	5-3
3	List of Machine Parameters	5-4
3.1	Measuring systems and machines	5-4
3.2	Positioning	5-8
3.3	Operation with feed forward control	5-10
3.4	Operation with servo lag	5-10
3.5	Spindle	5-11
3.6	Integrated PLC	5-13
3.7	Setting the data interfaces	5-14
3.8	Measuring with a 3D touch probe	5-16
3.9	Digitizing with TS 120 (only with digitizing option)	5-16
3.10	Tapping	5-17
3.11	Display and operation	5-17
3.12	Machining and program run	5-21
3.13	Hardware	5-23

1 What is a Machine Parameter?

A contouring control must have access to specific data (e.g., paths of traverse, 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 activate certain functions, which are possible with HEIDENHAIN contouring controls, but are required only on certain types of machines (e.g. automatic tool changing).

The machine parameters are grouped according to function:

Machine parameters	Functional group
0 to 999	Measuring systems and machine axes
1000 to 1399	Positioning
1400 to 1699	Operation with feed forward control
1700 to 1999	Operation with servo lag
3000 to 3999	Spindle
4000 to 4999	Integrate 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

The indices are assigned to the corresponding axes according to a fixed pattern.

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 MP 7330 / MP 7340 (see Chapter "Machine Integration", Section "Display and operation").

2 Input/Output of Machine Parameters

To call the machine parameter editor, press the MOD key and select "code number."

Enter the **code number 95148** to access the complete set of machine parameters.

The accesses a limited set of the machine parameters. This set of parameters can be safely entrusted to access by the control user (see User's Manual) The machine parameters that can be changed with the code number 123 are indicated in the list below with CN 123.

To exit the machine parameter editor, press the END ☐ key.

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 preset to RS-232-C format and FE1 mode. This default setting can be changed through the MOD functions (see Chapter "Data Interface").

A number is entered for each machine parameter. This value can be, for example, the acceleration in m/s² of an individual axis, or the analog voltage in volts.

Some machine parameters have multiple functions. The input values for these machine parameters are calculated according to the function to be activated.

Up to 16 different functions can be activated/deactivated in one machine parameter with bits 0 to 15. The input value is the sum of the decimal values of the bits representing the desired functions.

Bit	Significance	Decimal value
0	2 ⁰	1
1	2 ¹	2
2	2 ²	4
3	2 ³	8
4	2 ⁴	16
5	2 ⁵	32
6	2 ⁶	64
7	2 ⁷	128
8	2 ⁸	256
9	2 ⁹	512
10	2 ¹⁰	1 024
11	2 ¹¹	2 048
12	2 ¹²	4 096
13	2 ¹³	8 192
14	2 ¹⁴	16 384
15	2 ¹⁵	32 768

Incorrect entries are indicated with the error message "entry value incorrect."

3 List of Machine Parameters

3.1 Measuring systems and machines

Machine parameter	Function and input	Reaction/ CN 123	Page
MP10	Active axes Input range: 1 to 15 Bit 0 X axis +0 = not active +1 = active Bit 1 Y axis +0 = not active +2 = active Bit 2 Z axis +0 = not active +4 = active Bit 3 4th axis +0 = not active +8 = active	RESET	4-5
MP30	Checking the absolute position of the distance-coded reference marks Input range: 0 to 15 Bit 0 X axis +0 = not active +1 = active Bit 1 Y axis +0 = not active +2 = active Bit 2 Z axis +0 = not active +4 = active Bit 3 4th axis +0 = not active +8 = active		4-8
MP31	Checking the amplitude of the measuring system signals Input range: 0 to 31 Bit 0 X axis +0 = not active +1 = active Bit 1 Y axis +0 = not active +2 = active Bit 2 Z axis +0 = not active +4 = active Bit 3 4th axis +0 = not active +8 = active Bit 4 S axis +0 = not active +16 = active		4-8

Machine parameter	Function and input	Reaction/ CN 123	Page
MP32	Checking the edge separation of the measuring system signals Input range: 0 to 31 Bit 0 X axis +0 = not active +1 = active Bit 1 Y axis +0 = not active +2 = active Bit 2 Z axis +0 = not active +4 = active Bit 3 4th axis +0 = not active +8 = active Bit 4 S axis +0 = not active +16 = active		4-9
MP40	VDU display Input range: 1 to 15 Bit 0 X axis +0 = not active +1 = active Bit 1 Y axis +0 = not active +2 = active Bit 2 Z axis +0 = not active +4 = active Bit 3 4th axis +0 = not active +8 = active		4-13
MP50	Controlled axes Input range: 0 to 15 Bit 0 X axis +0 = not controlled +1 = controlled Bit 1 Y axis +0 = not controlled +2 = controlled Bit 2 Z axis +0 = not controlled +4 = controlled Bit 3 4th axis +0 = not controlled +8 = controlled		4-74
MP60	PLC axes Input: 0 to 15 Bit 0 Axis X +0 = normal NC axis +1 = PLC axis Bit 1 Axis Y +0 = normal NC axis +2 = PLC axis Bit 2 Axis Z +0 = normal NC axis +4 = PLC axis Bit 3 4th axis +0 = normal NC axis +8 = PLC axis		4-35

Machine parameter	Function and input	Reaction / CN 123	Page
MP110.0 to MP110.3	Assignment of the measuring system inputs to the axes Input range: 0 to 5 0 = measuring system input X1 1 = measuring system input X2 2 = measuring system input X3 3 = measuring system input X4 5 = measuring system input X6	RESET	4-12
MP120.0 to MP120.3	Assignment of the nominal value outputs Input range: 0 to 5 0 = output 1 1 = output 2 2 = output 3 3 = output 4 5 = output S	RESET	4-13
MP210	Count direction of the measuring system signals Input range: 0 to 15 Bit 0 X axis +0 = positive +1 = negative Bit 1 Y axis +0 = positive +2 = negative Bit 2 Z axis +0 = positive +4 = negative Bit 3 4th axis +0 = positive +8 = negative	RESET	4-7
MP310	Signal subdivision Input range: 0 to 3 0 = 256-fold 1 = 4-fold 2 = 2-fold 3 = 1-fold		4-6
MP330.0 to MP330.3	Signal period Input range 1000 to 1000,000 in [μm] or $\left[\frac{1^\circ}{1000} \right]$	REF	4-6
MP340.0 to MP340.4	Interpolation factor of the EXE at X6 Input: 0 to 5 0 = No measuring system on X6 1 = EXE without interpolation 2 = Spindle encoder with 2048 lines 3 = Spindle encoder with 4096 lines 5 = EXE with 5-fold interpolation		4-6

Machine parameter	Function and input	Reaction / CN 123	Page
MP410.3	Axis designation for axis 4 Input range: 0 to 5 0 = A 1 = B 2 = C 3 = U 4 = V 5 = W	RESET	4-12
MP710.0 to MP710.3	Backlash compensation Input range: –1.000 to +1.000 [mm]		4-21
MP711.0 to MP711.3	Size of the reversal peaks in circular motion Input range: –1.000 to 1.000 [mm]		4-22
MP712.0 to MP712.3	Compensation value per control loop cycle time Input range: 0.000 to 1.000 [mm]		4-23
MP720.0 to MP720.3	Linear axis-error compensation Input range: – 1.000 to + 1.000 in [mm/m]		4-23
MP730	Non-linear axis-error compensation Input range: 0 to 15 Bit 0 X axis +0 = not active +1 = active Bit 1 Y axis +0 = not active +2 = active Bit 2 Z axis +0 = not active +4 = active Bit 3 4th axis +0 = not active +8 = active		4-27
MP810	Modulo value for display of the PLC axes Input: 0.000 to 99 999999 [mm] MP810.0 Axis X MP810.1 Axis Y MP810.2 Axis Z MP810.3 Axis 4		4-35
MP910.0 to MP910.3	Traverse range 1: Maximum value Input range: –99 999.999 to +99 999.999 [mm] or [°]		4-15
MP911.0 to MP911.3	Traverse range 2: Maximum value Input range: –99 999.999 to +99 999.999 [mm] or [°]		4-15
MP912.0 to MP912.3	Traverse range 3: Maximum value Input range: –99 999.999 to +99 999.999 [mm] or [°]		4-16

Machine parameter	Function and input	Reaction / CN 123	Page
MP920.0 to MP920.3	Neg. software limit switch for traverse range 1: Input range: –99 999.999 to +99 999.999 [mm] or [°]		4-15
MP921.0 to MP921.3	Neg. software limit switch for traverse range 2: Input range: –99 999.999 to +99 999.999 [mm] or [°]		4-15
MP922.0 to MP922.3	Neg. software limit switch for traverse range 3: Input range: –99 999.999 to +99 999.999 [mm] or [°]		4-16
MP950.0 to MP950.3	Datum point for positioning blocks with M92 Input range: –99 999.999 to +99 999.999 [mm] or [°] Values referenced to machine datum		4-110

3.2 Positioning

Machine parameter	Function and input	Reaction / CN 123	Page
MP1010.0 to MP1010.3	Rapid traverse Input range: 80 to 300 000 [mm/min]		4-56
MP1020.0 to MP1020.3	Manual feed Input range: 80 to 300 000 [mm/min]		4-56
MP1030.0 to MP1030.3	Positioning window Input range: 0.001 to 2 [mm] or [°]		4-72
MP1040	Polarity of the nominal value voltage for the positive direction of traverse Input range: 0 to 15 Bit 0 X axis +0 = positive +1 = negative Bit 1 Y axis +0 = positive +2 = negative Bit 2 Z axis +0 = positive +4 = negative Bit 3 4th axis +0 = positive +8 = negative		4-7
MP1050.0 to MP1050.3	Analog voltage for rapid traverse Input range: 4.5 to 9 [V]		4-56
MP1060.0 to MP1060.3	Acceleration Input range: 0.001 to 3.0 [m/s²]		4-54
MP1070	Radial acceleration Input range: 0.001 to 3.0 [m/s²]		4-66
MP1080.0 to MP1080.3	Integral factor Input range: 0 to 65 535		4-64

Machine parameter	Function and input	Reaction / CN 123	Page
MP1110	Standstill monitoring Input range: 0.001 to 30 [mm]		4-72
MP1140	Movement monitoring Input value: 0.03 to 10 [V]		4-71
MP1150	Delay time before switching off the residual voltage on error message "Position error" Input value: 0 to 65.535 [s]		4-70
MP1220	Automatic cyclical offset adjustment Input value 0 to 65 536 [s] 0 = no automatic adjustment		4-63
MP1320	Direction for traversing the reference marks Input range: 0 to 15 Bit 0 Axis X +0 = positive +1 = negative Bit 1 Axis Y +0 = positive +2 = negative Bit 2 Axis Z +0 = positive +4 = negative Bit 3 4th axis +0 = positive +8 = negative		4-48
MP1330.0 to MP1330.3	Feed rate for traversing the reference marks Input range: 80 to 30 000 [mm/min]		4-48
MP1331.0 to MP1331.3	Feed rate for leaving the reference end position (only for rotary encoders MP1350 = 2) Input range: 80 to 500 [mm/min]		4-48
MP1340.0 to MP1340.3	Sequence for traversing reference marks Input range: 0 to 4 0 = no evaluation of the reference marks 1 = axis X 2 = axis Y 3 = axis Z 4 = 4th axis	REF	4-48
MP1350.0 to MP1350.3	Type of reference mark approach Input range: 0 to 3 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; after REF end position traverse direction is changed	REF	4-49

Machine parameter	Function and input	Reaction / CN 123	Page
MP1390	Control with lag or feed forward control Input range: 0 or 1 1 = operation with lag 0 = operation with feed forward control		4-52

3.3 Operation with feed forward control

Machine parameter	Function and input	Reaction / CN 123	Page
MP1410	Position monitoring in operation with feed forward (cancelable) Input value: 0.001 to 30 [mm]		4-70
MP1420	Position monitoring in operation with feed forward (EMERGENCY STOP) Input value: 0.001 to 30 [mm]		4-70
MP1510.0 to MP1510.3	k_v factor for feed forward control Input value: 0.1 to 10 [m/min/mm]		4-62
MP1520	Position approach speed Input value: 0.1 to 10 [m/min]		4-62
MP1530	Damping factor for transient response Input value: 0.01 to 0.999		4-62

3.4 Operation with servo lag

Machine parameter	Function and input	Reaction / CN 123	Page
MP1710	Position monitoring for operation with lag (cancelable) Input value: 0.001 to 1000 [mm]		4-69
MP1720	Position monitoring for operation with lag (EMERGENCY STOP) Input value: 0.001 to 1000 [mm]		4-69
MP1810.0 to MP1810.3	k_v factor for operation with lag Input range: 0.1 to 10 [m/min/mm]		4-56
MP1820	Multiplication factor for the k_v factor Input value: 0.001 to 1		4-58
MP1830	Kink point Input value: 0 to 100 [%]		4-58

3.5 Spindle

Machine parameter	Function and input	Reaction / CN 123	Page
MP3010	Spindle-speed output, Spindle orientation Input value: 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 = Analog output of the spindle speed, but gear-change signal only if the gear range changes 4 = Analog output of the spindle speed and gear-change signal at every TOOL CALL 5 = Analog output of the spindle speed and no gear-change signal 6 = As for input value 3, but with controlled spindle for orientation 7 = As for input value 4, but with controlled spindle for orientation 8 = As for input value 5, but with controlled spindle for orientation	RESET	4-79
MP3020	Definition of the spindle speed range Input value: 0 to 99999		4-91
MP3030	Axis-halt on TOOL CALL with only a spindle speed output Input value: 0 or 1 1 = No axis halt on TOOL CALL 0 = Axis halt on TOOL CALL		4-87
MP3120	Zero spindle speed permitted Input value: 0 or 1 0 = 0 rpm permitted 1 = 0 rpm not permitted		4-82

Machine parameter	Function and input	Reaction / CN 123	Page
MP3130	Polarity of the S-analog voltage Input value: 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		4-80
MP3140	Count direction of the measuring system signals for the spindle Input value: 0 or 1 0 = Positive count direction with M03 1 = Negative count direction with M03		4-81
MP3210.0 to MP3210.7	S-analog voltage for gear range Input value: 0 to 9.999 [V]		4-82
MP3240.0	Maximum S-analog voltage output Input: 0 to 9.000 [V]		4-82
MP3240.1	Minimum S-analog voltage output Input value: 0 to 9.999 [V]		4-82
MP3240.2	Jog voltage for gear change Input value: 0 to 9.999 [V]		4-86
MP3310.0	Limit with S override (maximum) Input value: 0 to 150 [%]		4-84
MP3310.1	Limit with S override (minimum) Input value: 0 to 150 [%]		4-84
MP3410.0	Ramp slope for spindle for M03, M04, M05 Input range: 0 to 1.999 [V/ms]		4-82
MP3410.1	Ramp slope of the analog voltage for Spindle orientation Input value: 0 to 1.999 [V/ms]		4-93
MP3410.2	Ramp slope for tapping Input value: 0 to 1.999 [V/ms]		4-100
MP3410.3	Accelerations ramp for rigid tapping Input value: 0 to 1.999 [V/ms]		4-104
MP3420	Positioning window for spindle Input range: 0 to 65 535 [increments] 1 increment is equivalent to about 0.088°		4-95
MP3430	Reference mark deviation from the desired position (Spindle preset) Input value: 0 to 360 [°]		4-95
MP3440.0 to MP3440.7	k _v factor for spindle orientation Input range: 0.1 to 10 [m/min/mm]		4-95

Machine parameter	Function and input	Reaction / CN 123	Page
MP3510.0 to MP3510.7	Spindle speed for gear ranges Input value: 0 to 99 999.999 [rpm]		4-82
MP3520.0	Spindle speed activated by marker M2501 Input value: 0 to 99 999.999 [rpm]		4-96
MP3520.1	Spindle speed for spindle orientation Input value: 0 to 99 999.999 [rpm]		4-93

3.6 Integrated PLC

Machine parameter	Function and input	Reaction / CN 123	Page
MP4010	PLC program from RAM or from EPROM Input: 0 or 1 0 = EPROM operation 1 = RAM operation	RESET	7-21
MP4060.0 to MP4060.4	Path dependent lubrication Input range: 0 to 65 535 (units of 65 536 µm)		4-18
MP4110.0 to MP4110.47	Time for timers Input range: 0 to 65 535 [PLC cycles] (TNC 360: 20 ms)		7-19
MP4120.0 to MP4120.31	Preset value for counters C0 to C31 Input range: 0 to 65 535 [PLC cycles]		7-21
MP4130	Fast PLC input to suppress the monitoring functions Input value: 0 to 255 [No. of the PLC input]		4-68
MP4131	Activation condition for fast PLC input from MP4130 Input value: 0 or 1		4-68
MP4210.0 to MP4210.47	Set a number in the PLC (D768 to W 988) Input range: –99 999.9999 to +99 999.9999 [mm] or [°]		7-17
MP4220.0 to MP4220.3	Set a number in the PLC in the word range W960 to W966 Input: 0 to 65 535		7-17

Machine parameter	Function and input	Reaction / CN 123	Page
MP4230.0 to MP4230.31 to MP4239.0 to MP4239.31	Set a number in the PLC via module 9032 Input value: -30 000 000 to + 30 000 000		7-17
MP4310.0 to MP4310.6	Set a number in the PLC, in the Word range W976 to W988 Input range: 0 to 65 535		7-17

3.7 Setting the data interfaces

Machine parameter	Function and input	Reaction / CN 123	Page
MP5010.0	Control characters for "blockwise transfer" Input range: 0 to 32 382 Character for program end and beginning The character for program end also applies for "Standard data interface"	CN 123	8-26
MP5010.1	ASCII characters for data input	CN 123	8-27
MP5010.2	ASCII characters for data output	CN 123	8-27
MP5010.3	ASCII characters for beginning and end of the command block	CN 123	8-26
MP5010.4	ASCII characters for positive acknowledgment or negative acknowledgment	CN 123	8-26
MP5010.5	ASCII character for data transmission ended	CN 123	8-26

Machine parameter	Function and input	Reaction / CN 123	Page															
MP5020	Operating mode EXT: interface configuration Input range: 0 to 255 Bit 0 7 or 8 Data bits +0 = 7 Data bits +1 = 8 Data bits Bit 1 Block Check Character +0 = BCC character optional +2 = BCC character not control character Bit 2 Transmission stop through RTS +0 = not active +4 = active Bit 3 Transmission stop through DC3 +0 = not active +8 = active Bit 4 Character parity +0 = even +16 = odd Bit 5 Character parity +0 = not required +32 = required Bit 6/7 Stop bits <table><tr><td></td><td><u>Bit 6</u></td><td><u>Bit 7</u></td></tr><tr><td>+0 = 1½ Stop bits</td><td>0</td><td>0</td></tr><tr><td>+64 = 2 Stop bits</td><td>1</td><td>0</td></tr><tr><td>+128 = 1 stop bit</td><td>0</td><td>1</td></tr><tr><td>+192 = 1 stop bit</td><td>1</td><td>1</td></tr></table> Bit 8 RTS in quiescent state low Bit 9 No EOT after ETX received		<u>Bit 6</u>	<u>Bit 7</u>	+0 = 1½ Stop bits	0	0	+64 = 2 Stop bits	1	0	+128 = 1 stop bit	0	1	+192 = 1 stop bit	1	1	CN 123	8-28
	<u>Bit 6</u>	<u>Bit 7</u>																
+0 = 1½ Stop bits	0	0																
+64 = 2 Stop bits	1	0																
+128 = 1 stop bit	0	1																
+192 = 1 stop bit	1	1																
MP5030.0	Operating mode EXT: data transmission protocol Input: 0 or 1 0 = "Standard data transmission" 1 = "Blockwise transfer"	CN 123	8-28															
MP5040	Data transfer rate for PLC interface Input range: 0 to 9 <table><tr><td>0 = 110 baud</td><td>5 = 2 400 baud</td></tr><tr><td>1 = 150 baud</td><td>6 = 4 800 baud</td></tr><tr><td>2 = 300 baud</td><td>7 = 9 600 baud</td></tr><tr><td>3 = 600 baud</td><td>8 = 19 200 baud</td></tr><tr><td>4 = 1200 baud</td><td>9 = 38 400 baud</td></tr></table>	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	CN 123	8-42					
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																	

3.8 Measuring with a 3D touch probe

Machine parameter	Function and input	Reaction / CN 123	Page
MP6010	Selection of the touch probe system Input: 0 or 1 0 = TS 120 1 = TS 511	CN 123	4-152
MP6120	Probing feed rate Input range: 80 to 30 000 [mm/min]	CN 123	4-152
MP6130	Maximum measuring range Input range: 0.001 to 99 999.999 [mm]	CN 123	4-152
MP6140	Safety clearance above measuring point Input range: 0.001 to 99 999.999 [mm]	CN 123	4-152
MP6150	Rapid traverse for probe cycle Input range: 80 to 30 000 [mm/min]	CN 123	4-152

3.9 Digitizing with TS 120 (only with digitizing option)

Machine parameter	Function and input	Reaction / CN 123	Page
MP6210	Number of oscillations per second in normal direction Input range: 0 to 65.535 [1/s]		4-156
MP6230	Feed rate in normal direction Input range: 0 to 10 000 [mm/min]		4-156
MP6240	Maximum deflection of stylus Input range: 0 to 10.000 [mm]		4-156
MP6250	Numerical value for M function for transferring modal values Input 0 to 89 0 = no transfer 1 ... 89 number of the M function		4-157
MP6260	Output of M90 for NC blocks with digitized data Input: 0 or 1 0 = no output of M90 1 = output of M90 in every NC block		4-157
MP6270	Rounding of trailing zeros Input: 0 to 1 0 = Output in 0.001 mm increments (1 µm) 1 = Output in 0.01 mm increments (10 µm)		4-157

3.10 Tapping

Machine parameter	Function and input	Reaction / CN 123	Page
MP7000	Acknowledgment of display "Power interruption" 0 = manual with "CE" 1 = automatic after 3 sec (for positioning module)		–
MP7110.0	Minimum for feed override when tapping Input range: 0 to 150 [%]		4-100
MP7110.1	Maximum for feed override when tapping Input range: 0 to 150 [%]		4-100
MP7120.0	Dwell time for change of direction of spindle rotation Input range: 0 to 65.535 [s]		4-100
	Advanced switch time of spindle for "tapping" cycle (only active with BCD output of spindle rpm) Input range: 0 to 65.535 [s]		4-102
MP7120.2	Spindle run-on time after reaching total hole depth Input range: 0 to 65.535 [s]		4-100
MP7130	Spindle run-in characteristic Input range: 0.001 to 10 [°/min] (matching MP1520)		4-104
MP7140	Transient response of spindle on acceleration Input range: 0.001 to 1 (matching MP1530)		4-104
MP7150	Positioning window for tool axis Input range: 0.001 to 2 [mm]		4-104
MP7160	Suppress spindle orientation at the beginning of Cycle 17 Input: 0 or 1 0 = Orient the spindle 1 = Suppress spindle orientation		4-104

3.11 Display and operation

Machine parameter	Function and input	Reaction / CN 123	Page
MP7210	Programming station Input values: 0 to 2 0 = Control and programming 1 = Programming station, "PLC active" 2 = Programming station "PLC inactive"	CN 123 RESET	4-134

Machine parameter	Function and input	Reaction / CN 123	Page
MP7215	Contouring control or point-to-point and straight-cut control Input range 0 to 3 0 = Contouring control (TNC 360) 1 = Point-to-point and straight cut (TNC 335) 2 = Position display inhibited, otherwise same as 1	CN 123	4-246
MP7220	Block number step size for ISO programs Input range: 0 to 255	CN 123	4-132
MP7222	Programming language selection: dialog or ISO Input: 0 to 2 0 = Selection via MOD 1 = Only dialog 2 = Only ISO	CN 123	4-132
MP7225	Generating blocks with the "ACTUAL POSITION CAPTURE" key (not in ISO programming) Input: 0 to 3 0 = no block generation possible 1 = block generation with the "ACTUAL POSITION CAPTURE" key 2 = block generation with M2829		4-121
MP7230	Switching the dialogue language Input: 0 or 1 0 = First dialog language 1 = Basic language (English)	CN 123	4-134
MP7235	Resetting the timing Input: 0 to 4 0 = No resetting 1 = Reset switch-on duration 2 = Reset run time of automatic operation 4 = Reset run time of spindle		4-118
MP7240	Inhibit program input for [Program name] = [OEM cycle number in EPROM]. Input: 0 or 1 0 = Inhibited 1 = Not inhibited		9-5
MP7245.0	Inhibit the HEIDENHAIN cycles 1 to 15 Input range: 0 to 65 534		4-126
MP7245.1	Inhibit the HEIDENHAIN standard cycles 16 to 30 Input range: 0 to 65 535		4-127
MP7250	Difference between Q parameter number for "CALL-active" and "DEF-active" block in OEM cycles Input 0 to 50		9-5
MP7251	Number of global Q parameters that can be transferred from OEM cycle to the calling program Input: 0 to 100		9-5

Machine parameter	Function and input	Reaction / CN 123	Page
MP7260	Number of tools in the tool table Input: 0 to 99 0 = No central tool file 1 - 99 number of tools, tool file is active	CN 123 RESET	4-196
MP7261	Number of pockets in the tool magazine Input: 0 to 99	CN123 RESET	4-214
MP7264	Number of pockets reserved next to special tool Input: 1 to 3	CN 123 RESET	4-197
MP7270	Display of the feed rates in manual operating modes (Manual operation, Electronic handwheel) Input: 0 or 1 Bit 0 = 0 Display of the axis feed rate only when an axis direction key is pressed Bit 0 = 1 Display of the axis feed rate before operating an axis-direction key (smallest value from MP1020.X for all axes) Bit 1 = 1 Abortion after NC stop in manual operating mode (control-in-operation symbol off)	CN 123	4-116
MP7274	Expanded spindle display Input: 0 to 3 Bit 0 Display of current gear range +0 No display +1 Display Bit 1 Display of spindle power +0 Acquisition via analog input +2 Acquisition via PLC word W600		4-120
MP7275	Reference value of displayed spindle power for acquisition via analog input Input: 0 to 4.999 [V] for acquisition via word W600 Input: 0 to 43 690 Input value is equivalent to 100%		4-120
MP7280	Decimal sign Input: 0 or 1 0 = Decimal comma 1 = Decimal point	CN 123	4-134
MP7285	Calculation of the tool length for the position display of the tool axis Input: 0 or 1 0 = Tool length is not calculated 1 = Tool length is calculated	CN 123	4-113

MP7290	Position-display step Input: 1 to 4 0 = 0.001 mm or 0.001° 1 = 0.005 mm or 0.005° 2 = 0.010 mm or 0.01° 3 = 0.050 mm or 0.05° 4 = 0.100 mm or 0.1°		4-113
MP7300	Cancel status display, Q parameters, and tool data Input: 0 to 7 0 = Status display, Q parameters and tool data canceled through program selection 1 = Status display, Q parameters and tool data canceled through M02, M30, END PGM and program selection 2 = Status display and tool data canceled through program selection 3 = Status display and tool data canceled through M02, M30, END PGM and program selection 4 = Status display and Q parameters canceled through program selection 5 = Status display and Q parameters canceled through M02, M30, END PGM and program selection 6 = Status display canceled through program selection 7 = Status display canceled through M02, M30, END PGM and program selection	CN 123	4-119
MP7310	Graphics display Input range: 0 to 3 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 + 2 = coordinate system rotated by +90°	CN 123	4-112
MP7330.0 to MP7330.15	Determination of the User Parameters Input range: 0 to 9999.00 (Number of the desired machine parameter)		4-133
MP7340.0 to MP7340.15	Dialogues for User Parameters Input range: 0 to 199 0 to 99 from PLC error messages 100 to 199 from dialogue texts		4-133

3.12 Machining and program run

Machine parameter	Function and input	Reaction / CN 123	Page
MP7410	"Scaling factor" cycle active in two or three axes Input: 0 or 1 0 = Cycle "Scaling factor" operates in all three principle axes 1 = Cycle "Scaling factor" only operates in the machining plane	CN 123	4-130
MP7411	Compensation data for TOUCH PROBE block Input: 0 or 1 0 = Compensation value from the calibration process 1 = Compensation value from the tool file		4-152
MP7420	Cycles for milling pockets with free-programmed contours Input range: 0 to 15 Bit 0 Slot milling direction for + 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 + 2 = First clear out pocket, then slot milling Bit 2 Merge programmed contours + 0 = Contours merged only if the tool center paths intersect + 4 = Contours merged if the programmed contours overlap Bit 3 Clear out and slot milling to pocket depth, or for each feed + 0 = Clearing out and slot milling performed in one operation for all feeds + 8 = For each peck, first perform slot-milling and then feed clearing out (depending on Bit 1) before next peck	CN 123	4-129
MP7430	Overlap factor for pocket milling Input: = 0.1 to 1.414	CN 123	4-128

Machine parameter	Function and input	Reaction / CN 123	Page
MP7440	Output of M functions Input range: 0 to 7 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 + 2 = modal cycle call M89 at end of block Bit 2 Program halt on M functions + 0 = Program halt until acknowledgment of M function + 4 = No program halt (does not wait for acknowledgment) Bit 3 and Bit 4: no function Bit 5 + 0 + 1 = Setting the marker axis in position between 2 NC blocks, if there is delay time here (e.g. programmed dwell time)	CN 123	4-140
MP7460	Constant contouring speed at corners Input: 0.001 to 179.999°	CN 123	4-67
MP7470	Display mode for rotary axis (not for PLC help axes) Input: 0 to 3 Bit 0 Display mode +0 = 0 to +359.999 [°] (no software limit switch) +1 = -30 000.000 to +30 000.000 (software limit switch active) Bit 1 Positioning the rotary axis +0 = according to the sign +2 = always by the shortest path (useful only when Bit 0 = 0)	CN 123	4-114
MP7480.0	Output of tool or pocket number in TOOL CALL block Input: 0 to 3 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		4-198

Machine parameter	Function and input	Reaction / CN 123	Page
MP7480.1	Output of tool or pocket number in TOOL DEF block Input: 0 to 3 0 = No output 1 = Tool number output only after tool number change (W262) 2 = Tool number output with each TOOL DEF block (W262) 3 = Output of pocket number (W262) and tool number (W264) only after tool number change		4-199

3.13 Hardware

Machine parameter	Function and input	Reaction / CN 123	Page
MP7620	Feed rate and spindle override Input: 0 to 15 Bit 0 Feed rate override is rapid traverse button is pressed in the "program run" mode +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 +4 = Override active Bit 3 Spindle and feed rate override in 1 % increments or according to a non-linear characteristic curve +0 = 1 % increments +8 = non-linear characteristic curve	CN 123	4-84 4-116
MP7640	Handwheel Input: 0 to 6 0 = No handwheel 1 = HR 330 2 = HR 130 3 = HR 330 with evaluation of keys via PLC 4 = HR 332 with evaluation of keys via PLC 5 = Up to three HR 150 via HRA 110 6 = HR 410	CN 123	4-167

Machine parameter	Function and input	Reaction / CN 123	Page																											
MP7641	Entry of interpolation factor and handwheel activation Input: 0 to 15 Bit 0 = 0 Entry of interpolation factor via TNC keyboard = 1 Entry of interpolation factor via PLC module 9036 Bit 1 = 1 Handwheel active in the Programming and editing mode Bit 2 = 1 Entry of interpolation factor via PLC module 9036 and TNC keyboard Bit 3 = 1 Polling handwheel keys from the PLC in every operating mode		4-107 4-172																											
MP7645.0	Assignment of handwheel keys for HR 332 if MP7640 = 4: Input value: 0 or 1 0 = All keys and LEDs, except axis selection keys and their LEDs, are selectable via PLC 1 = All 12 keys and LEDs are selectable via PLC If MP 7640 = 5: Assignment of 3rd handwheel via axis selection switch Input value: 0 to 2 <table><thead><tr><th></th><th><u>Switch</u></th><th><u>3rd handwheel</u></th></tr></thead><tbody><tr><td>0 =</td><td>1 (left stop)</td><td>Z axis</td></tr><tr><td></td><td>2</td><td>4th axis</td></tr><tr><td>1 =</td><td>1 (left stop)</td><td>X axis</td></tr><tr><td></td><td>2</td><td>Y axis</td></tr><tr><td></td><td>3</td><td>Z axis</td></tr><tr><td></td><td>4</td><td>4th axis</td></tr><tr><td>2 =</td><td>3</td><td>Z axis</td></tr><tr><td></td><td>4</td><td>4th axis</td></tr></tbody></table>		<u>Switch</u>	<u>3rd handwheel</u>	0 =	1 (left stop)	Z axis		2	4th axis	1 =	1 (left stop)	X axis		2	Y axis		3	Z axis		4	4th axis	2 =	3	Z axis		4	4th axis		4-169 4-172
	<u>Switch</u>	<u>3rd handwheel</u>																												
0 =	1 (left stop)	Z axis																												
	2	4th axis																												
1 =	1 (left stop)	X axis																												
	2	Y axis																												
	3	Z axis																												
	4	4th axis																												
2 =	3	Z axis																												
	4	4th axis																												
MP7645.1	Assignment of 3rd handwheel via machine parameter Input: 0 to 8 0 = Assignment of 3rd handwheel via MP7645.0 1 = 3rd handwheel X axis 2 = 3rd handwheel Y axis 4 = 3rd handwheel Z axis 8 = 3rd handwheel 4th axis		4-173																											
MP7645.2	Assignment of 3rd handwheel via axis selection switch or MP7645.1 Input: 0 to 1 0 = Assignment via axis selection switch according to MP 7645.0 1 = Assignment via MP 7645.1		4-173																											

Machine parameter	Function and input	Reaction / CN 123	Page
MP7650	Count direction for handwheel Input: 0 or 1 0 = Positive count direction 1 = Negative count direction		4-167
MP7660	Threshold sensitivity for electronic handwheel Input value: 0 to 65 535 [increments]		4-167
MP7670	Minimum interpolation factor for handwheel Input range: 0 to 10		4-167
MP7670	Interpolation factor for handwheel HR410 Input: 0 to 10		
MP7670.0	Interpolation factor at low speed		
MP7670.1	Interpolation factor at medium speed		
MP7670.2	Interpolation factor at high speed		
MP7671	Manual feed rate in "handwheel" operation mode with HR 410 Input: 0 to 1000 [% to MP1020]		
MP7671.0	Low speed		
MP7671.1	Medium speed		
MP7671.2	High speed		
MP7680	Memory function for axis-direction keys 0 = not stored 1 = stored		4-149
MP7690	Memory test at switch-on Input range: 0 to 3 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 + 2 = No memory test at switch-on		4-135

1 List of Markers

The markers listed in *italics* have been retained to insure 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	PLC	4-74
M2001	Axis enable Y	NC	PLC	4-74
M2002	Axis enable Z	NC	PLC	4-74
M2003	Axis enable 4	NC	PLC	4-74
M2004	S-analogue voltage not on ramp	NC	NC	4-82
M2005	S-analogue voltage = 0 V	NC	NC	4-82
M2007	Spindle in position	NC	NC	4-95
M2008	X axis in position	NC	NC	4-75
M2009	Y axis in position	NC	NC	4-75
M2010	Z axis in position	NC	NC	4-75
M2011	4th axis in position	NC	NC	4-75
M2012	Lubrication pulse X axis, since value of MP4060.0 was exceeded	NC	NC	4-18
M2013	Lubrication pulse Y axis, since value of MP4060.1 was exceeded	NC	NC	4-18
M2014	Lubrication pulse Z axis, since value of MP4060.2 was exceeded	NC	NC	4-18
M2015	Lubrication pulse 4th axis, since value of MP4060.3 was exceeded	NC	NC	4-18
M2022	Touch probe not ready (no standby signal at connector X12)	NC	NC	4-153
M2023	Stylus deflected before start of probing cycle	NC	NC	4-153
M2025	Stylus deflected (probing sequence is executed)	NC	PLC	4-153
M2026	Probe operation ended or interrupted	NC	NC	4-153
M2027	Battery voltage too low (battery warning on connector X12); evaluated only during the probe operation	NC	NC	4-153
<i>M2032</i>	<i>T Code 1st bit (lsb)</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
<i>M2033</i>	<i>T Code 2nd bit</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
<i>M2034</i>	<i>T Code 3rd bit</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
<i>M2035</i>	<i>T Code 4th bit</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
<i>M2036</i>	<i>T Code 5th bit</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
<i>M2037</i>	<i>T Code 6th bit</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
<i>M2038</i>	<i>T Code 7th bit</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
<i>M2039</i>	<i>T Code 8th bit (msb)</i>	<i>NC</i>	<i>NC</i>	<i>7-123</i>
M2040	The control functions as point-to-point and straight-cut control	NC	NC	4-246
M2041	Basic language (English) is selected	NC	NC	4-134
M2042	Analogue output of the spindle speed	NC	NC	4-80
M2043	Change signal for gear code	NC	NC	4-86
M2044	Change signal for S code	NC	NC	4-91
M2045	Change signal for M function	NC	NC	4-138

Marker	Function	Set	Reset	Page
M2046	Strobe signal T code (P code) with TOOL CALL	NC	NC	4-199
M2047	Strobe signal 2nd T code (2nd P code) with TOOL DEF	NC	NC	4-199
M2048	Tapping cycle called	NC	NC	4-100
M2049	Parallel editing	NC	NC	–
M2050	<i>Operating mode: Programming and editing</i>	NC	NC	7-123
M2051	<i>Operating mode: Manual</i>	NC	NC	7-123
M2052	<i>Operating mode: Electronic handwheel</i>	NC	NC	7-123
M2053	<i>Operating mode: Positioning with manual data input</i>	NC	NC	7-123
M2054	<i>Operating mode: Program run/single block</i>	NC	NC	7-123
M2055	<i>Operating mode: Program run/full sequence</i>	NC	NC	7-123
M2056	<i>Operating mode: Test run</i>	NC	NC	7-123
M2057	<i>Operating mode: Traversing the reference marks</i>	NC	NC	7-123
M2058	Repeat last operating mode key	NC	NC	–
M2060	ISO program is selected	NC	NC	4-132
M2061	END PGM, M02 or M30 has been executed	NC	NC	4-135
M2063	"PGM 0" is selected	NC	NC	4-199
M2064	<i>S Code 1st bit (lsb)</i>			7-123
M2065	<i>S Code 2nd bit</i>			7-123
M2066	<i>S Code 3rd bit</i>			7-123
M2067	<i>S Code 4th bit</i>			7-123
M2068	<i>S Code 5th bit</i>			7-123
M2069	<i>S Code 6th bit</i>			7-123
M2070	<i>S Code 7th bit</i>			7-123
M2071	<i>S Code 8th bit (msb)</i>			7-123
M2072	<i>M Code 1st bit (lsb)</i>			7-123
M2073	<i>M Code 2nd bit</i>			7-123
M2074	<i>M Code 3rd bit</i>			7-123
M2075	<i>M Code 4th bit</i>			7-123
M2076	<i>M Code 5th bit</i>			7-123
M2077	<i>M Code 6th bit</i>			7-123
M2078	<i>M Code 7th bit</i>			7-123
M2079	<i>M Code 8th bit (msb)</i>			7-123
M2080	<i>Minimum rpm from MP3020 1st bit (lsb)</i>			7-123
M2081	<i>Minimum rpm from MP3020 2nd bit</i>			7-123
M2082	<i>Minimum rpm from MP3020 3rd bit</i>			7-123
M2083	<i>Minimum rpm from MP3020 4th bit</i>			7-123
M2084	<i>Minimum rpm from MP3020 5th bit</i>			7-123
M2085	<i>Minimum rpm from MP3020 6th bit</i>			7-123
M2086	<i>Minimum rpm from MP3020 7th bit</i>			7-123
M2087	<i>Minimum rpm from MP3020 8th bit (msb)</i>			7-123
M2088	<i>Increment from MP3020 1st bit (lsb)</i>			7-123
M2089	<i>Increment from MP3020 2nd bit</i>			7-123
M2090	<i>Increment from MP3020 3rd bit</i>			7-123
M2091	<i>Increment from MP3020 4th bit (msb)</i>			7-123
M2092	Illegal spindle speed	NC	NC	4-87
M2093	Another T code (P code) follows with TOOL CALL	NC	NC	4-214
M2095	Rigid tapping is active	NC	NC	4-104

