SIEMENS

SINUMERIK

SINUMERIK 802D sl
Surface grinding

Programming and Operating Manual

Valid for controller
SINUMERIK 802D sl G/N

Software version
1.4

07/2009
6FC5398-5CP10-1BA0

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠DANGER
indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠WARNING
indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠CAUTION
with a safety alert symbol, indicates that minor personal injury can result if proper precautions are not taken.

CAUTION
without a safety alert symbol, indicates that property damage can result if proper precautions are not taken.

NOTICE
indicates that an unintended result or situation can occur if the corresponding information is not taken into account.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation for the specific task, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be adhered to. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of the Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
GERMANY

Ordernumber: 6FC5398-5CP10-1BA0
℗ 08/2009

Preface

Structure of the documentation

The SINUMERIK documentation is organized in 3 parts:

- General documentation
- User documentation
- Manufacturer/service documentation

Information on the following topics is available at http://www.siemens.com/motioncontrol/docu:

- Ordering documentation
  Here you can find an up-to-date overview of publications.
- Downloading documentation
  Links to more information for downloading files from Service & Support.
- Researching documentation online
  Information on DOConCD and direct access to the publications in DOConWEB.
- Compiling individual documentation on the basis of Siemens contents with the My Documentation Manager (MDM), refer to http://www.siemens.com/mdm

  My Documentation Manager provides you with a range of features for generating your own machine documentation.
- Training and FAQs
  Information on the range of training courses and FAQs (frequently asked questions) are available via the page navigation.

Target group

This publication is intended for programmers, planning engineers, machine operators and system operators.

Benefits

With the Programming and Operating Manual, the target group can develop, write, test and debug programs and software user interfaces.

In addition, it enables the target group to operate the hardware and software of a machine.

Standard scope

This documentation only describes the functionality of the standard version. Extensions or changes made by the machine tool manufacturer are documented by the machine tool manufacturer.

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

For the sake of simplicity, this documentation does not contain all detailed information about all types of the product and cannot cover every conceivable case of installation, operation, or maintenance.

Technical support

If you have any technical questions, please contact our hotline:

	Europe / Africa
Phone	+49 180 5050 222
Fax	+49 180 5050 223
€ 0.14/min. from German landlines, mobile phone prices may differ.	
Internet	http://www.siemens.com/automation/support-request

	America
Phone	+1 423 262 2522
Fax	+1 423 262 2200
E-mail	mailto:techsupport.sea@siemens.com

	Asia/Pacific
Phone	+86 1064 757575
Fax	+86 1064 747474
E-mail	mailto:support.asia.automation@siemens.com

Note

Country telephone numbers for technical support are provided under the following Internet address:

http://www.automation.siemens.com/partner

Questions regarding documentation

If you have any queries (suggestions, corrections) in relation to this documentation, please fax or e-mail us:

Fax	+49 9131 98 2176
E-mail	mailto:docu.motioncontrol@siemens.com

A fax form is available in the appendix of this document.

SINUMERIK Internet address

http://www.siemens.com/sinumerik

EC Declaration of Conformity

The EC Declaration of Conformity for the EMC Directive can be found/obtained

- on the internet:
  http://support.automation.siemens.com
  under the product/order No. 15263595
- at the relevant regional office of the I DT MC Business Unit of Siemens AG.

Table of contents

Description 1

1.1 Control and display elements

Operator control elements

The defined functions are called up via the horizontal and vertical softkeys. For a description, please refer to this manual:

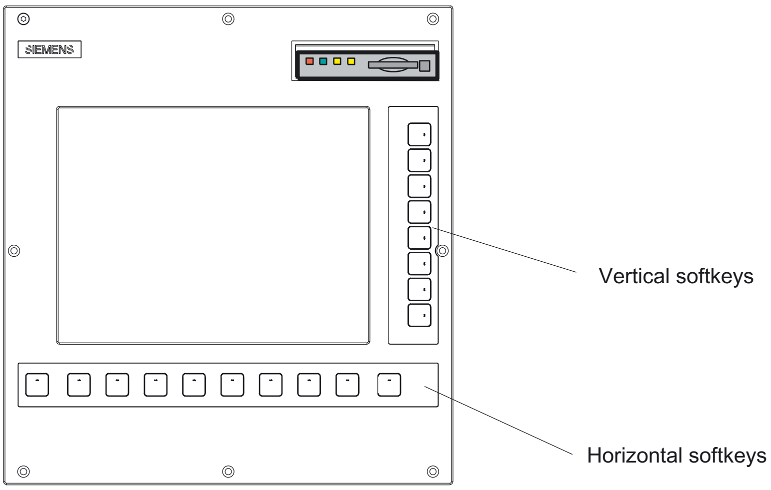
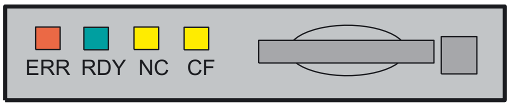


Figure 1-1 CNC operator panel

1.2 Error and status displays

LED displays on the CNC operator panel (PCU)

The following LEDs are installed on the CNC operator panel.



The individual LEDs and their functions are described in the table below.

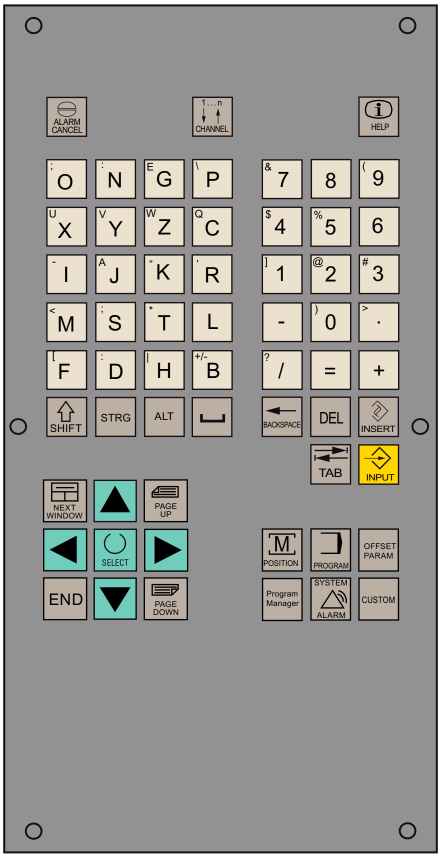
Table 1- 1 Status and error displays

LED	Significance
ERR (red)	Serious error, remedy through power OFF/ON
RDY (green)	Ready for operation
NC (yellow)	Signoflife monitoring
CF (yellow)	Reading from/writing to CF card

References

You can find information on error description in the SINUMERIK 802D sl Diagnostics Manual

1.3 Key definition of the full CNC keyboard (vertical format)



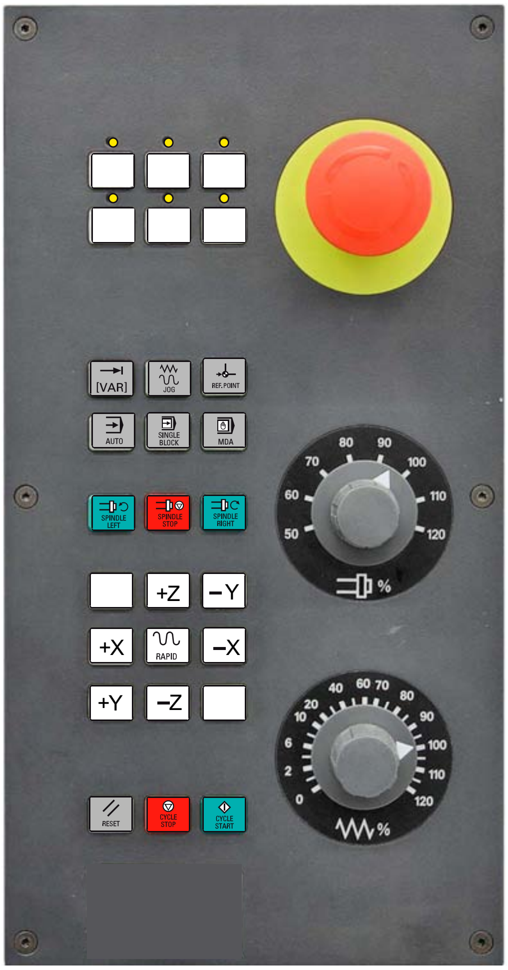
>	ETC key
Λ	Recall key
ALARM CANCEL	Acknowledge alarm key
1…n CHANNEL	No function
HELP	Info key
END	
SHIFT	Shift key
SELECT	Selection key / toggle key
CTRL	Control key
␣	Space
ALT	ALT key
BACKSPACE	Delete key (backspace)
DEL	Delete key
INSERT	Insert key
TAB	Tabulator
INPUT	ENTER / Input key
POSITION	POSITION operating area key (Position operating area)
PROGRAM	PROGRAM operating area key (Program operating area)
OFFSET PARAM	OFFSET PARAM operating area key (Parameter operating area)
PROGRAM MANAGER	PROGRAM MANAGER operating area key (Program Manager operating area)
SYSTEM ALARM	SYSTEM/ALARM operating area key (System/Alarm operating area)
CUSTOM	CUSTOM operating area (User operating area)
NEXT WINDOW	not assigned
PAGE UP, PAGE DOWN	Scroll keys
▲ ◀ ▶ ▼	Cursor keys
A J, W Z	Alphanumeric keys Double assignment at the Shift level
(9,) 0	Numeric keys Double assignment at the Shift level

Hot keys

In the part program editor and in the input fields of the HMI, the following functions can be carried out with certain key combinations on the full CNC keyboard:

Keystroke combination	Function
<CTRL> and <C>	Copy selected text
<CTRL> and <B>	Select text
<CTRL> and <X>	Cut selected text
<CTRL> and <V>	Paste copied text
<ALT> and <L>	Changeover to small letters
<ALT> and <H> or <HELP> key	Call help system
<ALT> and <S>	Switch-in and switch-out the Editor for Asian characters

1.4 Key definition of the machine control panel



RESET

CYCLE STOP
(NC STOP)

CYCLE START
(NC START)

EMERGENCY STOP

Spindle Speed Override
Spindle override

User-defined key with LED

User-defined key without LED

INCREMENT
Increment

JOG

REFERENCE POINT
Reference point

AUTOMATIC

SINGLE BLOCK
Single block

MANUAL DATA
Manual input

SPINDLE START CCW
Counterclockwise

SPINDLE STOP

SPINDLE START CW
Clockwise

RAPID TRAVERSE OVERLAY
Rapid traverse override

X axis

Y axis

Z axis

Feedrate override
Feedrate control

Note

This documentation assumes an 802D standard machine control panel (MCP). Should you use a different MCP, the operation may be other than described herein.

1.5 Coordinate systems

As a rule, a coordinate system is formed from three mutually perpendicular coordinate axes. The positive directions of the coordinate axes are defined using the so-called "3-finger rule" of the right hand. The coordinate system is related to the workpiece and programming takes place independently of whether the tool or the workpiece is being traversed. When programming, it is always assumed that the tool traverses relative to the coordinate system of the workpiece, which is intended to be stationary.

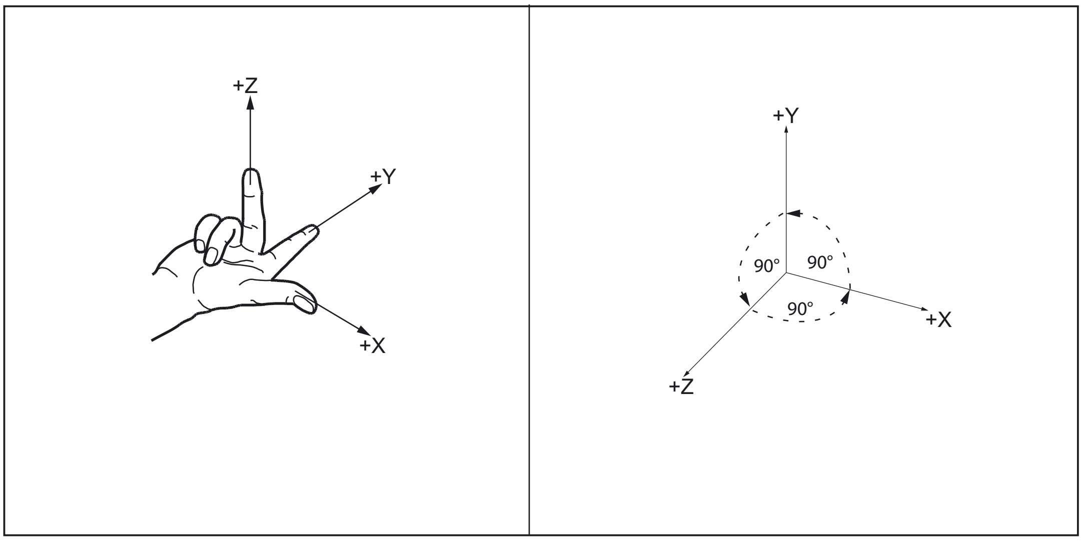


Figure 1-2 Determination of the axis directions to one another; coordinate system for programming

Machine coordinate system (MCS)

The orientation of the coordinate system relative to the machine depends on the respective machine type. It can be rotated in different positions.

The directions of the axes follow the "3-finger rule" of the right hand. Seen from in front of the machine, the middle finger of the right hand points in the opposite direction to the infeed of the main spindle.

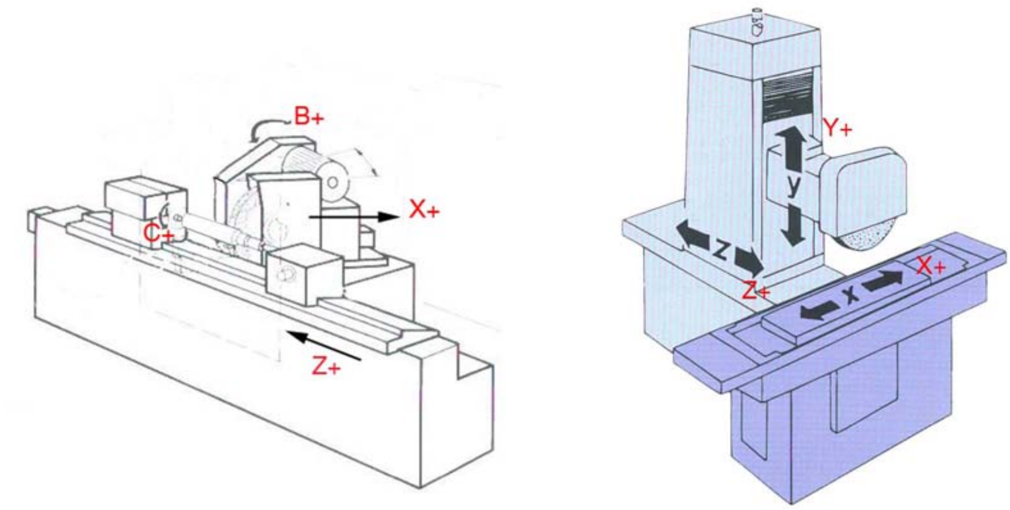


Figure 1-3 MCS for grinding (cylindrical grinding machine, surface grinding machine)

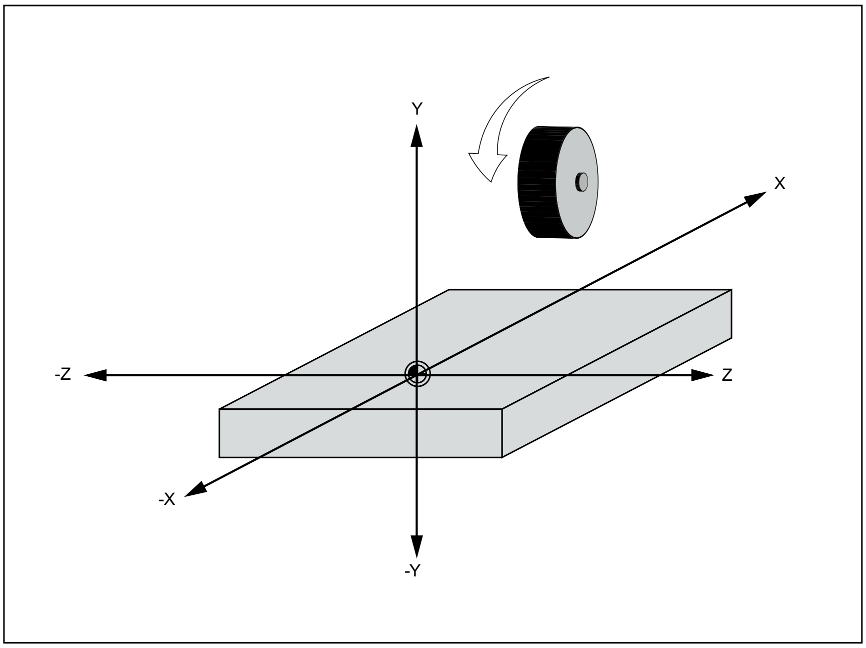


Figure 1-4 Machine coordinate system

The origin of this coordinate system is the **machine zero.**
This point is only a reference point which is defined by the machine manufacturer. It does not have to be approachable.

The traversing range of the **machine axes** can be in the negative range.

Workpiece coordinate system (WCS)

To describe the geometry of a workpiece in the workpiece program, a right-handed, right-angled coordinate system is also used.
The **workpiece zero** can be freely selected by the programmer in the Y axis. In the Z axis, it lies in the turning center.

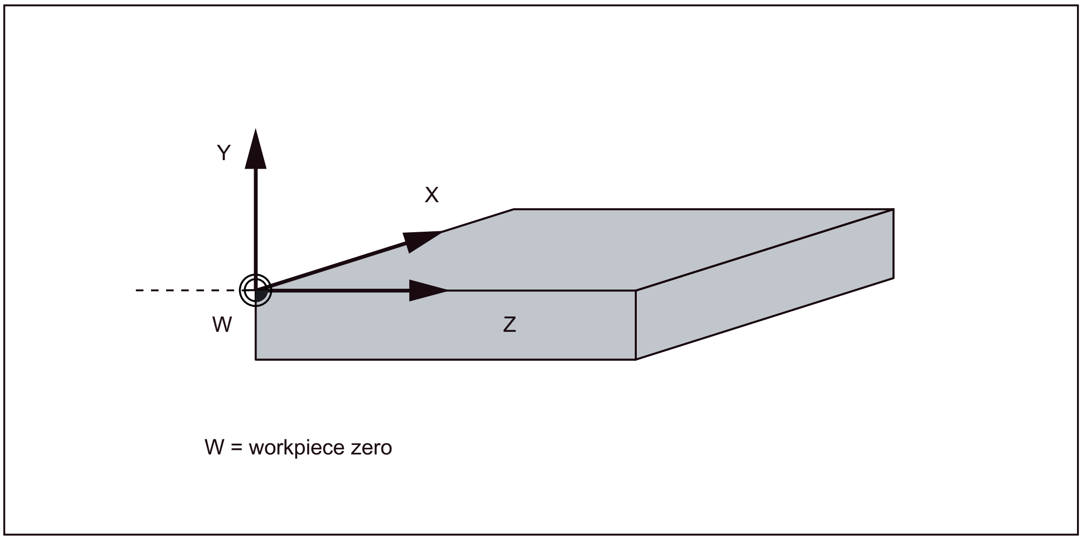


Figure 1-5 Workpiece Coordinate System

Relative coordinate system (REL)

In addition to the machine and workpiece coordinate systems, the control system provides a relative coordinate system. This coordinate system is used for setting reference points that can be freely selected and have no influence on the active workpiece coordinate system. All axis movements are displayed relative to these reference points.

Note

The actual value in the associated coordinate system can be activated and displayed in the "Position" operating area using the "MKS/WKS REL" vertical softkey.

Clamping the workpiece

For machining, the workpiece is clamped on the machine. The workpiece must be aligned such that the axes of the workpiece coordinate system run in parallel with those of the machine. Any resulting offset of the machine zero with reference to the workpiece zero is determined along the Y axis and entered in a data area intended for the **settable work offset**. In the NC program, this offset is activated during program execution, e.g. using a programmed **G54**.

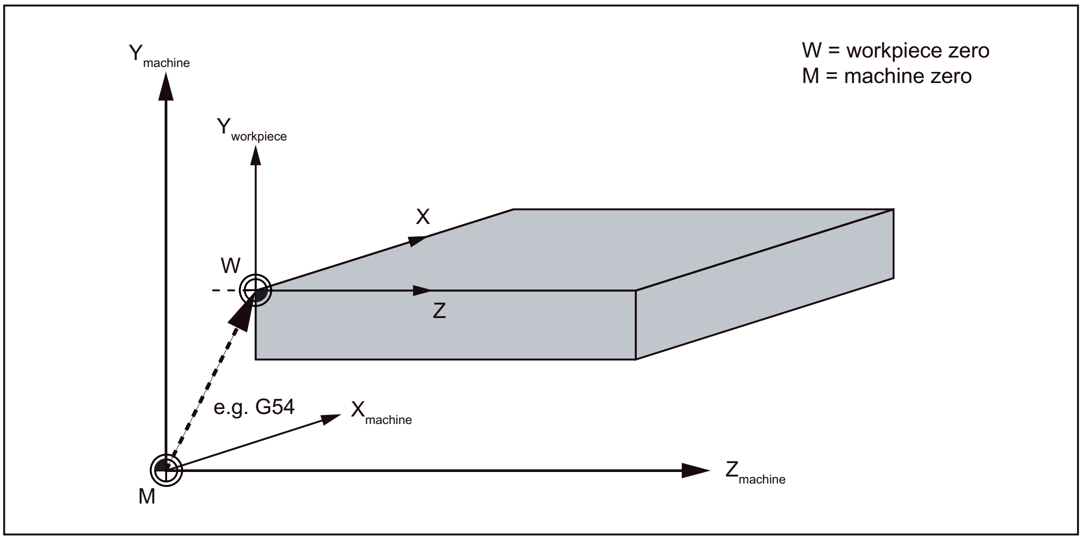


Figure 1-6 Workpiece on the machine

Current workpiece coordinate system

The programmed work offset TRANS can be used to generate an offset with reference to the workpiece coordinate system. resulting in the current workpiece coordinate system (see Section "Programmable work offset: TRANS").

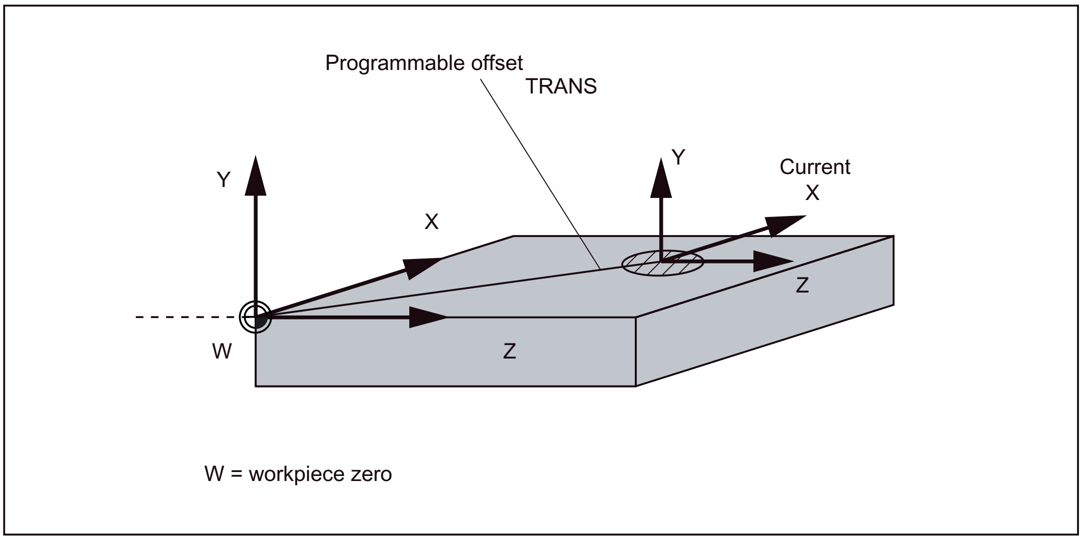


Figure 1-7 Coordinates on the workpiece; current workpiece coordinate system

2 Software user interface

2.1 Screen layout

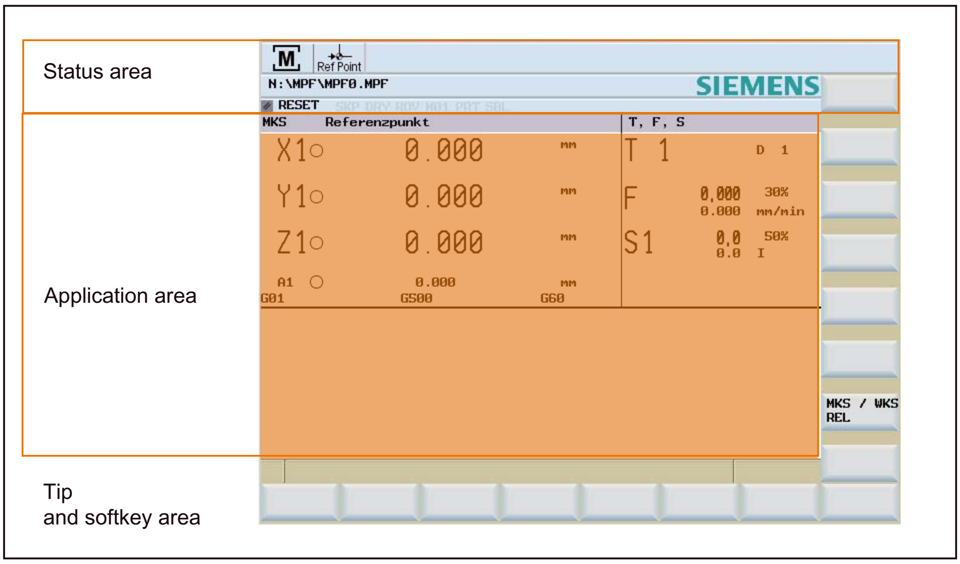


Figure 2-1 Screen layout

The screen is divided into the following main areas:

- Status area
- Application area
- Note and softkey area

Status area

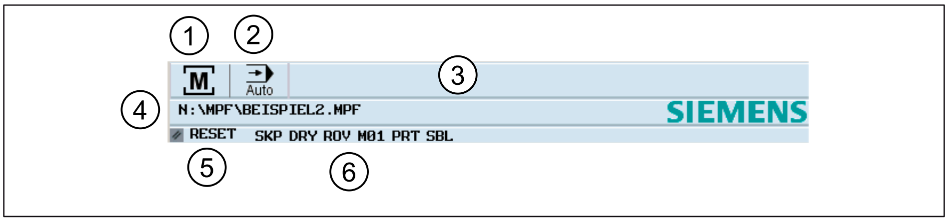
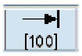


Figure 2-2 Status area

Table 2- 1 Explanation of the screen controls in the status area

Numbering	Display	Icon	Significance
①	Active operating area		Position (operating area key <POSITION>)
			System (operating area key <SYSTEM>)
			Program (operating area key <PROGRAM>)
			Program Manager (operating area key <PROGRAM MANAGER>)
			Parameter (operating area key <OFFSET PARAM>)
			Alarm (operating area key <ALARM>)
②	Active mode		Approaching a reference point
			JOG
			JOG INC; 1 INC, 10 INC, 100 INC, 1000 INC, VAR INC (incremental evaluation in the JOG mode)
			MDA

Numbering	Display	Icon	Significance
			AUTOMATIC
③	Alarm and message line		In addition, the following is displayed: 1. Alarm number with alarm text, or 2. Message text
④	Selected part program (main program)		
⑤	Program state	RESET	Program canceled / default state
		RUN	Program is running
		STOP	Program stopped
⑥	Program control in automatic mode		

Note and softkey area

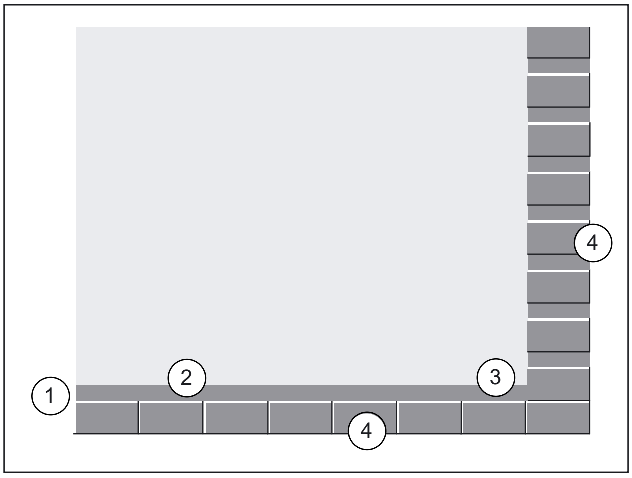
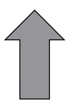
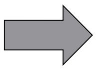
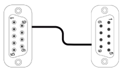
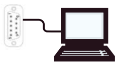


Figure 2-3 Note and softkey area

Table 2- 2 Explanation of the screen controls in the note and softkey area

Screen item	Display	Significance
①		RECALL symbol Pressing the <RECALL> key lets you return to the higher menu level.
②		Information line Displays notes and information for the operator and fault states

Screen item	Display	Significance
③		HMI status information
		ETC is possible (pressing this key displays the horizontal softkey bar providing further functions.)
		Mixed notation active (uppercase/lowercase letters)
		RS232 connection active
		Connection to commissioning and diagnostic tools (e.g. Programming Tool 802) active
		RCS network connection active
④		Softkey bar vertical and horizontal

Display of the softkeys in the document

To make the softkeys easier to locate, the horizontal and vertical softkeys are displayed in different basic colors.

Horizontal softkey

Vertical softkey

2.2 Operating areas

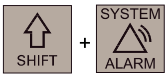
The functions of the control system can be carried out in the following operating areas:

POSITION — Machine operation

OFFSET PARAM — Entering the compensation values and setting data

PROGRAM — Creation of part programs

PROGRAM MANAGER — Part program directory



SYSTEM — Diagnostics, commissioning

ALARM — Alarm and message lists

CUSTOM — Users can call their own application

To change to another operating area, press the relevant key on the CNC full keyboard (hard key).

Protection levels

The SINUMERIK 802D sl provides a concept of protection levels for enabling data areas. The control system is delivered with default passwords for the protection levels 1 to 3.