Marker	Function	Set	Reset	Page
M2096	X key last pressed	NC	NC	4-113
M2097	Y key last pressed	NC	NC	4-113
M2098	Z key last pressed	NC	NC	4-113
M2099	Key IV last pressed	NC	NC	4-113
M2100	X axis is tool axis	NC	NC	4-13
M2101	Y axis is tool axis	NC	NC	4-13
M2102	Z axis is tool axis	NC	NC	4-13
M2103	4th axis is tool axis	NC	NC	4-13
M2104	G Code S Analog 1st bit (lsb)	PLC	PLC	4-86
M2105	G Code S Analog 2nd bit	PLC	PLC	4-86
M2106	G Code S Analog 3rd bit (msb)	PLC	PLC	4-86
M2112	<i>T Number (P Number) 1st decade (lsb)</i>			7-124
M2113	<i>T Number (P Number) 1st decade</i>			7-124
M2114	<i>T Number (P Number) 1st decade</i>			7-124
M2115	<i>T Number (P Number) 1st decade (msb)</i>			7-124
M2116	<i>T Number (P Number) 2nd decade (lsb)</i>			7-124
M2117	<i>T Number (P Number) 2nd decade</i>			7-124
M2118	<i>T Number (P Number) 2nd decade</i>			7-124
M2119	<i>T Number (P Number) 2nd decade (msb)</i>			7-124
M2128	X axis in motion	NC	NC	4-76
M2129	Y axis in motion	NC	NC	4-76
M2130	Z axis in motion	NC	NC	4-76
M2131	4th axis in motion	NC	NC	4-76
M2136	X axis reference mark not yet traversed	NC	NC	–
M2137	Y axis reference mark not yet traversed	NC	NC	–
M2138	Z axis reference mark not yet traversed	NC	NC	–
M2139	4th axis reference mark not yet traversed	NC	NC	–
M2149	Strobe marker for Q parameter transfer with FN19	NC	NC	–
M2150	Unit of measure of the NC program (0 = mm, 1 = inch)	NC	NC	–
M2160	Traverse direction X axis 0 = positive 1 = negative	NC	NC	4-7
M2161	Traverse direction Y axis 0 = positive 1 = negative	NC	NC	4-7
M2162	Traverse direction Z axis 0 = positive 1 = negative	NC	NC	4-7
M2163	Traverse direction 4th axis 0 = positive 1 = negative	NC	NC	4-7
M2176	<i>Code operating mode (lsb)</i>			7-124
M2177	<i>Code operating mode</i>			7-124
M2178	<i>Code operating mode</i>			7-124
M2179	<i>Code operating mode (msb)</i>			7-124
M2180	1st PLC scan after switch-on	NC	NC	–
M2182	Inhibited key was operated	NC	PLC	4-141
M2183	Program interruption ("Control operational" display flashes)	NC	NC	4-118
M2184	Control operational ("Control operational" display goes on or flashes)	NC	NC	4-118
M2185	1st PLC scan after interruption of the PLC program	NC	NC	–
M2186	<i>Code number 84159 entered</i>	NC	PLC	7-124
M2190	Non-flashing error message is displayed	NC	NC	4-122
M2191	"External EMERGENCY STOP" message is displayed	NC	NC	4-106

Marker	Function	Set	Reset	Page
M2192 to M2239	Markers controllable by MP4310.0, MP 4310.1 and MP 4310.2	NC	NC	7-124
M2240	Inhibit OEM cycle 68	PLC	PLC	9-5
M2241	Inhibit OEM cycle 69	PLC	PLC	9-5
M2242	Inhibit OEM cycle 70	PLC	PLC	9-5
M2243	Inhibit OEM cycle 71	PLC	PLC	9-5
M2244	Inhibit OEM cycle 72	PLC	PLC	9-5
M2245	Inhibit OEM cycle 73	PLC	PLC	9-5
M2246	Inhibit OEM cycle 74	PLC	PLC	9-5
M2247	Inhibit OEM cycle 75	PLC	PLC	9-5
M2248	Inhibit OEM cycle 76	PLC	PLC	9-5
M2249	Inhibit OEM cycle 77	PLC	PLC	9-5
M2250	Inhibit OEM cycle 78	PLC	PLC	9-5
M2251	Inhibit OEM cycle 79	PLC	PLC	9-5
M2252	Inhibit OEM cycle 80	PLC	PLC	9-5
M2253	Inhibit OEM cycle 81	PLC	PLC	9-5
M2254	Inhibit OEM cycle 82	PLC	PLC	9-5
M2255	Inhibit OEM cycle 83	PLC	PLC	9-5
M2256	Inhibit OEM cycle 84	PLC	PLC	9-5
M2257	Inhibit OEM cycle 85	PLC	PLC	9-5
M2258	Inhibit OEM cycle 86	PLC	PLC	9-5
M2259	Inhibit OEM cycle 87	PLC	PLC	9-5
M2260	Inhibit OEM cycle 88	PLC	PLC	9-5
M2261	Inhibit OEM cycle 89	PLC	PLC	9-5
M2262	Inhibit OEM cycle 90	PLC	PLC	9-5
M2263	Inhibit OEM cycle 91	PLC	PLC	9-5
M2264	Inhibit OEM cycle 92	PLC	PLC	9-5
M2265	Inhibit OEM cycle 93	PLC	PLC	9-5
M2266	Inhibit OEM cycle 94	PLC	PLC	9-5
M2267	Inhibit OEM cycle 95	PLC	PLC	9-5
M2268	Inhibit OEM cycle 96	PLC	PLC	9-5
M2269	Inhibit OEM cycle 97	PLC	PLC	9-5
M2270	Inhibit OEM cycle 98	PLC	PLC	9-5
M2271	Inhibit OEM cycle 99	PLC	PLC	9-5
M2400	Tool number 0 programmed	NC	NC	4-199
M2401	Active tool with pocket number (MP7260/MP7261)	NC	NC	4-214
M2402	Active tool without pocket number (MP7260/MP7261)	NC	NC	4-214
M2403	Active tool = special tool	NC	NC	4-214
M2408	Cycle 13 is executed			
M2448	NC start (edge evaluation)	PLC	PLC	4-149
M2449	Rapid traverse	PLC	PLC	4-149
M2450	Memory function for axis-direction keys	PLC	PLC	4-149
M2451	Feed release for all axes	PLC	PLC	4-74
M2452	Activate PLC positioning X axis	PLC	NC	7-124
M2453	Activate PLC positioning Y axis	PLC	NC	7-124
M2454	Activate PLC positioning Z axis	PLC	NC	7-124
M2455	Activate PLC positioning 4th axis	PLC	NC	7-124

Marker	Function	Set	Reset	Page
M2456	Manual traverse X+	PLC	PLC	4-149
M2457	Manual traverse X-	PLC	PLC	4-149
M2458	Manual traverse Y+	PLC	PLC	4-149
M2459	Manual traverse Y-	PLC	PLC	4-149
M2460	Manual traverse Z+	PLC	PLC	4-150
M2461	Manual traverse Z-	PLC	PLC	4-150
M2462	Manual traverse 4+	PLC	PLC	4-150
M2463	Manual traverse 4-	PLC	PLC	4-150
M2464	Complement – NC start	PLC	PLC	4-149
M2465	Complement – rapid traverse	PLC	PLC	4-149
M2466	Complement – memory function for axis-direction keys	PLC	PLC	4-149
M2467	Complement – feed release	PLC	PLC	4-75
M2468	<i>Complement PLC positioning X axis</i>	<i>NC</i>	<i>PLC</i>	<i>7-124</i>
M2469	<i>Complement PLC positioning Y axis</i>	<i>NC</i>	<i>PLC</i>	<i>7-124</i>
M2470	<i>Complement PLC positioning Z axis</i>	<i>NC</i>	<i>PLC</i>	<i>7-124</i>
M2471	<i>Complement PLC positioning 4th axis</i>	<i>NC</i>	<i>PLC</i>	<i>7-124</i>
M2472	Complement – manual traverse X+	PLC	PLC	4-149
M2473	Complement – manual traverse X-	PLC	PLC	4-149
M2474	Complement – manual traverse Y+	PLC	PLC	4-149
M2475	Complement – manual traverse Y-	PLC	PLC	4-149
M2476	Complement – manual traverse Z+	PLC	PLC	4-150
M2477	Complement – manual traverse Z-	PLC	PLC	4-150
M2478	Complement – manual traverse 4+	PLC	PLC	4-150
M2479	Complement – manual traverse 4-	PLC	PLC	4-150
M2480	Acknowledgment "Gear change completed"	PLC	PLC	4-86
M2481	S code acknowledgment	PLC	PLC	4-90
M2482	Acknowledgment of M function	PLC	PLC	4-138
M2483	Acknowledgment T code (P code) with TOOL CALL	PLC	PLC	4-199
M2484	Acknowledgment T code (P code) with TOOL DEF	PLC	PLC	4-199
M2485	Status display and sign of S analogue for M03	PLC	PLC	4-81
M2486	Status display and sign of S analogue for M04	PLC	PLC	4-81
M2487	Status display for M05 and spindle stop	PLC	PLC	4-81
M2488	NC stop ("0" signifies stop)	PLC	PLC	4-149
M2489	Change direction of spindle rotation	PLC	PLC	4-81
M2490	Spindle rotation left (for gear change)	PLC	PLC	4-86
M2491	Spindle rotation right (for gear change)	PLC	PLC	4-86
M2492	Await open control loop X axis	PLC	PLC	4-77
M2493	Await open control loop Y axis	PLC	PLC	4-77
M2494	Await open control loop Z axis	PLC	PLC	4-77
M2495	Await open control loop 4th axis	PLC	PLC	4-77
M2496	Enable marker for the decoded M code transfer to markers M1900 to M1999	PLC	PLC	4-138
M2497	Activate the edge evaluation for PLC inputs Rising-edge marker M1500 to M1659 Falling-edge marker M1700 to M1859	PLC	PLC	7-21
M2498	Release jog positioning	PLC	PLC	4-178
M2499	Open spindle control loop	PLC	PLC	4-93

Marker	Function	Set	Reset	Page
M2501	Activates spindle speed MP3520.0 and direction of rotation from marker M2656	PLC	PLC	4-96
M2502	NC STOP in all operating modes if stylus is deflected	PLC	PLC	4-153
M2503	Enable marker for probing functions	NC	PLC	4-153
M2508	Status display M08 or M09 0 = Status display M09 1 = Status display M08	PLC	PLC	4-117
M2510	Spindle potentiometer fixed to 100%	PLC	PLC	–
M2511	Feed rate potentiometer fixed to 100%	PLC	PLC	–
M2512	Start jog positioning axis X+	PLC	PLC	4-178
M2513	Start jog positioning axis X–	PLC	PLC	4-178
M2514	Start jog positioning axis Y+	PLC	PLC	4-178
M2515	Start jog positioning axis Y–	PLC	PLC	4-178
M2516	Start jog positioning axis Z+	PLC	PLC	4-178
M2517	Start jog positioning axis Z–	PLC	PLC	4-178
M2518	Start jog positioning axis 4+	PLC	PLC	4-178
M2519	Start jog positioning axis 4–	PLC	PLC	4-178
M2527	<i>Activate spindle orientation</i>	PLC	NC	7-124
M2528	Complement jog positioning axis X+	PLC	PLC	4-178
M2529	Complement jog positioning axis X–	PLC	PLC	4-178
M2530	Complement jog positioning axis Y+	PLC	PLC	4-178
M2531	Complement jog positioning axis Y–	PLC	PLC	4-178
M2532	Complement jog positioning axis Z+	PLC	PLC	4-178
M2533	Complement jog positioning axis Z–	PLC	PLC	4-178
M2534	Complement jog positioning axis 4+	PLC	PLC	4-178
M2535	Complement jog positioning axis 4–	PLC	PLC	4-178
M2543	<i>Complement spindle orientation</i>	NC	PLC	7-124
M2544	Open control loop X axis	PLC	PLC	4-77
M2545	Open control loop Y axis	PLC	PLC	4-77
M2546	Open control loop Z axis	PLC	PLC	4-77
M2547	Open control loop 4th axis	PLC	PLC	4-77
M2548	Reset of accumulated distance for lubrication X axis	PLC	PLC	4-18
M2549	Reset of accumulated distance for lubrication Y axis	PLC	PLC	4-18
M2550	Reset of accumulated distance for lubrication Z axis	PLC	PLC	4-18
M2551	Reset of accumulated distance for lubrication 4th axis	PLC	PLC	4-18
M2552	Actual – nominal value transfer X axis	PLC	PLC	4-77
M2553	Actual – nominal value transfer Y axis	PLC	PLC	4-77
M2554	Actual – nominal value transfer Z axis	PLC	PLC	4-77
M2555	Actual – nominal value transfer 4 axis	PLC	PLC	4-77
M2556	Reference end position for X axis	PLC	PLC	4-49
M2557	Reference end position for Y axis	PLC	PLC	4-49
M2558	Reference end position for Z axis	PLC	PLC	4-49
M2559	Reference end position for axis 4	PLC	PLC	4-49

Marker	Function	Set	Reset	Page
M2560 to M2589	BCD numerical values for PLC positioning, tool number, spindle orientation and Q parameters	PLC	PLC	7-124
M2594	Inhibit editing of central tool file (L and R possible)	PLC	PLC	–
M2597	Tool number: output mode 0 = binary 1 = BCD	PLC	PLC	–
M2600	Sequence of tool number or pocket number transfer (M2093 = 1)	PLC	PLC	4-214
M2601	Return special tool to its original pocket in spite of variable pocket coding	PLC	PLC	4-214
M2608	Inverse display of M03, M04, M05 and S analogue = 0 V	PLC	PLC	4-117
M2609	Status display M08, M09 inverse	PLC	PLC	4-117
M2610	Tool number shown inverted in status display	PLC	PLC	–
M2611	Acknowledgment of Q parameter transfer with FN19	PLC	PLC	–
M2612	Suppress exchange of pocket number	PLC	PLC	4–200
M2614	Suppress PLC functions (M/S/T/Q strobe) by the NC	PLC	PLC	–
M2615	Re-evaluate reference mark of the spindle	PLC	NC	
M2624	Limit switch X+	NC	NC	4-17
M2625	Limit switch X–	NC	NC	4-17
M2626	Limit switch Y+	NC	NC	4-17
M2627	Limit switch Y–	NC	NC	4-17
M2628	Limit switch Z+	NC	NC	4-17
M2629	Limit switch Z–	NC	NC	4-17
M2630	Limit switch 4+	NC	NC	4-17
M2631	Limit switch 4–	NC	NC	4-17
M2656	Spindle orientation from stop 0 = Orientation with M03 1 = Orientation with M04	PLC	PLC	4-93
M2688	No monitoring X axis	PLC	PLC	4-68
M2689	No monitoring Y axis	PLC	PLC	4-68
M2690	No monitoring Z axis	PLC	PLC	4-68
M2691	No monitoring axis 4	PLC	PLC	4-68
M2704	Activate PLC positioning X axis	PLC	NC	4-33
M2705	Activate PLC positioning Y axis	PLC	NC	4-33
M2706	Activate PLC positioning Z axis	PLC	NC	4-33
M2707	Activate PLC positioning 4th axis	PLC	NC	4-33
M2712	Activate PLC positioning for spindle orientation	PLC	NC	4-93
M2713	Activate the transfer of the value from D528 to the Q Parameter defined in W516	PLC	NC	4-135
M2714	Strobe marker write to tool file	PLC	NC	–
M2715	Strobe marker read from tool file	PLC	NC	–
M2716	Strobe marker for datum correction	PLC	NC	4-194
M2719	Deactivate the TNC 355 mode	PLC	PLC	7-121

Marker	Function	Set	Reset	Page
M2800 to M2807	Key code	PLC	PLC	7-124
M2808	Strobe marker for key code	PLC	PLC	7-124
M2809	Activate numerical value transfer from PLC to NC	PLC	PLC	7-124
M2810	Data format of a numerical value in M2560 to M2576			7-124
M2811	Data format of a numerical value in M2560 to M2576			7-124
M2812	Data format of a numerical value in M2560 to M2576			7-124
M2813	Activate the key from W516	PLC	NC	4-139
M2814	Activation of a gear range and spindle speed via PLC	PLC	NC	4-86
M2815	Flashing PLC error message	PLC	PLC	4-122
M2816	Select traverse range	PLC	PLC	4-16
M2817	Select traverse range	PLC	PLC	4-16
M2818	Q number (msb)	PLC	PLC	7-124
M2819	Activate datum correction	PLC	NC	7-124
M2824	Activation of the selected traverse range (M2816/M2817)	PLC	NC	4-16
M2825	External EMERGENCY OFF via PLC	PLC	PLC	–
M2826	Suppress handwheel pulses	PLC	PLC	4-167
M2827	Suppress EMERGENCY OFF, open the control loop, NC stop	PLC	PLC	–
M2829	NC block generation	PLC	NC	4-121
M2832 to M2839	Key code of the pressed inhibited key	NC	NC	7-124
M2855 to M2923	Inhibit keys	PLC	PLC	4-142
M2924 to M3023	Activate PLC error messages	PLC	NC; PLC	4-122
M3168	Overflow during multiplication	NC	PLC	7-59
M3169	Division by 0	NC	PLC	7-60
M3170	MODULO incorrectly executed	NC	PLC	7-61
M3171	M3171 is set when transfer incorrectly executed			7-103 - 118
M3172	Reserved for errors that the PLC programmer would like to intercept			–
M3200 to M3263	Values from MP4310.3 to MP4310.6			7-125

2 List of Words

Words	Function	Page
W256	G code for S analogue	4-86
W258	S code	4-91
W260	Code for M function	4-138
W262	Pocket number when MP7480 = 3 Tool number when MP7480 = 1 or 2	4-199
W264	Tool number when MP7480 = 3	4-199
W272	Operating mode 0 = Programming and editing 1 = Manual operation 2 = Electronic handwheel 3 = Positioning with manual entry 4 = Program run/single block 5 = Program run/full sequence 6 = Test run 7 = Pass over reference point	4-141
W274	Key code for the operated, inhibited key, Signal via M2182	4-141
D276	Code number entered in MOD mode	–
D280	1st value FN19	–
D284	2nd value FN19	–
W320	Nominal spindle speed	4-80
W322	Actual spindle speed	4-80
D356	Programmed spindle speed (NC → PLC)	4-80
D360	Programmed feed rate (NC → PLC)	–
D388	Current feed rate (NC → PLC)	–
W392	Analog voltage of the analog input (0 to 5 V)	–
W492	% factor – spindle override (NC → PLC)	4-84
W494	% factor – feed rate override (NC → PLC)	4-116
W512	T code	–
W514	P code	–
W516	Word with multiple function Key code for simulation of TNC keys Activate with M2813 Number of the Q parameter to be overwritten (Q100 to Q107 = 0 to 7)	4-135
W522	Monitoring functions suppressed if PLC input from MP4130 is activated	4-68
D528	Value to be transferred to the Q parameter Datum correction for X axis Position of X axis [1/1000 mm]	4-135 4-194 4-32
D532	Datum correction for Y axis Position Y axis	4-194 4-32
D536	Datum correction for Z axis Position Z axis	4-194 4-32
D540	Datum correction for 4th axis Position 4th axis	4-194 4-32
W560	Feed rate X axis [mm/min]	4-33
W562	Feed rate Y axis	4-33
W564	Feed rate Z axis	4-33

Words	Function	Page
W566	Feed rate 4th axis	4-33
W576 to W582	Follow up axis error compensation Input range: +32 767 to -32 768 [µm]	4-29
W576	Axis X	
W578	Axis Y	
W580	Axis Z	
W582	Axis 4	
D592	Position spindle orientation (Strobe M2712)	4-93
D596	Feed rate from PLC for feed rate limitation	4-116
W600	Value for spindle power display	4-120
W602	PLC text window	4-121
W604	Reserved	
W608	Axis specific inhibition of the reference pulse	–
D756	Set spindle speed from the PLC; programmed spindle speed	4-86
W764	% factor – spindle override (PLC → NC)	4-84
W766	% factor – feed rate override (PLC → NC)	4-116
D768 to D956	Input values from MP4210.0 to MP4210.47	7-15
W960 to W968	Input values from MP4220.0 to MP4220.4	7-17
W976 to W988	Input values from MP 4310.0 to MP4310.6	7-17
W1008	S code for minimum rpm	4-91
W1010	Number of tool magazine pockets	–

PLC Programming – Contents

1	PLC-functions	7-5	
1.1	Select PLC-operation	7-5	
1.2	PLC – Main menu	7-6	
1.2.1	PLC-program editing	7-6	
1.2.2	Erase PLC-program	7-8	
1.2.3	Transfer program from EPROM	7-8	
1.2.4	Translate PLC program	7-8	
1.2.5	Utilization	7-8	
1.3	Test functions for the PLC-program	7-9	
1.6.1	TRACE-function	7-9	
1.3.2	TABLE function	7-10	
1.3.3	Transferring the PLC-program	7-10	
2	Program creation	7-12	
2.1	Program structure	7-12	
2.1.1	Command	7-12	
2.1.2	Module technique	7-13	
2.2	Address allocation	7-14	
2.2.1	Operand directory	7-14	
2.2.2	Addressing the word memory	7-14	
2.3	Data transfer PLC → NC and NC → PLC	7-15	
2.4	Timers and counters	7-17	
2.4.1	Timers	7-17	
2.4.2	Counters	7-20	
2.5	Edge evaluation of the PLC-inputs	7-21	
2.6	EPROM-creation	7-21	
2.7	Error-messages	7-24	
2.7.1	Syntax errors within a command line	7-24	
2.7.2	Syntax errors in the course of a program	7-24	
2.7.3	Run-time errors	7-25	
3	Commands	7-27	
3.1	Load and Assign Commands	7-27	
3.1.1	LOAD	L	7-27
3.1.2	LOAD NOT	LN	7-29
3.1.3	LOAD TWO'S-COMPLEMENT	L–	7-31
3.1.4	LOAD BYTE	LB	7-32
3.1.5	LOAD WORD	LW	7-32
3.1.6	LOAD DOUBLEWORD	LD	7-32
3.1.7	ASSIGN	=	7-34
3.1.8	ASSIGN BYTE	B=	7-36
3.1.9	ASSIGN WORD	W=	7-36

3.1.10 ASSIGN DOUBLEWORD	D=	7-36
3.2 Set-commands		7-39
3.2.1 SET	S	7-39
3.2.2 RESET	R	7-40
3.2.3 SET NOT	SN	7-41
3.2.4 RESET NOT	RN	7-42
3.3 Logic Gates		7-44
3.3.1 AND	A	7-44
3.3.2 AND NOT	AN	7-46
3.3.3 OR	O	7-48
3.3.4 OR NOT	ON	7-50
3.3.5 EXCLUSIVE OR	XO	7-52
3.3.6 EXCLUSIVE OR NOT	XON	7-54
3.4 Arithmetic Commands		7-57
3.4.1 ADDITION	+	7-57
3.4.2 SUBTRACTION	-	7-58
3.4.3 MULTIPLICATION	x	7-59
3.4.4 DIVISION	/	7-60
3.4.5 REMAINDER	MOD	7-61
3.5 Comparisons		7-63
3.5.1 EQUAL TO	==	7-63
3.5.2 LESS THAN	<	7-64
3.5.3 GREATER THAN	>	7-65
3.5.4 LESS THAN OR EQUAL TO	<=	7-66
3.5.5 GREATER THAN OR EQUAL TO	>=	7-67
3.5.6 UNEQUAL	<>	7-68
3.6 Parentheses with logical gating		7-70
3.6.1 AND []	A[]	7-70
3.6.2 AND NOT []	AN[]	7-70
3.6.3 OR []	O[]	7-70
3.6.4 OR NOT []	ON[]	7-70
3.6.5 EXCLUSIVE OR []	XO[]	7-71
3.6.6 EXCLUSIVE OR NOT []	XON[]	7-71
3.7 Parentheses with arithmetic commands		7-74
3.7.1 ADDITION []	+ []	7-74
3.7.2 SUBTRACTION []	- []	7-74
3.7.3 MULTIPLICATION []	x []	7-74
3.7.4 DIVISION []	/ []	7-74
3.7.5 REMAINDER []	MOD []	7-75
3.8 Parentheses with comparison commands		7-78
3.8.1 EQUAL TO []	== []	7-78
3.8.2 LESS THAN []	< []	7-78
3.8.3 GREATER THAN []	> []	7-78
3.8.4 LESS THAN OR EQUAL TO []	<= []	7-78

3.8.5	GREATER THAN OR EQUAL TOL []	>= []	7-79
3.8.6	NOT EQUAL TO []	<> []	7-79
3.9	Shift Commands		7-82
3.9.1	SHIFT LEFT	<<	7-82
3.9.2	SHIFT RIGHT	>>	7-83
3.10	Bit Commands		7-85
3.10.1	BIT SET	BS	7-85
3.10.2	BIT RESET	BC	7-86
3.10.3	BIT TEST	BT	7-87
3.11	Stack Operations		7-89
3.11.1	Load Data onto the Data Stack	PS	7-89
3.11.2	Acquire data from the data stack	PL	7-90
3.11.3	Load logic accumulator onto the data stack	PSL	7-90
3.11.4	Load word accumulator onto the data stack	PSW	7-91
3.11.5	Acquire logic accumulator from the data stack	PLL	7-91
3.11.6	Acquire word accumulator from the data stack	PLW	7-91
3.12	Jump Commands		7-94
3.12.1	Unconditional Jump	JP	7-94
3.12.2	Jump if Logic Accumulator = 1	JPT	7-94
3.12.3	Jump if Logic Accumulator = 0	JPF	7-94
3.12.4	Call Module	CM	7-96
3.12.5	Call Module if Logic Accumulator = 1	CMT	7-96
3.12.6	Call Module if Logic Accumulator = 0	CMF	7-96
3.12.7	End of Module, Program End	EM	7-98
3.12.8	Jump Label	LBL	7-98
3.13	CASE statement		7-100
3.13.1	Indexed call module	CASE	7-100
3.13.2	End indexed Call Module	ENDC	7-100
4	PLC-Modules		7-103
4.1	Module 9000: Copy a marker-block		7-103
4.2	Module 9001: Copy a Word-block		7-103
4.3	Module 9010: Indexed reading of Byte Module 9011: Indexed reading of Word Module 9012: Indexed reading of Doubleword		7-104
4.4	Module 9020: Indexed writing of Byte Module 9021: Indexed writing of Word Module 9022: Indexed writing of Doubleword		7-105
4.5	Module 9032: Transfer machine parameter values to the PLC		7-106
4.6	Module 9035: Transfer status information to the PLC		7-107
4.7	Module 9036: Transfer PLC status information to the NC		7-108
4.8	Module 9040: Transfer coordinate values to the PLC		7-109
4.9	Data transfer by the PLC		7-109
4.9.1	Module 9100: Assign data interface		7-110
4.9.2	Module 9101: Enable data interface		7-110

4.9.3	Module 9102: Interface status	7-111
4.9.4	Module 9105: Transmit binary data via data interface	7-112
4.9.5	Module 9106: Receive binary data via data interface	7-113
4.9.6	Module 9120: Position an auxiliary axis	7-114
4.9.7	Module 9121: Stop an auxiliary axis	7-115
4.9.8	Module 9122: Poll the status of an auxiliary axis	7-115
4.9.9	Module 9123: Traverse the reference point of an auxiliary axis	7-116
4.9.10	Module 9124: Override for PLC axis	7-117
4.9.11	Module 9125: Run an NC block	7-118
5	Compatibility with TNC 355	7-121
5.1	PLC Program Conversion	7-122
5.2	Compatibility Markers	7-123
5.3	Incompatibility	7-125
5.3.1	PLC Macros	7-125
5.3.2	Non-implemented markers	7-128
5.3.3	PLC Cycle Time	7-128

1 PLC Functions

The integrated PLC in the TNC 360/TNC 335 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 can make it easier to find faults in the PLC program (see section "Test functions").

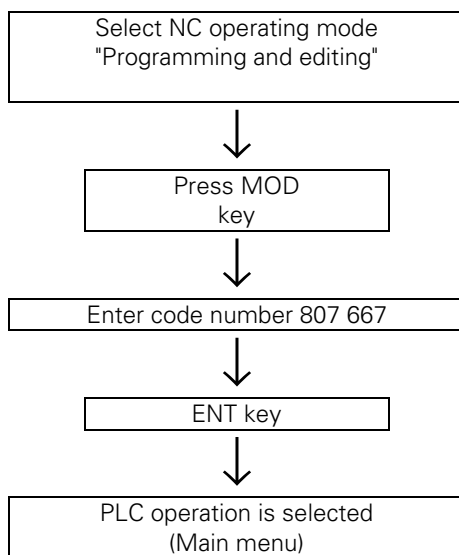
5 ms are available for a PLC run. Up to 4000 logic commands, equivalent to 28 KB, can be processed within this period (executable memory). A new PLC run commences every 24 ms, i.e. every 24 ms the inputs are read and outputs are set.

The following section shows how to start PLC operation.

1.1 Select PLC operation

PLC operation covers all functions for creating and testing the PLC programs.

It can be selected as follows, using the code number 807 667.



PLC operation can be stopped by pressing the END  key.

1.2 PLC – Main menu

Enter the code number 807 667 to call the following screen display (main menu):

ERASE PLC PROGRAM
TRANSFER PROGRAM FROM EPROM
PLC EDITING MODE
PLC PROGRAM TRACE MODE
TABLES I/O/C/T/M
TRANSLATE PLC PROGRAM
UTILIZATION

Select the required mode with the cursor keys ↑ and ↓ and activate by pressing ENT. To exit the menu and return to the original NC mode, press the END key again.

1.2.1 PLC program editing

In the operating mode PLC EDITING MODE an instruction list can be compiled or edited.

After the ENT key is pressed, the editor identifies itself with

"0 EM"

"0" stands for program line 0 and "EM" for End of Module.

When entering an instruction, "EM" automatically moves downward by one line.

With this control it is possible to reassign the functions of TNC operating keys by entering the code number 807 667. The new key assignments are indicated on a special tactile membrane that is placed over the TNC keyboard. The tactile membrane is included with this Technical Manual.

The PLC editing mode can be exited by pressing the END key.

A complete instruction list comprises:

- .Line numbers
- .Command from command store (see Section "Commands")
- .Operand type
- .Operand number
- .Comment

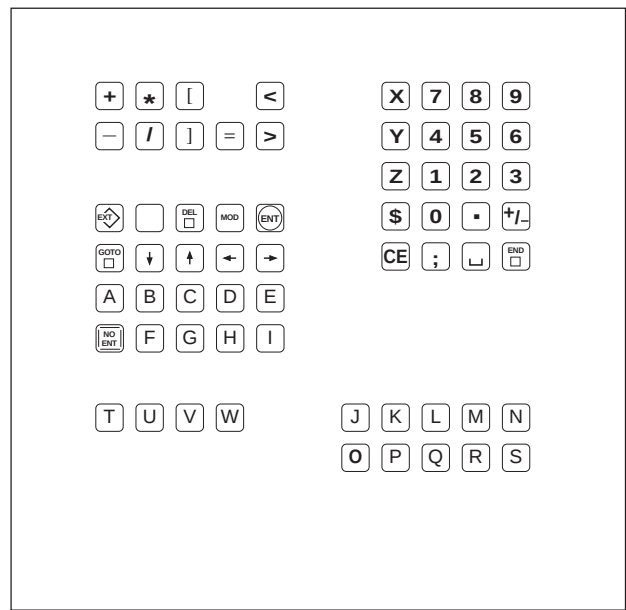
The line number is automatically generated on entering a command.

The command, operand type, operand number and the comment must be transferred with ENT. The cursor then jumps to the next positions within the instruction. The instruction is completed after the comments are entered. If no comment is to be entered, the position is simply concluded the ENT. Lines that are to contain only a comment should be started with the character ";," instead of a command. A comment can therefore cover several lines if each line commences with ";,". A line contains a maximum of 26 characters for the comment and may include all numeral, letters and other characters on the keyboard.

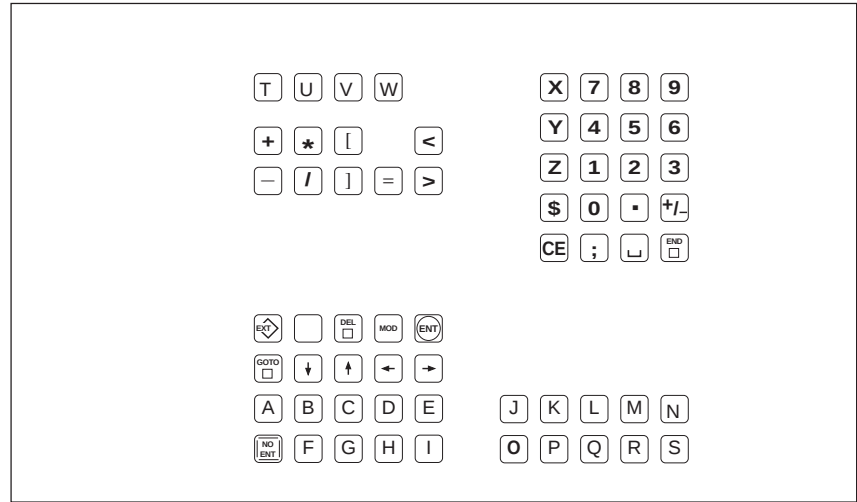
A "find function" makes it easier to find certain operands.

If you wish to find a certain marker in the program listing, use the arrow keys to move the cursor to a marker and press GOTO. Enter the number of the desired marker. Then press ENT to find the marker.

Key assignments for the PLC editor



TE 335 A / TE 355 A



TE 335 B/TE 335 B

1.2.2 Erase PLC program

In the operating mode ERASE PLC PROGRAM, a PLC program contained in the RAM can be deleted by pressing ENT.

1.2.3 Transfer program from EPROM

In the operating mode TRANSFER PROGRAM FROM EPROM, a program that is stored in the EPROM can be copied into the RAM by pressing ENT (see also MP 4010).

1.2.4 Translate PLC program

A PLC program created with the PLC editor and stored in RAM must first be compiled in the TRANSLATE PLC PROGRAM mode before it can be tested.

Press ENT to start translation.

If a PLC program is already in the RAM prior to TNC switch-on, it is automatically translated.

Errors in the program can be found during the translation run (see Section "Program creation", "Error messages").

1.2.5 Utilization

In the UTILIZATION mode of operation the occupied PLC processing time and the vacant PLC memory are displayed.

The vacant memory is derived from the total memory and the occupied memory.

The vacant PLC memory is displayed in bytes or in percent of the maximum memory (28 KB). 100 % processing time corresponds to 2.5 ms (from 24 ms PLC cycle time). The maximum possible processing time is determined by the manufacturer and equals 5 ms. The current processing time indicates the duration of the last PLC scan. If the current processing time exceeds 5 ms, the blinking error message

ERROR IN PLC PROGRAM 53

will appear.

Please note that the processing time increases by 20% during active V.24 data transfer, and by 5% during handwheel operation. A PLC program should therefore be tested during active V.24 and handwheel operation in order to ensure that it will never exceed maximum processing time.

1.3 Test functions for the PLC program

1.3.1 TRACE function

The TRACE function makes it possible to check the logical states of the markers, inputs, outputs, timers and counters.

If this function is selected, the following menu appears:

```
SELECT I/O/C/T/M
DISPLAY TRACE BUFFER
START TRACE
END TRACE
```

SELECT I/O/C/T/M

Those inputs (I), outputs (O), counters (C), timers (T) or markers (M) whose logical states are to be checked can be entered into a table in the SELECT I/O/C/T/M function. A maximum of 16 markers, timers etc. can be simultaneously verified. Each position is interrogated via dialogue. Erroneous entries can be deleted by pressing DEL.

The memory for the trace mode (TRACE BUFFER) has space for 1024 individual states per operand, i.e. 1024 PLC processes are recorded. In order to record the required time duration of the user, a TRIGGER condition can be entered for each operand:

"1" = Recording when operand is logically "1"

"0" = Recording when operand is logically "0"

If the trigger position is verified with the NOENT key, it means that a trigger condition is not required. 512 states are recorded before and after a trigger event. The trigger event is regarded as fulfilled only when the conditions for all operands are simultaneously fulfilled with TRIGGER.

```
Example:      M2064  1
              M2065  0
              M2066  1
              I5
              Trigger event:  M2064 logical "1"
                              M2065 logical "0"
                              M2066 logical "1"
```

The state of I5 has no significance for the trigger event: it is, however, recorded.

If no trigger condition is entered for the operands, the states of the operands are continually recorded and, after ending the trace mode, the last 1024 states are displayed.

To exit the SELECT I/O/C/T/M function, press END.

DISPLAY TRACE BUFFER

With the DISPLAY TRACE BUFFER function the logic states 0 or 1 of the selected operands are graphically displayed in a diagram. With simultaneous triggering, the counter is reset to 0 (upper left in screen). By using the cursor keys ← and →, 512 logic states can be observed before and after the trigger event.

This can be used to determine, for example, whether a marker, output etc. was set too late or too early. Taking the PLC processing time of 24 ms into account, a time displacement in ms can be detected.

START TRACE

With this mode, TRACE is started. Until the trigger event occurs, the display PC TR blinks in the status field. Then the display goes out.

END TRACE

If the trigger event does not occur, the trace can be aborted with the END TRACE function. In this case, the last 1024 states of the selected operands are stored.

1.3.2 TABLE function

In the TABLES I/O/C/T/M function the states of inputs (I), outputs (O), counters (C), timers (T) and markers (M) can be dynamically displayed on the screen. If the desired operand number is selected with the cursor keys, the operand can be changed with Set (S) or Reset (R), provided that it was not defined through the PLC program. The individual tables for inputs, outputs, etc. are selected with the corresponding letter.

1.3.3 Transferring the PLC program PLC program

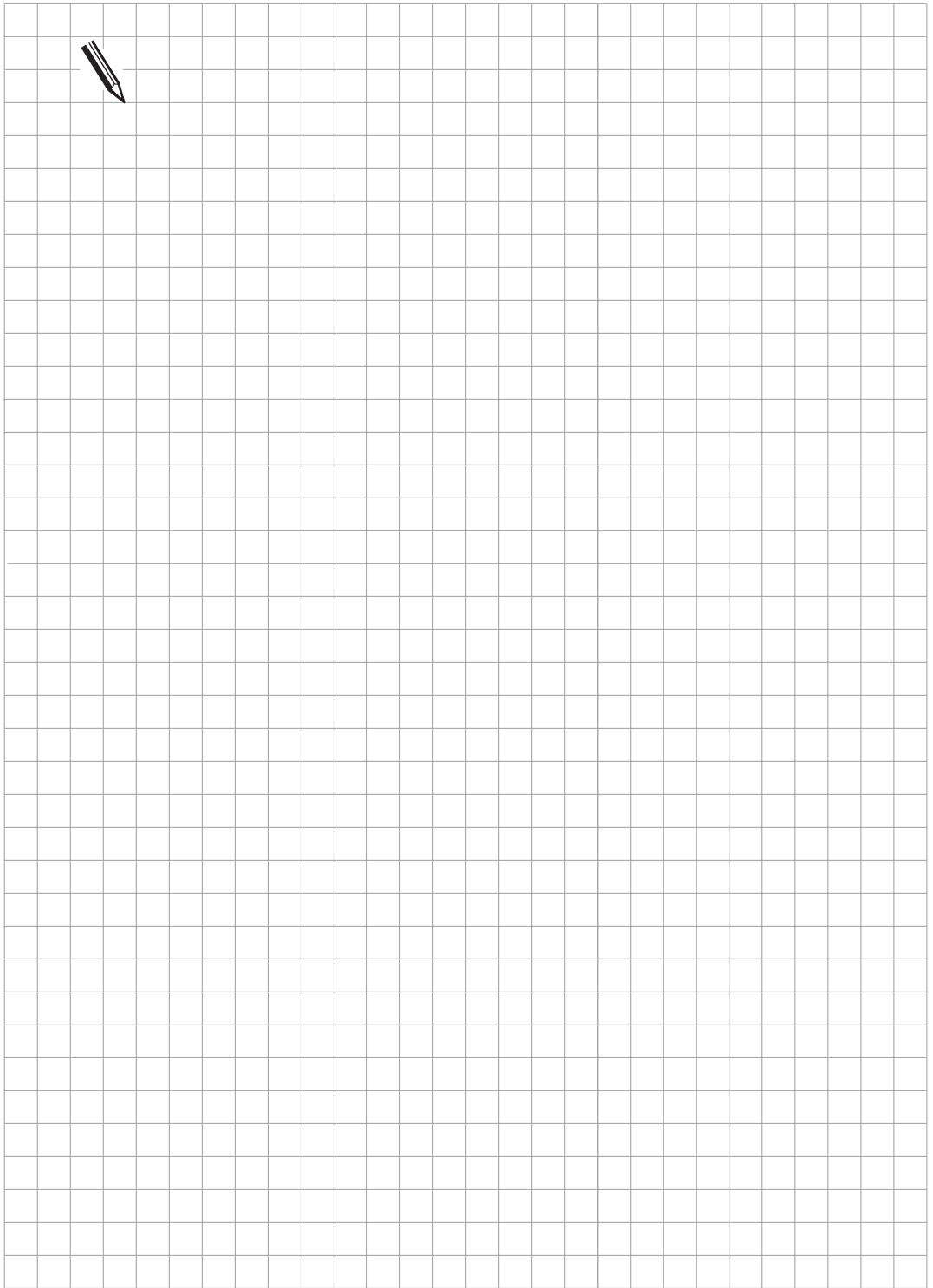
Data transmission can be activated with the EXT key from the main menu.