Protection level 1	Experts password
Protection level 2	Manufacturer password
Protection level 3	User password

These control the various access rigths.

In the menus listed below the input and modification of data depends on the protection level set:

- Tool offsets
- Work offsets
- Setting data
- RS232 settings
- Program creation / program correction

2.3 The help system

Comprehensive online help is stored in the control system. Some help topics are:

- Product brief of all important operating functions
- Overview and product brief of the NC commands
- Explanation of the drive parameters
- Explanation of the drive alarms

Operating sequence

You can call the help system from any operating area either by pressing the Info key or by using the key combination <ALT+H>.

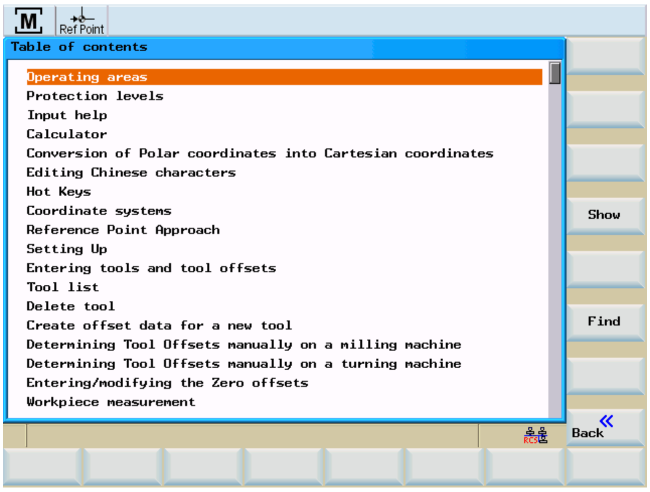


Figure 2-4 Help system: Table of contents

Softkeys

Show

This function opens the selected topic.

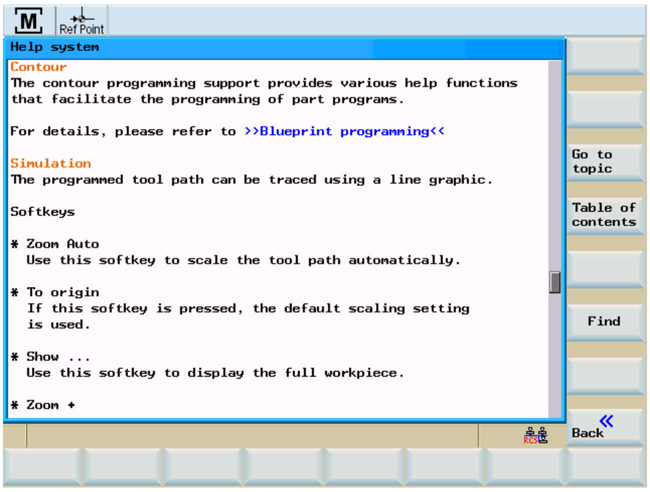


Figure 2-5 Help system: Description of the topic

Go to Topic

Use this function to select cross references. A cross reference is marked by the characters ">>....<<". This softkey is only displayed if a cross reference is displayed in the application area.

Back to Topic

If you select a cross-reference, the "Back to topic" softkey will also be displayed. Select this function to go back to the previous screen.

Find

Use this function to search for a term in the table of contents. Type the term you are looking for and start the search process.

Help in the "Program editor" area

The help system offers an explanation for each NC operation. To display the infotext directly, position the cursor after the appropriate operation and press the Info key. The NC instruction must be written using uppercase letters.

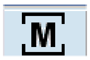
3 Turning on, reference point approach

Note

When you turn on the SINUMERIK 802D sl and the machine, please also observe the machine documentation, since turning on and reference point approach are machine-dependent functions.

Operating sequence

First, switch on the power supply for the CNC and the machine.



After the control system has booted, you are in the "Position" operating area, in the "Reference point approach" mode.

The "Reference point" window is active.

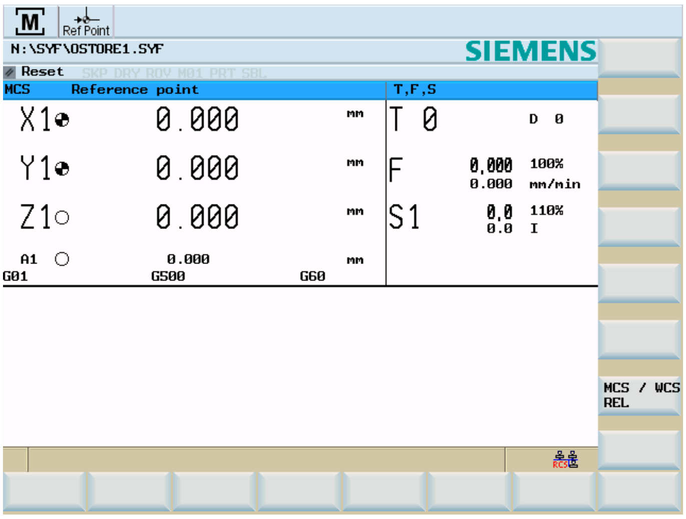


Figure 3-1 Reference-point approach start screen

The "Reference point" window displays whether the axes are referenced.

 Axis must be referenced

 Axis is referenced/synchronized

+X

+Y

Press the arrow keys.

-Z

If you select the wrong approach direction, no motion is carried out.

One after the other, move each axis to the reference point.
You can exit the function by selecting another operating mode (MDA, AUTOMATIC or JOG).

To access the functions described below, you need to select "Jog" mode.

Setup

4

4.1 Entering tools and tool offsets

Functionality

The "OFFSET PARAM" operating area allows you to store the parameters required for machine operation.

Operating sequences

OFFSET PARAM

Tool list

This function opens the "Tool offset data" window which contains a list of the tools created. Use the cursor keys and the <Page Up>/<Page Down> keys to navigate in this list.

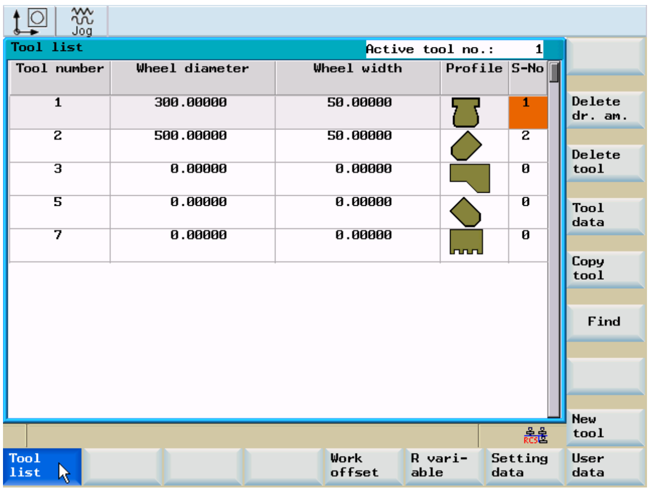


Figure 4-1 Tool list

To input the offsets, position the cursor bar on the tool to be changed and press the <Tool data> softkey.

Softkeys

Delete dr. am.	Clearing the calculated dresser data.
Tool Delete	Use this softkey to delete the tool.
Tool data	Opens a lower-level menu bar offering all functions required to create and display further tolol data.
Nominal dimension monitoring	This function is used to enter - guided by the menu - the nomiinal dimensions and monitoring data of the grinding wheel.
Geometry data	This function is used the enter the wheel geometry for the wheel type selected.
Techno-logy data	This function is used the enter the dressing technology for dressing the wheel type selected.
1st dresser	This function is used to enter/verify the dresser data of the first dresser. For dressers 2 and 3, it is selected through the respective softkeys.
Extended	This function is used to enter/verify all tool data (D1 through D9).
Tool copying	Use this function to copy an already existing tool.
Find	Use this function to search for a tool by its number.
Latest Tool	Use this softkey to create tool compensation data for a new tool.
R para-meters	This function is used to list and, if necessary, modify any R parameters that exist in the control system.
Setting data	Input of the setting data.
User data	This function is used to list and, if necessary, modify any user grinding data that exists in the controller.

4.2 Create new tool

Functionality

The tool offsets consist of various data describing the geometry, the wear and the tool type. Each tool contains a defined number of parameters, depending on the tool type. Tools are identified by a number (T number).

Operating sequences (general)

OFFSET PARAM

Press the <OFFSET PARAM> key.

Tool list

This function opens the "Tool list" window which contains a list of the tools created. Use the cursor keys and the <Page Up>/<Page Down> keys to navigate in this list.

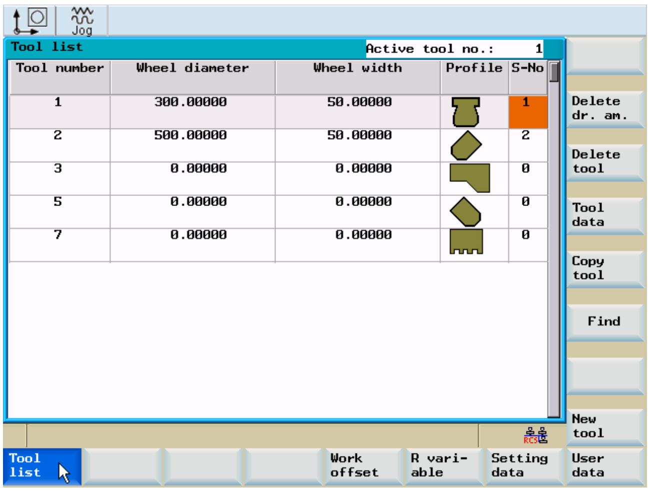


Figure 4-2 Tool list

Tool data

The offsets are entered by placing the cursor bar on the tool to be modified and pressing the <Tool data> softkey.

Operating sequences (new tool)

Latest Tool

This function opens an input screen in which the tool number, tool type, and grinding wheel shape are to be entered or selected.

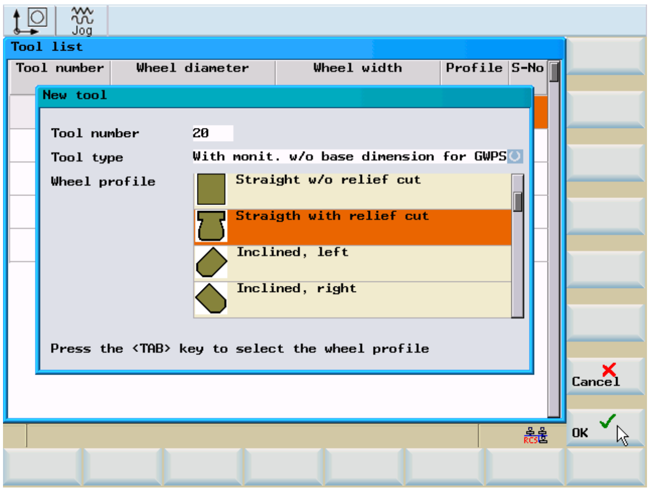


Figure 4-3 New tool

OK

Confirm your input using <OK>.

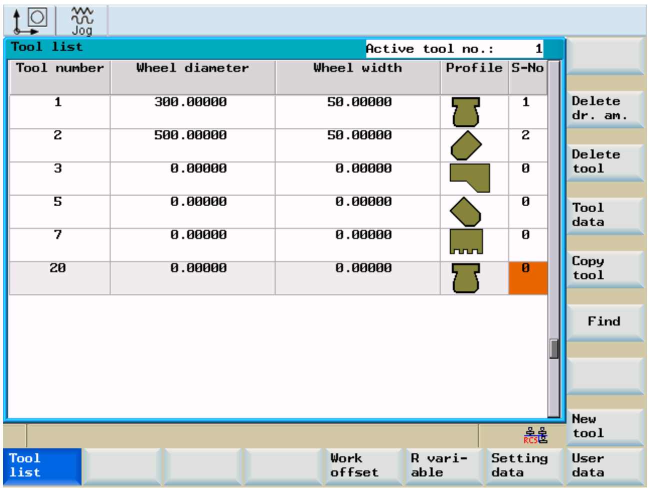


Figure 4-4 New tool inserted

A data record loaded with zero will be included in the tool list. This data block consists of 9 cutting edges (D fields). The first 6 cutting edges have a cutting edge type and are used as cutting edge geometry points.

The tool is assigned to a grinding spindle by an entry in the "S No" field. For values ≤0, an externally controlled grinding spindle is used, for values >0, the grinding spindles of the control system are known.

Note:
Cylindrical grinding begins with S2.
Flat grinding begins with S1.
The conversion is done internally, for an entered value of 1.

For standard wheels (vertical and inclined), the D numbers are assigned a fixed meaning (refer to the "Offset values" figure below). Based on the geometry data, this allocation is always set by default for setting up and dressing.

For wheel having a free contour, the user is always responsible for the cutting edges. Only when a wheel is newly created or for deleted wear values, the cutting edge values are set by default once, depending on the dressing angle. The default setting is made for angle = 0, as in the case of a simple vertical wheel, i.e. the odd cutting edges (D1, D3, D5) are on the left-hand side and the even cutting edges (D2, D4, D6) on the right-hand side, taking into account the entire wheel width.

The default setting for inclined wheels is arranged so that always all reference points are identical. There is no distinction between left-hand and right-hand sides. The user has the option of redefining the cutting edges in a dressing subroutine. For this, the NC syntax must be followed. Any changes will be accepted only after the first complete dressing stroke and not while shaping. Reference points are compensated as it is done for standard wheels.

Diameter and width monitoring will also be active only after both diameter and wear are included in the particular D number. Thus the user can modify additional reference point in the free contour. However, the left-hand and right-hand cutting edges regime must be maintained since the compensations are always taken into account (left-hand side negative, right-hand side positive) as they are for standard wheels.

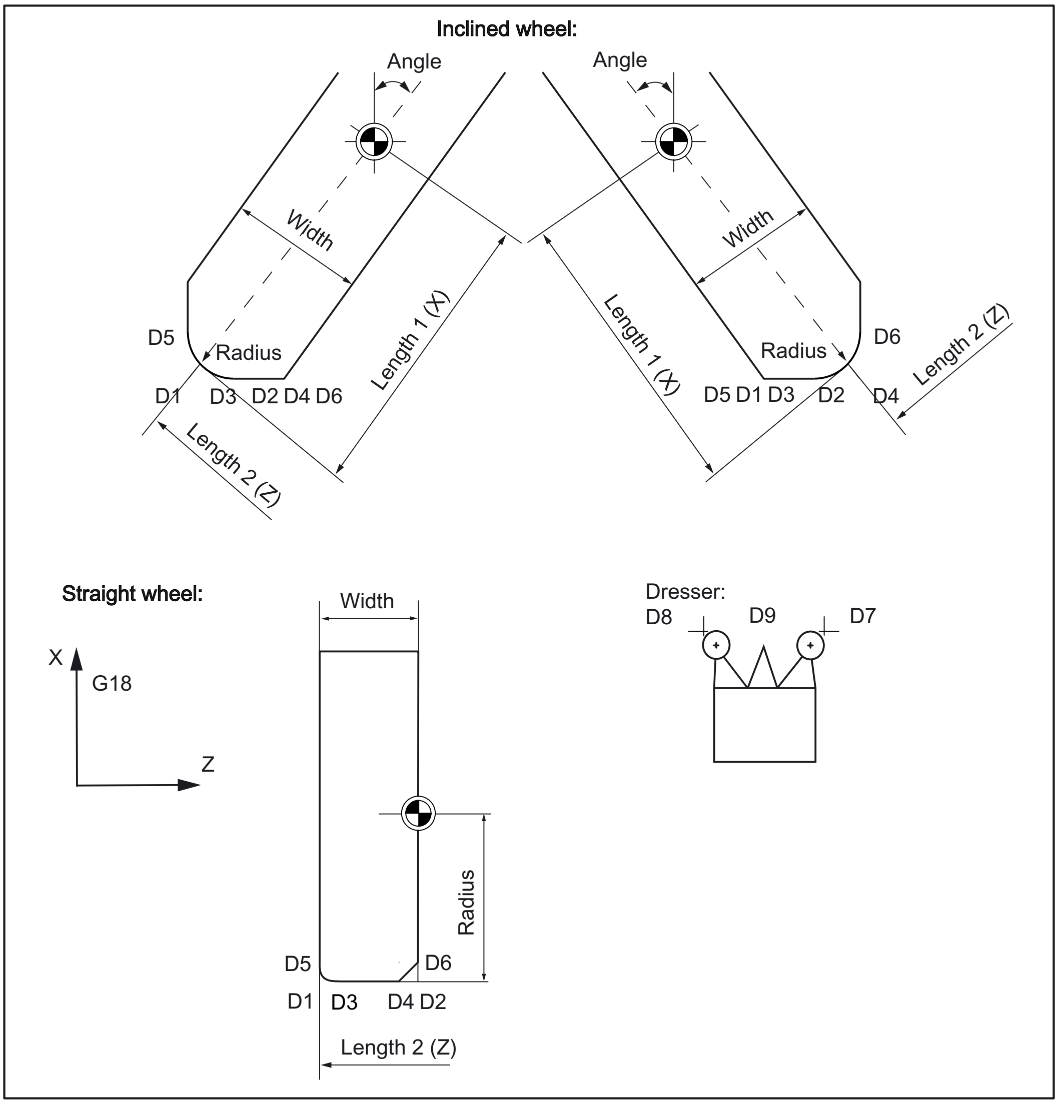


Figure 4-5 Corrective values

Cutting edges 7-9 are the three available dressing tools have a fixed allocation to the standard contour cutting edge.

Table 4- 1 Allocation of dressers

D field	Dresser	Assignment
D7	Dresser 1	Left-hand front cutting edge
D8	Dresser 2	Right-hand rear cutting edge
D9	Dresser 3	Optional for wheel diameter

Tool data

In the next step, the tool data are to be entered.

- Nominal dimensions for monitoring
- Geometry data
- Technological data
- Data for the dressers

Nominal dimensions and monitoring

Nominal dimension monitoring

This function opens in input screen into which grinding wheel nominal dimensions and monitoring data are entered.

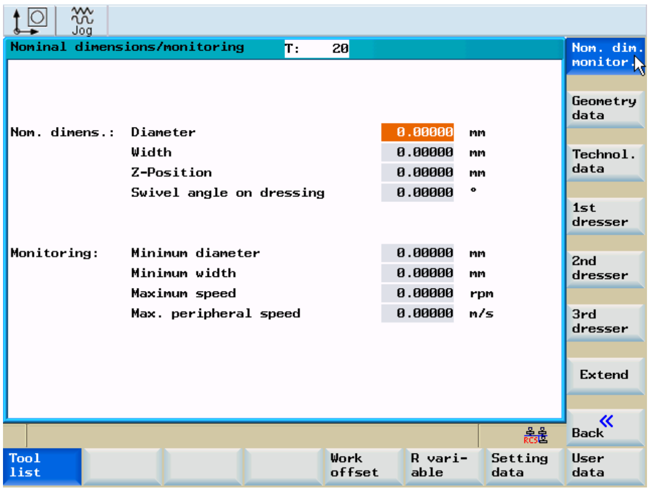


Figure 4-6 Grinding wheel nominal dimensions and monitoring data

Geometry data

Geometry data

This function is used the enter the wheel geometry for the wheel type selected.

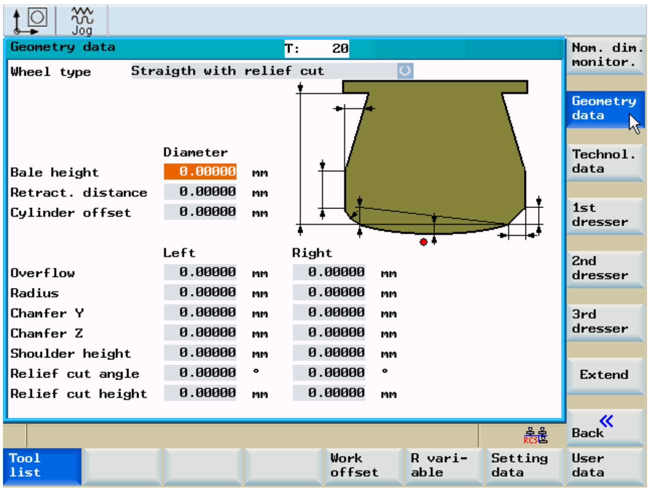


Figure 4-7 Geometry example data for a vertical wheel with back-slope

The following wheel types are available:

- Vertical wheel without back-slopes (type 1)
- Vertical wheel with back-slopes (type 2)
- Left-hand side inclined wheel (type 3)
- Right-hand side inclined wheel (type 4)
- Free contour (type 0)

The input screen is self-explaining.

Note

A red dot shown in the diagrammatic sketch indicates the geometry value just being entered.

Technological data

Techno-
logy data

By means of the technological data, the wheel type dependent dressing technology is defined.

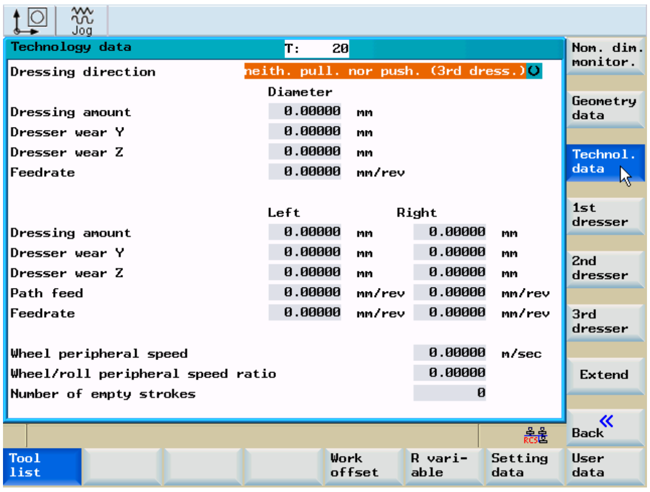


Figure 4-8 Technology example data for a vertical wheel with back-slope

Dresser

1st
dresser

Use the "1st dresser", "2nd dresser" or "3rd dresser" softkeys to access the dialog box for entering or verifying the dresser data.

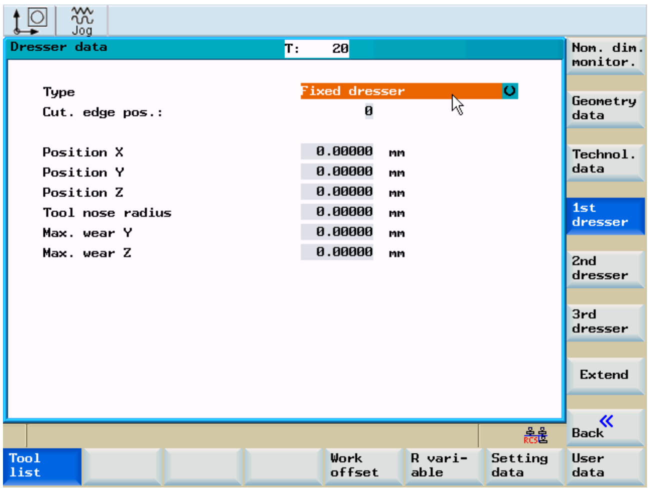


Figure 4-9 Fixed dresser

Use the "Type" toggle field to select the dresser type:

Fixed dresser:	Tile/Diamond
Form roll 1 to 3	
Diamond roll 1 to 3	

Enter the parameters depending on the selection made.

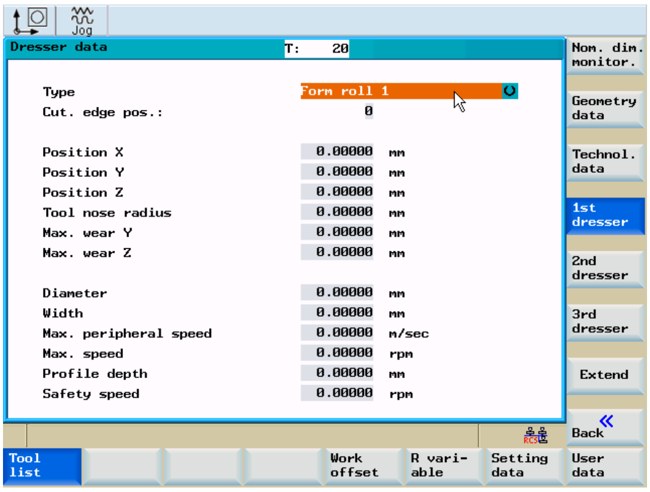


Figure 4-10 Form roll

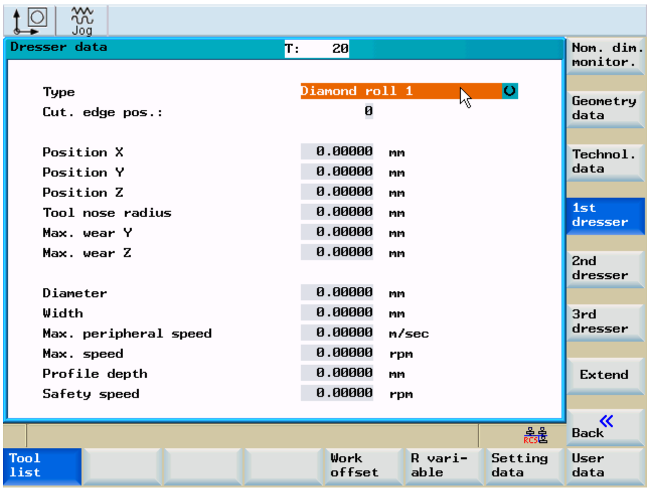


Figure 4-11 Diamond roll

Parameter tables

Extended

The function opens a summary of all cutting edge parameters.

Note: This function is available only with a password set (Customer).

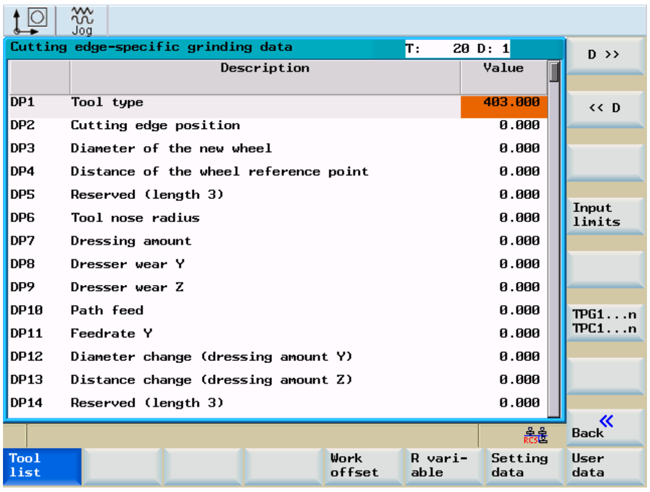


Figure 4-12 The following table contains all cutting edge data.

Tool offset data

See Chapter "Parameter tables of tool offset data" in the annex.

4.3 Register dresser

Functionality

This function is used to determine the dresser positions in the machine for dressers that are used by means of the geometry axes. The axis values are determinded in machine coordinates by the HMI and transmitted to the cycle.

Operation

The dresser is sensed in JOG mode.

Sense dresser

The input screen is opened.

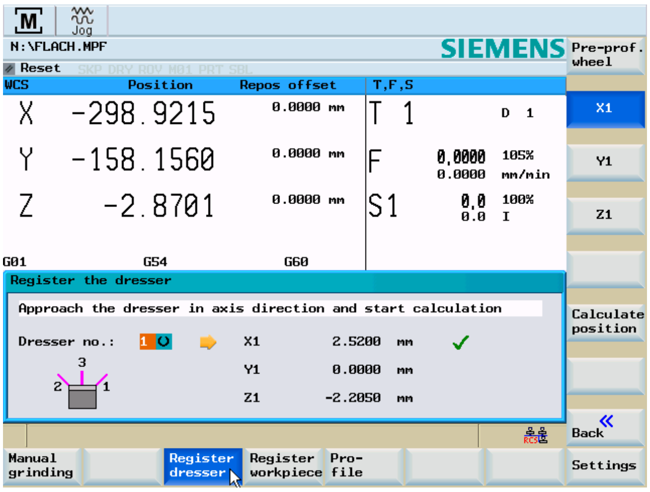
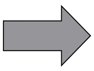


Figure 4-13 Register dresser

Use the "Dresser no.:" toggle field to select the dresser whose position you wish to register (e.g. "1"). The order is irrelevant.

Each axis can be registered independently of the others.

The steps required are shown in a text line.



The axis line to be processed is indicated by an arrow.

Note

For swiveling wheels, the wheel must already be set to its dressing angle.

Calculate position

Following scratching of the selected axis, select "Calculate position" to read the axis actual value and calculate it with the active tool.

The green check mark at the end of the line indicates this action.

For standard grinding wheels, the D number of the wheel is automatically selected in accordance with the choice of dresser for dressers 1 and 2.

For the 3rd dresser, select the D number or approach the edge of the active wheel as it is not automatically recognized.

<< Back

Exit the "Register dresser" function.

4.4 Sense workpiece

Functionality

This function is used to detect the workpiece position in the machine with respect to the particular axis. The HMI transmits both axis name and setpoint to the cycle.

Operation

The workpiece is sensed in JOG mode by scratching the respective axes.

Sense workpiece

The input screen is opened.

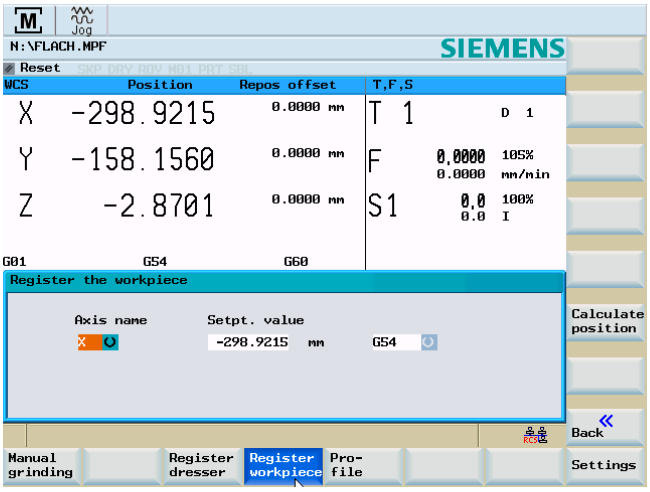


Figure 4-14 Sense workpiece

Use the "Axis name" toggle field to select the desired axis and enter the workpiece setpoint measured in the input field.