The following menu appears:

```
READ-IN SELECTED PROGRAM
OUTPUT ASCII
OUTPUT ASCII UNFORMATTED
OUTPUT ASCII WITH CROSS-REF
BINARY OUTPUT
```

PLC programs written on a computer can be transferred into the PLC RAM with the RS-232-C interface. To load a program, select READ IN SELECTED PROGRAM. With the OUTPUT ASCII function the PLC program can be transferred from the RAM to an external device. It can be transferred with the cross references of all its operands.

It is also possible to transfer the program in unformatted or binary form by selecting the corresponding menu items (see Section "Program creation", "EPROM creation").



2 Program Creation

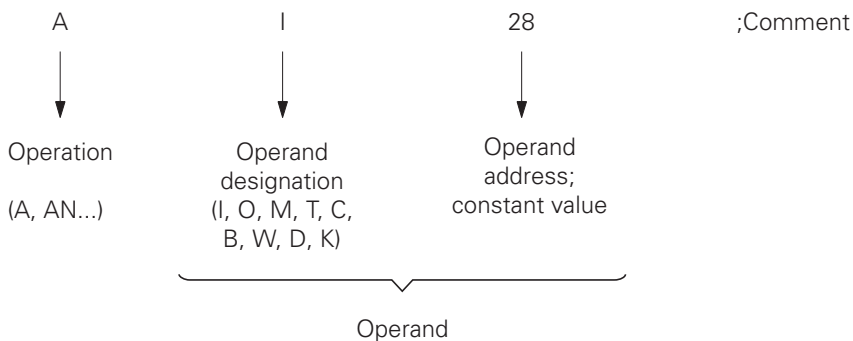
The PLC program can be created directly on the HEIDENHAIN contouring controls. For this, the PLC Editor must be called with the code number **807 667** (see Section "PLC functions").

The PLC program can also be written in ASCII code on a PC and then transferred to the control.

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.



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".

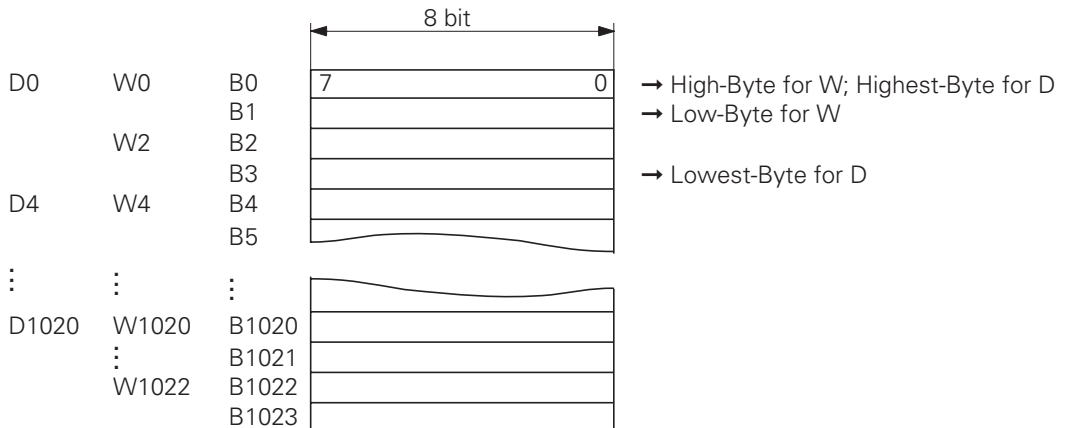
2.2 Address allocation

2.2.1 Operand directory

Operand	Abbreviation	Address range
Marker	M (Marker)	0 to 3263
Input	I (Input)	I0 to I31; I128 to I151; I64 to I126 (PL 400) or up to I127 (PL 410)
Output	O (Output)	O0 to O30; O32 to O62(PL 400/PL 410)
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 word 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 M2000 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 M2000 and Bytes B0 to B127 and the machine parameters 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 → PLC

2.3 Data transfer PLC → NC and NC → PLC

PLC → NC

The Q parameters Q100 to Q107 transfer numbers from the PLC to the part program. This means that Q100 to Q107 can be overwritten by the PLC. The numerical value is registered in Doubleword D528 and the Q parameter numbers 0 to 7 are defined in Word W516. The numbers 0 to 7 correspond to parameters Q100 to Q107. The transfer is activated with the strobe marker M2713. The Q parameter values are transferred with the next M/S/T strobe.

Marker	Function	Set	Reset
M2713	Activate the transfer of the value from D528 to the Q Parameter defined in W516	PLC	NC
Address	Function		
D 528	Doubleword with multiple function, here data for transfer from the PLC to the NC		
W516	Q Parameter No. for numerical transfer from PLC to NC (0 – 7 = Q100 – Q107)		

NC → PLC

Sixty different machine parameters are reserved for data transfer in the PLC. These machine parameters are kept in the Doublewords D768 to D956 and the Words W960 to W968 and W976 to W988. 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.

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

Address	Function
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
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

Address	Function
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**
Input range: -99 999.999 to +99 999.999

MP 4220 to MP4220.4 **Setting a number in the PLC:** In the Word range W960 to W968
Input range: 80 to 30 000

MP4230.0 to MP4230..31 to MP4239.0 to MP4239.31 **Set a number in the PLC via module 9032**
Input range: -30 000 000 to +30 000 000

MP4310.0 to MP4310.6 **Set a number in the PLC:** in the Word range W976 to W988
Input range: 0 to 65 535

2.4 Timers and counters

2.4.1 Timers

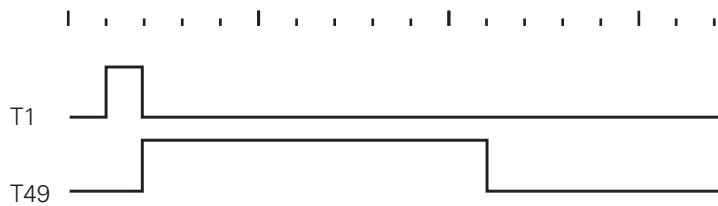
48 timers are available in the PLC. These 48 timers are controlled by special markers with the abbreviation symbol T. The time period for the timer is defined in the machine parameter MP4110.X. The time unit corresponds to the PLC cycle time.

The timers are started by setting the markers T0 to T47 which also sets the timers 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

MP4110.x

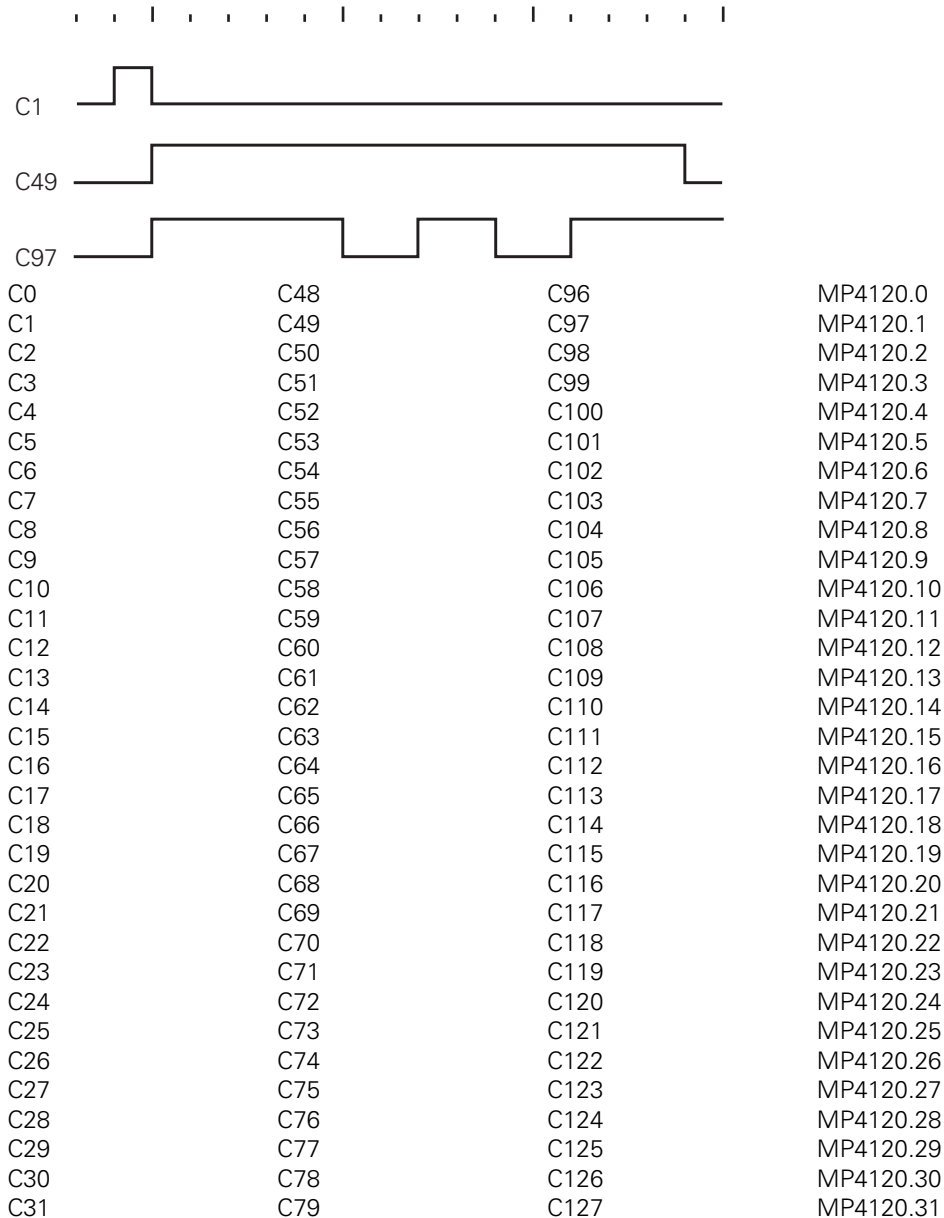
Time for timers

Input range: 0 to 65 535 [PLC cycles] (= 24 ms on TNC 360/TNC 335)

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)



MP4120.x

Pre-set value for counters C0 to C31

2.5 Edge evaluation of the PLC inputs

The edge evaluation for the PLC inputs can be activated by marker M2497. 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 (PL 400) or I127 (PL410)
M1628 to M1651	I128 to I151

Marker for falling edges at the PLC inputs:

Marker	PLC inputs
M1700 to M1731	I0 to I31
M1764 to M1826	I64 to I126 (PL 400) or I127 (PL410)
M1828 to M1851	I128 to I151

Marker	Function	Set	Reset
M2497	Activate the edge evaluation for PLC inputs Rising-edge marker M1500 to M1651 Falling-edge marker M1700 to M1851	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

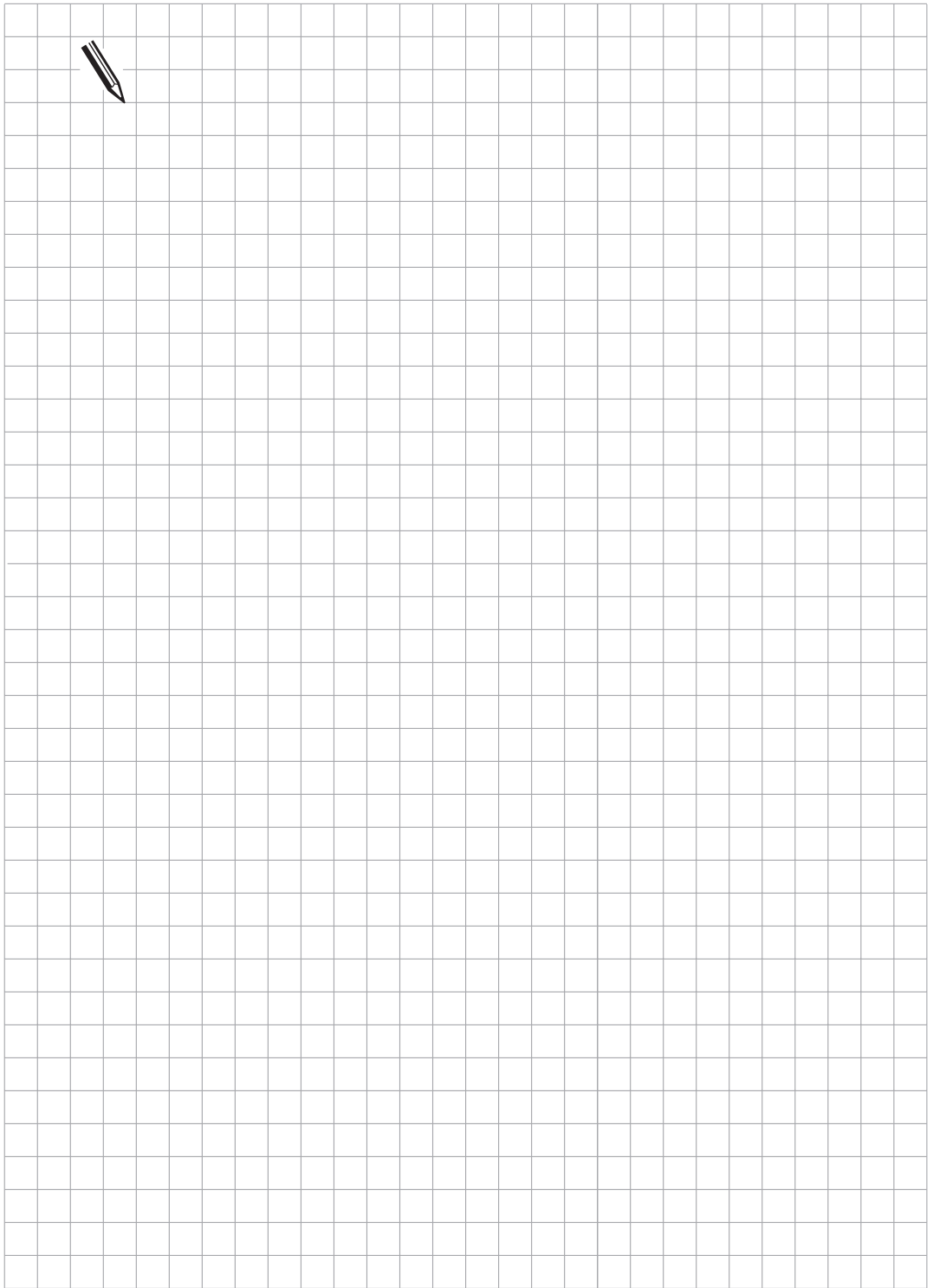
Input: 0 or 1
0 = EPROM operation
1 = RAM operation

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 in binary code (see Section "PLC functions").

The PLC files are then transferred to the PC in the Motorola EXORMAX S2 Record (MOD) format. With the MEGA PROMMER these data can be read in the MOD format and the EPROM programmed.

Please contact the HEIDENHAIN customer service if you have further questions.



2.7 Error messages

Error messages aid the programmer in creating the instruction list and testing of the program.

2.7.1 Syntax errors within a command line

These errors may occur when editing a line or on reading it in via the interface.

- 0 No valid command.
- 1 Operand for jump is not a Label. (Can only happen when reading in via the interface. For a Jump command a type abbreviation is available for the Operand).
- 2 Invalid Operand type (the command cannot be combined with this Operand).
- 4 Operand outside the permissible range (the stated number is too high, or odd address for Word or Doubleword).
- 5 No Limiter after command. (Can only happen when reading in via the interface. The comment after the command was not designated by ";" or "*").
- 6 Line end not found (Can only happen when reading in via the interface. Comment too long).

2.7.2 Syntax errors in the course of a program

These errors are recognized during the compilation process. The Editor points to the line where the error was found. If the PLC program is compiled on switching-on (for example, because the control was switched off immediately after editing the PLC program) then a flashing message "ERROR IN PLC PROGRAM" will be displayed. Remedy: switch off and on again, and call the PLC Editor with the code number. The Editor indicates the position of the error.

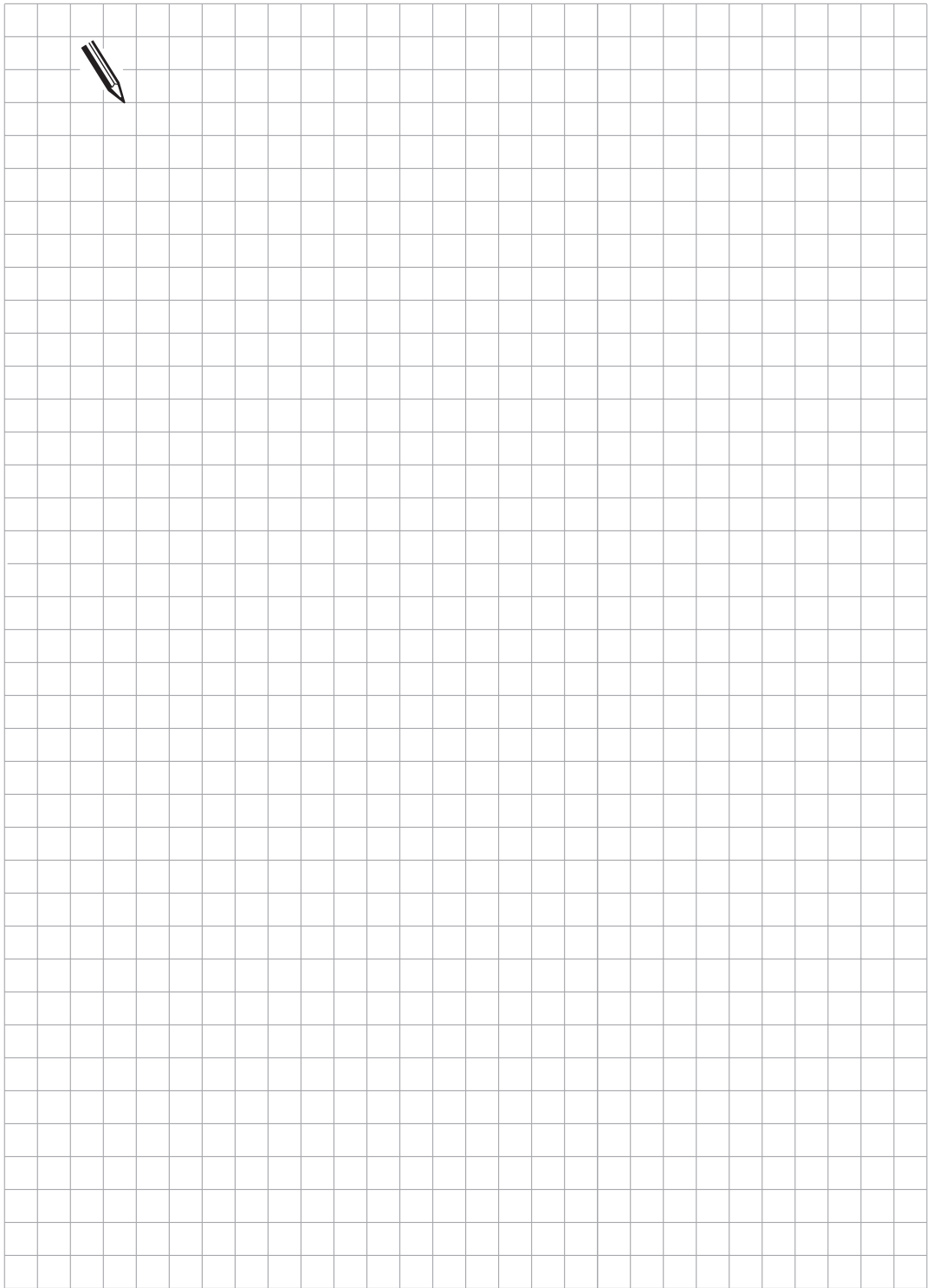
- 7 Called Label has not been defined.
- 8 No End program condition found (the program does not contain an EM instruction, or it contains a JP instruction without a following LBL instruction).
- 9 Program is too long (RAM overflow) (insufficient memory for the program code which is to be generated).
- 10 Assign within parentheses (an =, S, SN, R, RN, or PS instruction or a Jump command has been programmed, although arithmetic parentheses are open).
- 11 Excessive nesting of parentheses (more than 16 parentheses successively opened).
- 12 Jump within a gating sequence (an unconditional jump has been programmed, although the gating sequence was not closed with an Assign).
- 13 "Close parentheses" without "open parentheses" (a "close parentheses" command was programmed, although no parentheses were open).
- 14 Label within parentheses (a LBL instruction has been programmed, although parentheses are open).
- 15 Label within a gating sequence (a LBL instruction has been programmed, although the previous gating was not closed with an Assign).
- 16 Jump within parentheses (a jump instruction has been programmed, although parentheses are open).
- 17 Parentheses open at end of block (an EM instruction has been programmed, although parentheses are open).
- 18 Label defined twice.
- 19 Word Assign missing (a Logic instruction has been programmed, although the previous Word gating was not closed with an Assign).

- 20 Logic Assign missing (a Word instruction has been programmed, although the previous Logic gating was not closed with an Assign)
- 21 Word Accumulator not loaded (a Word Assign or gating has been programmed, although the Word Accumulator does not contain a definite value).
- 22 Logic Accumulator not loaded (a Logic Assign has been programmed, although the Logic Accumulator does not contain a definite value).
- 23 Accumulators not loaded on "open parentheses" (an A[, AN[, O[, ON[, or XON[command has been programmed, although neither the Word nor the Logic Accumulator has been gated or loaded).
- 24 Incorrect type of the parentheses result (a different type has been calculated in the parentheses from that which was defined at the "open parentheses" command, i.e. Logic instead of Word or vice versa).
- 25 Conditional jump with incorrect Logic Accumulator (a conditional jump has been programmed, although the Logic Accumulator does not contain a definite value).
- 26 Empty CASE instruction.
- 27 "END CASE" missing.

2.7.3 Run time errors

These errors only appear when the PLC program is executed. A flashing error message "ERROR IN PLC PROGRAM NR" is displayed. After switching the control off and on again, the Editor can be accessed by using the code number. The message "INPUT ERROR" is then displayed and the Editor stands at the erroneous line or, if the program run time has been exceeded, at the jump instruction which was last processed.

- 50 Excessive nesting (too many Modules nested inside one another).
- 51 STACK underflow (an attempt to acquire data from the STACK, although it was empty).
- 52 STACK overflow (an attempt to load too much data onto the STACK).
- 53 Time out (the permissible program run time has been exceeded by more than twice. Check the subprogram structure).
- 54 CASE Arguments are larger than the number of entries in the table.



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 x x x		
1	L K+54		... 0 0 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																0	0 1 0 0 0 1 0		00101010
3	= B8		... 0 0 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.

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																			
1	LN B6		... 0 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 0 1 0 0 0																		00101010	
3	= B8		... 0 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]	—	2.4/2.0	2.2	1.8
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																Operand Contents				
		Bit	31															7	0		7	0
			... 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]	30.0
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]	57.2
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]	117.2
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	
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.

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. The = command is only used at the end of a gating chain in order that a gating result is available. 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																	
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	

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.

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]	26.0
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]	50
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]	104.4
Number of bytes	14

Operands: M, I, O, T, C

Operation:

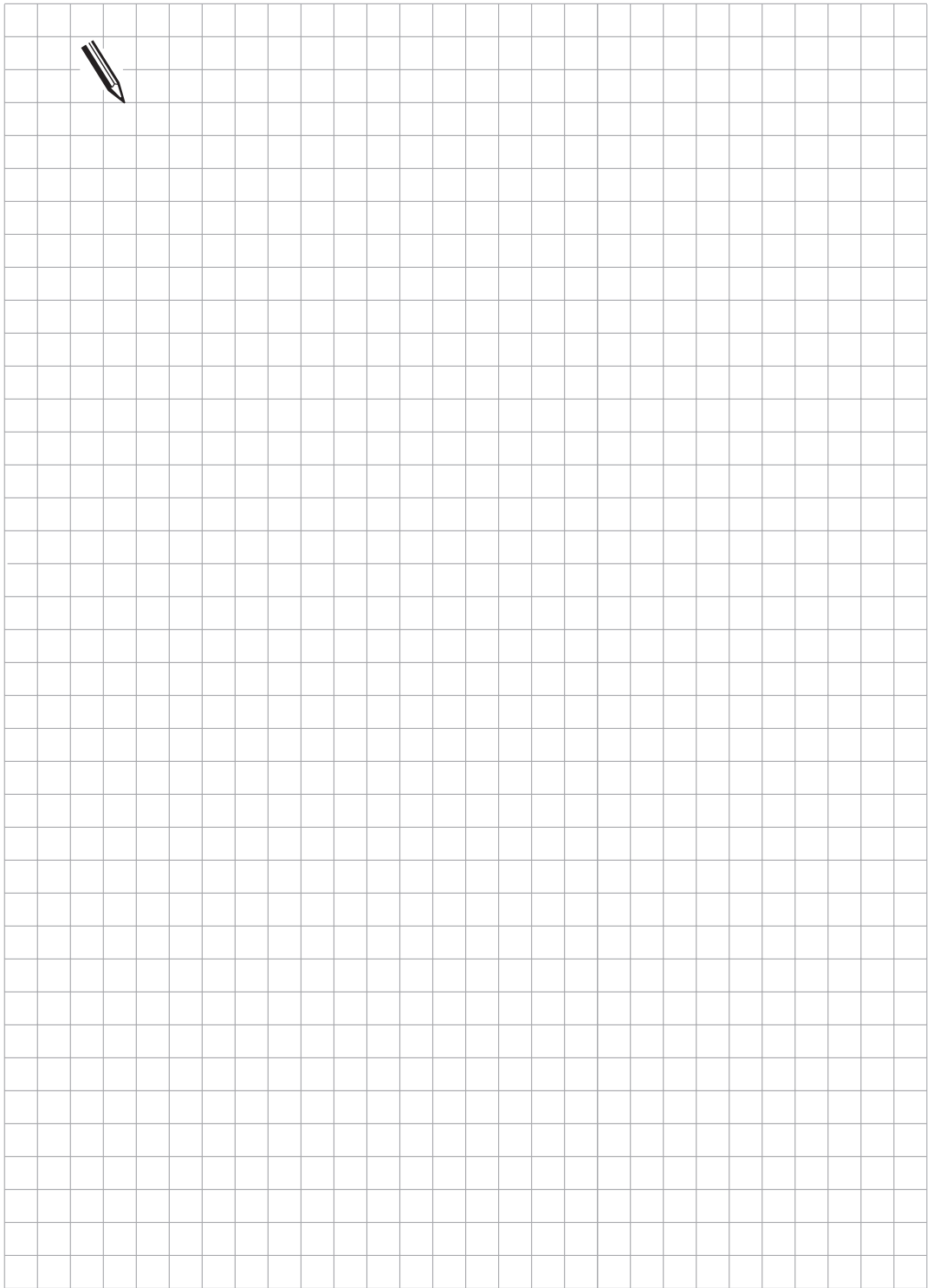
With the command D=, 32 bits are copied from the Word Accumulator to Markers, Inputs, Outputs, Timers or Counters with ascending numbering. Each bit corresponds to 1 operand. The LSB in the Accumulator is copied to the designated operand address, the LSB + 1 to the designated address + 1 and so on. The 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													
1	L W8		... 0 0 0 0 1 1 0 1 1 0 1										0011011011111111			
													O20 O5			
2	W= O5		... 0 0 0 0 1 1 0 1 1 0 1										0011011011111111			

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.



3.2 Set commands

3.2.1 SET (S)

Abbreviation for the PLC Editor: S (SET)

	Operand changed	Operand unchanged
Execution time [µs]	2.0 to 2.40	1.0 to 1.4
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	Operand Contents
		Bit 31 . . . 7 0	
		... x x x x x x x X x x x x x x x	
1	L I4	... x x x x x x x 1 x x x x x x x	1
2	O I5	... x x x x x x x 1 x x x x x x x	0
3	S O2	... x x x x x x x 1 x x x x x x x	1
4	S M500	... x 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]	2.0 to 2.4	1.0 to 1.4
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 x	
1	L I4	... x x x x x x x 1 x x x x x x x	1
2	O I5	... x x x x x x x 1 x x x x x x x	0
3	R O2	... x x x x x x x 1 x x x x x x x	0
4	R M500	... x x x x x x x 1 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 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]	2.0 to 2.4	1.0 to 1.4
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 x	
1	L I4	... x x x x x x x 0 x x x x x x x	0
2	O I5	... x x x x x x x 0 x x x x x x x	0
3	SN O2	... x x x x x x x 0 x x x x x x x	1
4	SN M500	... x 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]	2.0 to 2.4	1.0 to 1.4
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).

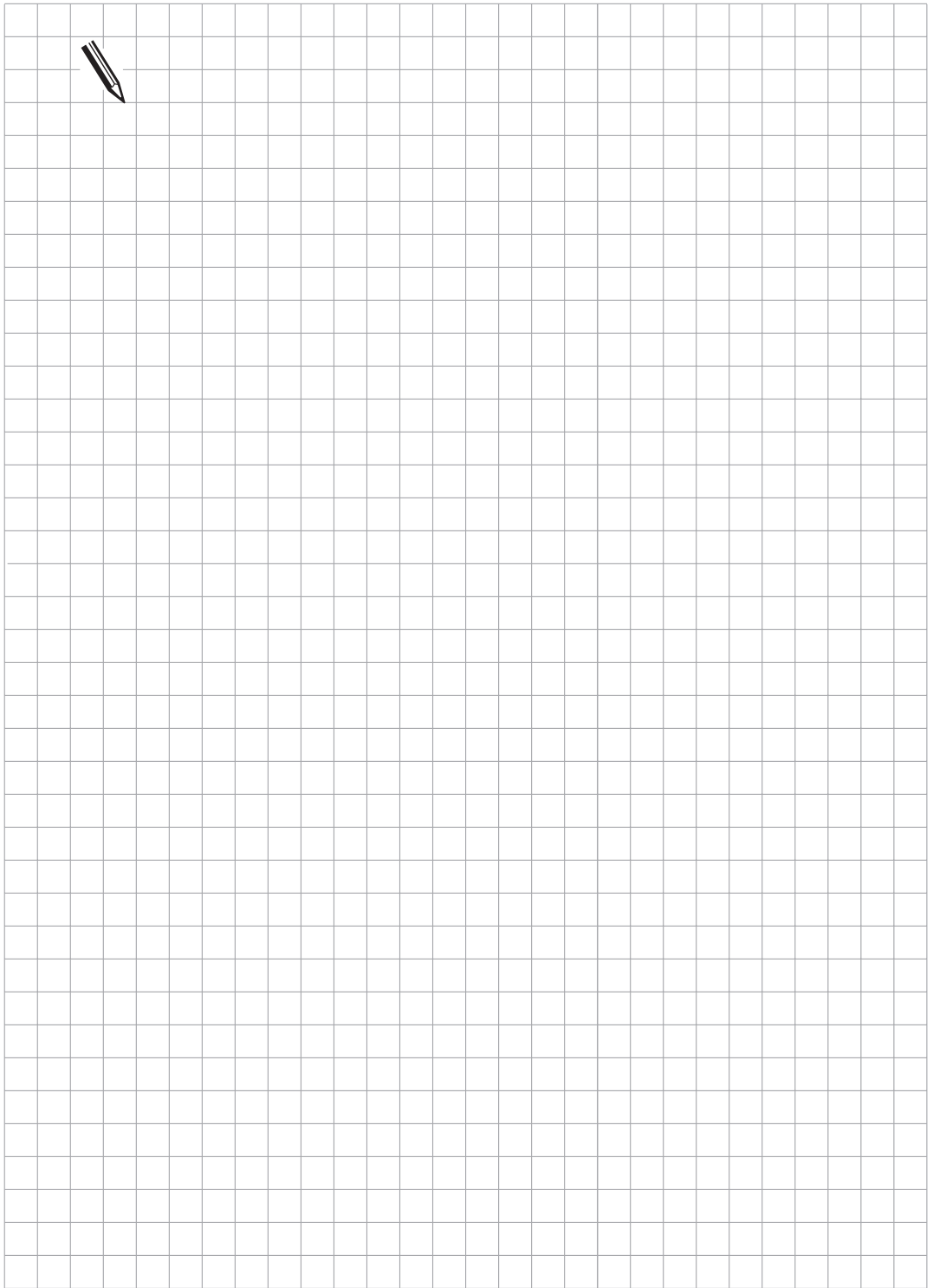
Example:
Input I4 and Input I5 are to be gated with OR.
If the gating result = 0, Output O2 and Marker M500 are reset.

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 x	
1	L I4	... x x x x x x x 0 x x x x x x x	0
2	O I5	... x x x x x x x 0 x x x x x x x	0
3	RN O2	... x x x x x x x 0 x x x x x x x	0
4	RN M500	... x 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 OR.
Line 3: The gating result = 0: Output O2 is reset.
Line 4: The gating result = 0: Marker M500 is reset.



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															7			0	15	8	7	0		
			... x x x x x x x 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 0 0 0 0																0	0 1 1 0 1 1 1 0						00110110	
2	A B5		... 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																0	0 1 0 0 0 1 0 0						00101010	
3	= B8		... 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0																0	0 1 0 0 0 1 0 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)

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	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			... x x x x x x x x x x x x x x x x																15	8	7	0												
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 1 0 1 0 0 0																00110110				10101010											
3	= W8		... 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 0 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]	1.2	1.2/1.2	1.8	1.6
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 360/TNC 335 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	Bit	Accumulator Contents	Operand Contents
		31	. . . 7 0	
			... x x x x x x x x x x x x x x x x	
1	L I4		... x x x x x x x 0 x x x x x x x x	0
2	O I5		... x x x x x x x 1 x x x x x x x x	1
3	= O2		... x x x x x x x 1 x 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	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	
1	L B6	... 0 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 0 1 1 1 1 1 0	00101010
3	= W8	... 0 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]	2.0	2.0/2.0	3.0	2.6
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 360/TNC 335 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 x x	
1	L I4	... x x x x x x x 0 x x x x x x x x	0
2	ON I5	... x x x x x x x 1 x x x x x x x x	0
3	= O2	... x x x x x x x 1 x 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	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			... 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																0011110000110110															
2	ON W4		... 1 1 1 1 1 1 1 1 0 1 0 1 1 1 0 1 1 1																0011011010101010															
3	= W8		... 1 1 1 1 1 1 1 1 0 1 0 1 1 1 0 1 1 1																1111110101110111															

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]	1.6	1.6/1.6	2.4	2.0
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 360/TNC 335 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 x x	
1	L I4	... x x x x x x x 1 x x x x x x x x	1
2	XO I5	... x x x x x x x 0 x x x x x x x x	1
3	= O2	... x x x x x x x 0 x 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	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
			... 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 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 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 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]	2.0	2.0	3.0	2.6
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 360/TNC 335 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.

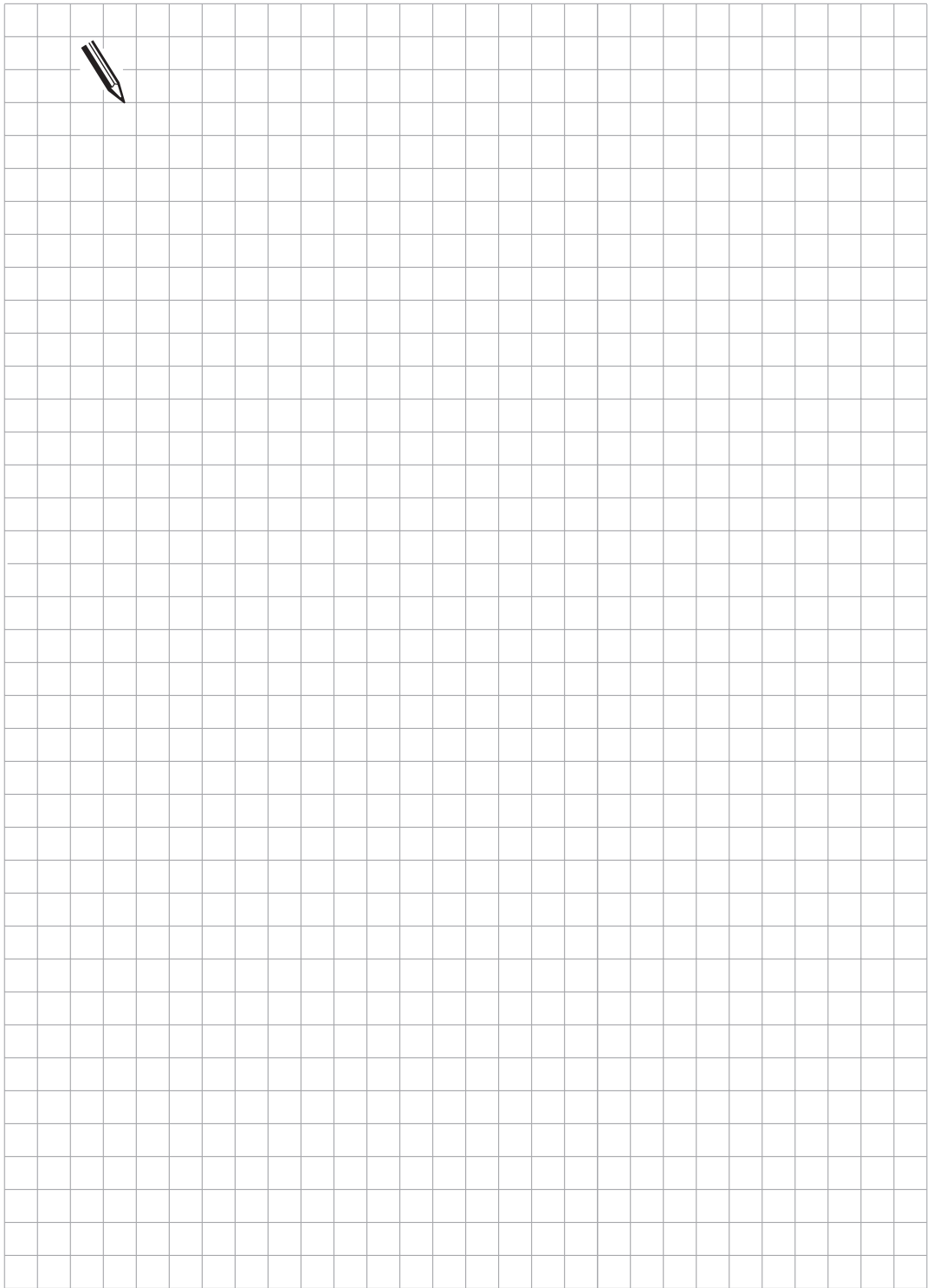
Example:

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																												
1	L W6		... 0 0 0 0 1 1 1 1 0 0 0 0 1 1 0 1 0																0011110000110110												
2	XON W4		... 1 1 1 1 1 1 1 0 1 0 1 0 1 0 1 1																0011011010101010												
3	= W8		... 1 1 1 1 1 1 1 0 1 0 1 0 1 0 1 1																1111010101100011												

- 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]	---	2.8/2.4	1.8	1.6
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 x x x x x x x x x	
1	L K100000	1 0 0 0 0 0 0	
2	+ W6	1 0 0 2 0 0	2 0 0
3	= D8	1 0 0 2 0 0	1 0 0 2 0 0

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]	---	2.8/2.4	1.8	1.6
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 x x x x x x x x x	
1	L K100000	1 0 0 0 0 0	
2	- W6	9 9 8 0 0	2 0 0
3	= D8	9 9 8 0 0	9 9 8 0 0

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]*	---	10.2/9.4	---	9.8
Number of bytes	---	14	10	14

* Only a maximum processing time can be entered with the arithmetic operations X, / and MOD.
The processing time may be shorter, depending on the operands.

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 result of multiplication causes an overflow, 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 x x x x x x x x	
1	L K100	1 0 0	
2	x W6	2 0 0 0	2 0
3	= D8	2 0 0 0	2 0 0 0

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] *) **)	---	20.2/19.8	—	19.8
Number of bytes	---	16	14	16

*) see Multiplication

**) An error in division and modulo (divisor = 0) results in a execution time of 3.8 µs.

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 division by 0 is attempted, 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 x x x x x x x x x	
1	L K100	1 0 0	
2	/ W6	5	2 0
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]	---	20.6 to 20.2		20.2
Number of bytes	---	18	16	12

*) see Multiplication

**) see Division

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	= ?	