Calculate position

Press the <Calculate position> softkey to calculate the setpoint.

Note

This procedure must be done for each axis separately.

<< Back

Exit the "Register workpiece" function.

Special features in connection with manual grinding

If you have interrupted manual grinding (Page 50) with the PLC key "Handwheel" during manual grinding, then the last position of the infeed axis can be calculated following "Measure workpiece" > "Calculate position".

Above the HMI, the following text appears:

"Accept setting value from manual grinding - continue with NC start".

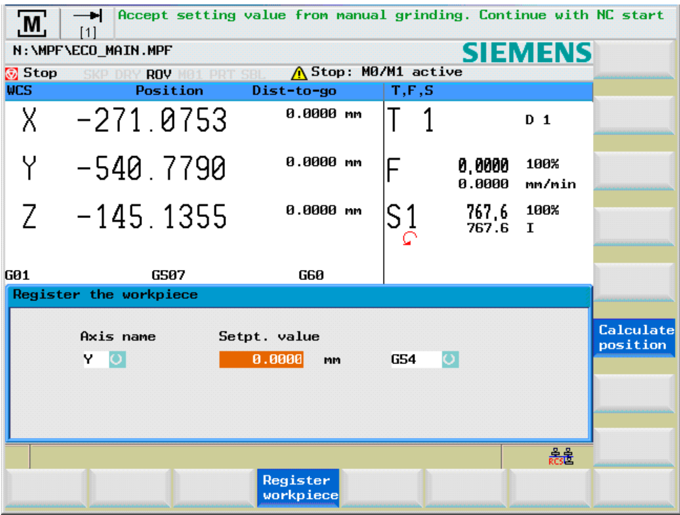


Figure 4-15 Measure workpiece after manual grinding

Measuring is possible only for the infeed axis from manual grinding and only once directly after manual grinding. If "Measure workpiece" is aborted or another axis is set as the last infeed axis, then every axis with any axis positions must be calibrated again.

4.5 Shaping/dressing

Functionality

This function is used to shape a "raw" grinding wheel without generating an NC program. The procedure always refers to the currently active tool.

Operation

Shaping is done in JOG mode.

Shaping

The input screen is opened.

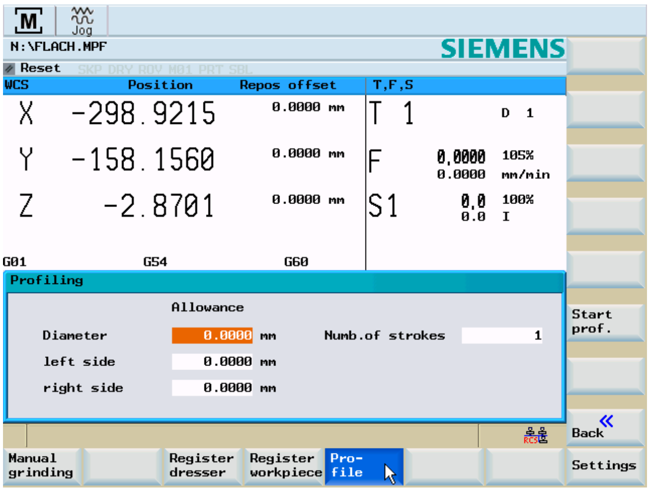


Figure 4-16 Shaping

The required shaper values that are machined in dressing strokes are entered using the input fields

For a new wheel (no wear), the shaper allowance is suggested by the control system. The number of dressing strokes can be freely selected.

Start prof.

When you press the <Start prof.> softkey, the following prompt will appear:

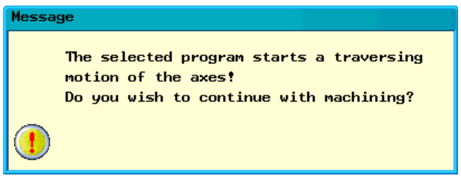
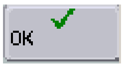


Figure 4-17 Prompt



How profiling is executed

In the cycle, the shaper allowance is machined first and then all dressing strokes are executed. The current state is shown in the fields.

The procedure can be aborted at any time.

Start prof.

To restart it, press the <Start prof.> softkey. Values can be modified.

<< Back

Exit the "Profiling" function.

4.6 Manual grinding

Functionality

This function is for grinding (precision grinding) with the handwheel. This function does not require a workpiece program.

Operation

Manual grinding is done in "Jog" mode.

Manual grind.

The input screen is opened.

Entry of parameters into the input screen for manual grinding (see figure below):

- T or D number
- Select oscillating motion via toggle field.

  The following oscillating motions are possible:
  - No function
  - Infeed Y axis no oscillation
  - Infeed Z axis no oscillation
  - Infeed Y axis oscillation X axis
  - Infeed Z axis oscillation X axis
  - Infeed Y axis oscillation X/Z axis
- Peripheral speed of tool (m/s)
- Workpiece speed (rpm)

Manual grinding, no oscillation

The figure below shows an input screen with parameters for manual grinding without oscillation:

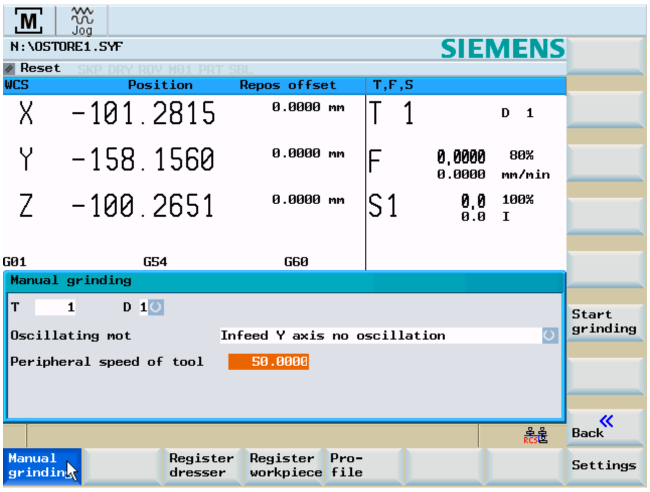


Figure 4-18 Manual grinding without oscillation

Start grind.

This function starts manual grinding with the handwheel. A prompt appears.

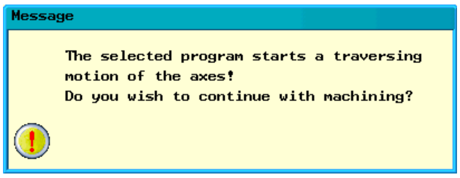


Figure 4-19 Prompt

OK

Execution of manual grinding with handwheel (without oscillation).

<< Back

Exit manual grinding.

Manual grinding, oscillation

The figure below shows an input screen with parameters for manual grinding with oscillation:

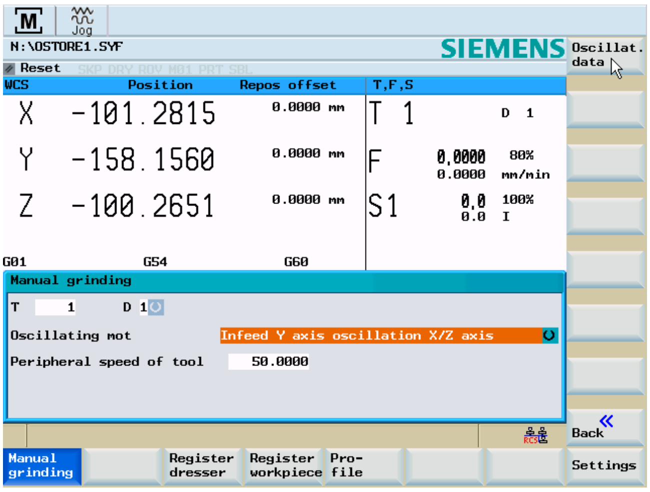


Figure 4-20 Manual grinding with oscillation

Oscillation data

If you have selected oscillation, then you should use this function to enter the oscillation data (see figure below):

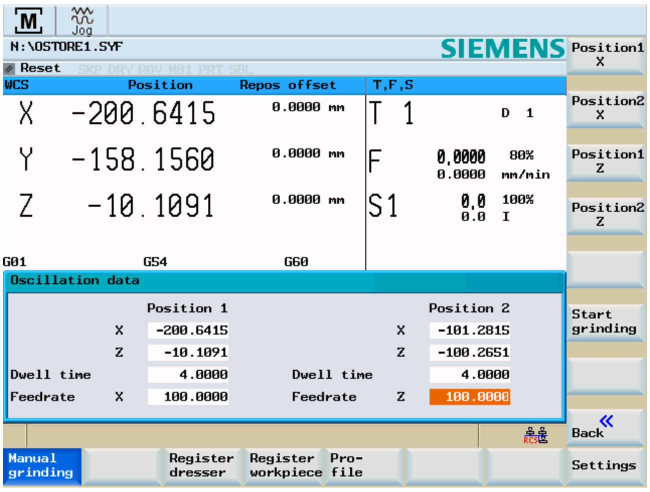


Figure 4-21 Manual grinding with oscillation data in X and Z

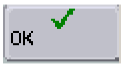
The following oscillation data is possible:

- Position 1 (start)/2 (end):
  - Use the numeric keypad to enter position 1/2 in the relevant input field.
  - Use traversing key <X> or <Z> on the machine control panel to approach position 1/2 and use vertical softkey "Position 1"/"Position 2" to transfer the position to the input field (teach in).
- Dwell time at reversal point position 1 (in seconds if there is a tool spindle present; otherwise, in revolutions)
- Feedrate X (mm/min)

- Feedrate Z (mm/stroke)
- Dwell time at reversal point position 2 (in seconds if there is a tool spindle present; otherwise, in revolutions)

This function starts manual grinding with the handwheel. The following prompt is displayed:

"The selected program starts a traversing motion of the axes! Do you wish to continue with machining?"



Execution of manual grinding with handwheel (oscillation)

Exiting manual grinding

Exit manual grinding.

Special features in connection with "Measure workpiece"

In order to be able to intervene in the grinding process during manual grinding, the PLC keys for "Interruption" and "Dressing" are active during manual grinding.

The PLC key "Handwheel" ends manual grinding on the starting position of the infeed axis. By aborting manual grinding with PLC key "Handwheel", the last position of the infeed axis is saved. This saved position of the infeed axis will be calculated with a following "Measure workpiece (Page 46)".

Measuring is possible only for the infeed axis from manual grinding and only once directly after manual grinding. If "Measure workpiece" is aborted or another axis is set as the last infeed axis, then every axis with any axis positions can be calibrated again.

4.7 Program setting data

Functionality

The setting data are used to define the settings for the operating states. These can be changed as necessary.

Operating sequence

OFFSET PARAM

These can be found in the <OFFSET PARAM> operating area.

Setting data

Press the "Setting data" softkey. The start screen "Setting data" is opened. Other softkey functions are available here with which you can set various control system options.

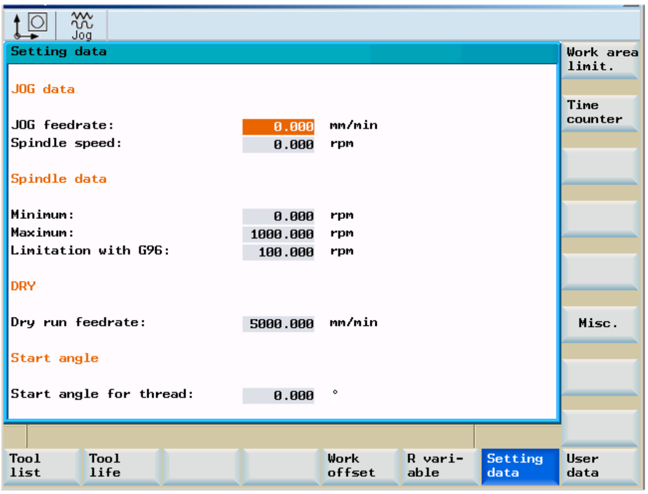


Figure 4-22 Setting data start screen

- **JOG feedrate**

  Feedrate value in JOG mode
  If the feedrate value is zero, the control system will use the value stored in the machine data.
- **Spindle**

  Spindle speed
- **Minimum / maximum**

  A limitation of the spindle speed in the "Max." (G26) / "Min." (G25) fields can only be performed within the limit values defined in the machine data.
- **Limitation using G96**

  Programmable upper speed limitation (LIMS) at constant cutting rate (G96).

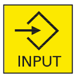
- **Dry run feed (DRY)**

  The feedrate which can be entered here will be used instead of the programmed feedrate in the AUTOMATIC mode if the "Dry run feed" function is selected.

- **Starting angle for thread (SF)**

  For thread cutting, a start position for the spindle is displayed as the start angle. A multiple thread can be cut by changing the angle when the thread cutting operation is repeated.

Place the cursor bar on the input field to be modified and enter the value.



Either press the <Input> key or move the cursor to confirm.

Softkeys

Limit working area

The working area limitation is active with geometry and additional axes. If you want to use a working area limitation, its values can be entered in this dialog box. Selecting the "Set active" softkey enables/disables the values for the axis highlighted by the cursor.

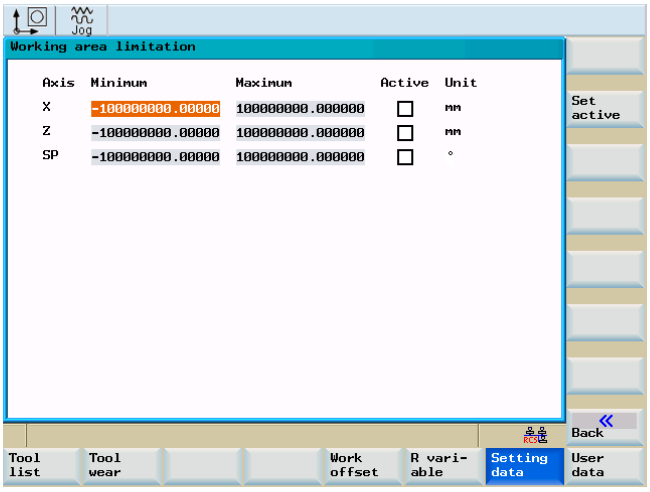


Figure 4-23 Working area limitation

Times Multiplier

Times Counters

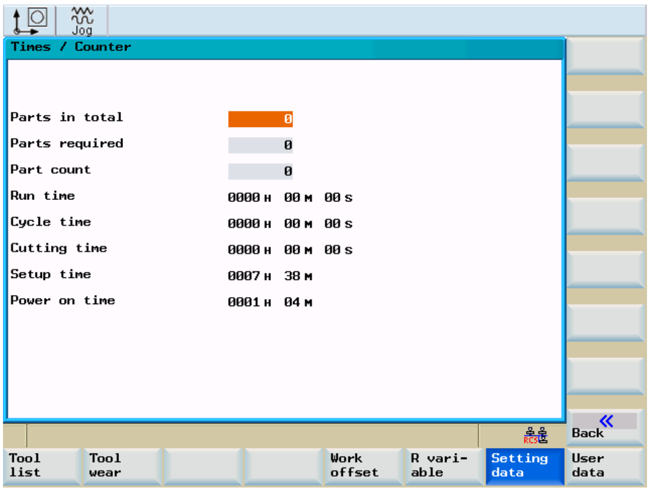


Figure 4-24 Times, Counters

Meaning:

- Total parts: Total number of workpieces produced (total actual)
- Parts requested: Number of workpieces required (workpiece setpoint)
- Number of parts: This counter registers the number of all workpieces produced since the starting time.

Note

The counter functionality is set using the following channel-specific machine data:

- MD27880 $MC_PART_COUNTER, the workpiece counter is activated
- MD27882 $MC_PART_COUNTER_MCODE[0-2], workpiece counting with user defined M command

---

- Total runtime: Total runtime of NC programs in AUTOMATIC mode

  In the AUTOMATIC mode, the runtimes of all programs between NC START and end of program / RESET are summed up. The timer is zeroed with each power-up of the control system.
- Program runtime Active tool operating times

  The runtime between NC Start and End of program / Reset is measured in the selected NC program. The timer is reset with the start of a new NC program.
- Feedrate runtime

  The runtime of the path axes is measured in all NC programs between NC START and end of program / RESET without rapid traverse active and with the tool active. The measurement is interrupted when a dwell time is active.

The timer is automatically reset to zero in the case of a "Control power-up with default values".

Misc.

Use this function to display all setting data for the control system in the form of a list. The setting data are divided up into general, axis-specific and channel-specific data.

They can be selected using the following softkey functions:

- "General"
- "Axis-spec."
- "Channel-spec."

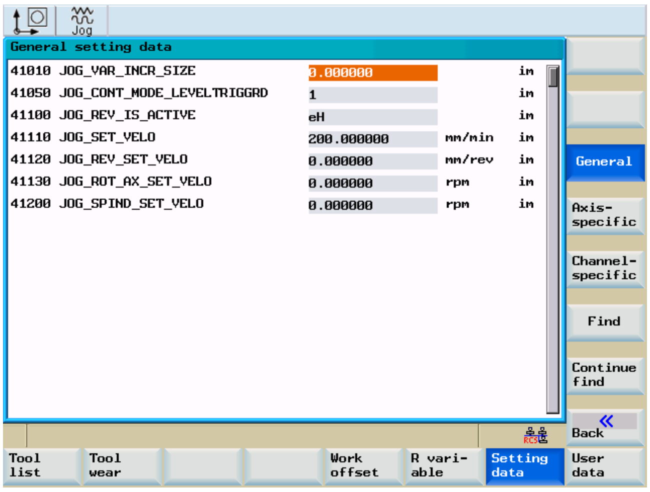


Figure 4-25 General setting data

4.8 Arithmetic parameter R

Functionality

In the "R parameters" start screen, any R parameters that exist within the control system are listed. These global parameters can be set or queried by the programmer of the part program for any purpose in the program and can be changed as required.

Operating sequence

OFFSET PARAM

These can be found in the <OFFSET PARAM> operating area.

R para-meters

Press the <R variable> softkey. The "R variables" start screen appears.

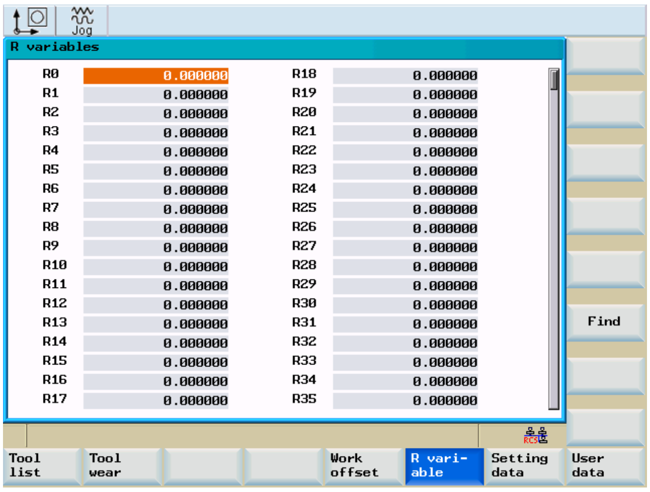
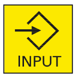


Figure 4-26 "R parameters" start screen

Place the cursor bar on the input field to be modified and enter the values.



Either press the <Input> key or move the cursor to confirm the entry.

Find

Searching for R variables

4.9 User data

Functionality

The user data is internally processed in the cycles. This data can be changed as necessary.

Operating sequences

OFFSET PARAM

These can be found in the <OFFSET PARAM> operating area.

User data

Press the <User data> softkey. This will open the "User data" start screen for the cycles.

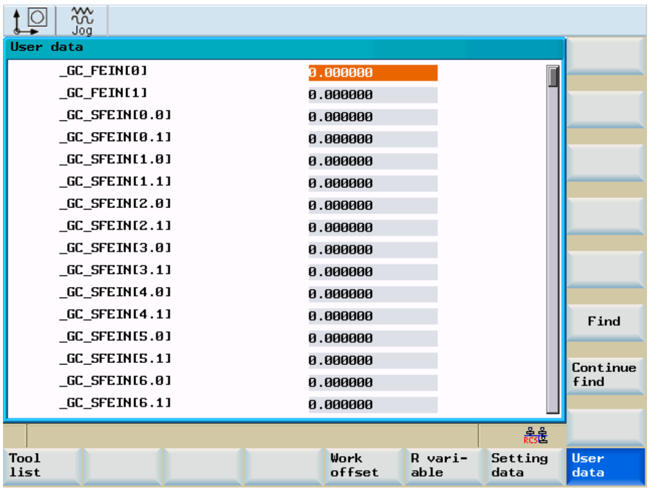


Figure 4-27 User data

Place the cursor bar on the input field to be modified and enter the values.

INPUT

Either press the <Input> key or move the cursor to confirm the entry.

Find

Continue find

Use this function to search for the user data.

See also

User data (Page 335)

5 Manual mode

5.1 Manual mode

Manual mode is supported by the JOG and MDA operating modes.

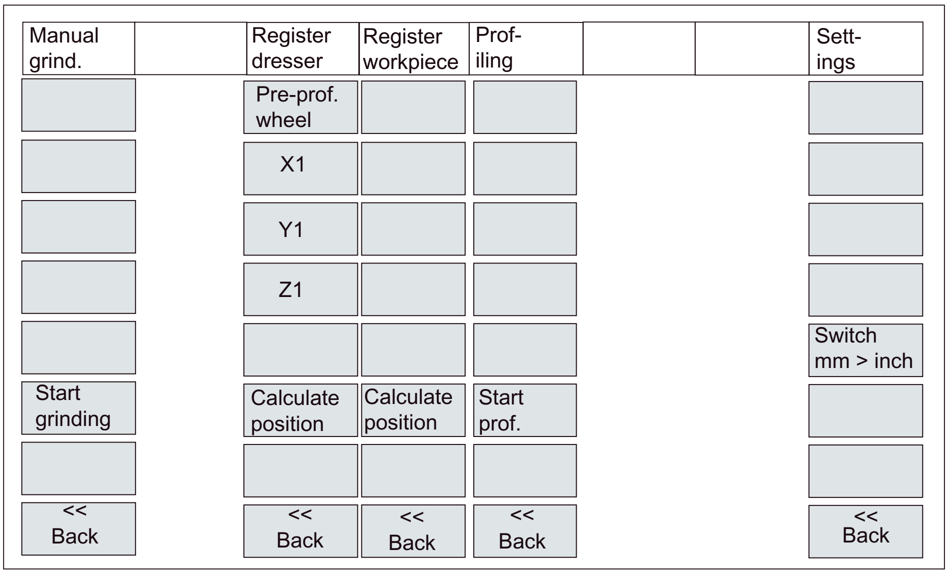


Figure 5-1 JOG menu tree, "Position" operating area

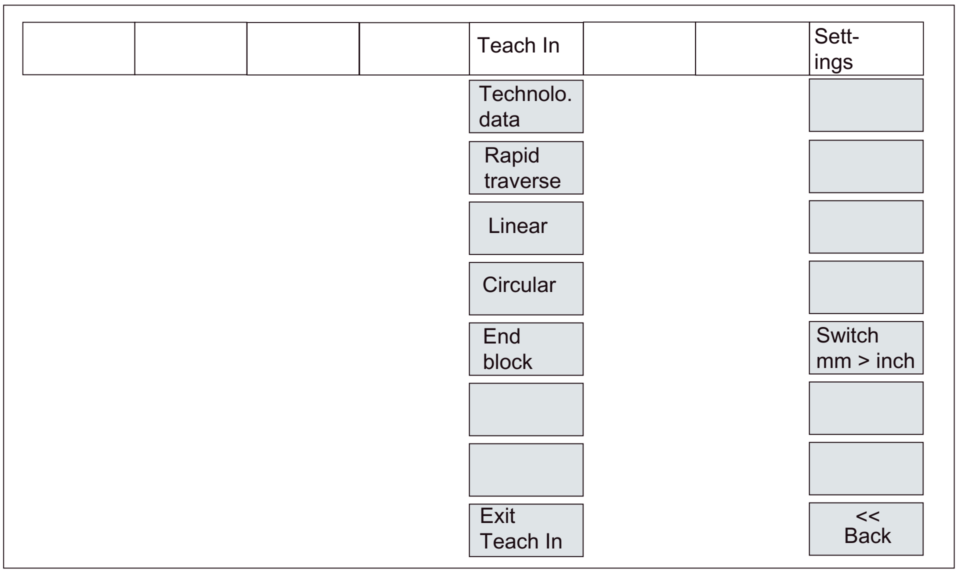


Figure 5-2 MDA menu tree, "Position" operating area

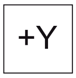
5.2 JOG mode - "Position" operating area

5.2.1 JOG mode

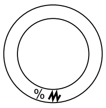
Operating sequences

Use the <JOG> key on the machine control panel to select the Jog mode.

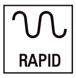
To traverse the axes, press the appropriate key for the X, Y, or Z axis.



The axes will traverse continuously at the velocity stored in the setting data until the key is released. If the value of the setting data is zero, the value stored in the machine data is used.



If necessary ,set the velocity using the override switch.



If you press the <Rapid traverse override> key at the same time, the selected axis will be traversed at rapid traverse speed while both keys are being held down.

In the "Increment" mode, you can traverse by adjustable increments using the same operating sequence. The set number of increments is displayed in the status area. To deselect, press <JOG> again.

The JOG start screen displays the position, feedrate and spindle values, as well as the current tool.

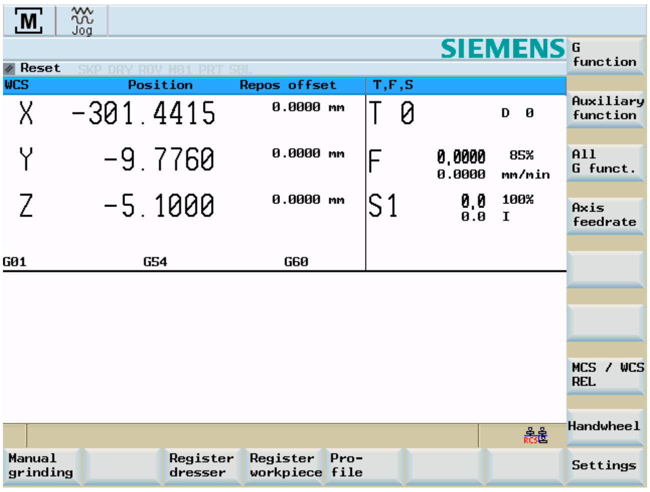


Figure 5-3 JOG mode start screen

Parameter

Table 5- 1 Description of the parameters in the JOG start screen

Parameter	Explanation
MCS X Y Z	Displays the axes existing in the machine coordinate system (MCS) or in the workpiece coordinate system (WCS)
+ X + Y - Z	If you traverse an axis in the positive (+) or negative () direction, a plus or minus sign will appear in the relevant field. If the axis is already in the required position, no sign is displayed.
Position mm	These fields display the current position of the axes in the MCS or WCS.
Repos. offset	If the axes are traversed in the "Program interrupted" condition in the *Jog* mode, the distance traversed by each axis is displayed referred to the interruption point.
G function	Displays important G functions
Spindle S r.p.m.	Displays the actual value and the setpoint of the spindle speed.
Feed F mm/min	Displays the path feedrate actual value and setpoint.
Tool	Displays the currently active tool with the current edge number

Note

If a second spindle is integrated into the system, the workspindle will be displayed using a smaller font. The window will always display the data of only one spindle.

The control system displays the spindle data according to the following aspects:

The master spindle (large display) is displayed:

- Idle,
- at spindle start
- with both spindles active

The workspindle (small display) is displayed:

- when starting the workspindle

The power bar applies to the spindle currently active. With both master spindle and workspindle active, the master spindle performance bar is displayed.

Softkeys

Note

The vertical softkeys are explained in the the section titled MDA mode (Page 66).

Manual grind.	This function is for grinding (precision grinding) with the handwheel. This function does not require a workpiece program.
Sense dresser	This function is used to determine the dresser positions in the machine for dressers that are used by means of the geometry axes.
Sense workpiece	This function is used to detect the workpiece position in the machine with respect to the particular axis.
Shaping	This function is used to shape a "raw" grinding wheel without generating an NC program.
Settings	**Note** The parameters within the "Settings" function do not affect grinding.
Switch mm > inch	Use this softkey to switch between the metric and the inch dimension systems

5.2.2 Assigning handwheels

Operating sequence

Select the "JOG" operating mode.

Handwheel

Press the <Handwheel> softkey. The "Handwheel" window appears on the screen.

After the window has been opened, all axis identifiers are displayed in the "Axis" column, which simultaneously appear in the softkey bar.

Select the desired handwheel using the cursor. Following this, assign or deselect as appropriate by pressing the relevant axis softkey for the desired axis.

The ☑ symbol appears in the window.

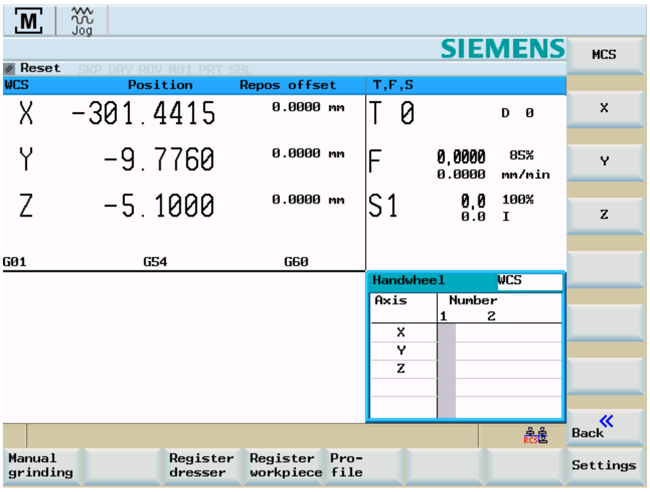


Figure 5-4 Handwheel menu screen

MCS

Use the <MCS> softkey to select the axes from the machine or workpiece coordinate system for hand wheel assignment.

The current setting is displayed in the window.

5.3 MDA mode (manual input) "Position" operating area

Functionality

In the MDA mode, you can create or execute a part program.

⚠ CAUTION
The Manual mode is subject to the same safety interlocks as the fully automatic mode. Furthermore, the same prerequisites are required as in the fully automatic mode.

Operating sequences

Use the machine control panel to select "MDA" mode.