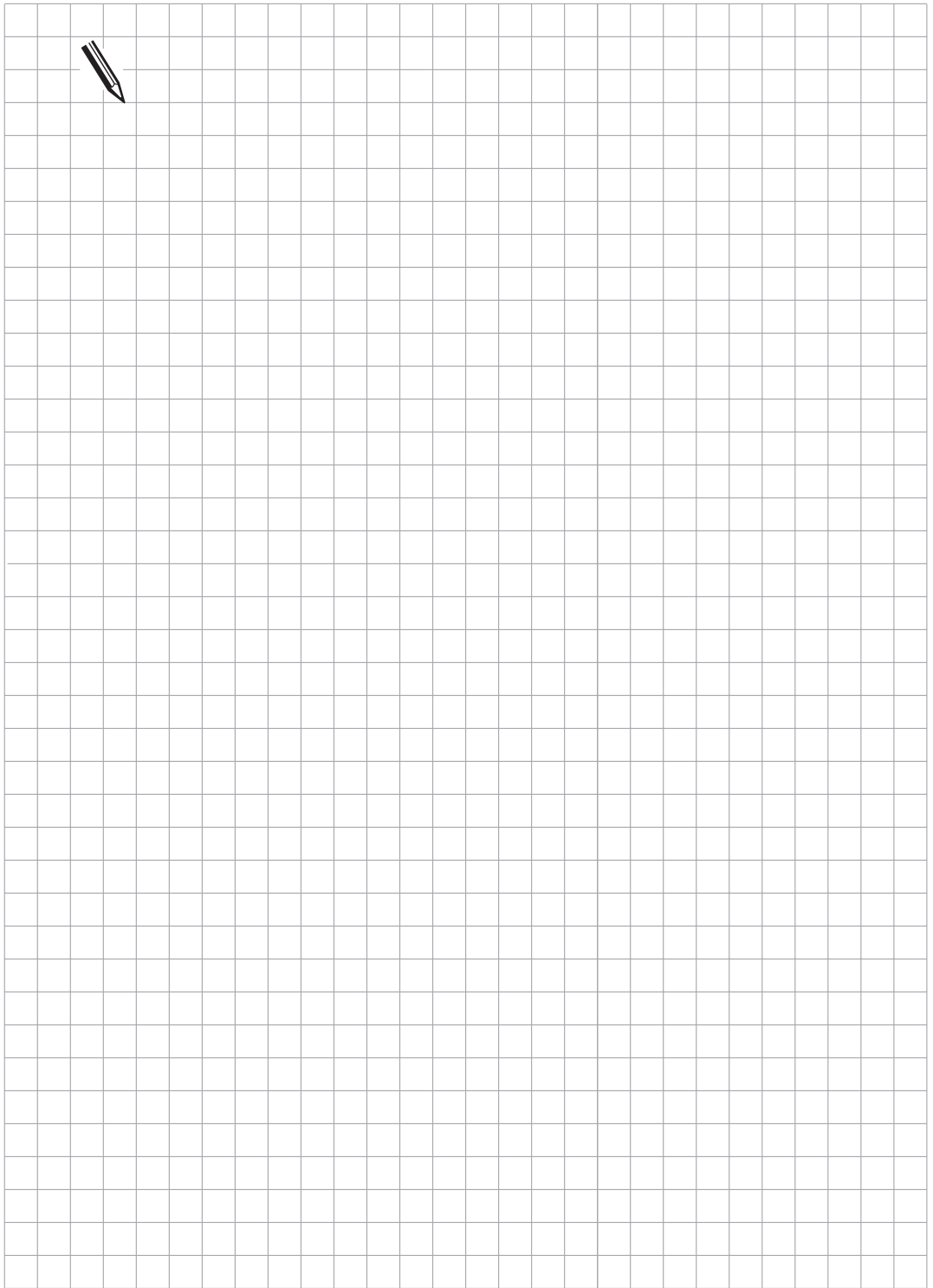
In the interests of clarity the contents of the Accumulator and the operand are shown in decimal notation. The 10 bit wide Accumulator allows the entry of the highest possible Accumulator contents (2 147 483 647).

Line	Instruction	Accumulator Contents	Operand Contents
		x x x x x x x x	
1	L W6	50	
2	MOD K15	5	15
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.



3.5 Comparisons

3.5.1 EQUAL TO (==)

Abbreviation for the PLC Editor: == (EQUAL)

	Byte/Word/Double	Constant
Execution time [µs]	1.8 to 2.4	2.0
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 x	
1	L K16000	1 6 0 0 0	
2	== D8	Bit 31 ... x x x x x x x 7 0 0	1 5 0 0 0
3	= M500	... x 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]	1.8 to 2.4	2.0
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 greater 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 x x x x x x x x x x x x	
1	L K16000	1 6 0 0 0 0	
		Bit 31 7 0	
1	< D8	... x x x x x x x 0 x x x x x x x x	1 5 0 0 0 0
2	= M500	... x x x x x x x 0 x 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 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]	1.8 to 2.4	2.0
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 x x x x x x x x x x x x	
1	L K16000	1 6 0 0 0 0	
		Bit 31 . . . 7 0	
1	> D8	... x x x x x x x 1 x x x x x x x x	1 5 0 0 0 0
2	= M500	... x x x x x x x 1 x 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 ($\leq$)

Abbreviation for the PLC Editor: <= (LESS EQUAL)

	Byte/Word/Double	Constant
Execution time [μs]	1.8 to 2.4	1.8 to 2.0
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 x x x x x x x	
1	L K16000	1 6 0 0 0	
1	<= D8	Bit 31 ... x x x x x x x 0 x x x x x x x x 0	1 5 0 0 0
2	= M500	... x x x x x x x 0 x x x x x x x x 0	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]	1.8 to 2.4	1.8 to 2.0
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 x x x x x x x x x x x x	
1	L K16000	1 6 0 0 0 0	
		Bit 31 . . . 7 0	
1	>= D8	... x x x x x x x 1 x x x x x x x x	1 5 0 0 0 0
2	= M500	... x x x x x x x 1 x 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]	1.8 to 2.4	1.8 to 2.0
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:

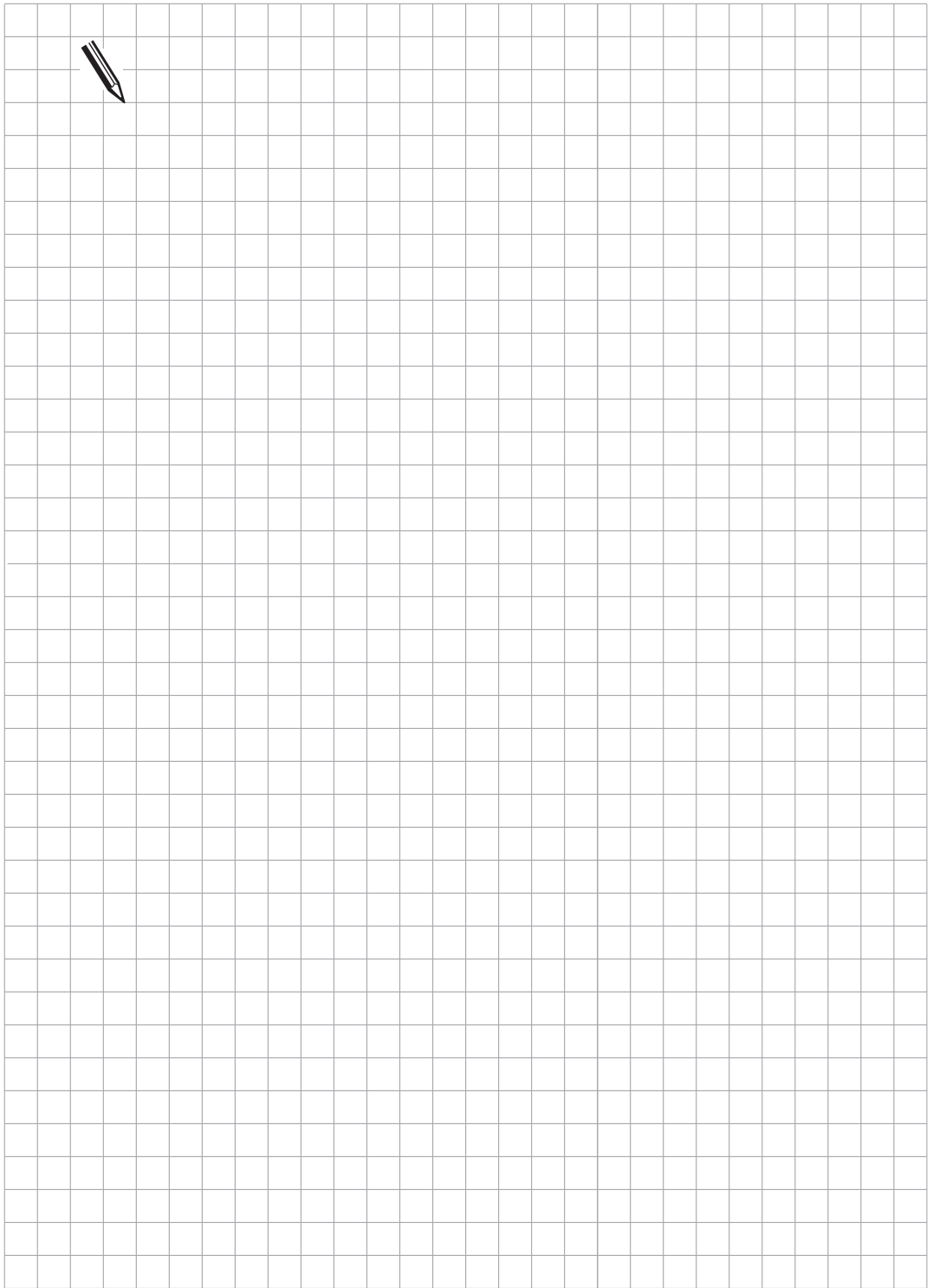
A constant is to be compared with the contents of Doubleword D8. The result is then assigned to marker M500.

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 x x x	
1	L K16000	1 6 0 0 0 0	
		Bit 31 . . . 7 0	
1	<> D8	... x x x x x x x 1 x x x x x x x x	1 5 0 0 0 0
2	= M500	... x x x x x x x 1 x 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 summarized 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]	1.6	2.6
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]	2.0	3.2
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]	1.6	2.6
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]	2.0	3.2
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]	2.0	3.2
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]	2.4	3.8
Number of bytes	8	8

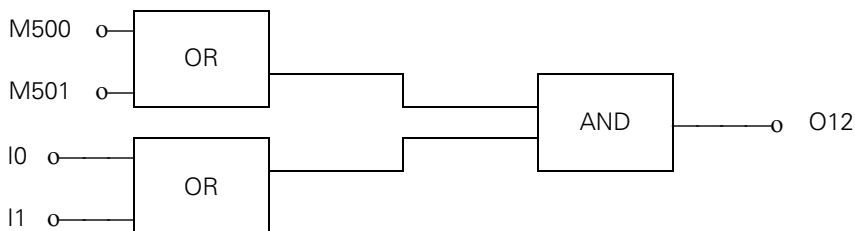
Operands: none

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.

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

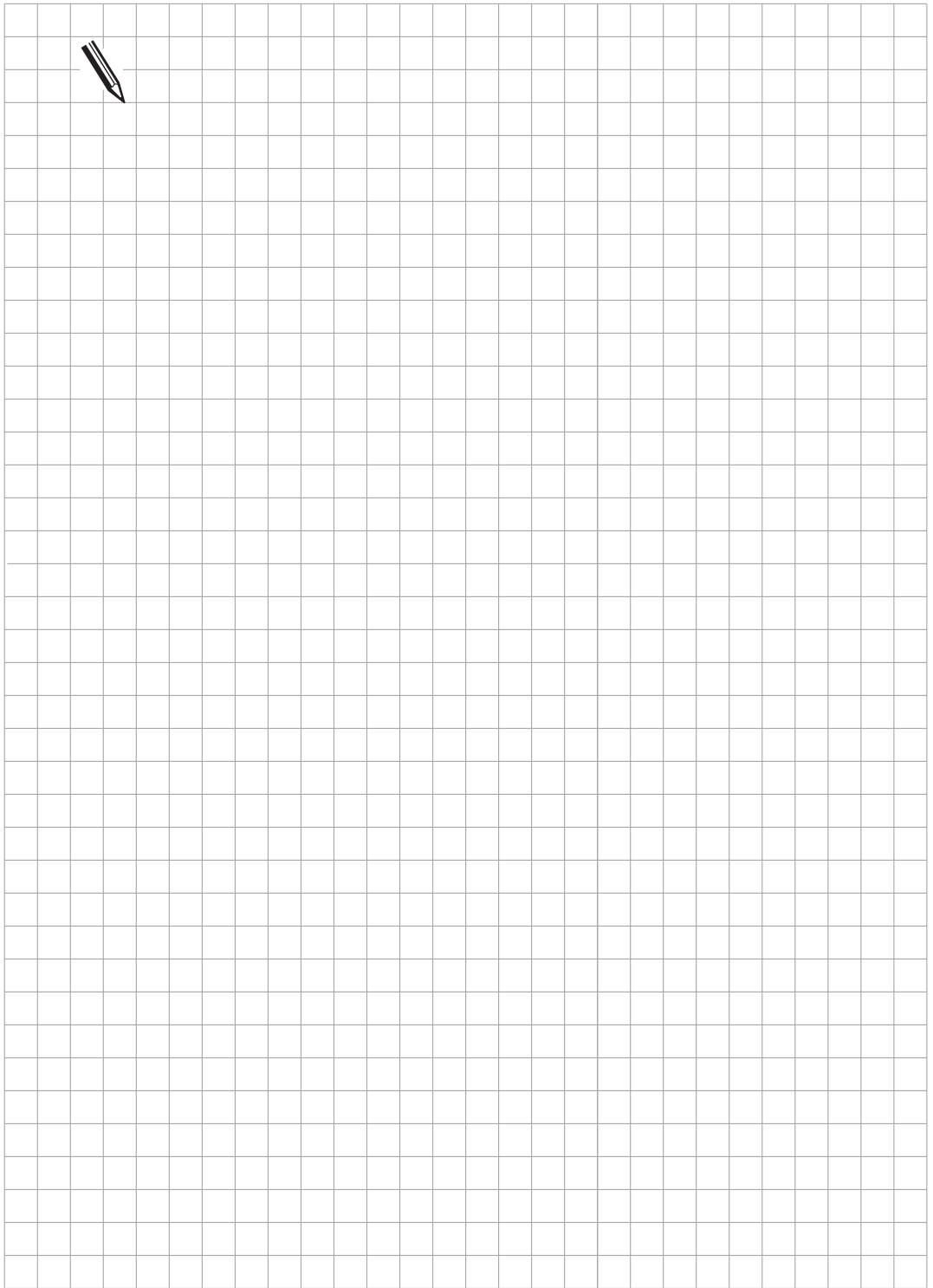
Line	Instruction	Accumulator Contents	Operand Contents
------	-------------	----------------------	------------------

Diagram illustrating the Program Stack (Memory Stack) after the first instruction:

- The stack grows downwards (higher memory addresses at the top, lower memory addresses at the bottom).
- Stack contents (from top to bottom):
 - Address 31: Instruction `L M500` (bit 7 = 0)
 - Address 30: Instruction `O M501` (bit 7 = 1)
 - Address 29: Instruction `A [` (bit 7 = 1)
 - Address 28: Instruction `L I0` (bit 7 = 0)
 - Address 27: Instruction `O I1` (bit 7 = 1)
 - Address 26: Instruction `]` (bit 7 = 1)
 - Address 25: Instruction `= O12` (bit 7 = 1)
- The instruction `]` at address 26 has its bit 7 (the carry) set to 1, which is pushed onto the stack at address 25.
- The stack grows downwards, with higher addresses at the top and lower addresses at the bottom.

Line 7: The result of the complete logical process is assigned to Output O12.

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 summarized respectively for the "open parentheses" and corresponding "close parentheses" commands.

3.7.1 ADDITION [] (+[])

Abbreviation for PLC Editor: + [] (PLUS [])

	Logic	Byte/Word/Double
Execution time [μs]	—	2.6
Number of bytes	—	6

Operands: none

3.7.2 SUBTRACTION [] (-[])

Abbreviation for PLC Editor: – [] (MINUS [])

	Logic	Byte/Word/Double
Execution time [μs]	—	3.2
Number of bytes	—	6

Operands: none

3.7.3 MULTIPLICATION [] (x[])

Abbreviation for PLC Editor: x [] (MULTIPLY [])

	Logic	Byte/Word/Double
Execution time [μs] *)	—	11.0
Number of bytes	—	12

Operands: none

3.7.4 DIVISION [] (/[])

Abbreviation for PLC Editor: / [] (DIVIDE [])

	Logic	Byte/Word/Double
Execution time [μs] *) **)	—	20.2
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

*) See Multiplication

**) See Division

3.7.5 REMAINDER [] (MOD[])

Abbreviation for PLC Editor: MOD [] (MODULO [])

	Logic	Byte/Word/Double
Execution time [µs] *) **)	—	20.6
Number of bytes	—	14

*) See Multiplication

**) See Division

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 is possible. 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.

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)
	DoublewordD12	= 15000	(decimal)
	DoublewordD36	= 100	(decimal)
	DoublewordD100	= ?	

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).

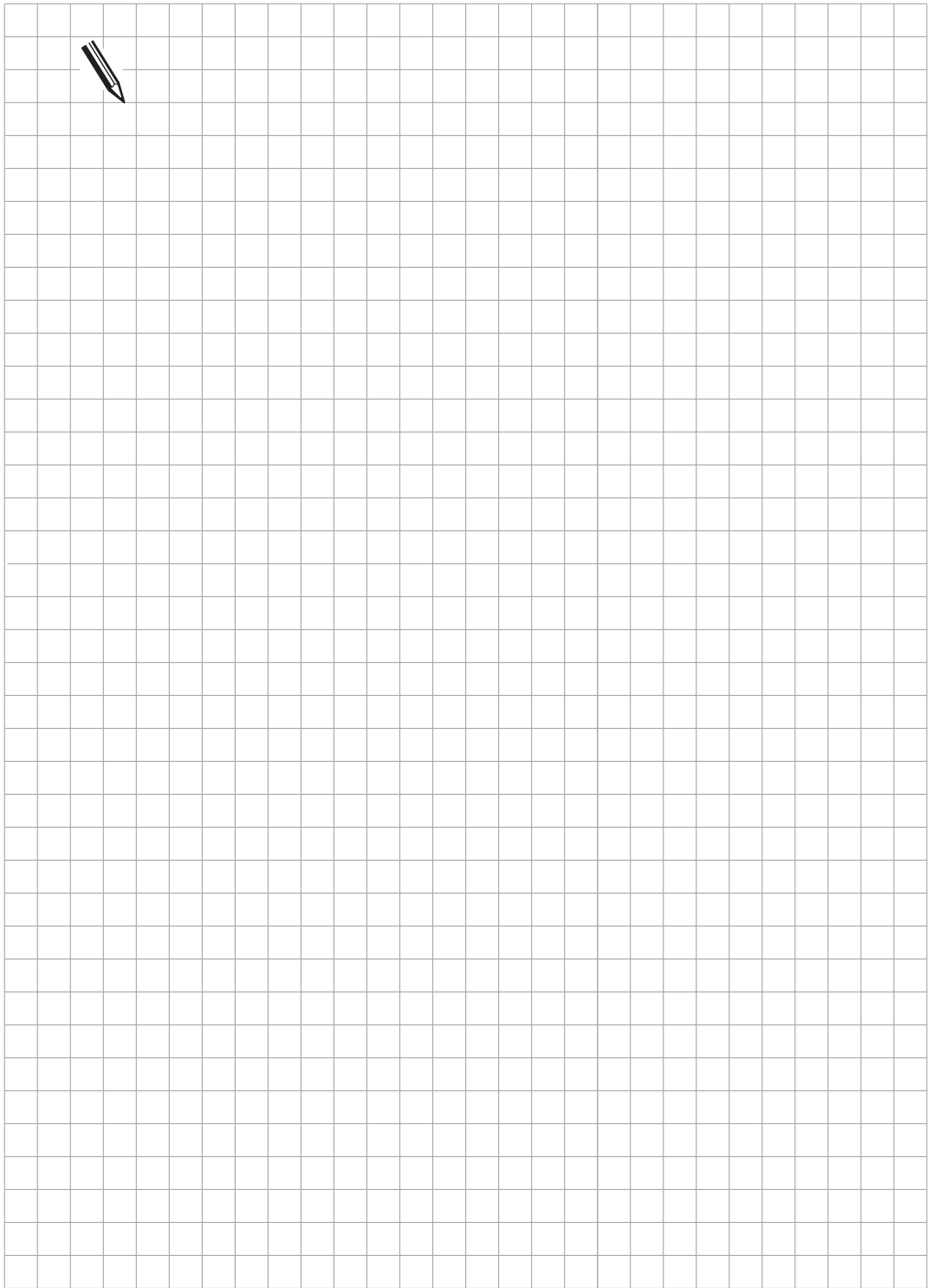
Command sequence without parentheses:

Line	Instruction	Accumulator Content	Operand Content
		x x x x x x x x x x	
1	L D12	1 5 0 0 0	1 5 0 0 0
2	+ K1000	1 6 0 0 0	
3	/ D36	1 6 0	1 0 0
4	= D100	1 6 0	1 0 0

Commend sequence with parentheses:

Line	Instruction	Accumulator Contents	Operand Contents
		x x x x x x x x x x	
1	L D12	1 5 0 0 0	1 5 0 0 0
2	+ [	1 5 0 0 0	----- Program-stack: 1 5 0 0 0
3	L K1000	1 0 0 0	
4	/ D36	1 0	
5	]	1 5 0 1 0	
6	= D100	1 5 0 1 0	

- 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]	---	3.0 to 3.2
Number of bytes	---	6

Operands: none

3.8.2 LESS THAN [] (<[])

Abbreviation for PLC Editor: < [] (LESS THAN [])

	Logic	Byte/Word/Double
Execution time [μs]	---	3.0 to 3.2
Number of bytes	---	6

Operands: none

3.8.3 GREATER THAN [] (>[])

Abbreviation for PLC Editor: > [] (GREATER THAN [])

	Logic	Byte/Word/Double
Execution time [μs]	---	3.0 to 3.2
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]	---	3.0 to 3.2
Number of bytes	---	6

Operands: none

3.8.5 GREATER THAN OR EQUAL TO [] (>=[])

Abbreviation for PLC Editor: >= [] (GREATER EQUAL [])

	Logic	Byte/Word/Double
Execution time [µs]	---	3.0 to 3.2
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]	---	3.0 to 3.2
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.

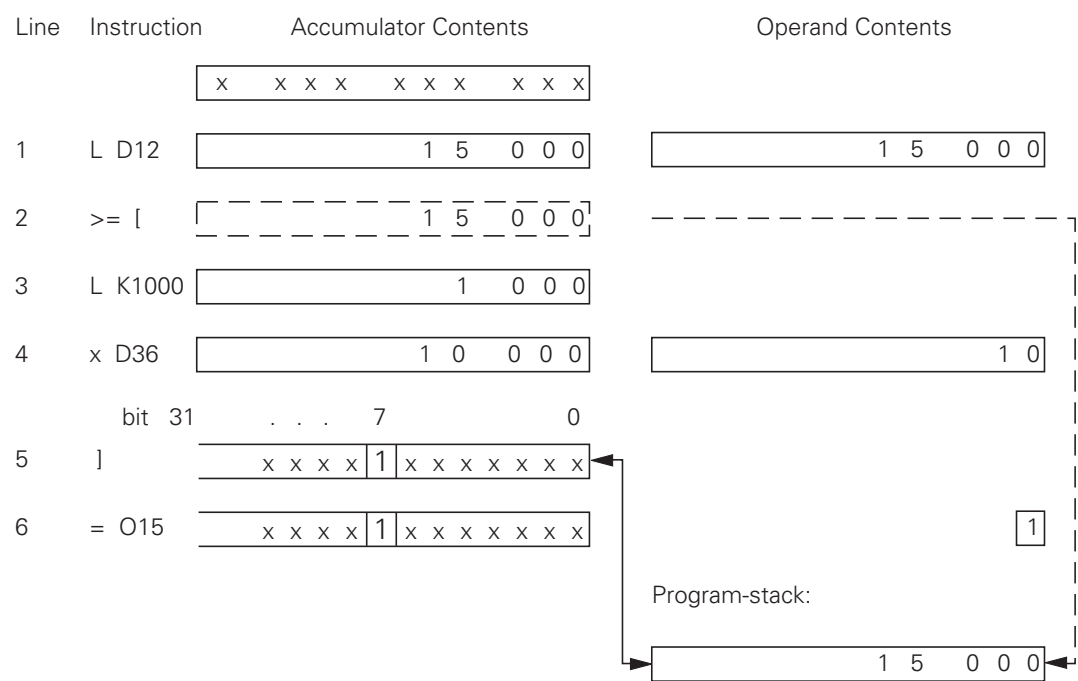
A direct transition from Word to Logic execution takes place with comparison commands. If the comparison condition is "true", the Logic Accumulator is set to "1". If the condition is not fulfilled, the Logic Accumulator is set to "0".

Example:

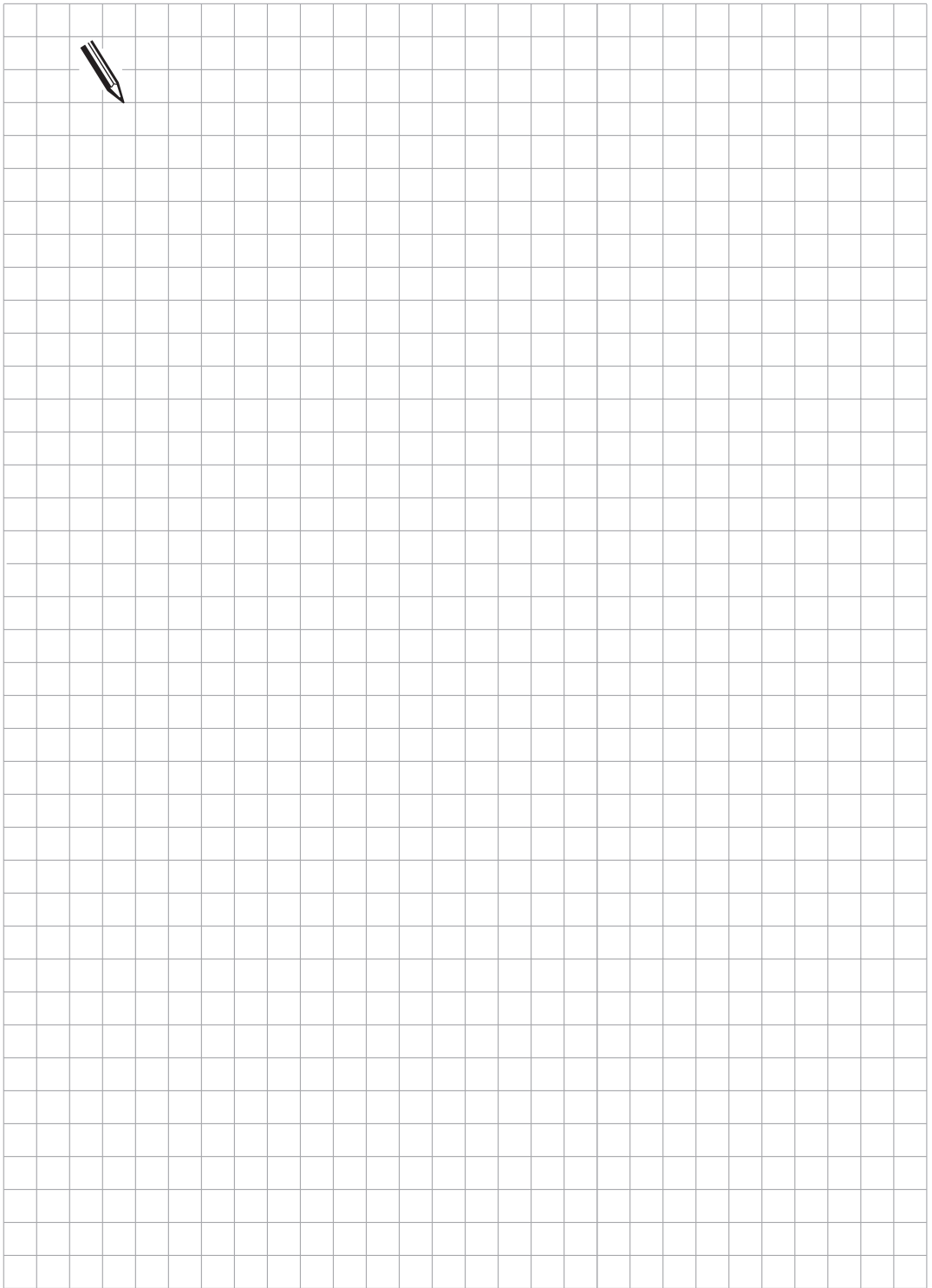
Initial state: Constant = 1000 (decimal)
 Doubleword D12 = 15000 (decimal)
 Doubleword D36 = 10 (decimal)
 Output O15 = ?

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]	$2.0 + 0.2 \times n$	$2.0 + 0.2 \times n$
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 number of the shift operations is determined via the operand. Thus the set bits, which are shifted beyond the Accumulator to the left, are lost; the Accumulator is filled with nulls from the right-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 left 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	<< 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]	$2.0 + 0.2 \times n$	$2.0 + 0.2 \times n$
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 = ?

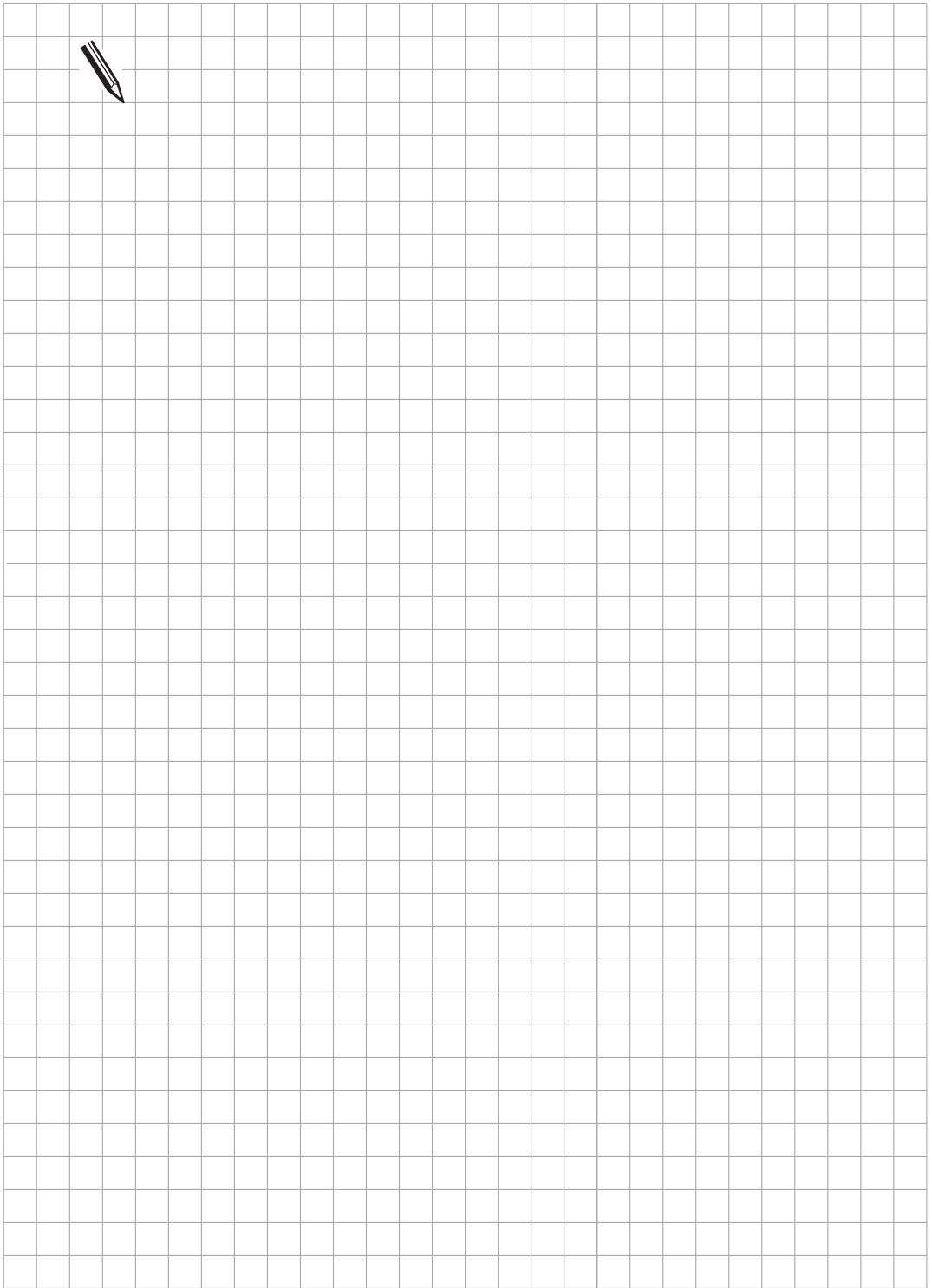
The Accumulator content is shown here in binary notation and the operand content in hexadecimal notation.

Line	Instruction	Accumulator Content	Operand Content
		xxxxxxx xxxxxxx xxxxxxx xxxxxxx	
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]	2.0 to 2.4	2.0
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								
		<table><tr><td>xx</td><td>xx</td><td>xx</td><td>xx</td></tr></table>	xx	xx	xx	xx					
xx	xx	xx	xx								
1	L D8	<table><tr><td>00</td><td>00</td><td>3E</td><td>80</td></tr></table>	00	00	3E	80	<table><tr><td>00</td><td>00</td><td>3E</td><td>80</td></tr></table>	00	00	3E	80
00	00	3E	80								
00	00	3E	80								
2	BS K+0	<table><tr><td>00</td><td>00</td><td>3E</td><td>81</td></tr></table>	00	00	3E	81					
00	00	3E	81								
3	= D12	<table><tr><td>00</td><td>00</td><td>3E</td><td>81</td></tr></table>	00	00	3E	81	<table><tr><td>00</td><td>00</td><td>3E</td><td>81</td></tr></table>	00	00	3E	81
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]	2.2 to 2.6	2.2
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								
		<table border="1"><tr><td>xx</td><td>xx</td><td>xx</td><td>xx</td></tr></table>	xx	xx	xx	xx					
xx	xx	xx	xx								
1	L D8	<table border="1"><tr><td>00</td><td>00</td><td>3E</td><td>81</td></tr></table>	00	00	3E	81	<table border="1"><tr><td>00</td><td>00</td><td>3E</td><td>81</td></tr></table>	00	00	3E	81
00	00	3E	81								
00	00	3E	81								
2	BC K+0	<table border="1"><tr><td>00</td><td>00</td><td>3E</td><td>80</td></tr></table>	00	00	3E	80					
00	00	3E	80								
3	= D12	<table border="1"><tr><td>00</td><td>00</td><td>3E</td><td>80</td></tr></table>	00	00	3E	80	<table border="1"><tr><td>00</td><td>00</td><td>3E</td><td>80</td></tr></table>	00	00	3E	80
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]	2.2 to 2.8	2.2 to 2.4
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 = ?

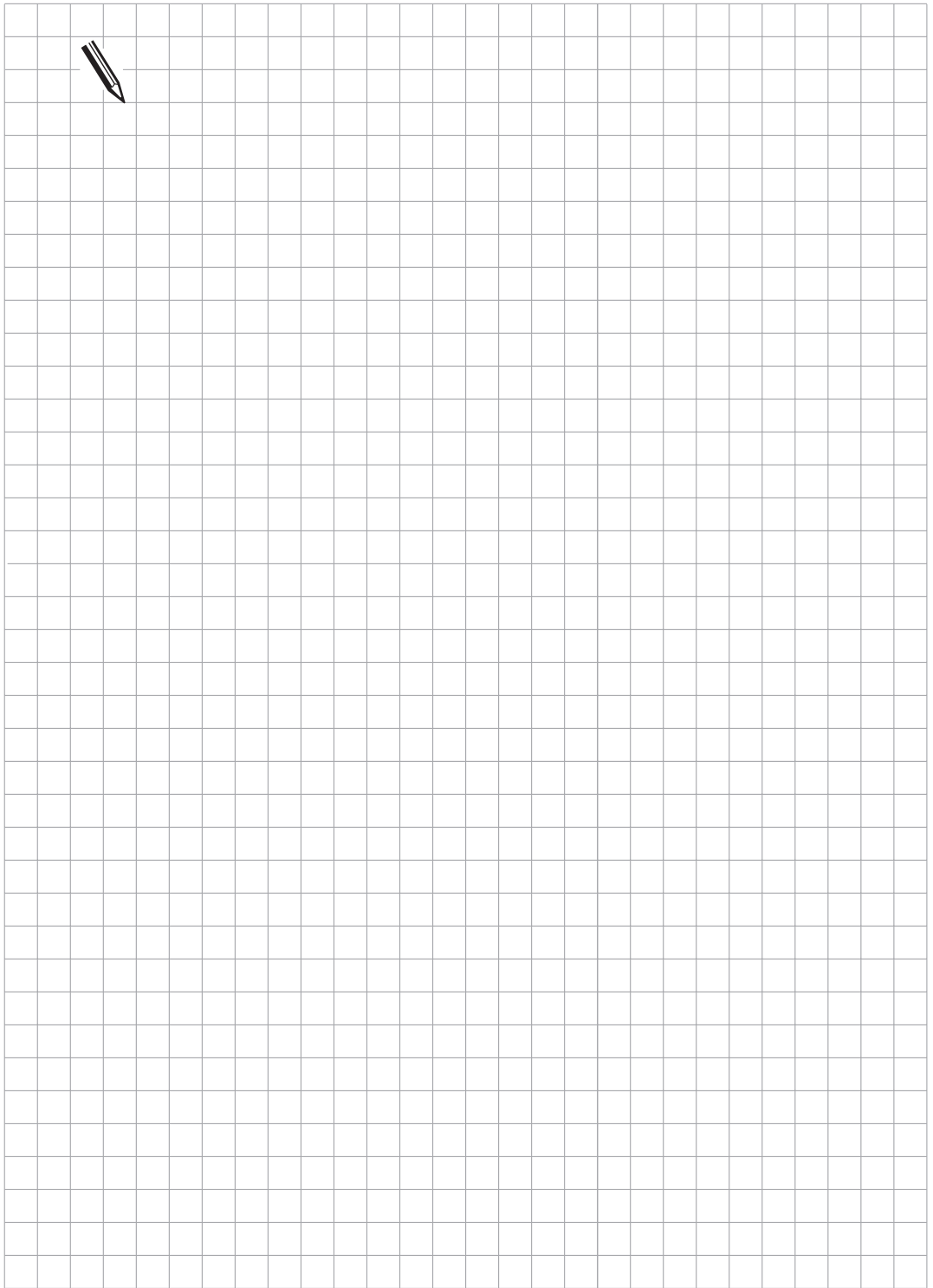
Word Accumulator and operand contents are shown here in hexadecimal notation, the Logic Accumulator in binary representation.