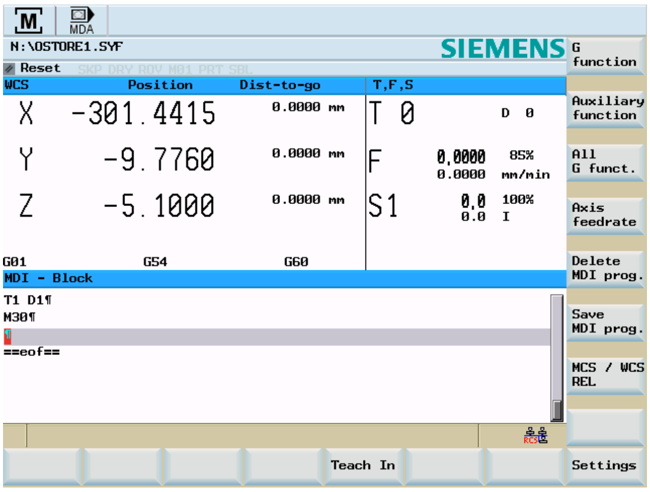


Figure 5-5 MDA mode start screen

Enter one or several blocks using the keyboard.

By pressing <NC START> machining is started. During machining, editing of the blocks is no longer possible.

After machining, the contents are preserved so that the machining can be repeated by pressing <NC START> once more.

Parameter

Table 5- 2 Description of the parameters in the MDA working window

Parameter	Explanation
MCS X Z	Displays the existing axes in the MCS or WCS
+X -Z	If you traverse an axis in the positive (+) or negative () direction, a plus or minus sign will appear in the relevant field. If the axis is already in the required position, no sign is displayed.
Position mm	These fields display the current position of the axes in the MCS or WCS.
Distance-to-go	This field displays the distance to go of the axes in the MCS or WCS.
G function	Displays important G functions
Spindle S r.p.m.	Displays the actual value and the setpoint of the spindle speed.
Feedrate F	Displays the path feedrate actual value and setpoint in mm/min or mm/rev.
Tool	Displays the currently active tool with the current edge number (T..., D...).
Editing window	In the "Stop" or "Reset" program state, an editing window serves to input a part program block.

Note

If a second spindle is integrated into the system, the workspindle will be displayed using a smaller font. The window will always display the data of only one spindle.

The control system displays the spindle data according to the following aspects:

The master spindle is displayed:

- Idle,
- at spindle start
- with both spindles active

The workspindle is displayed:

- when starting the workspindle

The power bar applies to the spindle currently active.

Softkeys

The horizontal softkeys are explained in the section titled "JOG mode - 'Position' operating area" (Page 62).

G-function

The G function window displays G functions whereby each G function is assigned to a group and has a fixed position in the window.

Use the <PageUp> or <PageDown> keys to display additional G functions. Selecting the softkey repeatedly will close the window.

Auxiliary function

This window displays the auxiliary and M functions currently active. Selecting the softkey repeatedly will close the window.

All G-functions

All the G functions are displayed.

Axis feedrate

Use this softkey to display the "Axis feedrate" window.
Pressing the softkey repeatedly will close the window.

Delete MDA progr.

Use this function to delete blocks from the program window.

Save MDA prog.

Enter a name in the input field for saving the MDA program in the program directory.
Alternatively, you may select an existing program from the list.
Use the Tab key to toggle between the input field and program list.

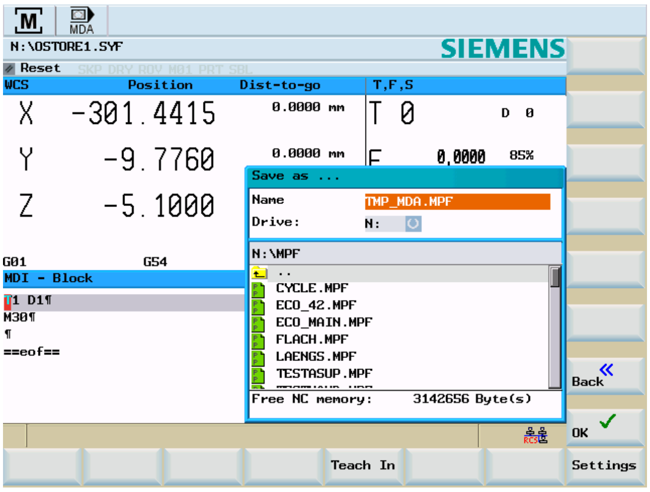


Figure 5-6 Saving an MDA program

MKS/WKS REL

The actual values for the MDA mode are displayed depending on the selected coordinate system. Use this softkey to switch between the two coordinate systems.

See also

JOG mode (Page 62)

5.3.1 Teach In

Functionality

You can use the "Teach In" function to create and change simple traversing blocks. You can transfer axis position values directly into a newly generated or changed part program record.

The axis positions are reached by traversing with the axis direction keys and transferred into the part program.

Operating sequence

In the <POSITION> operating area, use the machine control panel to select <MDA> mode.

Teach In

Press the <Teach In> softkey.

In the "Teach In" submode, assume the following start screen:

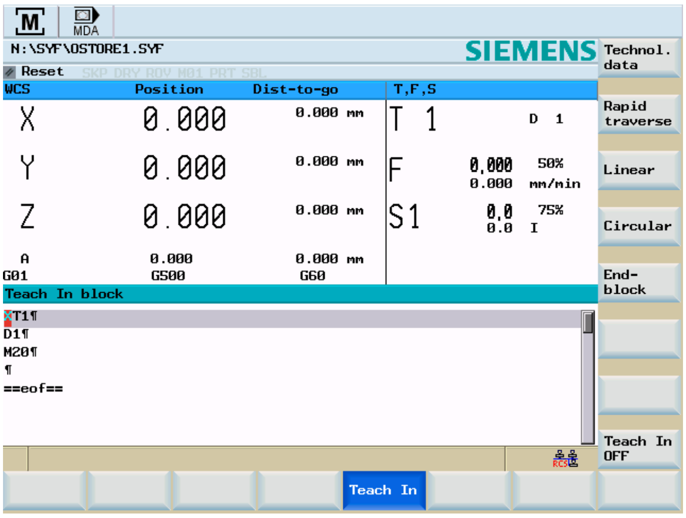


Figure 5-7 Teach In start screen

General sequence

1. Use the arrow keys to select the program block that you want to edit or that is to have the new traversing block inserted in front of it.
2. Select the appropriate softkey.

Technol. data

– "Technological data"

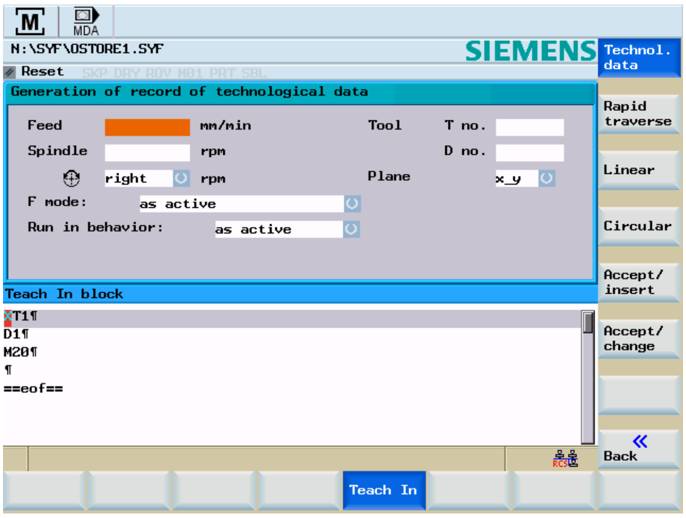


Figure 5-8 Technological data

Enter the appropriate technological data (e.g. feedrate: 1000).

Insert transfer

Click "Insert transfer" to add a new part program block. The new part program block will be added in front of the block selected with the cursor.

Change transfer

Click "Change transfer" to change the selected part program block.

<< Back

Use "<<Back" to return to the "Teach In" start screen.

Rapid traverse

– "Rapid feed"

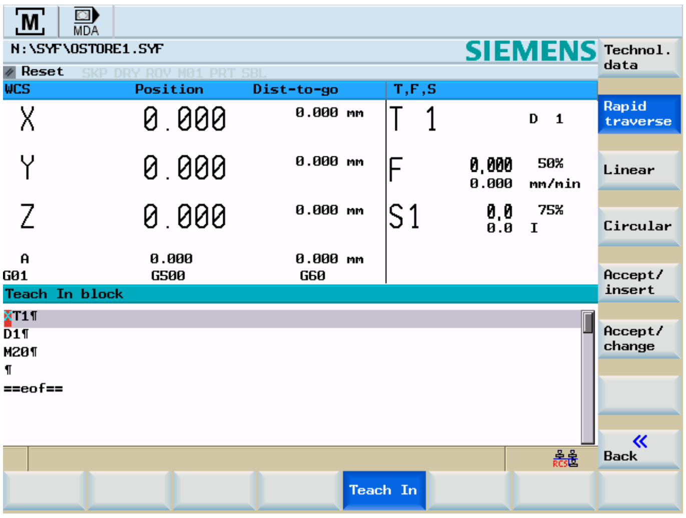


Figure 5-9 Rapid traverse

You traverse the axes and teach-in a rapid traverse block with the approached positions.

Linear

– "Linear"

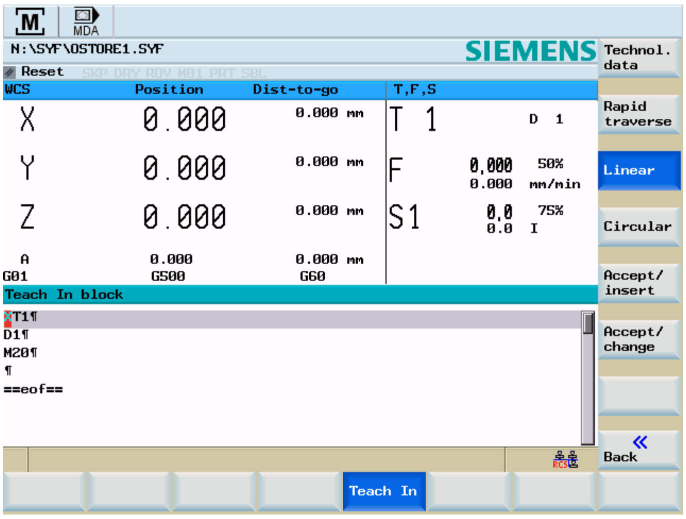


Figure 5-10 Linear

You traverse the axes and teach in a linear block with the approached positions.

Circular

– "Circular"

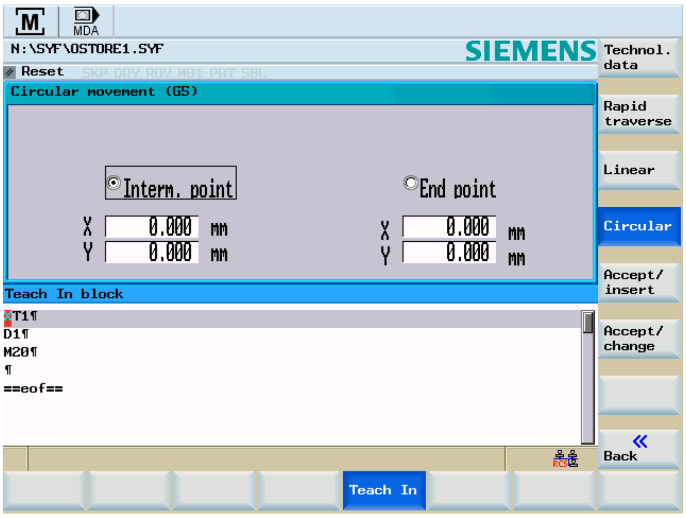


Figure 5-11 Circular

You teach in an intermediate point and an end point for a circle.

Operation in the "Rapid traverse", "Linear", and "Circular" dialogs

+X

+Y

1. Use the axis keys to traverse the axes to the required position that you want to add/change in the part program.

Insert transfer

2. Click "Insert transfer" to add a new part program block. The new part program block will be added in front of the block selected with the cursor.

Change transfer

3. Click "Change transfer" to change the selected part program block.

<< Back

Use "<<Back" to return to the "Teach In" start screen.

Exit Teach In

Use "Exit Teach In" (see "Start screen") to leave the "Teach In" submode.

Automatic mode 6

6.1 Automatic mode

Menu tree

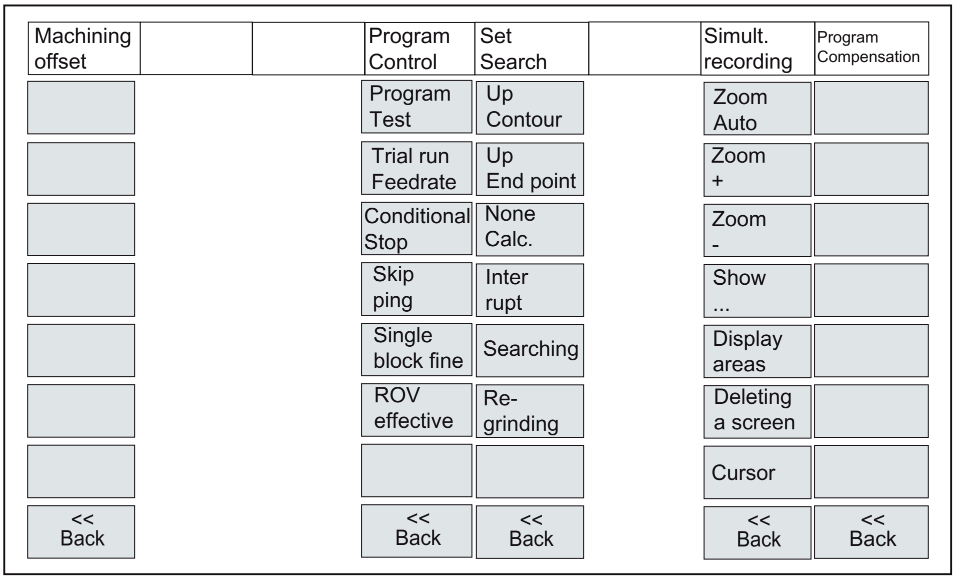
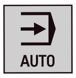


Figure 6-1 Automatic menu tree

Preconditions

The machine is set up for the AUTOMATIC mode according to the specifications of the machine manufacturer.

Operating sequence



Select Automatic mode by pressing the <Automatic> key on the machine control panel.

The Automatic start screen appears, displaying the position, feedrate, spindle, and tool values, as well as the currently active block.