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 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)

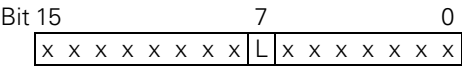
	Logic	Byte/Word	Double	Constant
Execution time [µs]	4.2	5.4/5.0	5.5	4.2
Number of bytes	24	26	22	24

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 organized 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:

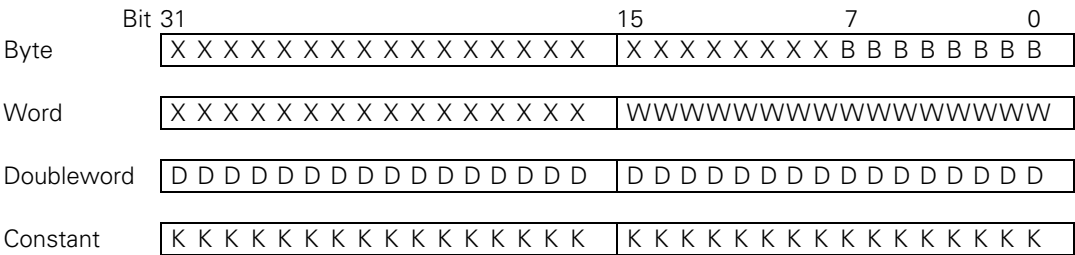


Word Execution with the PS Command

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:



3.11.2 Acquire data from the data stack (PL)

Abbreviation for PLC Editor: PL (PULL)

	Logic	Byte/Word	Double	Constant
Execution time [μs]	3.4	3.8	4.2	---
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)

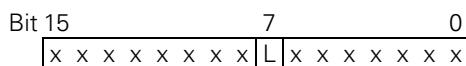
Execution time [μs]	3.0
Number of bytes	20

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 organized 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:



3.11.4 Load word accumulator onto the data stack (PSW)

Abbreviation for PLC Editor: PSW (PUSH WORDACCU)

Execution time [µs]	3.4
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]	2.2
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]	2.6
Number of bytes	16

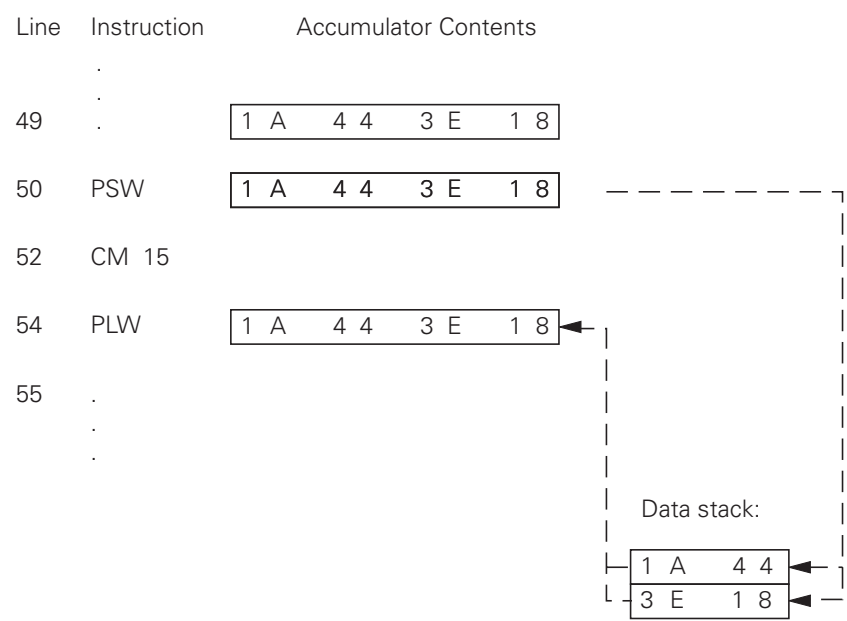
Operands: none

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 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]	2.2	---
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]	3.0 to 3.4	1.0 to 1.4
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]	3.0 to 3.4	1.0 to 1.4
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	70	
		...	x x x x x x x X x x x x x x x	
1	L I5	...	x x x x x x x 1 x x x x x x x	1
2	JPT 10	...	x 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 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]	5.0	---
Number of bytes	22	
Special Library Call:		
Execution time [µs]	6.2	---
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]	6.8 to 7.2	1.0 to 1.4
Number of bytes	26 (24)	
Special Library Call:		
Execution time [µs]	7.4	1.4
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]	6.8 to 7.2	1.0 to 1.4
Number of bytes	26 (24)	
Special Library Call:		
Execution time [µs]	7.4	1.4
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 Content	Operand Content
	Bit	31 7 0	
		... x x x x x x x X x x x x x x x	
1	L I5	... x x x x x x x 0 x x x x x x x	0
2	CMF 10	... x x x x x x x 0 x x x x x x x	
3	L M100	... x 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 x 0 x x x x x x x	0
502	OM 500	... x x x x x x x 1 x x x x x x x	1
503	= O20	... x 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]	2.0
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 Jump Label (LBL)

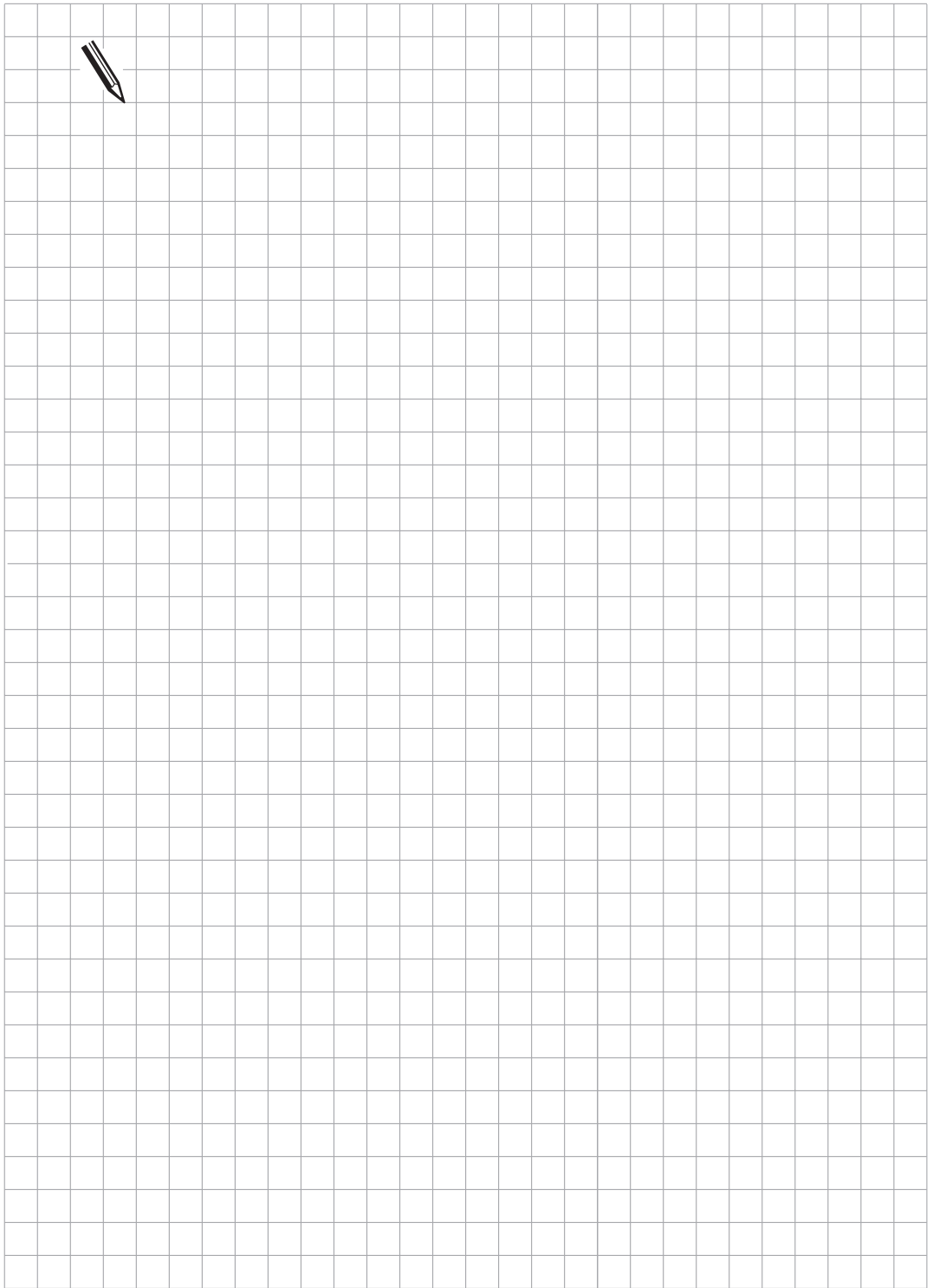
Abbreviation for PLC Editor: LBL (LABEL)

Execution time [μs]	0
Number of bytes	0

Operands: 0 to 511

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.



3.13 CASE statement

3.13.1 Indexed call module (CASE)

Abbreviation for PLC Editor: CASE (CASE OF)

	Byte	Word
Execution time [μs]	12.0	12.0
Number of bytes	46	44

4 Bytes must be added to the length for each entry in the table (CM).

Operands: B, W

Operation:

With the CASE command, a specific subprogram may be selected from a list of Call Modules (CM). These CM commands immediately follow the CASE command and are internally numbered in an ascending sequence from 0 to a maximum of 127. The content of the operands (B, W) addresses the required Module.

3.13.2 End indexed Call Module (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 in conjunction with the CASE command. It must immediately follow the list of CM commands.

Structure of a CASE statement:

```
Internal Addressing (0 to max. 127)
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 loaded into the operand

Line 2: Call Module for operand contents 0

Line 3: Call Module for operand contents 1

Line 4: Call Module for operand contents 2

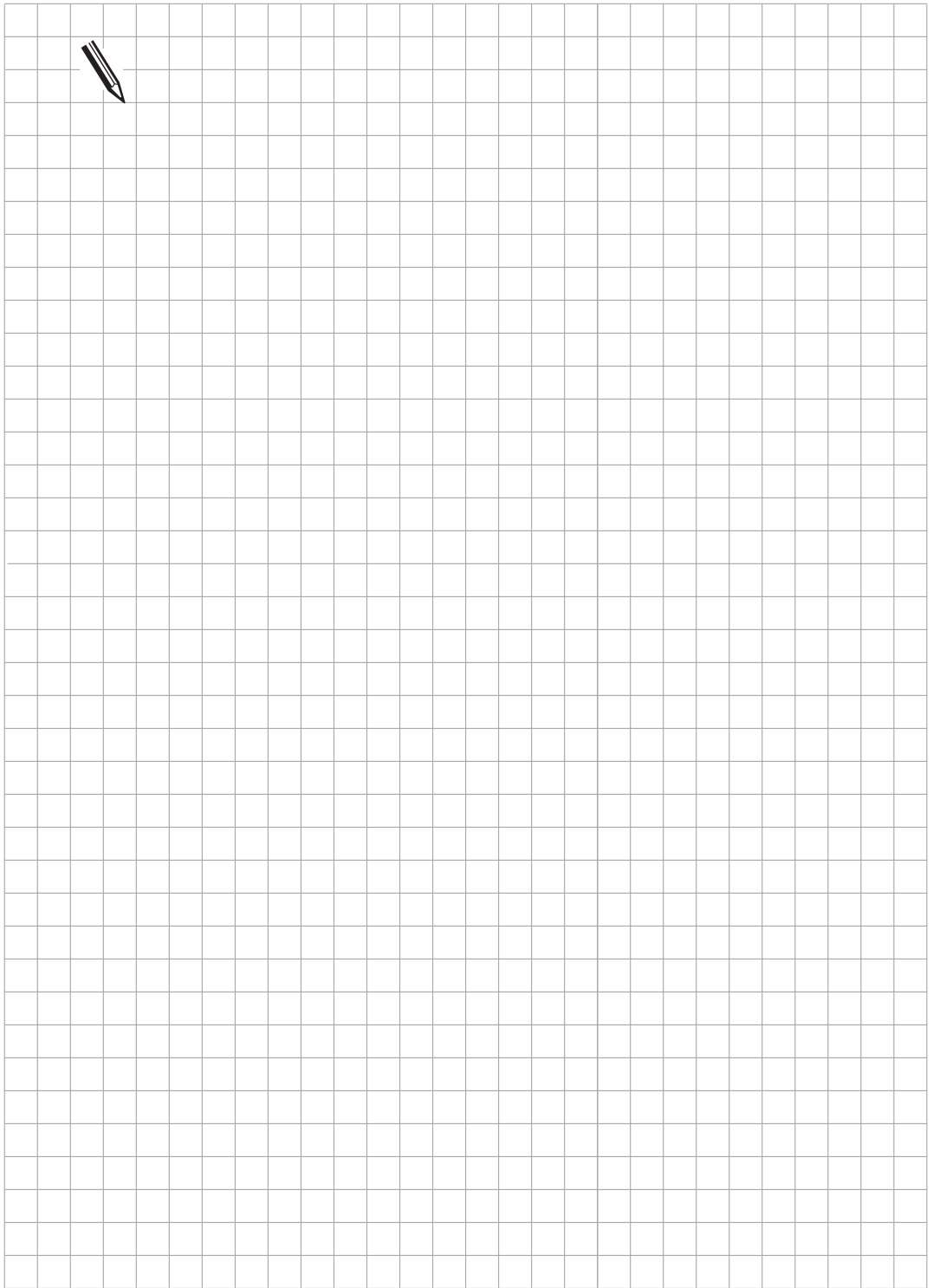
Line 5: Call Module for operand contents 3

Line 6: Call Module for operand contents 4

Line 7: Call Module for operand contents 5

Line 8: Call Module for operand contents 6

Line 9: End of the CASE statement



4 PLC Modules

A range of PLC Modules is available for certain PLC functions that cannot be performed with the PLC commands, or only in a complicated way. They are described below in more detail.

4.1 Module 9000: Copy a marker block

Module 9000 can be used to copy a block with a certain number of markers, beginning with the start address, in ascending order to another marker range, beginning at a certain target address.

Care must be taken that the blocks do not overlap, and that there is adequate space for the copied block after the target address.

Module call:

PS K/B/W/D	1st Marker of the block to be copied (start address)
PS K/B/W/D	1st Marker of the target address for copying
PS K/B/W/D	Length of the marker block
CM 9000	

If the block is not copied correctly, marker M3171 will be set.

4.2 Module 9001: Copy a Word block

Module 9001 can be used to copy a block with a certain number of Bytes, Words or Doublewords, beginning with the start address, in ascending order to another area, beginning at a defined target address.

The length is always given in Bytes.

Care must be taken that the blocks do not overlap, and that there is adequate space for the copied block after the target address.

Module call:

PS K/B/W/D	1st Byte of the block which is to be copied (start address)
PS K/B/W/D	1st Byte of target address for copying
PS K/B/W/D	Length of the block in Bytes
CM 9001	

If the block is not copied correctly, marker M3171 will be set.

4.3 Module 9010: Indexed reading of Byte

Module 9011: Indexed reading of Word

Module 9012: Indexed reading of Doubleword

Modules 9010, 9011 and 9012 can be used to read a Byte, Word or Doubleword, and load the result onto the STACK. This allows indexed reading of the Word memory when a Variable is used to define the memory location.

Module call:

PS K/B/W/D Address of a Byte, where the contents address another Byte, the contents of which are to be copied to the target address.

CM9010
PL B Target address - Byte

PS K/B/W/D Address of a Word, where the contents address another Word, the contents of which are to be copied to the target address.

CM9011
PL W Target address - Word

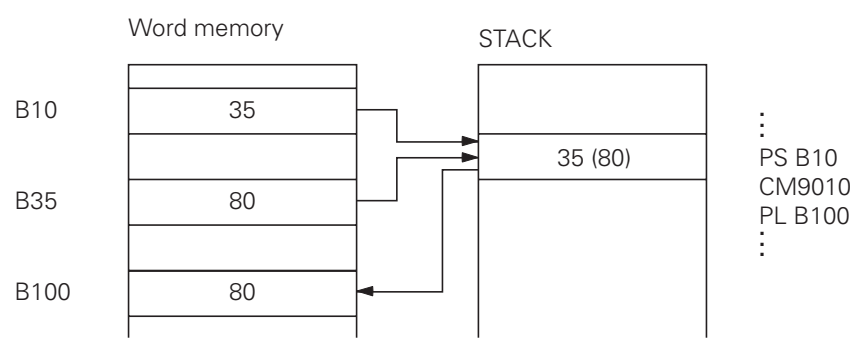
PS K/B/W/D Address of a Doubleword, where the contents address another Doubleword, the contents of which are to be copied to the target address.

CM 9012
PL D Target address - Doubleword

The addresses which are given must lie within the valid range (0 to 1023). For Word addressing the address must be divisible by two, for Doubleword addressing by four.

If the Byte, Word or Doubleword is not read correctly, marker M3171 will be set.

Example for Module 9010:



4.4 Module 9020: Indexed writing of Byte

Module 9021: Indexed writing of Word

Module 9022: Indexed writing of Doubleword

Modules 9020, 9021 and 9022 can be used to write the contents of a Byte, Word or Doubleword to a given address in the Word memory. This makes indexed writing possible in the Word memory, when a variable is used to define the memory cell.

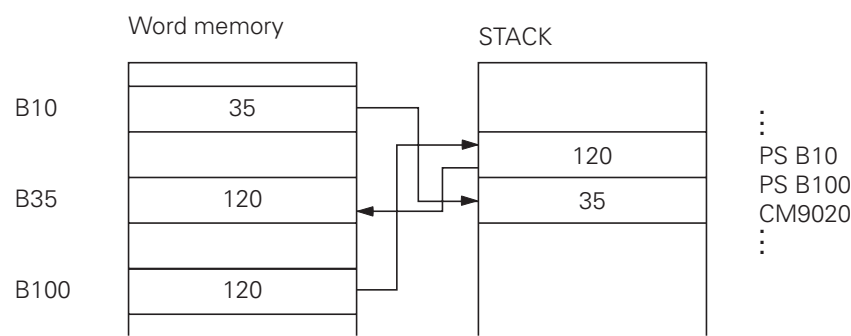
Module call:

PS K/B/W/D	Address of a Byte whose contents address another Byte into which the value of a second Byte is entered.
PS K/B/W/D CM 9020	Address of the Byte where the value to be transferred is stored.
PS K/B/W/D	Address of a Word, where the contents address another Word, where a value is to be entered from a second Word.
PS K/B/W/D CM 9021	Address of the Word, where the value to be transferred is stored.
PS K/B/W/D	Address of a Doubleword, where the contents address another Doubleword where a value is to be entered from a second Doubleword.
PS K/B/W/D CM 9022	Address of the Doubleword, where the value to be transferred is stored.

The addresses which are given must lie within the valid range (0 to 1023). For Word addressing, the address must be divisible by two, and for Doubleword addressing by four.

If the Byte, Word or Doubleword is not transferred correctly marker M3171 will be set.

Example for Module 9020:



4.5 Module 9032: Transfer machine parameter values to the PLC

Module 9032 can be used to transfer the value of a machine parameter, designated by its number and index, to the PLC.

The value of the machine parameter is transferred to the PLC as an integer. The decimal point is shifted by the number of possible places after the decimal point. An index must always be defined. For machine parameters without index, enter the value zero.

Module call:

PS K/B/W/D	Number of the machine parameter, the value of which is to be read
PS K/B/W/D	Machine parameter index
CM 9032	
PL B/W/D	Address for machine parameter value

If the machine parameter is read incorrectly, marker M3171 will be set.

4.6 Module 9035: Transfer status information to the PLC

Module 9035 can be used to transfer NC status information such as the selected handwheel axis, number and type of handwheel axes (bit coded) and handwheel interpolation factor to the PLC.

Module call:

PS B/W/D/K	Type of NC status information
CM 9035	
PL B/W/D	NC status value

If the data are incorrectly transferred, marker M3171 is be set.

Type of NC status information:	NC status value:
0 ... 8	Not yet implemented
9: Selected handwheel axis	-1: No axis or several axes 0: X axis 1: Y axis 2: Z axis 3: IV axis
10: Number and type of handwheel axes	Bit 0 +0 = X axis with handwheel axis +1 = X axis is handwheel axis Bit 1 +0 = Y axis is not handwheel axis +2 = Y axis is handwheel axis Bit 2 +0 = Z axis is not handwheel axis +4 = Z axis is handwheel axis Bit 3 +0 = IV axis is not handwheel axis +8 = IV axis is handwheel axis
11: Handwheel interpolation factor X axis	0 ... 10
12: Handwheel interpolation factor Y axis	0 ... 10
13: Handwheel interpolation factor Z axis	0 ... 10
14: Handwheel interpolation factor IV axis	0 ... 10

4.7 Module 9036: Transfer PLC status information to the NC

Module 9036 transfers all or individual axes from the PLC to the NC (for display of interpolation factors and handwheel functions).

Module call:

PS B/W/D/K	Type of PLC status information
PS B/W/D/K	PLC status value
CM 9036	
PL B/W/D	Status messages that are filed in the STACK:
	0: PLC status information was transferred
	1: Incorrect type of PLC status information
	2: Transferred PLC status information incorrect
	3: Write inhibited (e.g. through MP)

When the PLC status information has been transferred, Marker M3171 is reset.

Type of NC status information: PLC status value:

0: Handwheel interpolation factor 0 ... 10	
X axis	
1: Handwheel interpolation factor 0 ... 10	
Y axis	
2: Handwheel interpolation factor 0 ... 10	
Z axis	
3: Handwheel interpolation factor 0 ... 10	
IV axis	
4: Handwheel interpolation factor 0 ... 10	
all axes	
5: Handwheel interpolation factor 0 ... 10	
all axes	
6 to 9 reserved	
10: jog distance limitation	0 to 50 mm
	<0 or >50 = limit canceled

4.8 Module 9040: Transfer coordinate values to the PLC

Module 9040 transfers axis coordinates for all the NC axes to the PLC. Actual value, nominal value, reference value, servo lag and distance-to-go can be transferred. The values are loaded in the format 1/1000 mm or 1/1000°, beginning at the given target address, in five Doublewords for the X, Y, Z, and 4th axes.

Module call:

PS K/B/W/D	Target address for axis coordinates
PS K/B/W/D	Coordinate type
	0 = Actual value
	1 = Nominal value
	2 = Reference value
	3 = Servo lag
	4 = Distance-to-go

CM 9040

Regardless of whether individual axes are disabled by MP10, the values for all axes will always be read in. The coordinate value for an axis is undefined until the reference point for this axis has been passed over.

If the transfer is incorrect, marker M3171 will be set.

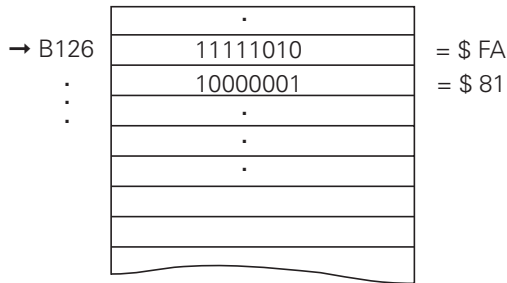
4.9 Data transfer by the PLC

The modules described in this section enable the PLC to transfer data via the RS-232-C/V.24 or RS-422/V.11 data interfaces (see Chapter "Data Interface" for transfer parameters).

The PLC assigns and enables the data interfaces with modules 9100 and 9101. The current status of the data interface is interrogated by module 9102.

The transmitting and receiving buffers for the PLC are both 128 characters long. Each STRING terminates in an END character in the transmitting or receiving buffer and therefore a STRING can contain a maximum of 127 characters. Modules 9105 and 9106 can be used to transfer a block of binary values (bytes) from the word memory.

Example: Transferring a binary block



When transferring binary data from address B126 in word memory, the ASCII characters "F", "A", "8", "1" etc. are transmitted in succession via the interface.

When binary data are transferred, every byte contains two ASCII characters. Therefore the transmitting and receiving buffers are 63 bytes long.

4.9.4 Module 9105: Transmit binary data via data interface

Module 9105 transfers a block of binary data from the word memory via the interfaces.

The appropriate interface must be assigned to the PLC with module 9100 before module 9105 is called.

Module call:

PS K/B/W/D Interface	0 = RS-232-C/V.24 (1 = RS-422/V.11)
PS K/B/W/D	Address of first byte in binary block (start address: 0 to 1023)
PS K/B/W/D CM 9105	Length of binary block in bytes (0 to 63)

If the module is incorrectly processed the block of binary data is not transferred and marker M3171 is set.

Potential errors:

- The call parameters are outside the permitted range.
- The interface is not assigned to the PLC.
- The transmitting buffer is not empty.
- A part of the binary block is above address 1024
(e.g. Start of binary block: 1020
Length of binary block: 60
=> End address 1080 is above the upper limit of 1024)
- The previous transfer contained an error but the interface was not re-assigned with module 9100.

4.9.5 Module 9106: Receive binary data via data interface

Module 9106 reads a block of binary data from the receiving buffer of an interface into the word memory of the PLC. This process clears the receiving buffer.
The length of the received binary block is fed back as the output.

The appropriate interface must be assigned to the PLC with module 9100 before module 9106 is called.

Module 9106 can only be run within a SUBMIT program.

Module call:

PS K/B/W/D	Interface	0 = RS-232-C/V.24 (1 = RS-422/V.11)
PS K/B/W/D	Address of first byte in binary block (Start address: 0 to 1023)	
CM 9106		
PL B/W/D	Length of binary block in bytes	

If the module is incorrectly processed the block of binary data is not read into the word memory and marker M3171 is set.

Potential errors:

- The call parameters of the module are outside the permitted range.
- The interface is not assigned to the PLC.
- The receiving buffer contains no data.
- The number of characters in the receiving buffer is uneven or greater than 126.
- The binary block cannot be stored at the specified address because it is too long (start address + length > 1024).
- The data in the receiving buffer contain characters that cannot be interpreted as hexadecimal values. Only the characters 0 to 9, A to F and control characters are permitted.

4.9.6 Module 9120 Position an auxiliary axis

A positioning movement is started by entering a target position (in the reference system), a feed rate, and a flag register. The axis is positioned in complete dependence on other processes in the control. However, it's path is not interpolated with other axes.

Constraints:

- The entered axis must have been activated through MP10 and declared as auxiliary axis through MP60.
- The values for rapid traverse, analog voltage for rapid traverse, acceleration etc. must be correctly set in the machine parameters.
- For axes with automatic display value reduction (modulo value in MP810.x) the axis is always moved in the direction of the shorter path to the target position, except when the target position is entered as an incremental value.
- There is no monitoring to prevent violation of the limit switch limits!
- The axis must be stationary. Any movement must first be canceled with module 9121.
- If the axis was in reference run mode, this condition is canceled. The positioning movement is always added to the current counter value.
- If the modules 9120, 9121 and 9122 are called repeatedly for the same axis during a PLC scan, the last command is executed.
- If in this axis the "positioning error" status is set, it is erased.

Possible errors:

- The transferred axis does not exist.
- The transferred axis was not declared through MP10 and MP60 to be an auxiliary axis.
- The axis is already being positioned.

Call:

```
PS      B/W/D/K <Axis>   (0 to 3 for X/Y/Z/4)
PS      B/W/D/K <Target position>   (in the reference system, format 0.001 mm)
PS      B/W/D/K <Feed rate>   (mm/min)
PS      B/W/D/K <Flag register> Bit 0   = 1: Target position incremental
                                           = 0: Target position absolute
```

CM9120

```
PL      B/W/D   <Error code>
          0: Positioning was started
          1: The transferred axis does not exist
          2: The axis is not configured as an auxiliary axis
          3: The axis was already positioned
          4: The absolute position is outside the modulo range
```

Error status after call: M3171 = 0: Positioning was correctly started
 = 1: Faulty positioning

4.9.7 Module 9121 Stop an auxiliary axis

A positioning movement that has been started with Module 9120 or Module 9123 can be canceled at any point with Module 9121.

Constraints:

- The entered axis must have been activated through MP10 and declared as auxiliary axis through MP60.
- If the modules 9120, 9121 and 9122 are called repeatedly for the same axis during a PLC scan, the last command is executed.

Possible errors:

- The transferred axis does not exist.
- The transferred axis was not declared through MP10 and MP60 to be an auxiliary axis.
- The entered axis is already stationary.

Call:

```
PS      B/W/D/K<Axis> (0 to 3 for X/Y/Z/4)
CM9121
PL      B/W/D  <Error code>
          0: Positioning movement is canceled
          1: Transferred axis does not exist
          2: Axis is not configured as an auxiliary axis
          3: The entered axis was already stationary
```

Error status after call: M3171 = 0: Positioning movement is stopped
= 1: Faulty execution

4.9.8 Module 9122 Poll the status of an auxiliary axis

For a certain axis a bit-encoded status word is transferred that provides information on the momentary condition of this axis.

Constraints:

- Status changed resulting from commands sent by the PLC to control the auxiliary axes (Modules 9120, 9121, 9123) are not recognized until the next PLC scan.
- After switch-on, Bit 1 (axis over reference point) is erased.
- It is possible to position the axis even without first crossing over the reference mark.

Possible errors:
The transferred axis does not exist.

Call:

```
PS      B/W/D/K <Axis> (0 to 3 for X/Y/Z/4)
CM9122
PL      B/W/D  <Status>
        Bit 0: 1= Axis is auxiliary axis
        Bit 1: 1= Axis has traversed the reference point
        Bit 2: 1= Axis is positioned
        Bit 3: 1= Axis direction is negative
        Bit 4: 1= Positioning error has occurred
```

Error status after call: M3171 = 0: Status was transferred
 = 1: Faulty execution

4.9.9 Module 9123 Traverse the reference point of an auxiliary axis

This module starts a positioning movement in a given direction that continues until a reference point has been traversed or the positioning movement has been canceled with the module 9121. Because of the possible problems finding a target position during referencing (direction reversal, etc.) it is necessary to stop when the reference point has been found.

Constraints:

- The entered axis must have been activated through MP10 and declared as auxiliary axis through MP60.
- The values for rapid traverse, analog voltage for rapid traverse, acceleration etc. must be correctly set in the machine parameters.
- There is no monitoring to prevent violation of the limit switch limits!
- The axis must be stationary. Any movement must first be canceled with module 9121.
- Feed rate override is not accounted for.
- The condition "reference mark not yet traversed" is set for the axis.
- An already existing reference point in this axis is erased, but not the position value of the axis. This is not re-initialized until the reference point has been traversed.
- If the modules 9120, 9121 and 9122 are called repeatedly for the same axis during a PLC scan, the last command is executed.
- If in this axis the "positioning error" status is set, it is erased.
- The positioning movement is immediately canceled when the reference point is reached. Due to the braking path of the axis, however, it stops somewhat behind the reference point in the direction of traverse.

Possible errors:

- The transferred axis does not exist.
- The transferred axis was not declared through MP10 and MP60 to be an auxiliary axis.
- The axis is already being positioned.

Call:

```
PS      B/W/D/K <Axis>  (0.to.3 for X/Y/Z/4)
PS      B/W/D/K <Feed rate>      (mm/min)
PS      B/W/D/K <Flag register>  Bit 0    = 1: negative traverse direction
                                           = 0: positive traverse direction
```

CM9123

```
PL      B/W/D  <Error code>
          0: Positioning was started
          1: The transferred axis does not exist
          2: The axis is not declared as an auxiliary axis
          3: The axis was already positioned
```

Error status after call: M3171 = 0: Positioning was started
= 1: Faulty execution

4.9.10 Module 9124 Override for PLC axis

An override value can be entered to influence the traversing speed of a PLC axis.

Constraints:

- The entered axis must have been activated through MP10 and declared as auxiliary axis through MP60.
- The override value can lie between 0% and 100.00% (resolution 0.01%) and must be transferred as an integer (0 ... 10 000).
- When movement starts, the last transferred override value is considered.
- After a reset or after an interruption of the PLC program the override value in each PLC axis is set to 100.00%.
- It is also possible to call the module while a PLC axis is moving.
- The module can be called in the same PLC scan in addition to a module from the group (9120/9121/9123). It does not become effective, however, until the previously called module is finished.

Possible errors:

- The transferred axis does not exist.
- The transferred axis was not declared through MP10 and MP60 to be an auxiliary axis.
- The transferred override value is incorrect.

Call:

```
PS B/W/D/K <Axis> (0..3 for X/Y/Z/IV)
PS B/W/D/K <Override> (Format 0,01%)
CM 9124
PL B/W/D <Error code>
0: Override was set
1: Transferred axis is invalid
2: The axis is not defined as a PLC axis
3: Override value is incorrect
```

Error status after Call: M3171 =0: Override was set
 =1: For error condition see above

4.9.11 Module 9150 Run an NC block

With module 9150 the PLC can, during an active M/S/T output, define an NC block that is then executed after acknowledgment of the M/S/T strobe before the control continues with the NC program. An NC program can also be defined if a program is not being run. It is then executed immediately.

Call:

```
PS B/W/D/K <Command code>
PS B/W/D/K <Address the parameter block in the B/W/D range>
CM 9150
PL B/W/D <Error code>
```

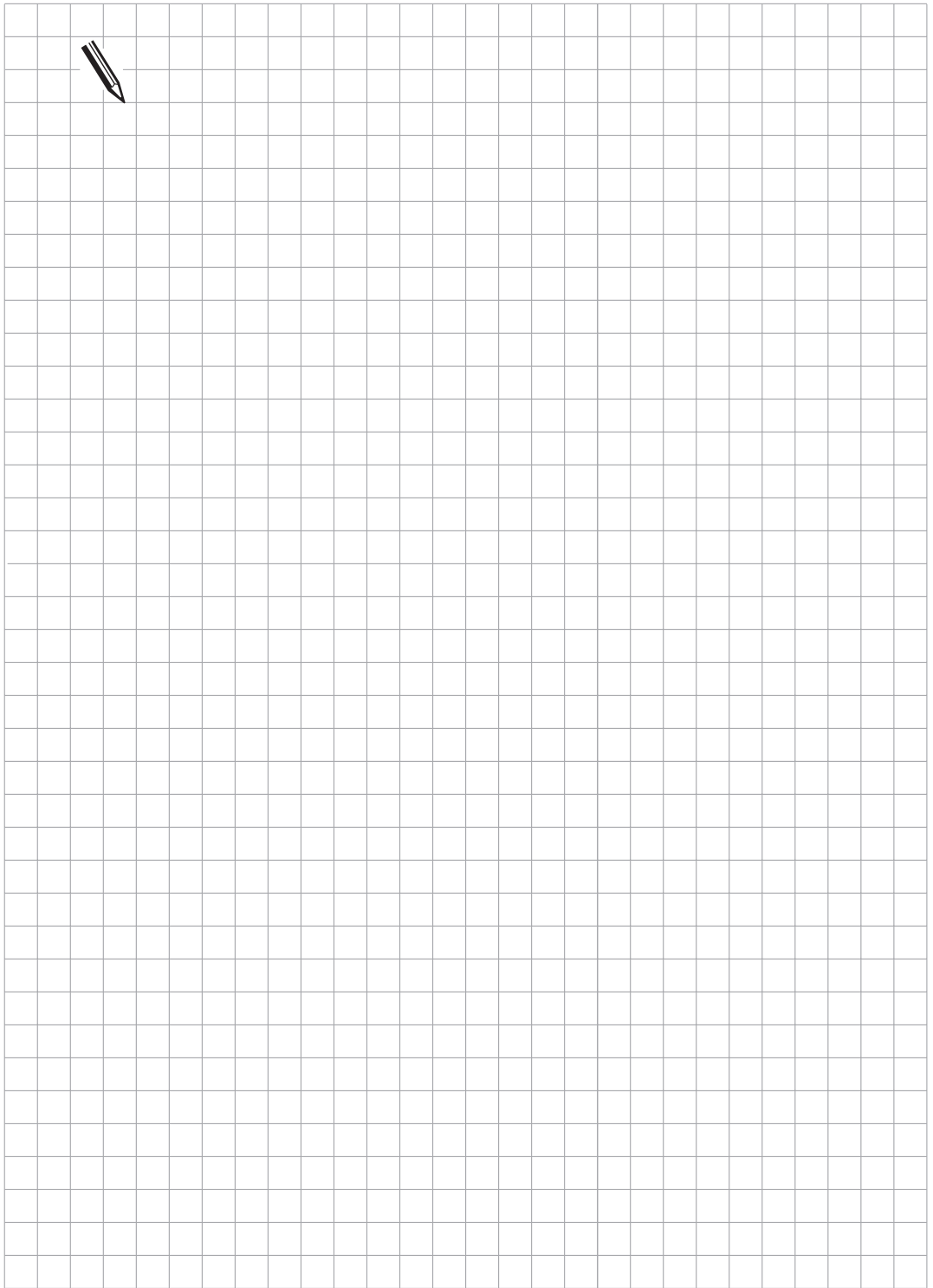
Error code: 0 = NC block was inserted
 1 = NC program was started, but no M/S/T strobe
 2 = Unknown error code
 3 = False address in the B/W/D range

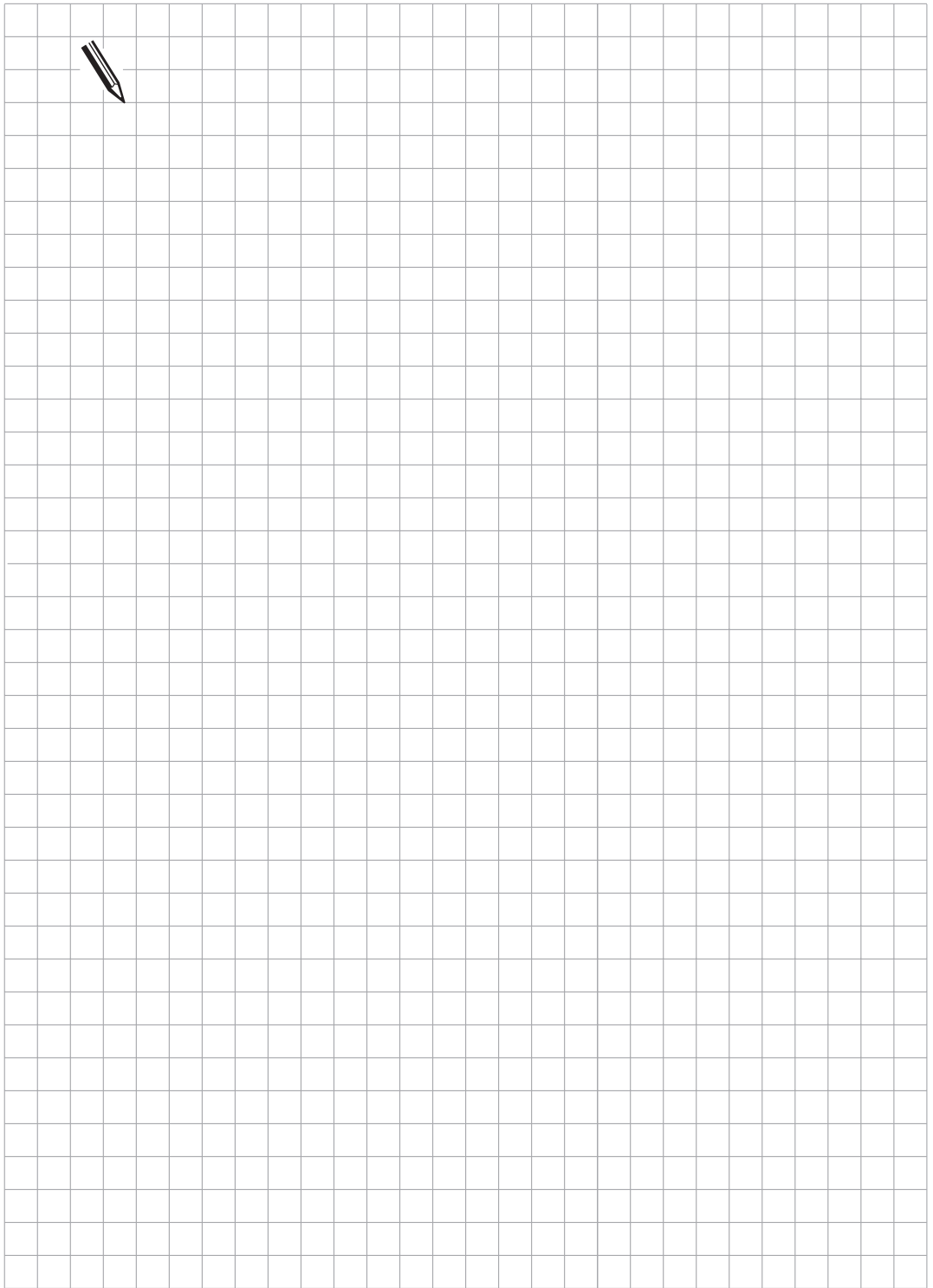
Error status after call: M3171 = 0 NC block was inserted
 = 1 for error condition see above

At present the command code <0> is implemented for TOOL CALL

Parameters :

B<Adr+0> active elements bit-coded
Bit 0 =1: Tool number, otherwise modal
Bit 1 =1: Tool axis, otherwise modal
Bit 2 =1: Spindle rpm, otherwise modal
Bit 3 =0
Bit 4 =0
B<Adr+1> Tool axis (0/1/2/3 = X/Y/Z/IV)
W<Adr+2> Tool number
D <Adr+ 4> Spindle rpm (format 0.001 rpm)





5 Compatibility with TNC 355

The flag ranges for the TNC 355 have also been used for the TNC 360 to enable PLC routines created for the TNC 355 to be run on the TNC 360 as well.

However there are many functions that can be programmed much more simply with PLC words with the TNC 360. PLC marker M2719 must be set to activate the strobes for word processing.

Marker	Function	Set	Reset
M2719	Deactivate the TNC 355 mode 0 = Activate strobes for marker range 1 = Activate strobes for word processing	PLC	PLC

Meaning	M2719 = 1		M2719 = 0	
	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	–	–	2599	M2560 to M2567 M2112 to M2119
Read in strobe tool number	–	–	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.

5.1 PLC Program conversion

PLC programs created for the TNC 355 can also be used on the TNC 360.

When these PLC programs are downloaded to the TNC 360 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 360 using the appropriate PLC words.

Marker	Function	Set	Reset	TNC 360/335
M2032	T Code 1st bit (lsb)	NC	NC	W262 W264
M2033	T Code 2nd bit	NC	NC	
M2034	T Code 3rd bit	NC	NC	
M2035	T Code 4th bit	NC	NC	
M2036	T Code 5th bit	NC	NC	
M2037	T Code 6th bit	NC	NC	
M2038	T Code 7th bit	NC	NC	
M2039	T Code 8th bit (msb)	NC	NC	
M2050	Operating mode: Programming and editing	NC	NC	W272
M2051	Operating mode: Manual	NC	NC	
M2052	Operating mode: Electronic handwheel	NC	NC	
M2053	Operating mode: Positioning with manual input	NC	NC	
M2054	Operating mode: Program run/Single block	NC	NC	
M2055	Operating mode: Program run/Full sequence	NC	NC	
M2056	Operating mode: Test run	NC	NC	
M2057	Operating mode: Traversing the reference marks	NC	NC	
M2064	S Code 1st bit (lsb)			W258
M2065	S Code 2nd bit			
M2066	S Code 3rd bit			
M2067	S Code 4th bit			
M2068	S Code 5th bit			
M2069	S Code 6th bit			
M2070	S Code 7th bit			
M2071	S Code 8th bit (msb)			
M2072	M Code 1st bit (lsb)			W260
M2073	M Code 2nd bit			
M2074	M Code 3rd bit			
M2075	M Code 4th bit			
M2076	M Code 5th bit			
M2077	M Code 6th bit			
M2078	M Code 7th bit			
M2079	M Code 8th bit (msb)			
M2080	Minimum rpm from MP3020 1st bit (lsb)			W1008
M2081	Minimum rpm from MP3020 2nd bit			
M2082	Minimum rpm from MP3020 3rd bit			
M2083	Minimum rpm from MP3020 4th bit			
M2084	Minimum rpm from MP3020 5th bit			
M2085	Minimum rpm from MP3020 6th bit			
M2086	Minimum rpm from MP3020 7th bit			
M2087	Minimum rpm from MP3020 8th bit (msb)			
M2088	Increment from MP3020 1st bit (lsb)			—
M2089	Increment from MP3020 2nd bit			
M2090	Increment from MP3020 3rd bit			
M2091	Increment from MP3020 4th bit (msb)			

Marker	Function	Set	Reset	TNC 360/335
M2104	G Code S Analog 1st bit (lsb)			W256
M2105	G Code S Analog 2nd bit			
M2106	G Code S Analog 3rd bit (msb)			
M2112	T Number (P number) 1st decade (lsb)			W262 W264
M2113	T Number (P number) 1st decade			
M2114	T Number (P number) 1st decade			
M2115	T Number (P number) 1st decade (msb)			
M2116	T Number (P number) 2nd decade (lsb)			
M2117	T Number (P number) 2nd decade			
M2118	T Number (P number) 2nd decade			
M2119	T Number (P number) 2nd decade (msb)			
M2176	Code operating mode (lsb)			W272
M2177	Code operating mode			
M2178	Code operating mode			
M2179	Code operating mode (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 W986
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 4th axis	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 4th axis	NC	PLC	
M2527	Activate spindle orientation	PLC	NC	M2712
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	—
M2598	Transfer of the tool number	PLC	PLC	
M2599	Transfer of the tool number	PLC	PLC	
M2800 to M2807	Key code	PLC	PLC	W516
M2808	Strobe marker for key code	PLC	NC	M2813
M2809	Activate numerical value transfer from PLC to NC	PLC	PLC	M2713
M2810	Data format of a numerical value in M2560-M2576	PLC	PLC	—
M2811	Data format of a numerical value in M2560-M2576	PLC	PLC	—
M2812	Data format of a numerical value in M2560-M2576	PLC	PLC	—
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 pressed inhibited key	NC	NC	W274

Marker	Function	Set	Reset	TNC 360/335
M3200 to M3263	Values from MP4310.3 to MP4310.6	NC	NC	W 976 to W 968

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 360 both in memory organization and ergonomics.

5.3.1 PLC Macros

The following macro programs are available in the TNC 355 for controlling the toolchanger.

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 360/TNC 335. 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+O
=        M3043    ;T code == O
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   ;
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    ;
JPT     211
L       B255     ;load number of mag. pockets
B=      M3032    ;as actual value
LBL221
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 Non-implemented markers

Marker	Function
M2021	Nominal/actual speed difference over ranged
M2024	Touch probe ready
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
M2510	Deactivate spindle override
M2511	Deactivate feed rate override
M2526	Switch X/Y or Z to 4th axis
M2542	Complement for M2526
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
M2596	Select central tool file during active program run mode
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
M2667	No standstill monitoring axis 4
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 analog

5.3.3 PLC Cycle time

The contouring controls of the TNC 355, TNC 360 and TNC 335 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 360	24 ms
TNC 335	24 ms

Data Interface – Contents

1	Introduction	8-3
1.1	Principles of data transfer	8-4
1.1.1	Serial/parallel	8-4
1.1.2	Asynchronous data format	8-5
1.1.3	Checking data	8-7
1.1.4	Data transfer rate	8-8
1.2	Handshaking	8-9
1.2.1	Hardware handshaking	8-9
1.2.2	Software handshaking	8-9
2	TNC data interfaces	8-10
2.1	General	8-10
2.2	RS-232-C/V.24 interface	8-10
2.2.1	Hardware	8-10
2.2.2	Signal levels	8-11
2.2.3	Signal designation	8-11
2.2.4	Pin layouts	8-13
2.3	Data interface functions	8-14
2.3.1	Saving/reading files	8-14
2.3.2	Output to external devices	8-15
2.3.3	Reading in and simultaneously executing programs (DNC operation)	8-15
2.3.4	Communication between TNCs	8-15
2.4	Data transmission protocols	8-15
2.4.1	Standard transmission protocol	8-16
2.4.2	Data transfer with Block Check Character (BCC)	8-17
2.5	Configuration of the interface	8-22
2.5.1	Selection of the interface	8-22
2.5.2	Freely configurable interfaces	8-22
2.6	External programming	8-28
2.7	Interfacing with other equipment	8-28
3	Standard Data Transmission Protocol	8-29
3.1	General information	8-29
3.1.1	Calling the program directory	8-29
3.1.2	Outputting a selected program	8-31
3.1.3	Outputting all programs	8-31
3.1.4	Reading in selected program	8-31
3.1.5	Reading in all programs	8-33
3.1.6	Reading in an offered program	8-34

4	Data transfer with BCC	8-35
4.1	General	8-35
4.1.1	Calling a program directory	8-36
4.1.2	Outputting a selected program	8-37
4.1.3	Outputting all programs	8-37
4.1.4	Reading in a selected program	8-38
4.1.5	Reading in all programs	8-39
4.1.6	Reading in an offered program	8-40
5	Data transfer by PLC	8-42
5.1	Configuration of PLC data interface	8-42
6	Error messages	8-43
6.1	TNC error messages	8-43
6.2	HEIDENHAIN peripherals' error codes	8-44
6.3	Data transmission software error messages	8-45

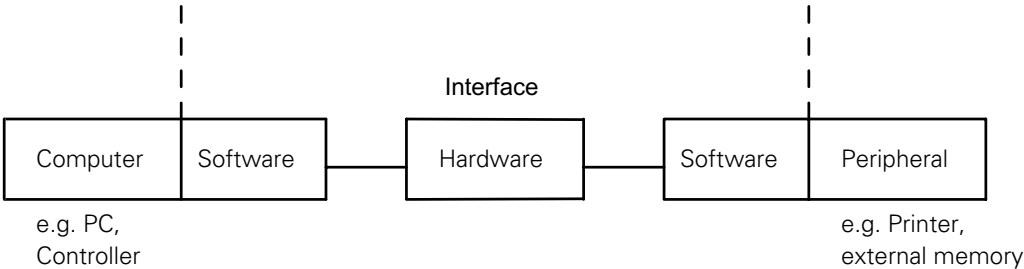
1 Introduction

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 used in conjunction with the Central Processing Unit (CPU).

A data interface makes it possible for the CPU and the peripherals to communicate.

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 and is associated with both the operating software 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 interface, which is 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.

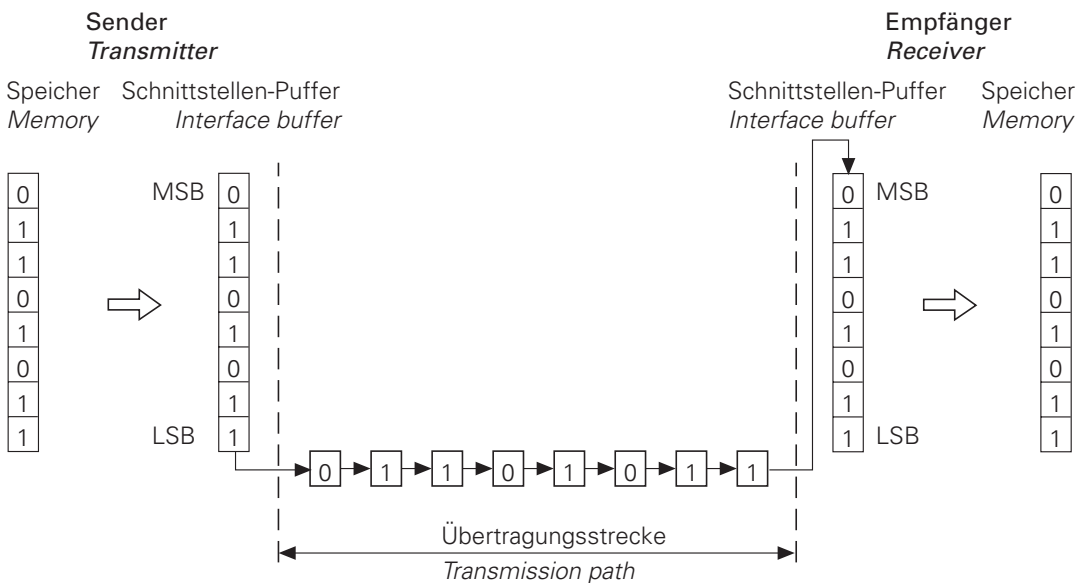
1.1.1 Serial/parallel

Data can be transmitted in either serial or parallel format.

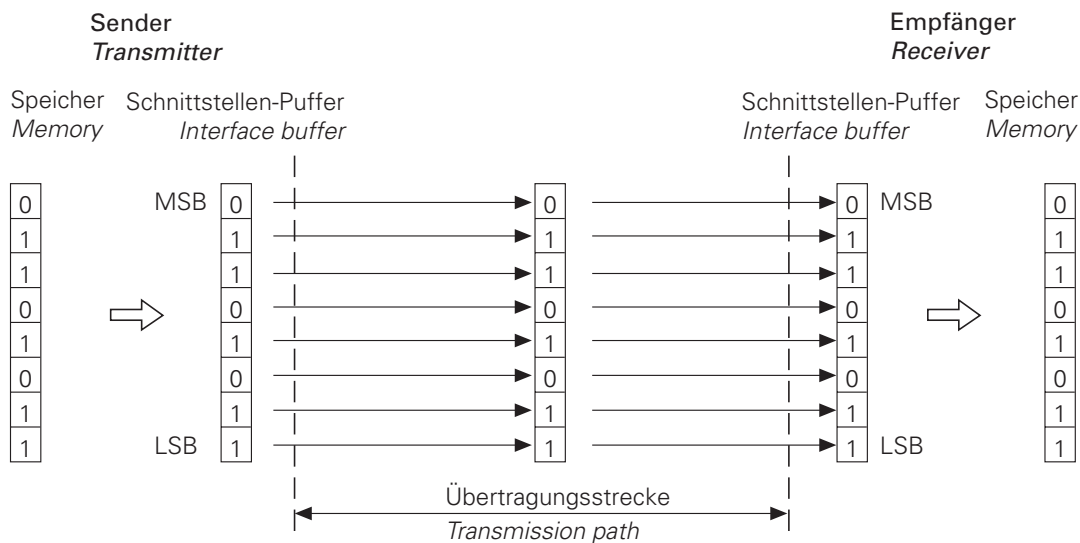
Basically, data is coded in the computer system, e.g. as bytes (8 bits) and supplied to the interface in parallel.

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 meters.



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:

$$\begin{array}{ccccccc}
 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
 \text{MSB} & & & & & & \text{LSB}
 \end{array}
 = 10 \text{ dec} = 0A \text{ hex}$$

The letter 'z' is represented by the following combination of bits:

$$\begin{array}{ccccccc} 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ \text{MSB} & & & & & & \text{LSB} \end{array} = 122 \text{ dec} = 7A \text{ hex}$$

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.

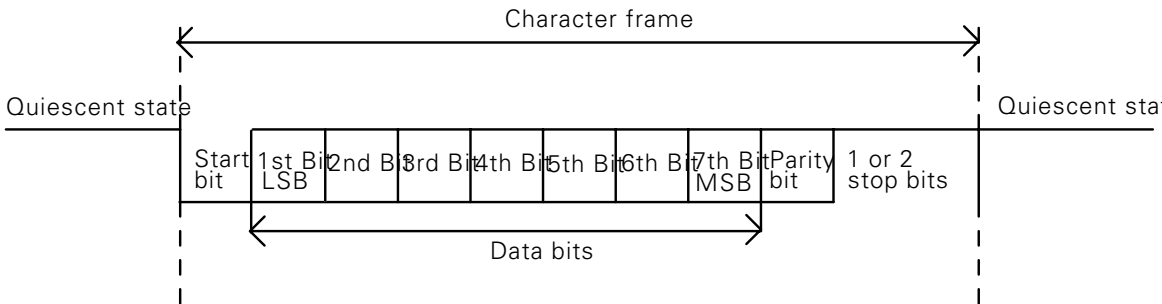
A quiescent state exists before switch-on and is reverted to after each transmission. Before a data bit can be transmitted this has to be communicated to the receiver. Otherwise, if the first bit of the data word has the same value as the quiescent state, the receiver will not notice any difference from 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 Section 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 recognize the transmitter again before the start of the next character. Synchronization is repeated before each character.

The synchronization is repeated before every word and is valid for one 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.

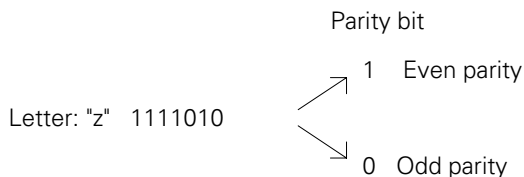
– Even parity

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:



1.1.4 Data transfer rate

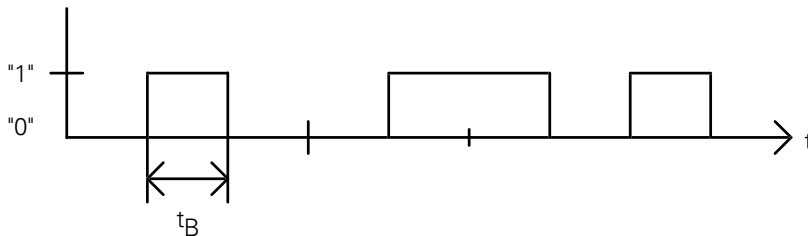
The data transfer rate of an interface is given in 'baud' and indicates the number of bits of data transmitted in one second.

$$1 \text{ baud} = \left[1 \frac{\text{Bit}}{\text{s}} \right]$$

Common baud rates are:

110, 150, 300, 600, 1200, 2400, 4800, 9600, 19 200 and 38 400 baud.

The time taken to transmit one bit (t_B) can be calculated from the baud rate:



$$t_B = \frac{1}{\text{Baud rate} \left[\frac{\text{bits}}{\text{s}} \right]}$$

For example, a baud rate of 19,200 baud will have a bit duration of $t_B = 52.083 \mu\text{s}$.

The number of characters transmitted per second can be calculated from the baud rate and the transmission format:

$$\text{Characters per second} = \frac{\text{Baud rate} \left[\frac{\text{bits}}{\text{s}} \right]}{\text{Number of bits per character}}$$

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 signaled 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

The TNC 360 has a data interface, the RS-232-C/V.24. The HEIDENHAIN FE 401 floppy disk unit, ME 101 magnetic tape unit and external devices with appropriate data interfaces (computers, printers, readers, punches) can be connected via the RS-232-C/V.24 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 this protocol.

2.2 RS-232-C/V.24 interface

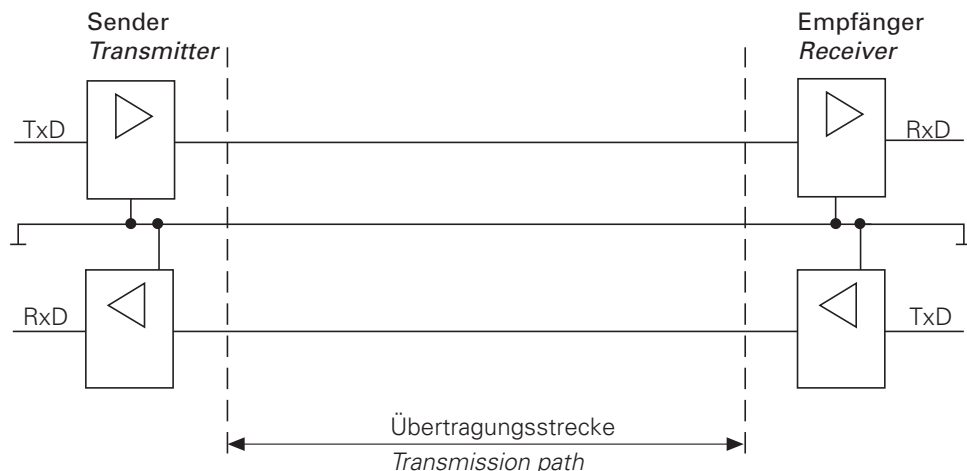
RS-232-C is the designation of a serial interface based on the EIA standard of the same name and is usable for transmission rates up to 19 200 bits/sec. Data transfer is executed asynchronously, with a start bit before each character and one or two stop bits after each character. The interface is designed for transmission distances of up to 20 meters.

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.

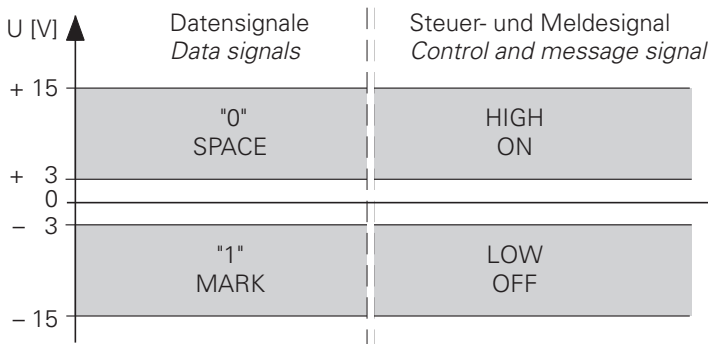
Data lines:

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 designation

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):

Hardware handshake:
Output of the receiving unit

CTS (Clear to Send):

Hardware handshake:
Output of the transmitting unit

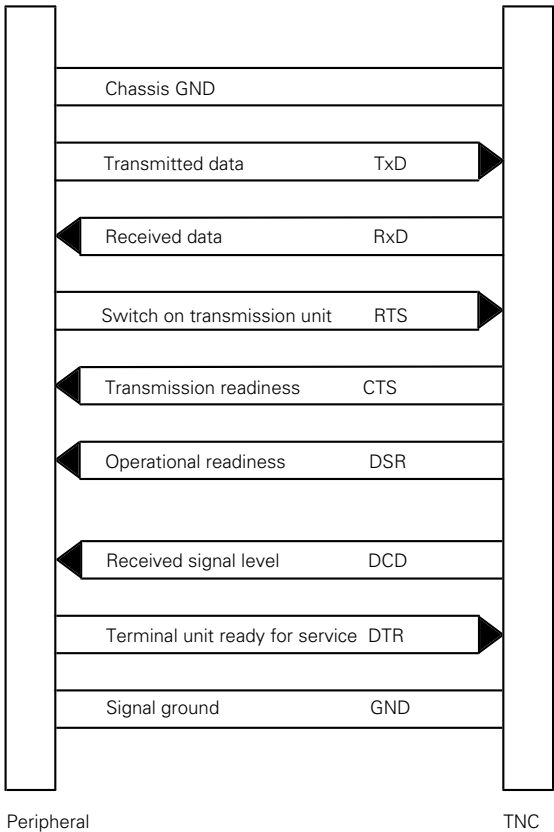
Earth conductor (cables for power supply):

Chassis GND:

Casing connection

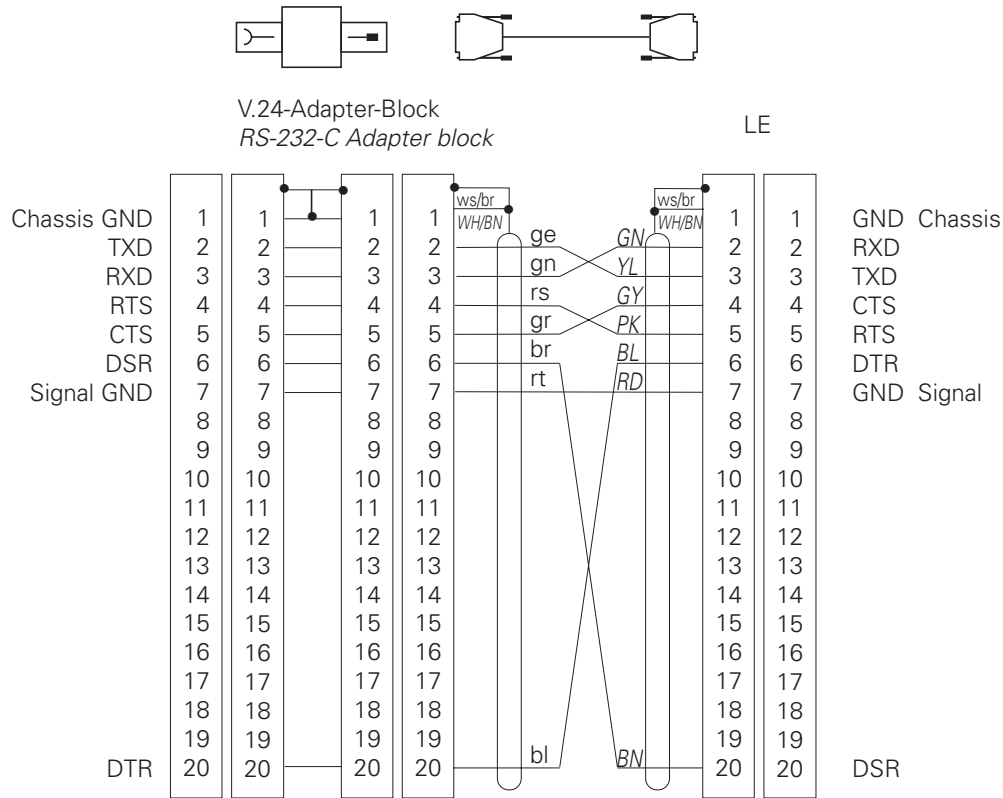
Signal GND:

0-Volt lines for all signals



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 Chapter "Mounting and electrical installation").



Note: The LE and adapter block are connected with crossed lines.

A 9-pin plug on a PC should have the following pin layout:

Pin	Allocation
1	Do not use
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	Do not use

2.3 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.

2.3.1 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	Identification
HEIDENHAIN-dialog NC program	H
ISO NC program	D
Machine parameters	M
Correction table	S
PLC program	P

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 ISO program is stored, it is given the file extension *.DNC.

2.3.2 Output to external devices

Any external device, e.g. computers, printers, readers and punches, can be addressed via the interface. For this purpose, the TNC has a freely configurable interface mode (EXT) which, within certain limits, permits 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.

2.3.3 Reading in and simultaneously executing programs (DNC operation)

In PROGRAM RUN mode, machine programs can be transmitted with "Block transfer" from an external memory via the serial data interface 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.

If the program blocks are read into the TNC faster than they can be executed, the entire NC memory is initially fully loaded. If it is full, transfer is suspended until a storage space of one NC block is vacated by execution. Only then is transfer resumed, until once again the memory is filled.

For DNC operation the data transfer protocol must be set with Block Check Character BCC (data checking).

2.3.4 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 the RS-232-C/V.24 interface.

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 (ME mode) must be set at both control units and transfer started. It must be ensured that the receiving 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 by PLC").

2.4 Data transmission protocols

The TNC enables data and files to be transferred using two different protocols (which can be selected via the interface setup or the machine parameters).

These transmission protocols can be selected in three different operating modes, as follows:

- ME Standard transmission protocol to match HEIDENHAIN magnetic tape unit (7 data bits, 1 start bit, 1 stop bit)
Data transmission rate: 2400 baud
- FE Transmission with Block Check Character and with fixed control characters (7 data bits, 1 start bit, 1 stop bit)
Freely configurable baud rate
- EXT Freely configurable operating modes: data format, transmission protocol and control characters can be freely set via machine parameters. Freely configurable baud rate.

The following applies to data transmission protocols:

- If a file which is read in is already stored in the TNC, it displays the message

"ERASE=ENT / OVERREAD=NOENT".

In this case the TNC aborts transmission with the appropriate handshake and does not continue transfer until after acknowledgment.

In the event of an attempt to erase write-protected files, the error message "PROTECTED PGM" is displayed and the dialog "OVERREAD=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 MP5010 in operating modes EXT).
- If a transmission is terminated with the END-key, the error message "PROGRAM INCOMPLETE" is issued.

2.4.1 Standard transmission protocol

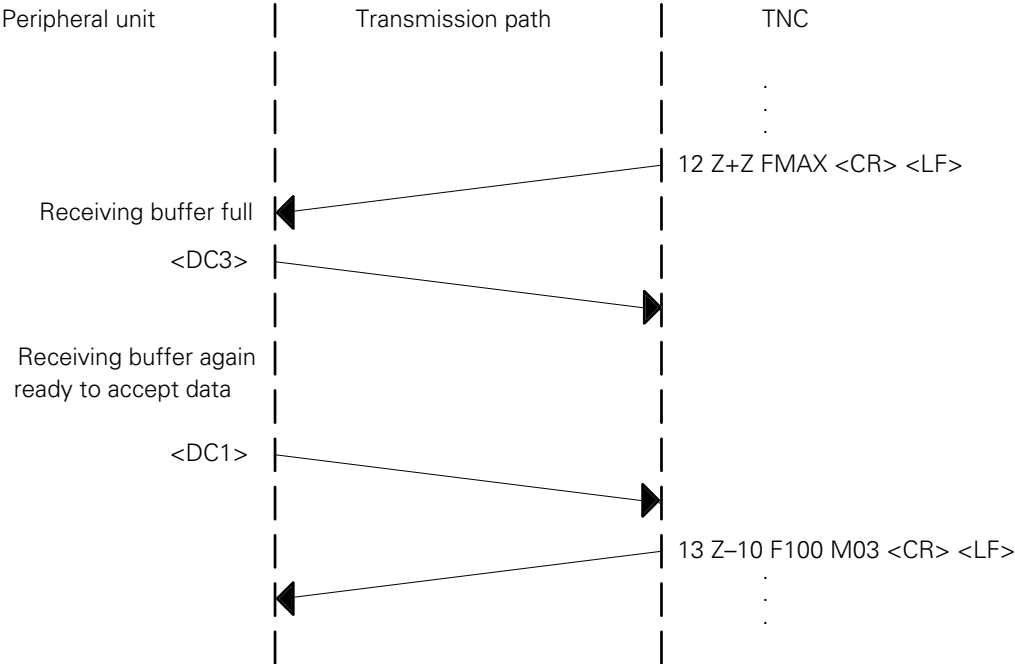
With this protocol, first of all the TNC transmits the character <NUL> exactly 50 times. This is followed by the individual program blocks, which each end with the characters <CR> and <LF>. These blocks are transmitted in order, but they are not error checked. If the receiver's data buffer is full, the receiver has two alternatives for stopping and recommencing transmission:

- Software handshake - Stopping transfer by sending character <DC3> (XOFF), continuing transfer by sending character <DC1> (XON).
- Hardware handshake - By putting appropriate voltage level on the RTS and CTS control and signal lines of the RS-232-C/V.24 interface.

Example: protocol for dialog program:

<NUL><NUL><NUL><NUL><NUL><NUL><NUL>.. 50 times	
0 BEGIN PGM 1 MM<CR><LF>	1st program block
1 TOOL DEF 1 L+0 R+3<CR><LF>	2nd program block
.	
.	
.	
26 END PGM 1 MM <CR><LF>	Program end
<ETX><EOT>	Close data transfer menu

Example of software handshake:



Hardware handshake (see Section "Freely configurable interfaces").

2.4.2 Data transfer with Block Check Character (BCC)

This protocol, specific to HEIDENHAIN, works with different control characters and with additional data checking when transmitting.



In the freely configurable operating modes (EXT), the following control characters - <SOH>, <ETB>, <STX>, <ACK>, <NAK>, <ETX>, <EOT> - can be defined as arbitrary ASCII characters (see Section "Freely configurable interfaces").

When a file is transferred, the first block, called the "header", is sent. It consists of the following characters:

<SOH>"H" "Name" "M" <ETB>BCC<DC1>

<SOH> (Start of Header): This character indicates the start of the header.

The header contains 'H' - the code for the program (see Section "Saving/reading files"), 'Name' - the program name and 'M' - the transmission mode(E=input/A=output).

This header ends with character <ETB>, ending a data transfer block.

The subsequent BCC (Block Check Character) provides additional confidence.

In addition to the parity check of the individual characters (see Section "Checking data"), a parity check is carried out on a complete transmitted block. The BCC always supplements the individual bits of the transmitted characters of a data transfer block to give even parity.

Example for formation of BCC:

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, the HEIDENHAIN dialog (code 'H') has been used to write program '15' which is read in via the data interface ('E'). A parity bit is also formed for the BCC (with even parity, the parity bit of the BCC in this example is given a value '1').

The character <DC1> is sent after the BCC. This character (XON) is needed for a few devices, explicitly to demand a reply from them in order to start transfer again.

At the end of each block, a check is conducted to see whether the block has been correctly transferred. To do this, the receiver forms a BCC from the received block and compares it with the received BCC. If the received and calculated BCCs are identical, the receiver sends character <ACK> (= positive ACKnowledgment), i.e. the data block has been transferred without error.

If the two BCCs are not identical, the receiver sends character <NAK> (= Not AcKnowledgeed), i.e. the data block has been incorrectly transferred and the same block must be sent again. This process is repeated up to three times. An error message ("TRANSFERRED DATA INCORRECT N") is displayed, and transfer is aborted.

If, however, this header is acknowledged with <ACK>, the first data block can be transmitted: <STX>0 BEGIN PGM 1 MM <ETB> BCC <DC1>.

The start of a data block is always indicated by control character <STX>. The other control characters in this block are identical to the control characters of the header.

If the block is acknowledged with <ACK>, the next program block is sent. In the event of a <NAK>, the same block must be repeated, and so on...

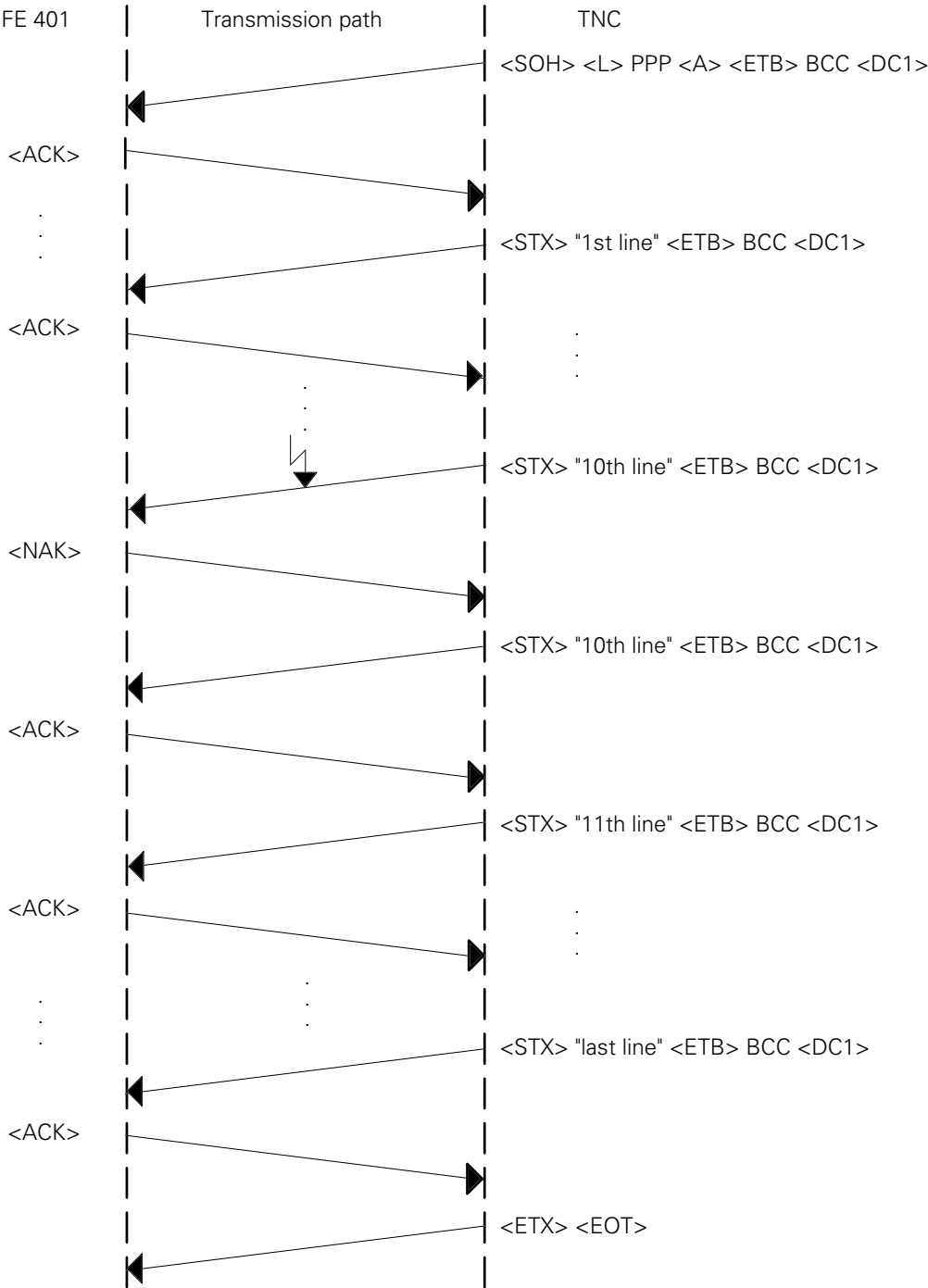
If the last program block has been sent successfully (acknowledged with <ACK>), transmission ends with characters <ETX> (End of Text) and <EOT> (End of Transmission).

Table of control characters:

Character	Meaning	Description
SOH	Start of Header	SOH indicates the start of the transfer of the data header. The header is a sequence of characters which contains the program number and information concerning the type of program and transmission mode.
STX	Start of Text	STX indicates the start of the program block.
ETB	End of Text Block	ETB ends a data transfer block. The character following ETB is used for data checking (BCC).
DC1	Start data transfer (XON)	DC1 starts data transfer after a stop.
DC3	Stop data transfer (XOFF)	DC3 stops data transfer.
ETX	End of Text	ETX is sent at the end of the program.
EOT	End of Transmission	EOT ends data transmission and produces the quiescent state. This character is sent by the TNC at the end of program input, and as an error to external devices
ACK	ACKnowledged	ACK is sent by receiver when a data block has transferred without error.
NAK	Not AcKnowledgeed	NAK is sent by the receiver when a data block has transferred incorrectly. The transmitter must transmit the data block again.

Example:

A pallet file with the name "8455" is to be sent to a peripheral unit (e.g. FE 401).



The software handshake is very easy to achieve when transmitting with a BCC. The receiver sends neither a positive (<ACK>) nor a negative acknowledgment (<NAK>), and the transmitter waits until it receives one of these characters. When the buffer in the receiver is again capable of accepting data, it again sends an <ACK> and the transmitter continues its data transmission.

It is also possible, however, to conduct the software handshake with control characters <DC1> and <DC3>. If hardware handshake (EXT) has been selected, it is identical to standard data transfer and to data transfer with a BCC (see Section "Freely configurable interfaces").

2.5 Configuration of the interface

2.5.1 Selection of the interface

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 EXT, there are two fixed modes (ME and FE). The FE mode must be set if the HEIDENHAIN floppy disk unit or an external computer using the TNC.EXE transmission software is connected. In this mode, the transmission protocol with Block Check Character is rigidly defined.

The ME operating mode matches the HEIDENHAIN ME 101/ME 102 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 operating mode EXT is 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

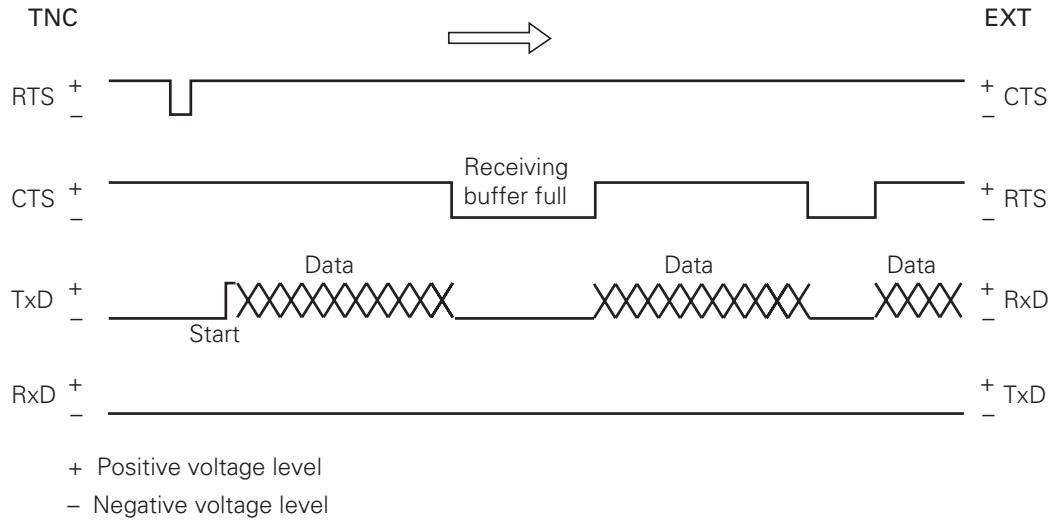
If calculation of the BCC produces a number less than \$20 (i.e. a control character) then a "Space" character (\$20) is sent in addition immediately before <ETB>. The BCC will consequently always be greater than \$20 and cannot therefore be recognized as a control character.

Hardware handshaking

Bit 2 can be set to determine whether the TNC stops transfer from an external device by using RTS.

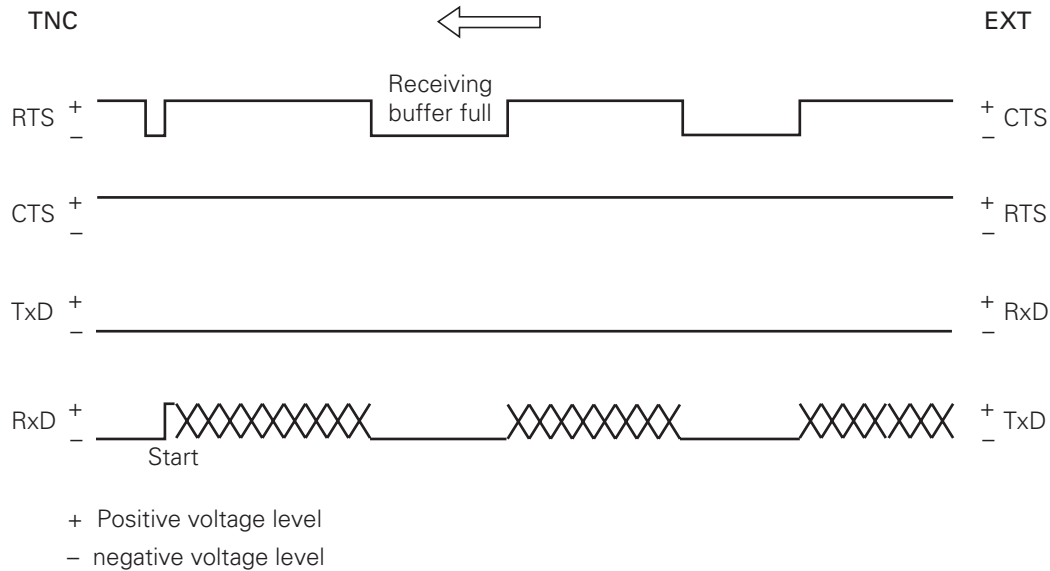
– **Data output TNC -> EXT**

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.



– **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.



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).

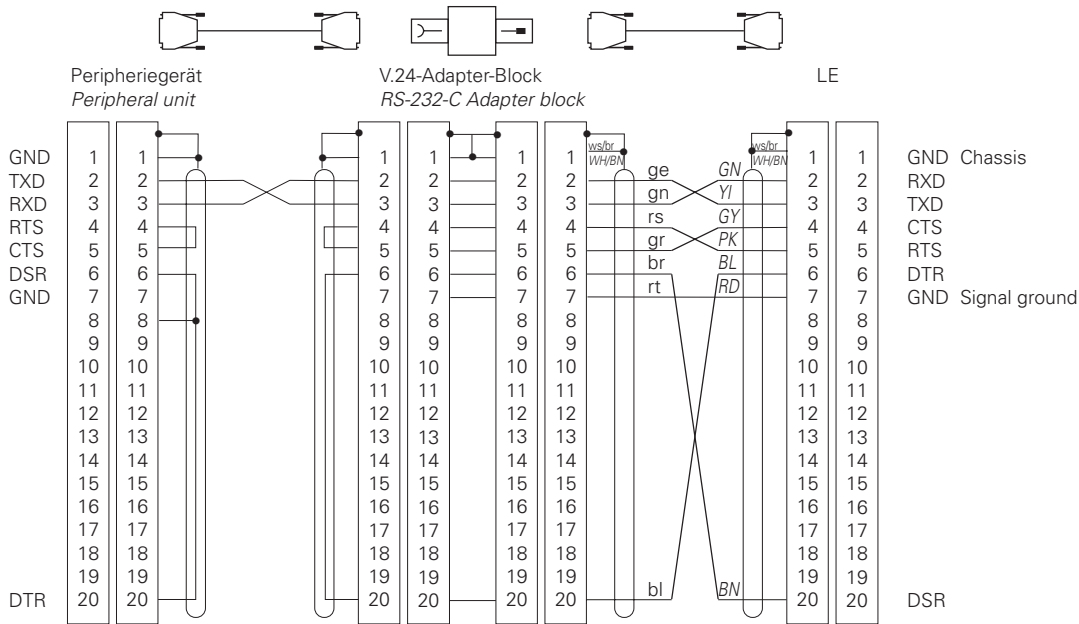
DTR: Polled by peripheral; it is logic "1" if TNC is ready for service.
DSR: Polled by TNC.
LOW level => ext. data input/output not ready.
HIGH level => ext. data input/output ready.

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.

If the TNC is transmitting data, it reacts both to hardware and software handshakes, regardless of the setting in the MP5020.

If the TNC is receiving data, and no transmission stop is set in the MP5020, 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 Section "Checking data").

Stop bits

Bits 6 and 7 determine the number of stop bits sent at the end of a character.

MP 5020 Operating mode EXT: Interface configuration
Input range: 0 to 255

Bit 0	7 or 8 data bits		
	+0 = 7 data bits		
	+1 = 8 data bits		
Bit 1	Block check character		
	+0 = any BCC		
	+2 = BCC not control character		
Bit 2	Transmission stop by RTS		
	+0 = not active		
	+4 = active		
Bit 3	Transmission stop by DC3		
	+0 = not active		
	+8 = active		
Bit 4	Character parity		
	+0 = even		
	+16 = uneven		
Bit 5	Character parity		
	+0 = not desired		
	+ 32 = desired		
Bit 6/7	Stop bits	Bit 6	Bit 7
	+0 = 1 1/2 stop bits	0	0
	+64 = 2 stop bits	1	0
	+128 = 1 stop bit	0	1
	+192 = 1 stop bit	1	1

For operating modes EXT, MP5030 defines the transmission protocol.

MP5030.0 Operating mode EXT: Data transfer protocol
Input: 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 zero, the standard settings given in brackets are active (as for FE operation).

Two control characters are encoded in one machine parameter.

The input is as follows:

Input value Decimal code (1st control character) + 256 * Decimal code (2nd control character)

MP5010 Control characters for data transmission protocol

MP5010.0 1st control character for End of Text (ETX)
 2nd control character for Start of Text (STX)
 Input range: 0 to 32 382

MP5010.3 1st control character for End of Transmission Block (ETB)
 2nd control character for Start of Heading (SOH)
 Input range: 0 to 32 382

MP5010.4 1st control character for "Acknowledgment of reception" (ACK)
 2nd control character for Start of Heading (SOH)
 Input range: 0 to 32 382

MP5010.5 Control character for "End of Transmission" (EOT)
 Input range: 0 to 32 382

Example: The following ASCII characters are to be transmitted for start and end of text:
 End of text: ? = 63 dec
 Start of text: ! = 33 dec

Input:
 MP5010.0 = 63 + 256 * 33 = 8511

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.

For the file type, when transferring with Block Check Characters (see Section 2.3.1 "Saving/reading files"), an ASCII character must be entered for the file which is being output or input. With an input value of zero, the TNC automatically enters the correct type of file in the file header.

The ASCII characters for input and output identification can also be freely defined.

MP5010.1 1st ASCII character for data file during data input
 2nd ASCII character for input identification (E)
 Input range: 0 to 32 382

MP5010.2 1st ASCII character for file type during data output
 2nd ASCII character for output identification (A)
 Input range: 0 to 32 382

Example: If an ISO program should be output in the EXT mode, then:
 MP5010.2 = 68
 (or MP5010.2 = 0)

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.
- Blank characters between the individual words can be omitted in NC programs (HEIDENHAIN dialog).
- 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 are stored only in PLC programs
- 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 EXT. For this purpose, machine parameters 5020 to 5030 permit relatively free adjustment of the data format, the data transmission protocol and the control characters.

Consider the example of interfacing EXT 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)
- 9600 baud

The following settings (EXT) are made at the TNC:

MP5020 = 169	8 data bits (+1)
	Any BCC character (+0)
	Transfer stop by RTS not active (+0)
	Transfer stop by DC3 active (+8)
	Character parity even (+0)
	Character parity wanted (+32)
	1 stop bit (+128)
MP5030 = 0	Standard data transfer

In the TNC, the EXT operating mode must still be assigned to the RS-232 interface and the baud rate set to 9600 (see the TNC 360 User's Manual).

3 Standard Data Transmission Protocol

3.1 General information

This protocol is set as standard in operating mode ME and can also be optionally selected for operating modes EXT 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 EXT, 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 EXT 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 signaled to the TNC, the following sequence of instructions must be sent:

<ESC> <1> 'ERROR NUMBER'

Listed below are the transmission protocols for the various alternatives for data output and input.

EXT mode is set:

- Control character for "End of Text" - <ETX>
- Control character for "End of Transmission" - <EOT>
- Software handshake

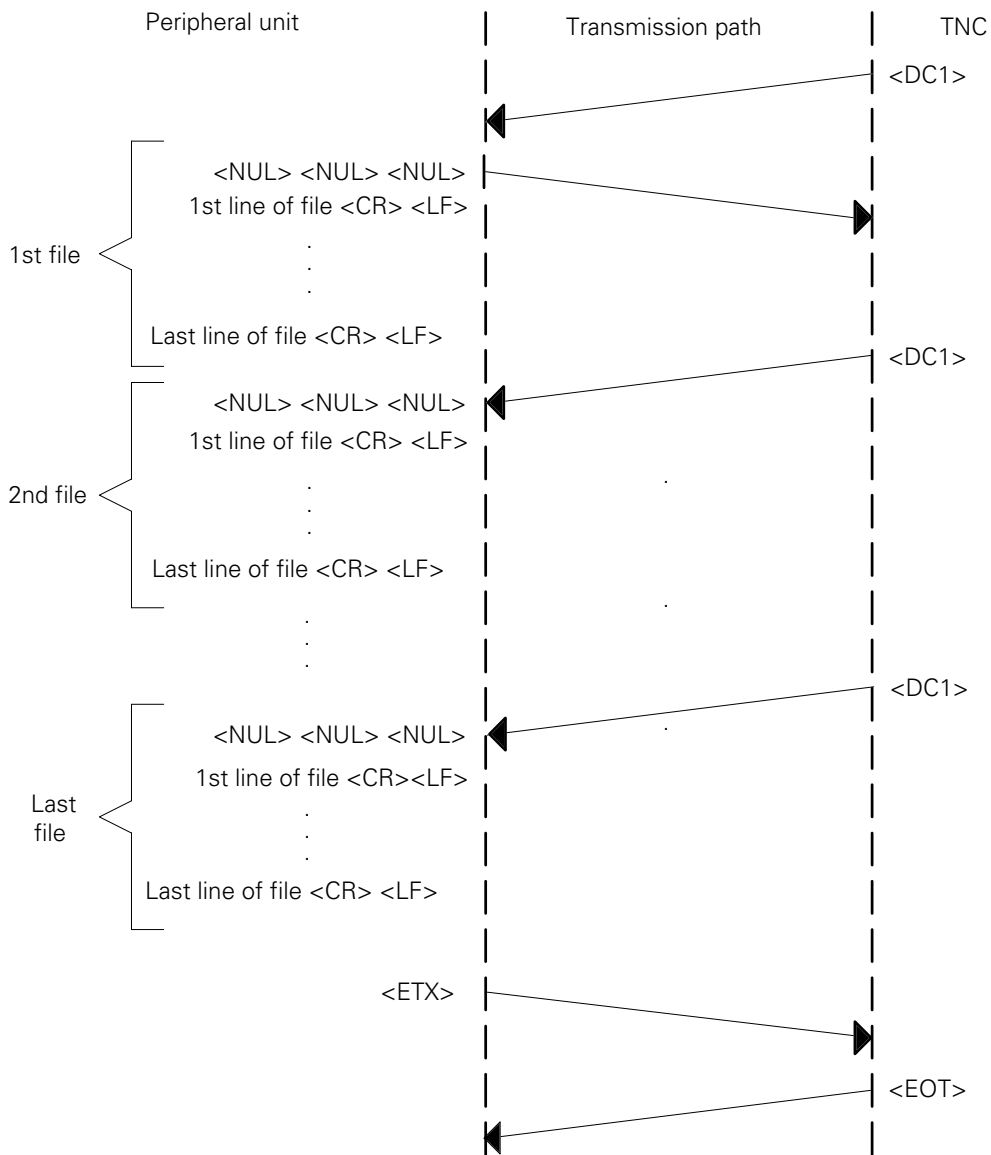
3.1.1 Calling the program directory

Using the menu item "Program 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>.

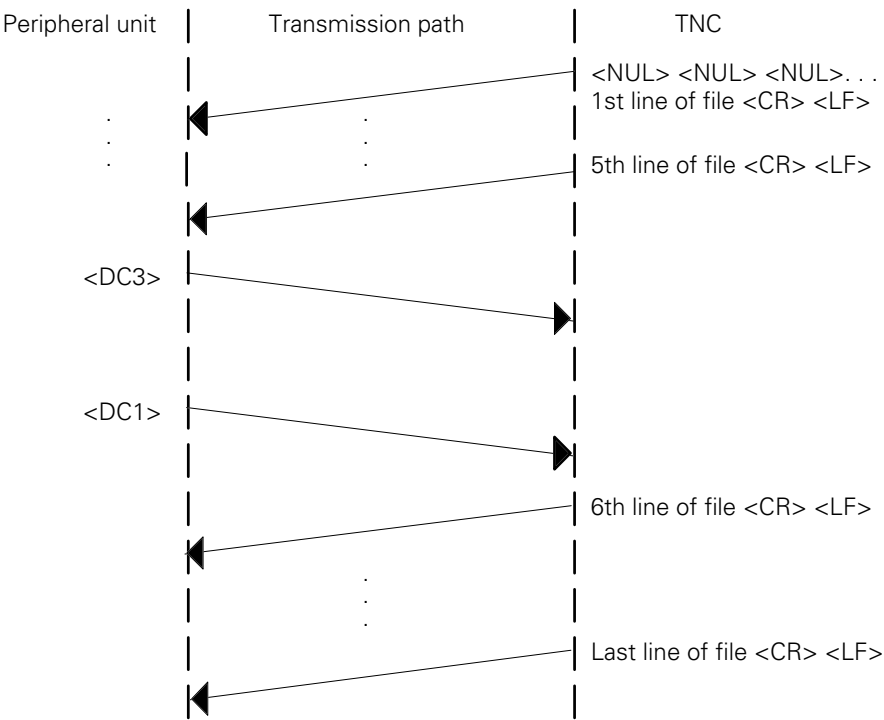
If the request is immediately interrupted with the END key, the TNC sends characters <ETX><EOT> and no directory is read in.

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.



3.1.2 Outputting a selected program

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>.



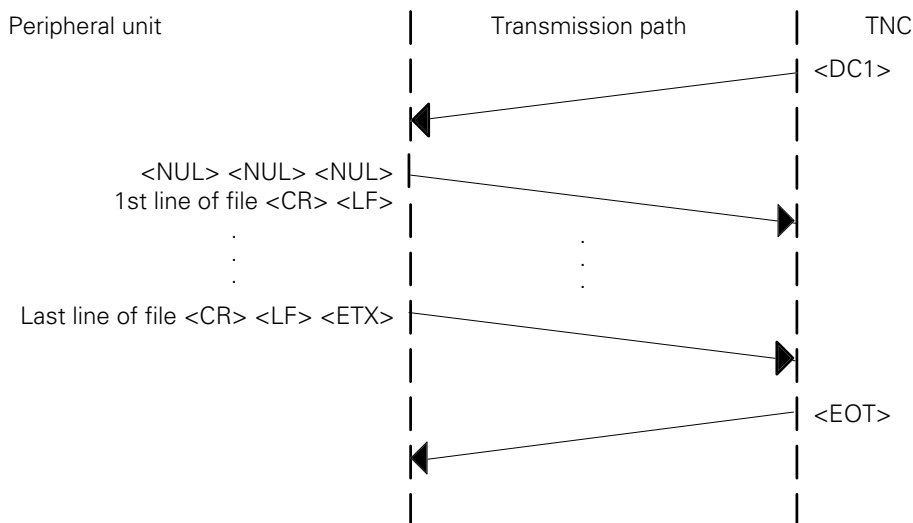
3.1.3 Outputting all programs

The procedure is similar to the protocol described in Section 3.1.2 "Outputting a selected program". The TNC arranges all the programs in order and transmits them. No control character is sent between the individual files.

3.1.4 Reading in selected program

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.

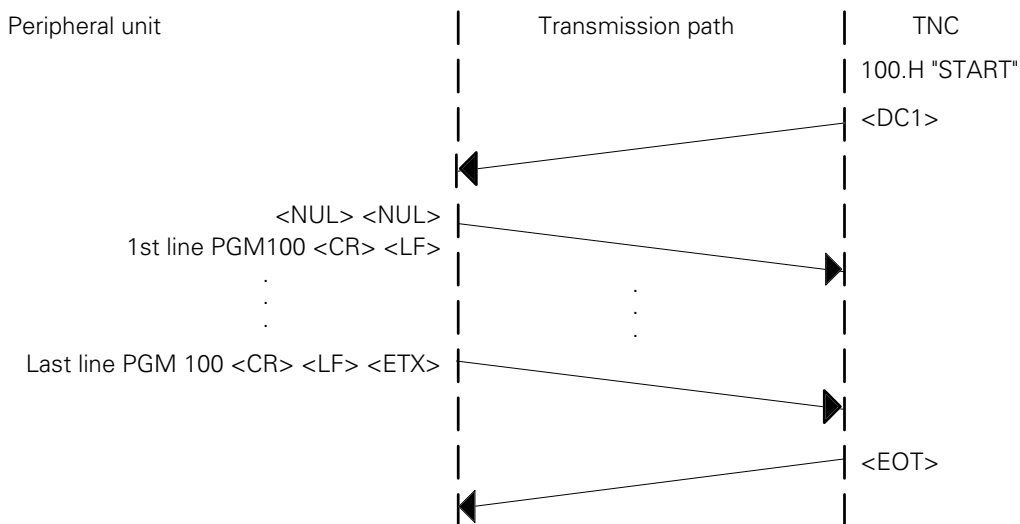
When the entire file has been transferred, the TNC sends character <EOT>.



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.

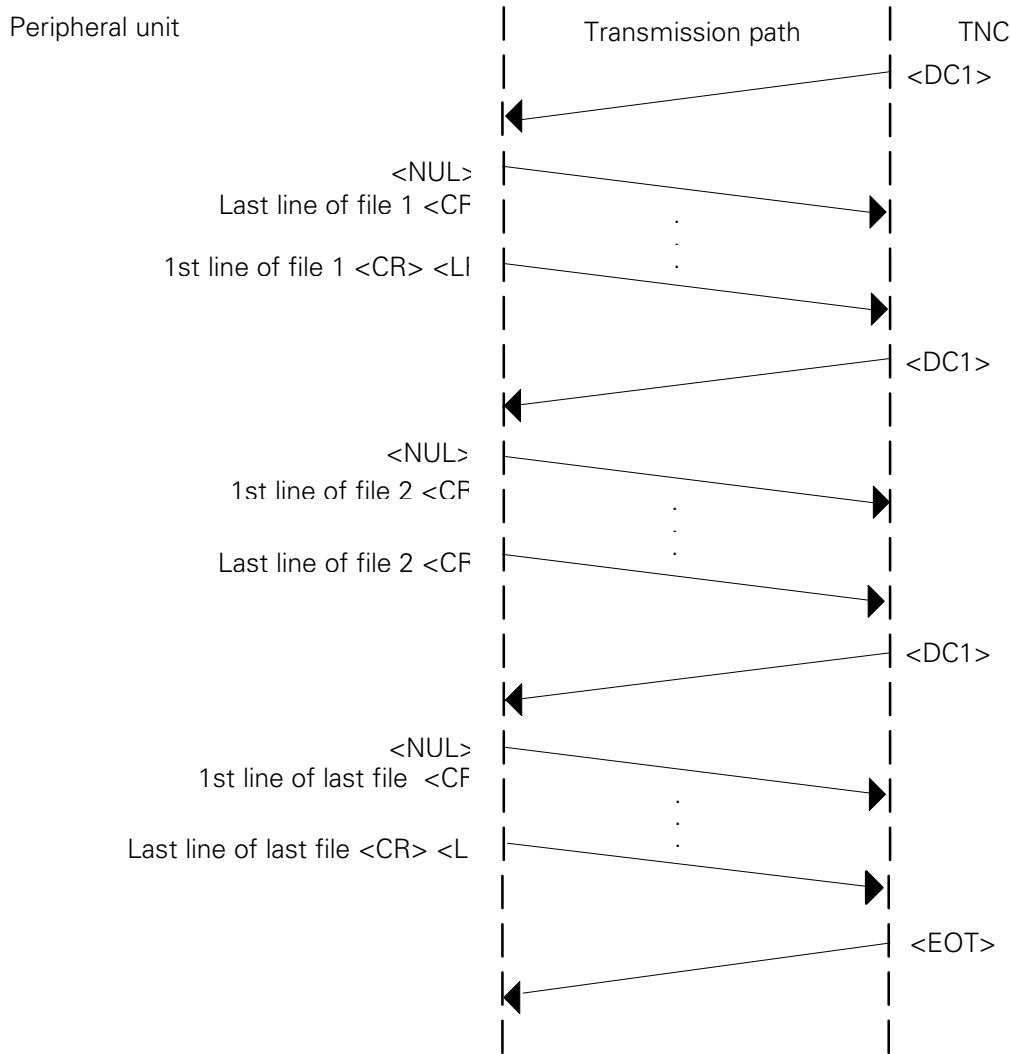
Example: Reading in program 100.H.



If, in this case, the last PGM-block ends without the <ETX> character, transfer is terminated without an error message but the data is not stored. This means that a program on the data medium with incorrect program name will be overlooked.

3.1.5 Reading in all programs

If both the peripheral unit and the TNC have been started, the following protocol is followed:

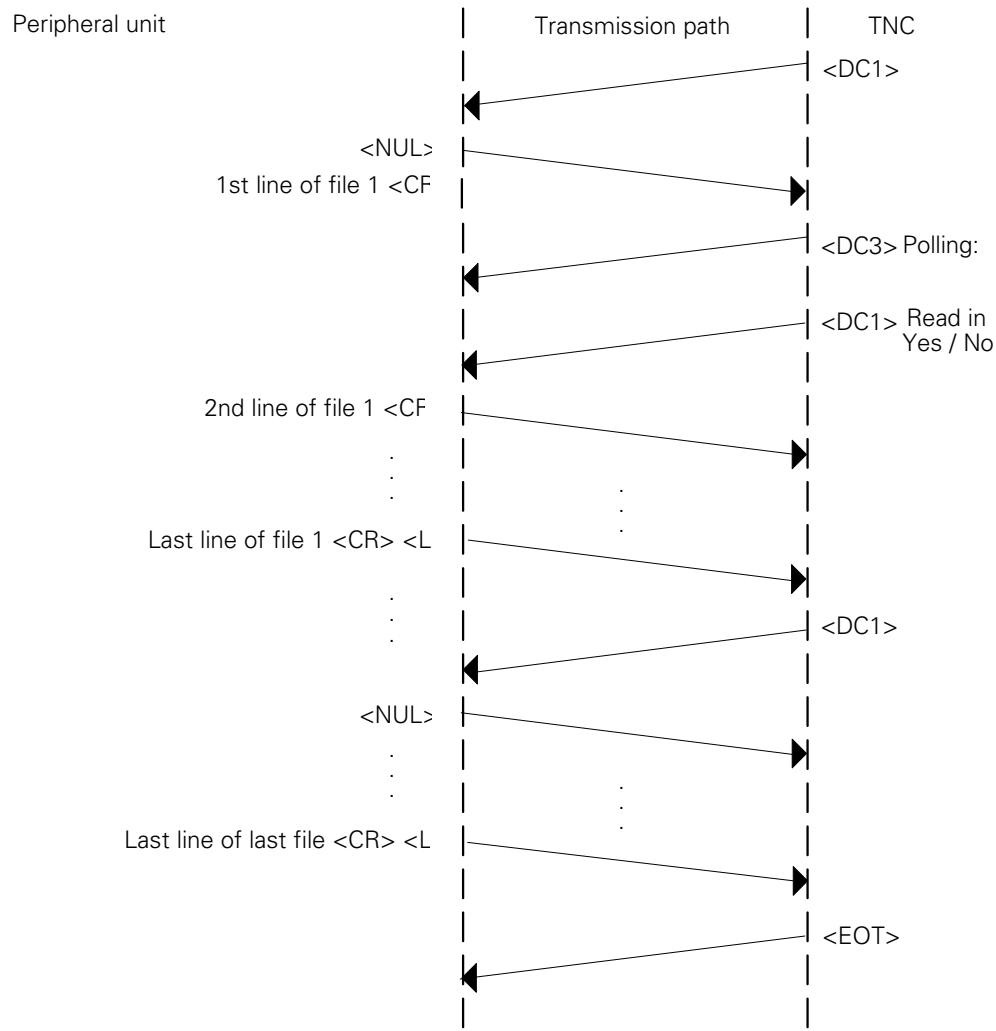


If several programs are gathered together in a file which ends with **<ETX>** then these programs are read in without being requested by **<DC1>**.

Data read-in does not stop until a program has ended with **<ETX>**.

3.1.6 Reading in an offered program

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 acknowledgment 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.



4 Data transfer with BCC

4.1 General

HEIDENHAIN TNCs, as well as allowing data transfer with the standard data transmission protocol, also allow data transfer with a Block Check Character (BCC).

This protocol is set for the following modes:

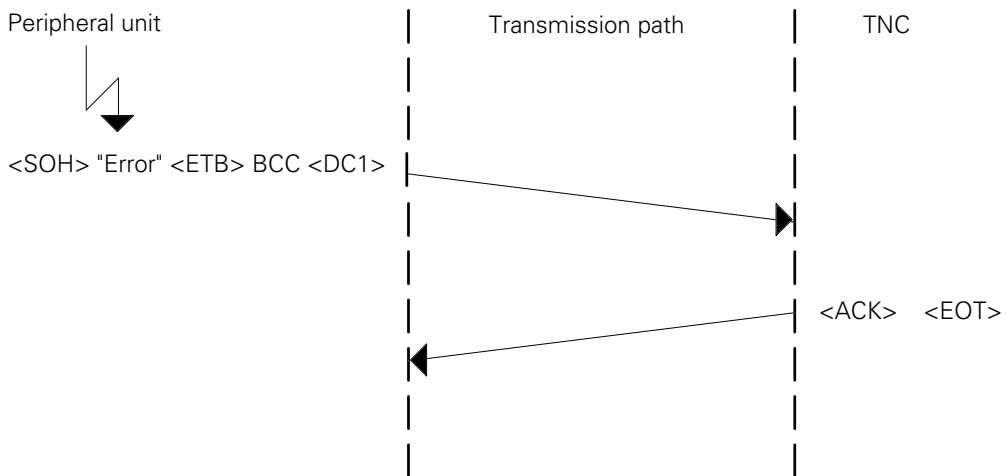
- FE mode
- EXT mode (choice)

The data transmission protocols are identical in these modes.

In the following, the transmission protocols are listed for the various file input and output possibilities. FE mode is set.

If an error occurs at a peripheral device, the following block must be sent to the TNC.

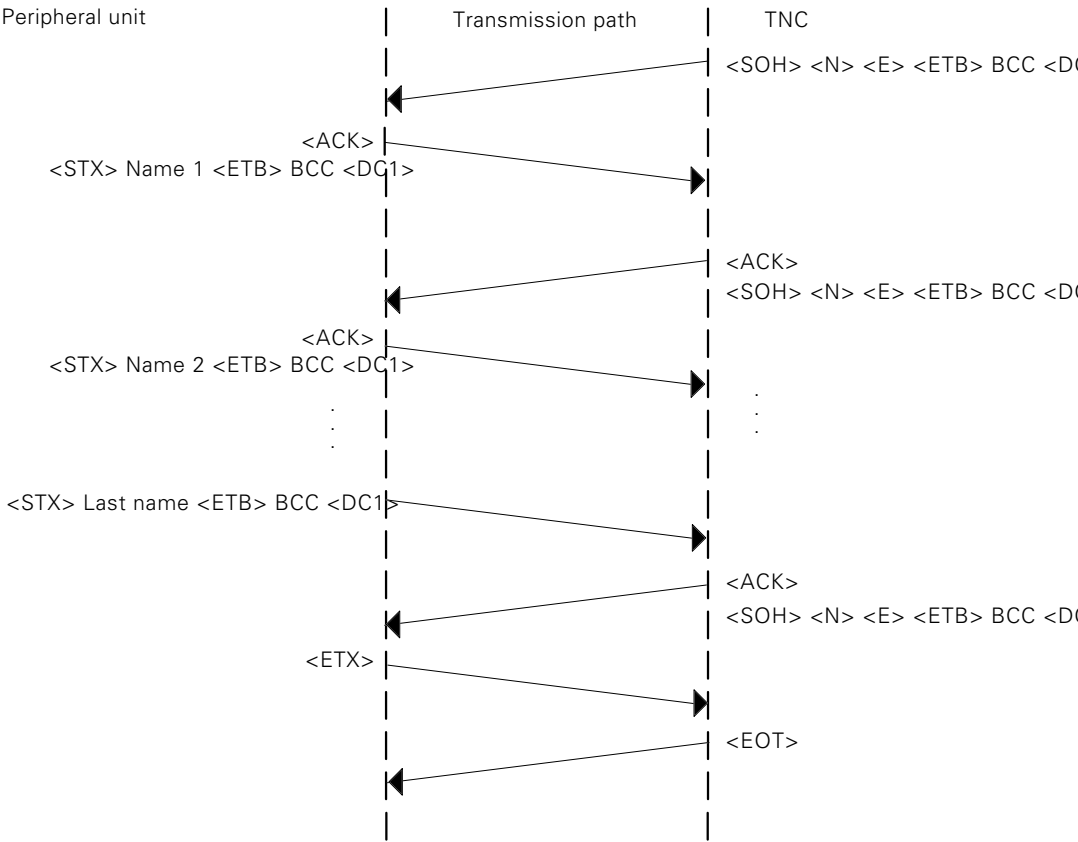
<SOH>"Error text"<ETB>BCC



The received error message is displayed in the TNC, but can be acknowledged and erased with the CE key.

4.1.1 Calling a program directory

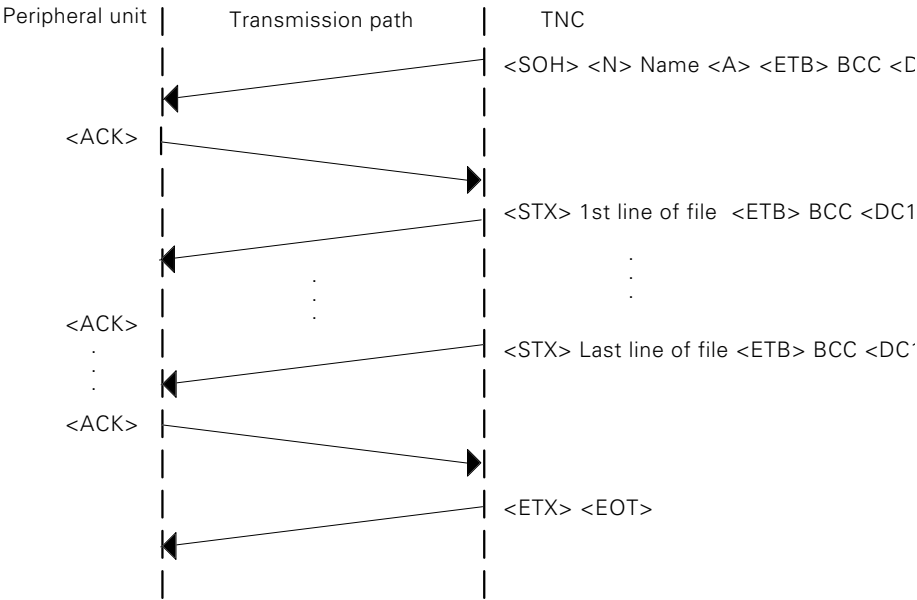
If the "Program Directory" menu item is selected, the TNC outputs a header without a program name. The peripheral then starts to output the files to the TNC, but, after the first file block has been sent, the TNC immediately requests the next file.



The program names read in are displayed in the TNC.

4.1.2 Outputting a selected program

The following protocol is followed:



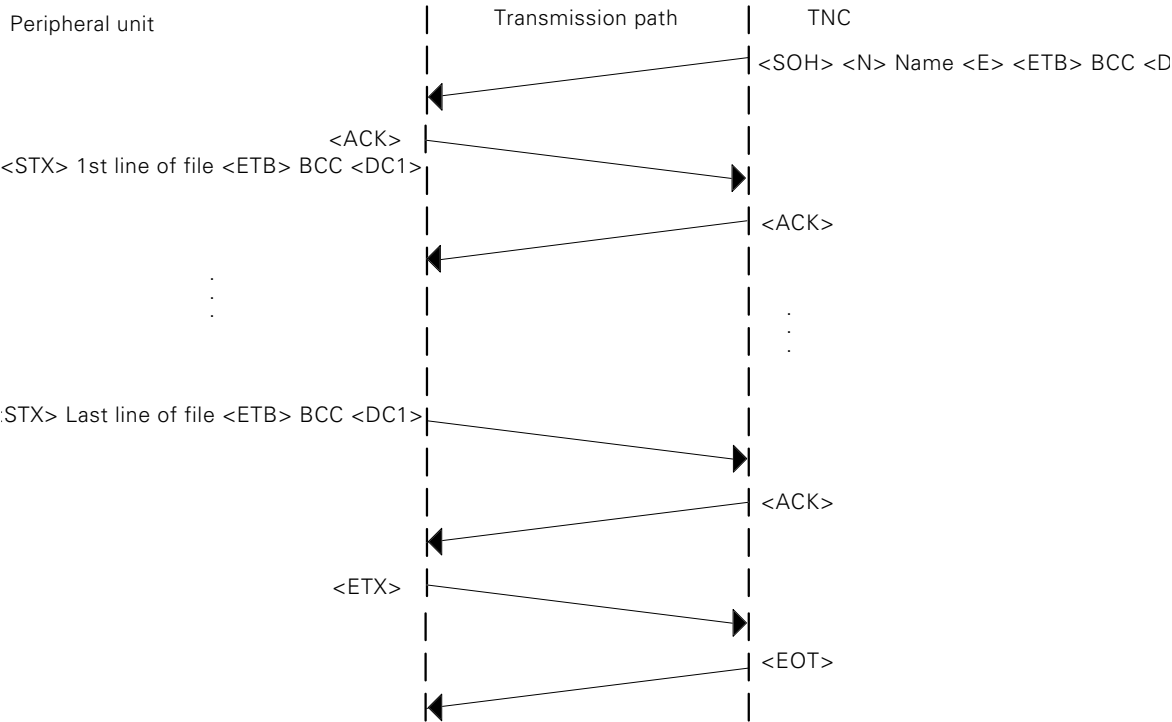
The program name may contain up to eight characters.

4.1.3 Outputting all programs

The files are output in order, as in Section 4.1.2 "Outputting a selected program". Control characters <ETX><EOT> are sent to the peripheral device between files.

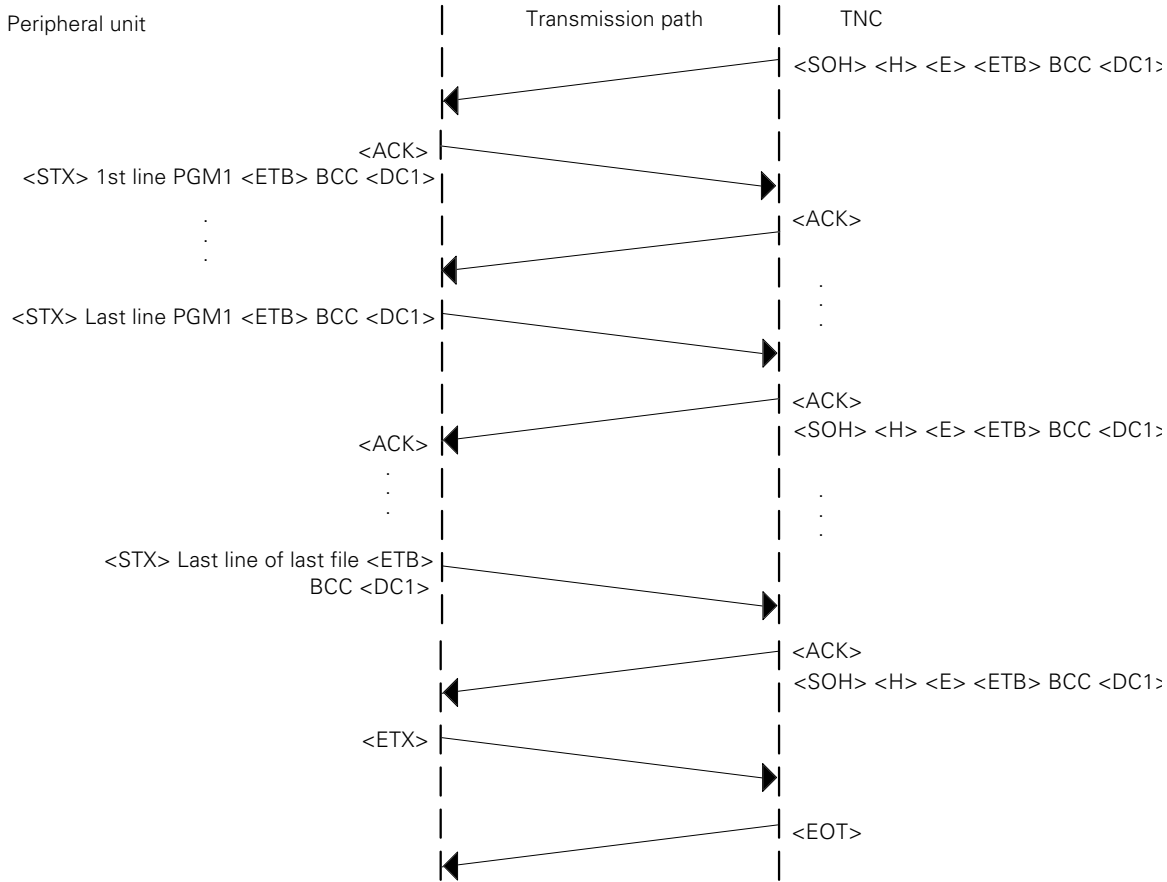
4.1.4 Reading in a selected program

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.



4.1.5 Reading in all programs

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.

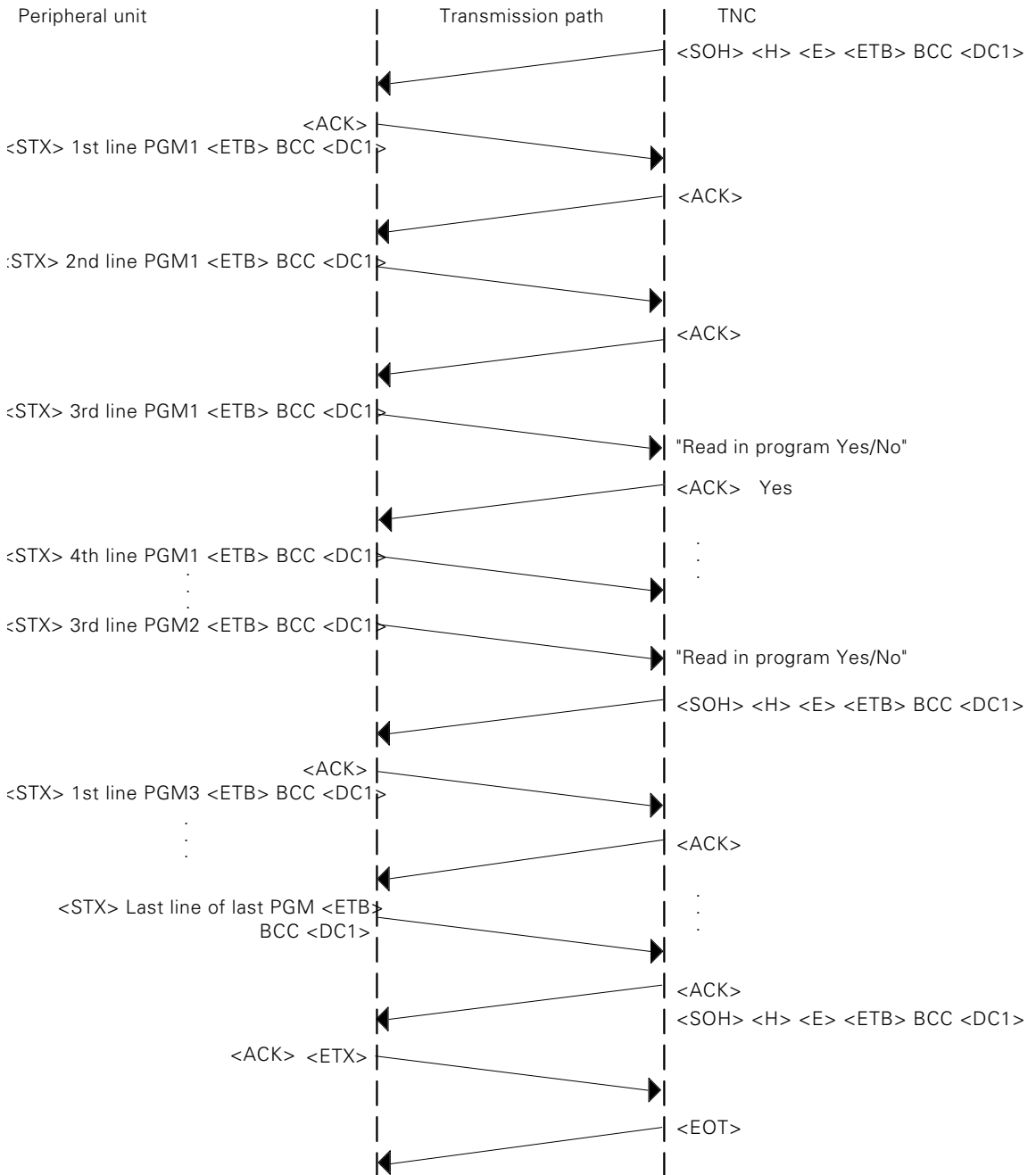


If the control is set to ISO programming, the central tool file (if activated via machine parameters) will also be read in.

4.1.6 Reading in an offered program

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 acknowledgment (no <ACK>) and awaits acknowledgment.

If a positive acknowledgment 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 acknowledgment will depend on the transfer rate set. At a low data transfer rate, the TNC will stop after the first block.

5 Data transfer by PLC

Using PLC modules (for description see Chapter "PLC programming", Section "PLC modules"), data can be transferred by the PLC via the RS-232-C/V.24 or RS-232-C/V.24 data interface. These modules, for example, permit communication between two logic units at PLC level via the interface.

5.1 Configuration of PLC data interface

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 FE or ME mode, or to configure it freely.

The appropriate control characters and the data format for EXT are set in the machine parameters, which are described in detail in Section "Freely configurable interfaces". In addition, the baud rate for transfer can be selected via MP5040.

MP5040

Data transfer rate for PLC interface

Input range: 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

6 Error messages

6.1 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.

Error messages occurring only in ME mode:

WRONG MODE
TRANSFERRED DATA INCORRECT
WRONG PROGRAM DATA
ME: END OF TAPE
DATA CARRIER FULL
DATA CARRIER EMPTY
DATA CARRIER WRITE-PROTECTED

General error messages:

INTERFACE ALREADY ASSIGNED
Transfer is already taking place via interface, or data transfer has not been completed.
PROGRAM INCOMPLETE
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 or faulty.
TRANSFERRED DATA INCORRECT X
X can assume the values A to H, K or L (error codes).

The error message TRANSFERRED DATA INCORRECT N is displayed if, in the case of data transfer with Block Check Character (BCC), a Not acknowledged (control character <NAK>) has been received three times.

The remaining error codes (A to H) in this error message indicate that an error has been detected in the received module. The error can have the following causes, with no assignment to the code letters:

- Same baud rate not set at TNC and peripheral unit
- Parity bit wrong
- Incorrect data frame (e.g. no stop bit)
- Receiving module of interface faulty

Error code K or L is only displayed for transfer with standard data transmission protocol:

- K: When an error was transmitted to TNC, character <1> not sent after the <ESC> character.
L: The wrong error number was received after the error sequence <ESC><1>. (Error number range: 0 to 7).

6.2 HEIDENHAIN peripherals' error codes

These error messages refer to the FE 401 floppy disk unit and to magnetic tape unit ME 101/ ME 102. With the FE 401 floppy disk unit 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.

6.3 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:

DATA CARRIER @: IS FULL

Data carrier >@:< full.

FILE NAME NOT PROGRAM NAME

Name of NC program and name of file do not match.

INSTRUCTION NOT ALLOWED

Request instruction issued by control unit is not allowed.

PROGRAM INCOMPLETE

NC program does not contain an end block.

PROGRAM NOT PRESENT

File requested by control unit does not exist in currently configured access path.

PROTECTED FILE

File which is protected with read-only or hidden attribute is likely to be overwritten.

SEARCH FEATURE NOT ALLOWED

Search feature not included in number of acceptable characters.

TRANSFERRED DATA INCORRECT

Attempts to transmit block to control unit has failed four times.

TRANSFERRED DATA INCORRECT N

There was no acknowledgment ACK within a defined time

A precise description of this software is given in the User's Manual of the transmission software.

OEM Cycles – Contents

1	Introduction	9-2
1.1	Creating OEM cycles	9-3
1.2	Dialogues for OEM cycles	9-6
1.3	Output in binary code	9-6
2	"Bolt Hole Circle" OEM-Cycle Example	9-7
3	Calling OEM Cycles	9-8
3.1	Calls in a HEIDENHAIN dialogue program	9-8
3.2	Calls in an ISO program	9-9

1 Introduction

OEM cycles (customized macros) are programmed in **HEIDENHAIN dialogue** as NC programs.

OEM cycles cannot be created in ISO format!

By using these cycles in a main program created in the HEIDENHAIN dialogue or in accordance with ISO, repetitive machining tasks or machine-specific functions can be executed with a single call. The execution of OEM cycles can be influenced by parameter transfer (see also TNC User's Manual).

OEM cycles can be divided into "DEF-active" and "CALL-active" OEM cycles. The DEF-active cycles are effective immediately upon definition, without CYCL CALL, M89 or M99. The CALL-active cycles must be called with a CYCL CALL, M89 or M99.

Up to 32 different OEM cycles 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 their dialogues, it is possible to store this data in the PLC EPROM.

Permitted functions in OEM cycles

- 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).
- Calling OEM cycles from a main programs that is transmitted "blockwise" and executed simultaneously. The OEM cycles must be stored in the control unit memory (EPROM or RAM).
- Calling HEIDENHAIN dialogue or ISO programs in OEM cycles. The called programs are not displayed.

Functions not permitted in OEM cycles

- M functions M02, M30, M06 with program-run stop.
- Programmed STOP block.
- 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 repeats 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 Creating OEM cycles

Programming an OEM-cycle and is only possible if the program number is in the range 99 999 968 to 99 999 999. Each of these program names is permanently assigned a cycle number (e.g. program number 999999**68** represents OEM-cycle **68**).

After creating a program with the above number and pressing LBL SET and then ENT, the control asks whether you wish to program a DEF-active or CALL-active cycle.

DEF-active cycle

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 coordinate transformation.

CALL-active cycle

If the OEM cycles are to be activated later in the NC program via CYCL-CALL, M98 or M99, a CALL-active OEM-cycle is programmed with the NO ENT key, e.g. for a machining cycle.

Up to 15 dialogue numbers can then be input. The first dialogue 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 dialogues are to be programmed, the dialogue block can end with END. If more than 15 parameters are to be entered (maximum 32), the DLG DEF block or the DLG CAL block must be entered repeatedly in the OEM cycle. Only five entries are evaluated in the third DLG block

The parameter assignment can be shifted via MP7250 so that DEF-active and CALL-active cycles do not address the same parameters Q1 to Q14.

Example cycle:

```
0 BEGIN PGM 99999968 MM
1 DLG-DEF 0/2/8/99
.
.
.
15 END PGM 99999968 MM
```

Using machine parameter MP7240, it is possible to inhibit program input for [Program number] = [OEM cycles number].

If MP7240 has a value 0, no program with the program number 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-number 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 dialogue support, Q parameters are assigned specified input values. The Q-parameter numbers are automatically generated by the TNC (Q1 to Q32). In order to prevent the same Q-parameter numbers being generated for DEF-active and CALL-active cycles, the difference between Q-parameter numbers can be specified with MP7250.

The input values of the OEM cycles are assigned in ascending order of magnitude to the Q parameters Q1 to Q14. The input values for the cycle are assigned to Q parameters Q[1 + MP7250] to Q[14 + MP7250].

Example for MP7250 = 30

	Parameter number in OEM-cycle with CALL-active	DEF-active
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 calling 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".

Example for global local Q parameters:
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 = local	Q60 = global	Q1 = local	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				
END PGM 100 MM	+2	+15	+2	+5

- MP7240** **Inhibit program input for [Program number] = [OEM-cycle number in EPROM].**
Input value 0 or 1
0 = inhibit
1 = not inhibit
- MP7250** **Difference between Q-parameter number for "CALL-active" and "DEF-active" 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 or non-existent OEM cycles are transferred to the control unit, an error message "PGM 999999.. UNAVAILABLE" is generated during program call. The program cannot be run.

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 Dialogues for OEM cycles

The dialogue numbers defined in the OEM-cycle determine the text to be displayed from the PLC EPROM

The following dialogue texts for the designation of the OEM-cycle and the input parameters are files in the standard PLC-EPROM.

Dialogue number in OEM-cycle	Standard dialogue in PLC EPROM
0	USER CYCL
1	CYCL PARAMETER 01
2	CYCL PARAMETER 02
.	.
.	.
.	.
99	CYCL PARAMETER 99

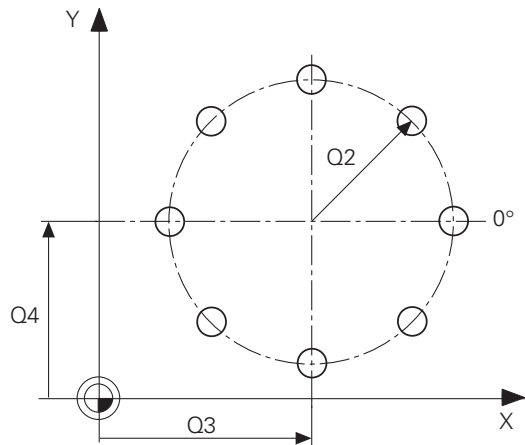
Instead of these standard dialogues up to 100 customized dialogues can be filed in the PLC-EPROM.

1.3 Output in binary code

If the OEM cycles have been fully tested, they can be output together with the PLC program in binary form for EPROM programming. It is possible to output both the files located in the PLC EPROM and those in the NC program memory in binary code. An accurate description of file output via the data interface is given in Chapter "PLC-programming".

2 "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 coordinate of center of bolt hole circle
Q4 = Y coordinate of center 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 99999968 MM
1 DLG-CALL 0/1/2/3/4/5/6/7
2 FN1: Q6 = +Q6 + +Q5
3 FN4: Q50 = +360 DIV + Q1
4 FN0: Q60 = +0
5 CC X+Q3 Y+Q4
6 LBL 11
7 LP PR +Q2 PA +Q60 R0 FMAX
8 L IZ +Q6 FQ7
9 L IZ -Q6 FMAX
10 FN 1: Q60 = +Q60 + +Q50
11 FN12: IF +Q60 LT +360 GOTO LBL 11
12 END PGM 99999968 MM
```

Dialogue block
Distance traversed in Z
Angle increment
Start angle
Center of bolt hole circle
Jump label
Approach drilling position
Drilling with feed
Clear
Next angle
Last bore?

Dialogues for "Bolt hole circle" OEM-cycle

Dialogue No.	DIALOGUE
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	DRILLING FEED RATE?

3 Calling OEM Cycles

OEM cycles in the NC program memory or PLC EPROM can be defined, called and executed both in HEIDENHAIN dialogue programs and in ISO programs.

3.1 Calls in a HEIDENHAIN dialogue program

In the HEIDENHAIN dialogue program, OEM cycles are defined as standard cycles (see "Dialogue Programming" in the TNC 360 User's Manual).

The dialogue 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 numeric 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	

3.2 Calls in an ISO program

In an ISO program, OEM cycles are not called via a G- function but via key "D" and the cycle number (e.g. D88).

The individual parameters are input via the numeric 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 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	Call of cycle 68 "Bolt hole circle"
P4+50 P5-2 P6-20 P7+100*	
N70 G79*	Cycle active
N99999 % 1000 G71 *	

Positioning Module – Contents

1	PLC Positioning Module	10-2
1.1	Introduction	10-2
1.2	Hardware	10-3
1.3	EMERGENCY STOP routine	10-5
1.4	Reference signal evaluation	10-6
1.5	Connecting the positioning module	10-6

1 PLC Positioning Module

1.1 Introduction

A hardware version of the LE 360 – **the LE 234.003** – can be used in conjunction with the TNC 407/TNC 415 as a positioning module.

In this way the TNC 407/TNC 415 can be expanded by up to 4 auxiliary axes. PLC inputs and outputs of the LE 234.003 can also be used.

The positioning module can be used for controlling tool changers and pallet feed as well as rotary axes and swivel heads.

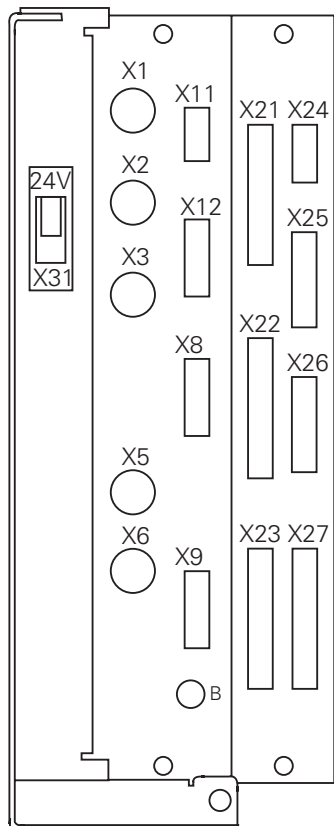
The NC software of the TNC 360 can also be run on the LE 234.003, i.e. all the functions of the TNC 360 are also possible with the positioning module.

This Technical Manual is therefore also applicable to the LE 234.003.

1.2 Hardware

The differences in the hardware of the LE 234.003 from the LE 360 C are:

- the connector X25 contains the RS-232/V.24 and the RS-422/V.11 interfaces
- The measuring system input X4 (sine-wave) has been dropped
- The measuring system input X5 (square-wave) has been added



Control-loop board

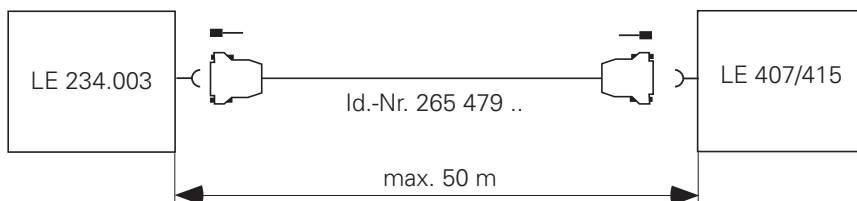
- X1 = Measuring system 1 (–)
- X2 = Measuring system 2 (–)
- X3 = Measuring system 3 (–)
- X5 = Measuring system 5 (□)
- X6 = Measuring system S (□)
- X8 = Nominal value output 1,2,3,4,S
- X9 = Visual display unit (only for commissioning)

B = Signal ground

- X21 = PLC output
- X22 = PLC input
- X23 = TNC keyboard (TE)
(only for commissioning)
- X24 = Power supply 24 V for PLC
- X25 = Data interface RS-422/V.11
(RS-232/V.24)
- X31 = Power supply 24 V for LE

X11, X12 and X26 are not needed,
X4 is not available.

Connecting cable



Pin layout

X25 Data interface RS-422/V.11 and RS-232/V.24-C

Flange socket with female insert (25-pin)

Pin Number	Assignment RS-422/V.11	Assignment RS-232/V.24-C
1		Screen
2		RxD
3		TxD
4		CTS
5		RTS
6		DTR
7		GND Signal
8	Vacant	
9	$\overline{\text{RxD}}$	
10	CTS	
11	$\overline{\text{TxD}}$	
12	RTS	
13	DSR	
14	DTR	
15	DTR	
16, 17, 18	Vacant	
19	GND Signal	
20	DSR	DSR
21	RxD	
22	CTS	
23	TxD	
24	RTS	
25 Housing	Ext. screen GND housing	

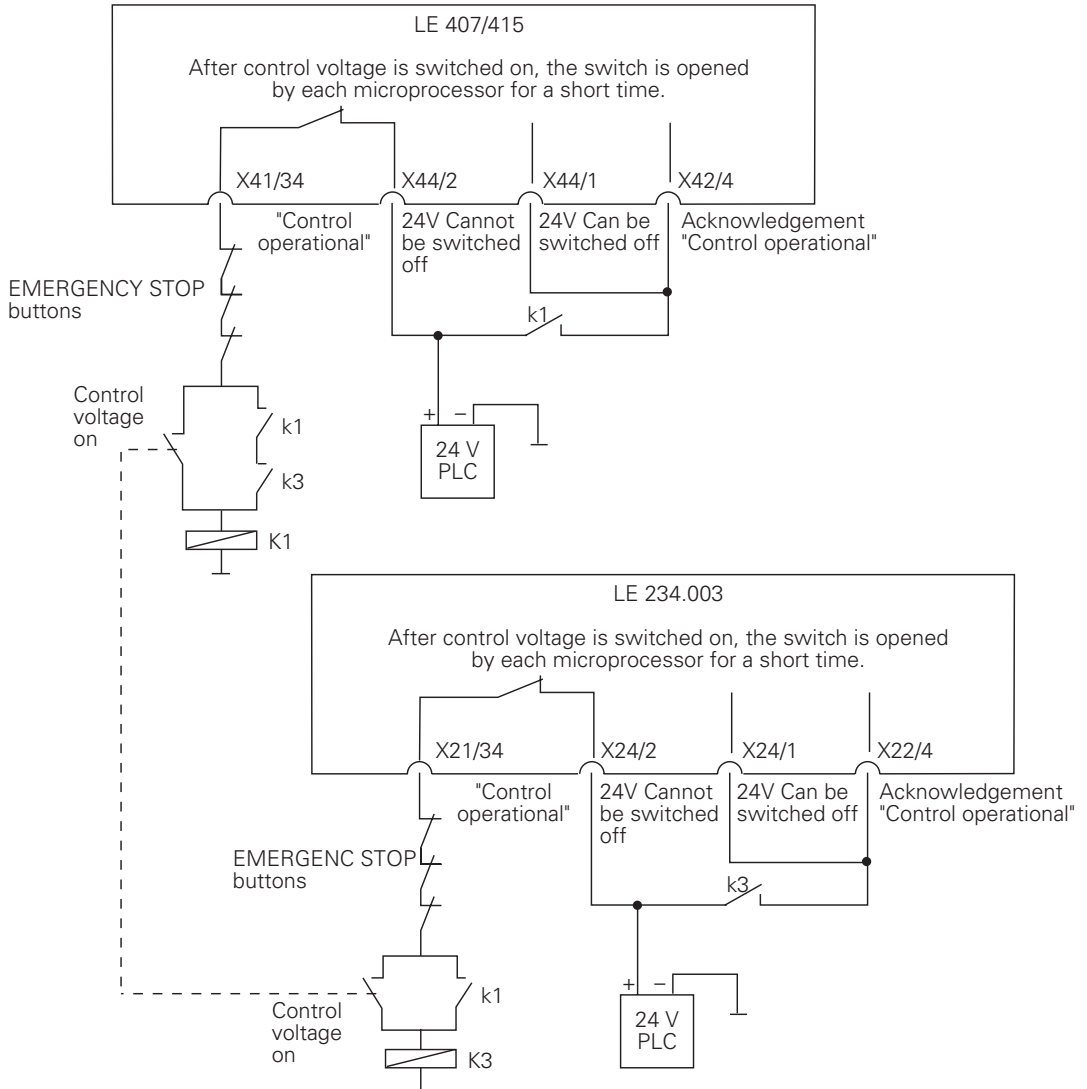
X5 Measuring system input 5 (□)

Pin Number	Assignment
1	Ua2
2	+ 5V (U _P)
3	Ua0
4	Ua0
5	Ua1
6	Ua1
7	$\overline{\text{UaS}}$
8	Ua2
9 (contact spring)	Screen = Housing
10	0 V (U _P)
11	0 V (U _P)
12	+ 5V (U _P)

1.3 EMERGENCY STOP routine

After the TNC has been switched on, the internal EMERGENCY STOP switch-off is checked, i.e. for each processor the power supply is briefly switched off.

Appropriate circuitry must ensure that the positioning module does not function if an error occurs during the NC check, i.e. that the voltage for the acknowledgment is mutually switched off. The circuit illustrated below can fulfill this function. The check is started at the same time that the control voltage is switched on.



1.4 Reference signal evaluation

After the reference signal evaluation in the NC axes, the reference signal evaluation of the PLC axes must be started, either automatically or (e.g. with NC switch-on) or manually with special keys. PLC positioning is not possible until the reference signals have been evaluated.

1.5 Connecting the positioning module

The positioning module hardware is connected at the RS-422/V.11 interface with a special cable (Id.-Nr. 265 479).

The software is connected through the PLC of the master control (TNC 407 or TNC 415). The modules 9100, 9101, 9102, 9105 and 9106 were developed for this purpose. They make it possible to activate the RS-422/V.11 data interface and to transmit and receive binary data.

These modules are described in Chapter 7 "PLC-programming".

The PLC program for the positioning module can be written either with the LE 234.003 itself (which must be connected to the BE 212 visual display unit and the TE 355 A/B keyboard), or on a PC with the PLC programming software from HEIDENHAIN.

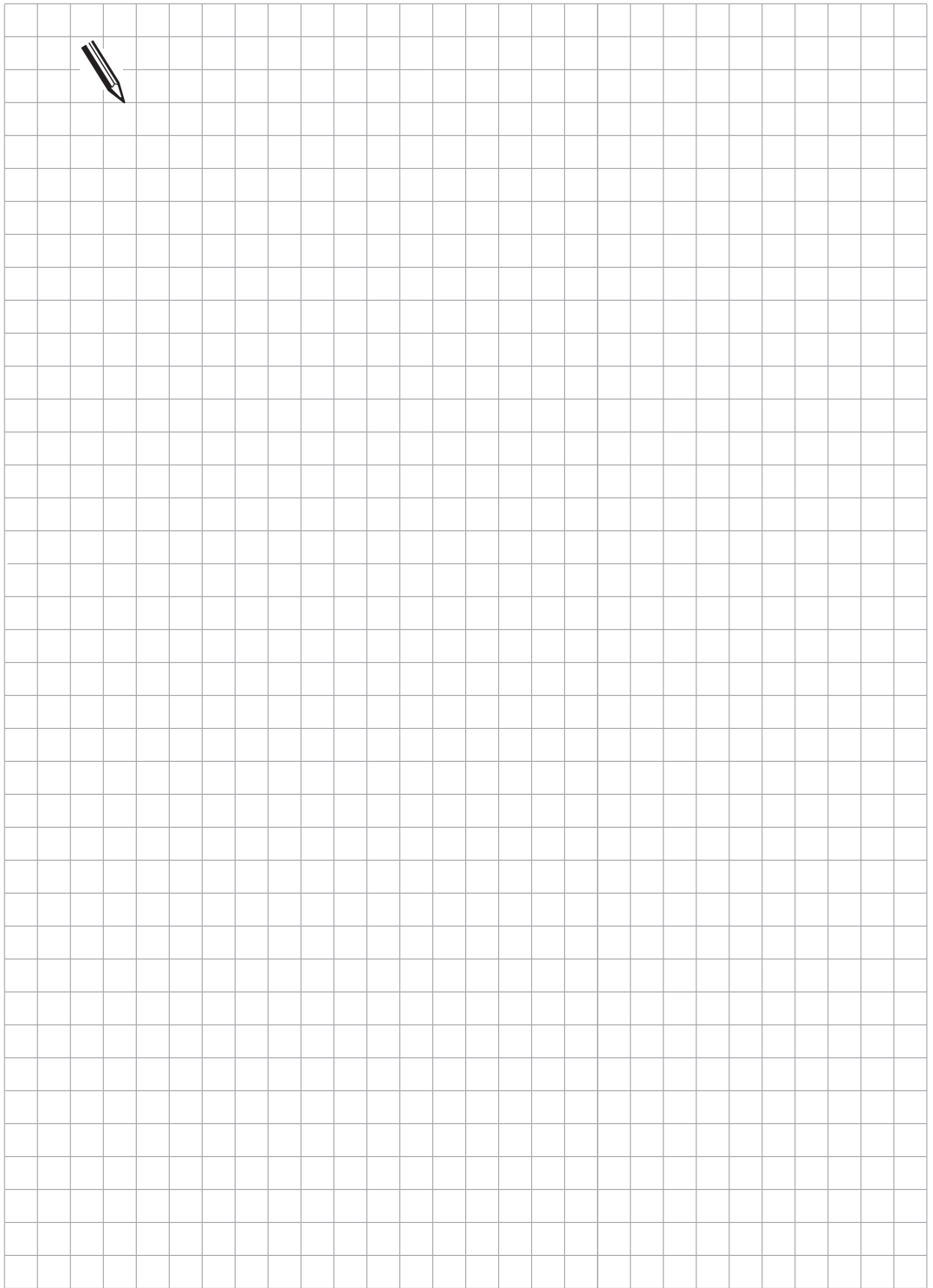
Please contact your HEIDENHAIN representative if you have any questions regarding the PLC programming software.

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



Subject Index

A

Acceleration.....	4-55
Acceleration, determination.....	4-229
Acceleration, optimizing	4-238
Acknowledgment of S-code	4-92
Active axes	4-6
Actual - nominal value transfer	4-78
Actual position capture	4-121
ADDITION (+)	7-57
ADDITION [] (+[])	7-74
Address allocation	7-14
Addressing the word memory.....	7-14
Ambient temperature	3-8
Analog inputs.....	4-176
Analog voltage.....	4-56
AND (A).....	7-44
AND [] (A[]).....	7-70
AND NOT (AN).....	7-46
AND NOT [] (AN[]).....	7-70
Angle encoders.....	3-19
Angular measurement.....	4-7
APE.....	3-30
Arithmetic Commands.....	7-57
Assembly hints	3-8
ASSIGN (=)	7-34
ASSIGN BYTE (B=).....	7-36
ASSIGN DOUBLEWORD (D=)	7-36
ASSIGN WORD (W=)	7-36
Assignment of the analogue outputs	4-14
Assignment of the measuring system inputs	4-13
Automatic cyclical offset adjustment	4-64
Axes in motion.....	4-77
Axes in position	4-76
Axis designation	4-12
Axis-direction keys, memory function for.....	4-149
Axis-enable	4-75
Axis-error compensation	4-22

B

Backlash compensation.....	4-22
Baud	8-8
Baud rate	8-8
BCC	8-18
BIT RESET (BC)	7-86
Bit Commands.....	7-85
BIT SET (BS)	7-85
BIT TEST (BT)	7-87
Block Check Character	8-18
Block-number step size for ISO programs	4-133
Buffer battery	3-16

C

Call Module (CM).....	7-96
Cascade control.....	4-52
CASE Statement.....	7-100
Change signal, S-code.....	4-92
Checking absolute position of distance-coded REF marks.....	4-9
Checking amplitude of measuring system signals.....	4-9
Checking data.....	8-7
Checking edge separation of measuring system signals.....	4-10
CN 123.....	5-14 – 5-23
Code for M function.....	4-138
Code number.....	4-134; 5-3
Coded output of spindle speed.....	4-92
Commands.....	7-27
Commissioning.....	4-227
Comparisons.....	7-63
Compatibility markers.....	7-123
Compatibility with TNC 355.....	7-121
Compensation for reversal errors.....	4-23
Configuration of the interface.....	8-22
Connection box.....	3-23
Constant feed rate in corners.....	4-67
Constant feed rate in corners with M90.....	4-68
Contour Lines.....	4-158
Control is ready.....	4-107
Control with feed forward control.....	4-61
Control with feed forward control, optimizing.....	4-237
Control with lag, optimizing.....	4-233
Control with servo lag.....	4-53
Controlled.....	4-75
Cooling.....	3-8
Correction table.....	4-25; 4-26
Counters.....	7-20
Counting direction.....	4-8
Current tool axis.....	4-14
Cycle call.....	4-140
Cycle inhibit.....	4-127
Cycles.....	4-127

D

Data bits.....	8-22
Data interface.....	3-32
Data transfer.....	8-4
Data transfer by PLC.....	8-42
Data transfer by the PLC.....	7-109
Data transfer PLC.....	7-15
Data transfer rate.....	8-8
Data transfer with Block Check Character (BCC).....	8-17
Data transmission protocol.....	8-29
Data transmission protocols.....	8-15
Datum.....	4-110
Datum correction.....	4-194
DBB Double Ball Bar system.....	4-23

Decimal sign	4-135
Degree of protection	3-11
Dialog language	4-135
Dialogues for OEM cycles	9-6
Digitizing feed rate in normal direction	4-156
Digitizing with TS 120	4-155
Digitizing, maximum deflection of stylus	4-156
Digitizing, number of oscillations in normal direction	4-156
Digitizing, optimizing the scanning sequence	4-161
Digitizing, scanning Cycles	4-158
Dimensions	3-58
Direction of traverse	4-8
Direction of traverse for referencing	4-49
Display "Control is ready"	4-119
Display of current gear range	4-121
Display spindle power	4-120
DIVISION (/)	7-60
DIVISION [] (/ [])	7-74

E

Edge evaluation of the PLC inputs	7-21
Electrical noise immunity	3-8
EMERGENCY STOP	4-109
EMERGENCY STOP DEFECTIVE	4-109
EMERGENCY STOP routine	4-107
EMERGENCY STOP routine, testing	4-231
Encoder cables	3-21
Encoder inputs	3-20
Encoder monitoring	4-9
Encoders	3-19; 4-6
End indexed Call Module (ENDC)	7-100
End of Module, Program End (EM)	7-98
EPROM creation	7-21
EPROM operation	7-21
EPROM sockets	2-9
EPROM test	4-136
EQUAL TO (==)	7-63
EQUAL TO [] (=[])	7-78
Error messages	4-122
EXCLUSIVE OR (XO)	7-52
EXCLUSIVE OR [] (XO[])	7-71
EXCLUSIVE OR NOT (XON)	7-54
EXCLUSIVE OR NOT [] (XON[])	7-71
EXT operating mode	8-28
External EMERGENCY STOP	4-107
External programming	8-28

F

Feed for PLC positioning	4-33
Feed potentiometer	4-116
Feed rate display	4-116
Feed rate enable	4-75
Feed rate for leaving the reference end-position	4-49

Feed rate for traversing the reference marks	4-49
Feed rate override	4-117
File types	4-133
File types – disable	4-133
Flashing PLC error message	4-123

G

Gear change	4-86
Gear change, Acknowledgment of	4-87
Gear change, jog voltage for	4-87
Gear code	4-86
Gear range	4-82
Gear range, S-analogue voltage for	4-83
Gear ranges, spindle speed for	4-83
Gear-code change signal	4-87
Graphic simulation	4-113
Graphics display	4-113
GREATER THAN (>)	7-65
GREATER THAN [] (>[])	7-78
GREATER THAN OR EQUAL TO (>=)	7-67
GREATER THAN OR EQUAL TOL[] (>=[])	7-79
Gross positioning error A	4-71
Gross positioning error B	4-72
Gross positioning error C	4-72
Gross positioning error D	4-73
Gross positioning error E	4-64
Gross positioning error A	4-70

H

Handshaking	8-9
Handwheel adapter HRA 110	3-37; 4-172
Handwheel HR 130	3-34; 4-169
Handwheel HR 330	3-33; 4-169
Handwheel HR 332	3-35; 4-169
Handwheel input	3-33
Handwheel keys	4-169
Handwheel, count direction for	4-167
Handwheel, minimum subdivision factor for	4-167
Handwheel	4-166
Hardware components	3-4
Hardware handshaking	8-9
Hardware version	2-7
Heat generation	3-8
Hirth coupling	4-183
Humidity	3-9

I

Incremental jog positioning	4-178
Indexed Call Module (CASE)	7-100
Inhibited key	4-141
Integral factor	4-65
Integral factor, optimizing	4-242

Interface RS-232-C/V.24	8-10
Interpolation factor	4-7; 5-6
ISO programs	4-133

J

Jump Commands	7-94
Jump if Logic Accumulator = 0 (JPF)	7-94
Jump if Logic Accumulator = 1 (JPT)	7-94
Jump Label (LBL).....	7-98

K

Key code.....	4-141
Key simulation	4-141
Kink point.....	4-59
kv factor.....	4-55; 4-63

L

Lag monitoring.....	4-70
Lead-screw pitch error.....	4-25
LESS THAN (<)	7-64
LESS THAN [] (<[])	7-78
LESS THAN OR EQUAL TO (<=)	7-66
LESS THAN OR EQUAL TO [] (<=[])	7-78
Linear axis-error compensation	4-24
Linear encoders	3-19
Linear measurement	4-6
LOAD (L).....	7-27
LOAD BYTE (LB).....	7-32
Load command.....	7-27
LOAD DOUBLEWORD	7-32
LOAD NOT (LN).....	7-29
Load Stack (PS).....	7-89
Load Stack (PSL).....	7-90
Load Stack (PSW).....	7-91
LOAD TWO'S COMPLEMENT (L-)	7-31
LOAD WORD (LW).....	7-32
Logic Gates.....	7-44
Logic unit.....	3-4
Lubrication pulse	4-19

M

M function, acknowledgment of	4-138
M functions	4-118; 4-137
Machine axes	4-6
Machine control panel	3-53; 4-149
Machine datum.....	4-50; 4-110
Machine parameters	5-3; 5-4
Manual feed.....	4-57
Meander	4-158
Measuring system monitoring	4-9
Measuring systems	3-19; 4-6
Mechanical defects	4-22

Mechanical vibration.....	3-9
Memory test.....	4-136
Modul 9120 Position an auxiliary axis.....	7-114
Module 9000\ Copy a marker block.....	7-103
Module 9001\ Copy a Word block.....	7-103
Module 9010\ Indexed reading of Byte.....	7-104
Module 9011\ Indexed reading of Word.....	7-104
Module 9012\ Indexed reading of Doubleword.....	7-104
Module 9020\ Indexed writing of Byte.....	7-105
Module 9021\ Indexed writing of Word.....	7-105
Module 9022\ Indexed writing of Doubleword.....	7-105
Module 9032\ Transfer machine parameter values to the PLC.....	7-106
Module 9035\ Transfer status information to the PLC.....	7-107
Module 9036\ Transfer PLC status information to the NC.....	7-108
Module 9040\ Transfer coordinate values to the PLC.....	7-109
Module 9100\ Assign data interface.....	7-110
Module 9101\ Enable data interface.....	7-110
Module 9102\ Interface status.....	7-111
Module 9105\ Transmit binary data via data interface.....	7-112
Module 9106\ Receive binary data via data interface.....	7-113
Module 9121 Stop an auxiliary axis.....	7-115
Module 9122 Poll the status of an auxiliary axis.....	7-115
Module 9123 Traverse the reference point of an auxiliary axis.....	7-116
Module 9124 Override for PLC axis.....	7-117
Module 9150 Run an NC block.....	7-118
Module call if Logic Accumulator = 0 (CMF).....	7-96
Module call if Logic Accumulator = 1 (CMT).....	7-96
Module technique.....	7-13
Monitoring functions.....	4-69
Monitoring the analog voltage.....	4-72
Mounting position.....	3-9
Movement-monitoring.....	4-72
MULTIPLICATION (x).....	7-59
MULTIPLICATION [] (x[]).....	7-74
Multiplication factor.....	4-59

N

NC power supply.....	3-14
Nominal value output.....	3-23
Nominal value potential.....	3-23
Nonlinear axis error compensation.....	4-25
Normal direction.....	4-156
NOT EQUAL TO [] (<>[]).....	7-79

O

OEM cycles.....	9-3
OEM cycles, calling.....	9-8
Offset adjustment.....	4-64; 4-243
Offset adjustment by code number.....	4-64
Offset adjustment with integral factor.....	4-65
Open control loop.....	4-78
Operand directory.....	7-14

Operating mode.....	4-141
OR (O).....	7-48
OR [] (O[]).....	7-70
OR NOT (ON).....	7-50
OR NOT [] (ON[]).....	7-70
orientation.....	4-94
Overlap factor	4-129

P

Parentheses.....	7-70; 7-74
Path dependent lubrication.....	4-19
PLC – Main menu.....	7-6
PLC auxiliary axes.....	4-36
PLC axes.....	4-36
PLC Cycle time	7-128
PLC error messages	4-122; 7-24
PLC Functions	7-5
PLC I/O unit	3-45
PLC I/O unit PL 410	3-49
PLC input, fast.....	4-69
PLC Input/Output board.....	3-11
PLC Input/Output unit.....	3-5
PLC inputs	3-40
PLC Modules	7-103
PLC operation, select	7-5
PLC outputs.....	3-40
PLC positioning.....	4-33
PLC positioning module.....	3-2
PLC power supply	3-15
PLC Program conversion	7-122
PLC program creation.....	7-12
PLC program editing.....	7-6
PLC program Transferring	7-10
PLC program, erase.....	7-8
PLC program, transfer from EPROM	7-8
PLC program, translate.....	7-8
PLC TABLE function	7-10
PLC text window	4-122
PLC TRACE function.....	7-9
PLC utilization, processing time, memory	7-8
Pocket coding	4-199
Pocket coding, fixed	4-199
Pocket coding, variable.....	4-199
Pocket milling	4-129
Point-to-point and straight-cut control	4-246
Polarity of the nominal value potential	4-8
Polarity of the S-analogue voltage	4-81
Position approach, optimizing.....	4-240
Position control loop	4-52
Position display.....	4-114
Position display step.....	4-114
Position loop gain	4-55
Position monitoring.....	4-70; 4-71

positioning error.....	4-70; 4-71
Positioning module	3-2
Positioning window	4-73
Potential steps.....	4-58
Power consumption	2-5
Power supply	3-14
PROCESSOR CHECK ERROR M	3-15
Program, end of.....	4-136
Programming station	4-135
Pull Stack (PLL).....	7-91
Pull Stack (PLW).....	7-91
Pull Stack (PL).....	7-90

Q

Q parameters, overwriting	4-136
---------------------------------	-------

R

Radial acceleration.....	4-67
RAM operation	7-21
RAM test	4-136
Ramp slope for S-analogue voltage	4-82
Rapid traverse.....	4-55
Reference end-position	4-39; 4-50
Reference marks	4-37
reference marks, passing over	4-39
Reference point.....	4-110
RELAY EXT. DC VOLTAGE MISSING.....	4-109
REMAINDER (MOD)	7-61
REMAINDER [] (MOD[I])	7-75
RESET (R).....	7-40
RESET NOT (RN)	7-42
Residual voltage	4-71
Rigid tapping.....	4-104
Rigid tapping, Acceleration ramp for	4-105
Rotary axes.....	4-115
Rotary axis, display mode for	4-115
Rotational axis	4-12
RS-232-C/V.24	3-32
Run duration timers.....	4-118
Run duration, display of.....	4-118
Run-in behavior.....	4-62

S

S code.....	4-92
Sag.....	4-25
Scale datum point.....	4-110
Scaling factor.....	4-131
Scanning range	4-158
S-code table.....	4-93
Secondary linear axis	4-13
Sequence for traversing reference marks	4-49
Servo accuracy	4-57

Servo lag.....	4-55
Servo positioning of the NC-axes	4-52
SET (S).....	7-39
Set commands	7-39
SET NOT (SN).....	7-41
Shift Commands.....	7-82
SHIFT LEFT (<<).....	7-82
SHIFT RIGHT (>>)	7-83
Signal period	4-6
Signal subdivision	4-7
Software handshaking.....	8-9
Software limit switch.....	4-16
Software limit switch ranges.....	4-233
Software number	2-6
Software option	2-8
Software protection module.....	2-8
Software version	2-6
Special Tools.....	4-197
Specifications	2-3
Spindle.....	4-80
Spindle control loop, open	4-94
Spindle display, expanded	4-120
Spindle orientation.....	4-80; 4-94
Spindle orientation, kv factor for	4-96
Spindle orientation, ramp slope	4-94
Spindle override.....	4-85
Spindle positioning window.....	4-96
Spindle power.....	4-120
Spindle preset.....	4-96
Spindle rotation, direction of.....	4-81
Spindle rpm, analog output.....	4-81
Spindle run-in characteristic	4-105
Spindle speed	4-81
Spindle, count direction of the measuring system signals.....	4-82
Spindle, transient response on acceleration.....	4-105
Spindle-enable	4-82
Spindle-speed for spindle-orientation	4-94
Spindle-speed output.....	4-80
Stack Operations	7-89
Standard transmission protocol.....	8-16
Standstill monitoring.....	4-73
Status display	4-114
Status display, cancel	4-120
Stiction.....	4-23
Strobe signal for M function	4-138
Subdivision factor	4-166
SUBTRACTION (-)	7-58
SUBTRACTION [] (-[])	7-74
Summary of connections	3-12

T

Tapping.....	4-100
--------------	-------

Tapping with floating tap holder	4-100; 4-103
Tapping without floating tap holder	4-104
Tapping, ramp slope for	4-101
Temperature compensation	4-30
Thread pitch	4-104
Timers	7-17
TNC keyboard	3-4; 3-56
Tool changer	4-196
Tool length, calculation of	4-114
Tool magazine, number of pockets in	4-214
Tool table	4-196
Touch Probe	4-151
Touch probe feed rate	4-152
Touch probe maximum measuring range	4-152
Touch probe system input	3-28
Touch probe system, selection of	4-152
TRACE BUFFER, DISPLAY	7-10
TRACE, END	7-10
TRACE, START	7-10
Transient response	4-62
Transient response, optimizing	4-239
Transmission format	8-8
Traverse direction, testing	4-232
Traverse ranges	4-16
TS 120	3-29
TS 511	3-30
Tuning of the drive amplifier	4-228

U

Unconditional jump (JP)	7-94
UNEQUAL (<>)	7-68
User parameters	4-133; 5-2
User parameters, dialogs for	4-134

V

VDU display	4-14
Visual display unit	3-5; 3-26
VM 101A	4-25
voltage step	4-58

W

Weight	2-5
Wrong spindle speed	4-88