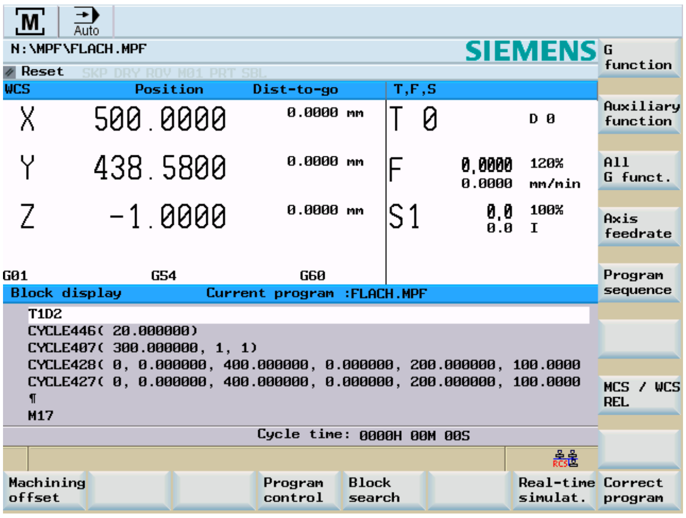


Figure 6-2 Automatic start screen

Parameter

Table 6- 1 Description of the parameters in the working window

Parameter	Explanation
MCS X Z	Displays the existing axes in the MCS or WCS
+ X - Z	If you traverse an axis in the positive (+) or negative () direction, a plus or minus sign will appear in the relevant field. If the axis is already in the required position, no sign is displayed.
Position mm	These fields display the current position of the axes in the MCS or WCS.
Distance-to-go	These fields display the current position of the axes in the MCS or WCS.
G function	Displays important G functions
Spindle S r.p.m.	Displays the actual value and the setpoint of the spindle speed.
Feed F mm/min or mm/rev	Displays the path feedrate actual value and setpoint.

Parameter	Explanation
Tool	Displays the currently active tool with the current edge number (T..., D...).
Current block	The block display displays seven subsequent blocks of the currently active part program. The display of one block is limited to the width of the window. If several blocks are to be executed in quick succession, you are recommended to switch to the "Program progress" window. To switch back to the seven-block display, use the <Program sequence> softkey.

Note

If a second spindle is integrated into the system, the workspindle will be displayed using a smaller font. The window will always display the data of only one spindle.

The control system displays the spindle data according to the following aspects:

The master spindle is displayed:

- Idle,
- at spindle start
- with both spindles active

The workspindle is displayed:

- when starting the workspindle

The power bar applies to the spindle currently active. With both master spindle and workspindle active, the master spindle performance bar is displayed.

Softkeys

G-
function

Opens the G functions window to display all G functions currently active.

The G functions window displays all the G functions that are currently active, with each G function assigned to a group and having a fixed position in the window.

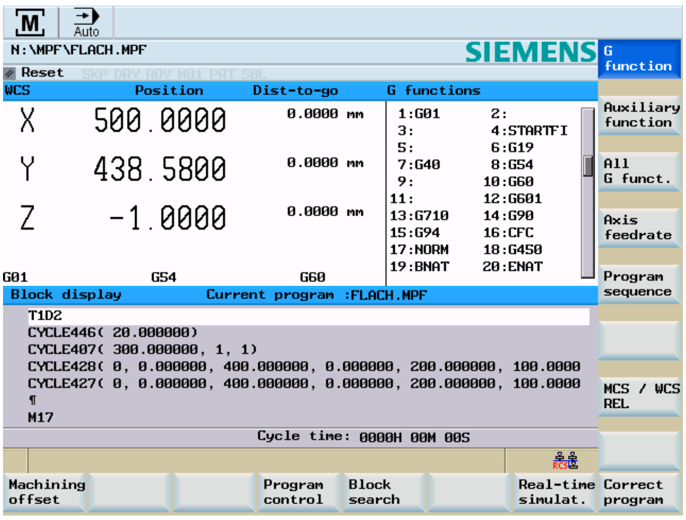


Figure 6-3 G Functions

Use the <PageUp> or <PageDown> keys to display additional G functions.

Auxiliary function

This window displays the auxiliary and M functions currently active.
Selecting the softkey repeatedly will close the window.

All G-functions

All the G functions are displayed.

Axis feedrate

Use this softkey to display the "Axis feedrate" window.
Pressing the softkey repeatedly will close the window.

Program sequence

Use this softkey to switch from the seven-block to the three-block display.

MKS/WKS REL

Switches the axis value display between the machine, workpiece and relative coordinate systems.

Machining offset

Use this softkey to display the "Machining offset" window.

Fine offsets can be entered in Y and Z, globally for each seat or individually for a specific seat.

From then on, these offsets will always be used for the grinding work (seat).

<< Back

Use this softkey to close the screen. Your offsets are saved.

Program control

The program control softkeys are displayed (e.g. "Skip block", "Program test").

- "Program test":

  If "Program test" is selected, the output of setpoints to axes and spindles is disabled. The set point display "simulates" the traverse movements.
- "Dry run feedrate":

  If you select this softkey, all traversing motions will be performed with the feedrate setpoint specified via the "Dry run feed" setting data. The dry run feedrate function replaces the programmed travel commands.
- "Conditional stop":

  When this function is active, processing of the program is stopped at every block in which miscellaneous function M01 is programmed.
- "Skipping":

  Program blocks, the block number of which is preceded by a slash, are skipped during program execution (e.g. "/N100").

- "Single block, fine":

  If this function is active, the part program blocks are executed as follows: Each block is decoded separately, and a stop is performed at each block; an exception are only the thread blocks without dry run feedrate. In such blocks, a stop is only performed at the end of the current thread block. "Single block, fine" can only be selected in the RESET status.
- "ROV effective":

  The feedrate override switch will also act on the rapid traverse override.

<< Back

Use this softkey to close the screen.

Block search

Use the block search function to go to the desired program location.

To contour

Forward block search with calculation
During the block search, the same calculations are carried out as during normal program operation, but the axes do not move.

To end point

Forward block search with calculation to the block end point
During the block search, the same calculations are carried out as during normal program operation, but the axes do not move.

Without calculat.

Block search without calculation
During the block search, no calculation is carried out.

Interr. point

The cursor is placed on the main program block of the interrupt point.

Find

The "Find" softkey provides the functions "Find line", "Find text" etc.

Re-grind.

Use this softkey to display the "Regrinding" window.

Enter the offset values for regrinding. When you select <OK>, the parameters will be inserted in the program after the selected block.

Simultaneous recording

It is possible to simultaneously record when the part program is being executed (see Chapter "Simultaneous recording (Page 84)").

Correct program

Use this softkey to correct a fault program passage. Any changes will be stored immediately.

6.2 Machining offset

Functionality

Fine offsets can be entered in Y and Z, globally for each seat or individually for a specific seat.

From then on, these offsets will always be used for the grinding work (seat).

Operating sequence

Machining offset

The "Automatic" start screen will display a window for the machining offsets.

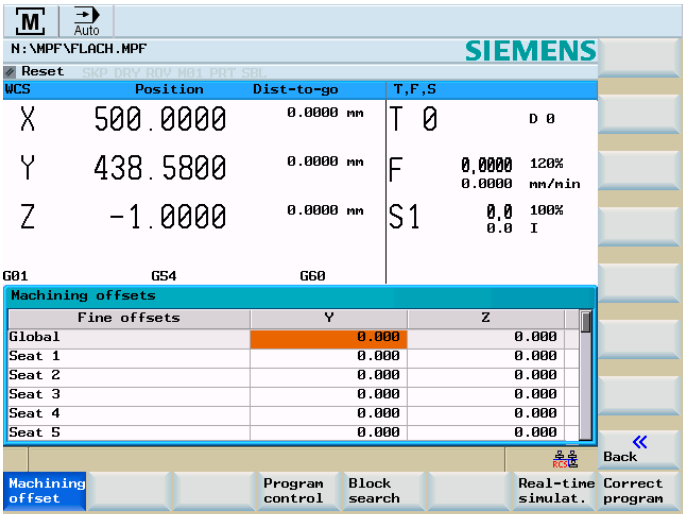


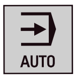
Figure 6-4 Machining offsets

6.3 Selection and start of a part program

Functionality

Before starting the program, make sure that both the control system and the machine are set up. Observe the relevant safety notes of the machine manufacturer.

Operating sequence



Select Automatic mode by pressing the <Automatic> key on the machine control panel.

The Program Manager is opened. Use the <NC directory> (default selection) or <Customer CF card> softkeys to enter the appropriate directories.

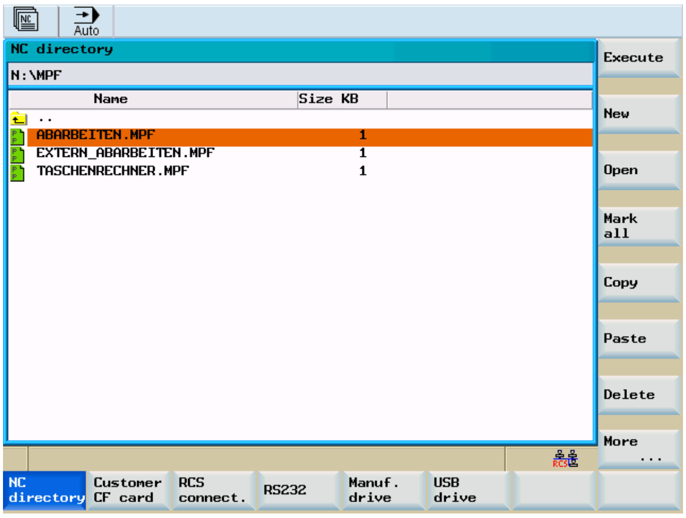


Figure 6-5 "Program Manager" start screen

Place the cursor bar on the desired program.

Exe-
cute

Use the <Execute> softkey to select the program to be executed (see also "External execution"). The name of the selected program will appear in the "Program name" screen line.

Program control

If desired, here you can specify how you want the program to be executed.

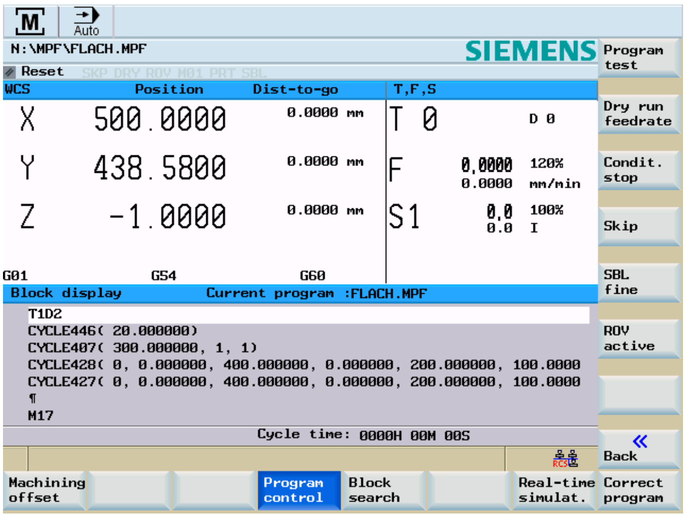


Figure 6-6 Program control

Press <NC START> to start executing the part program.

6.4 Block search

Operating sequence

Prerequisite: The desired program has already been selected, and the control system is in the RESET state.

Block search	The block search function provides advance of the program to the required block in the part program. The search target is set by positioning the cursor bar directly on the required part program block in the part program.

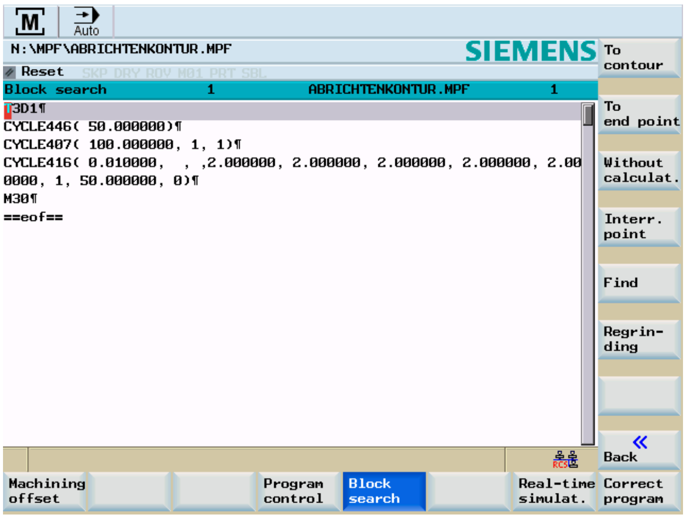


Figure 6-7 Block search

To contour	Block search to block start
To end point	Block search to block end
Without calculat.	Block search without calculation
Interr. point	The interruption location is loaded.

Find

Use this softkey to perform the block search by entering a term you are looking for.

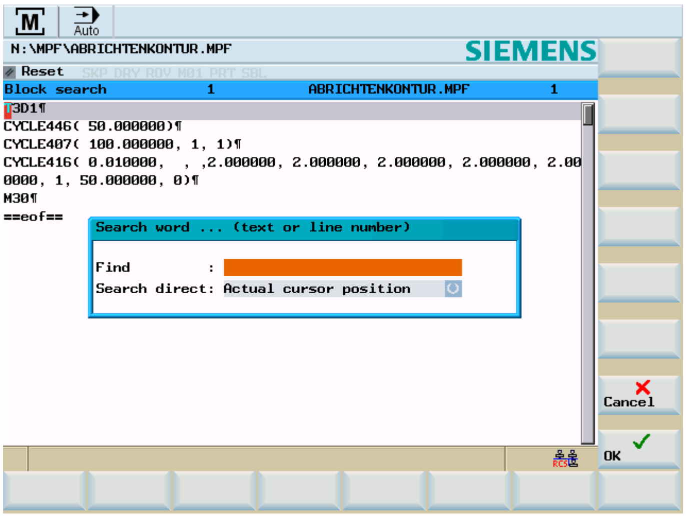


Figure 6-8 Entering the searched term

You can use the toggle field to define the starting position for the search.

Search result

The required part program block is displayed in the "Current block" window.

Note

For "Execute externally", **no** block search is possible.

Regrinding

"Regrinding" enables you to remachine the "seat" of a workpiece that has already been machined, either with or without an offset, but always with the same technological values.

Re-grind.

Use this softkey to display the "Regrinding" window.

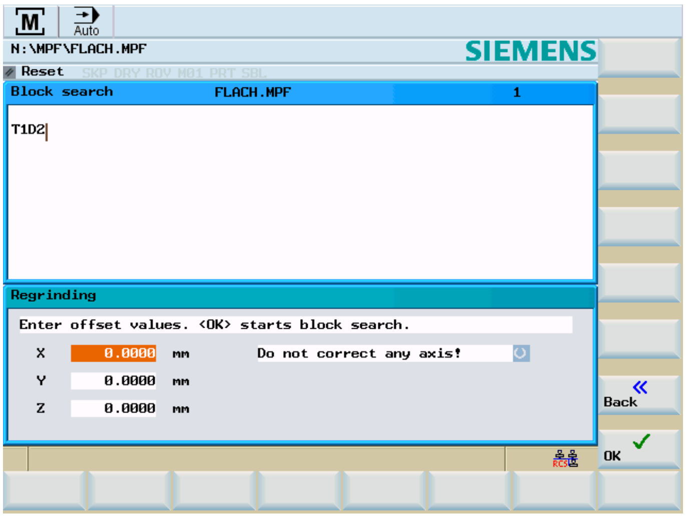


Figure 6-9 Regrinding

Enter the offset values for regrinding.

Choose between the following options in the toggle field:

- Do not correct any axis
- Correct tool
- Machining offsets

When you select <OK>, the parameters will be inserted in the program after the selected block.

The block search starts.

6.5 Simultaneous recording

Operating sequence

You have selected a part program to be executed and have pressed <NC START>.

Simultaneous recording

Execution of the part program is simultaneously recorded on the HMI using the "Simultaneous recording" function.

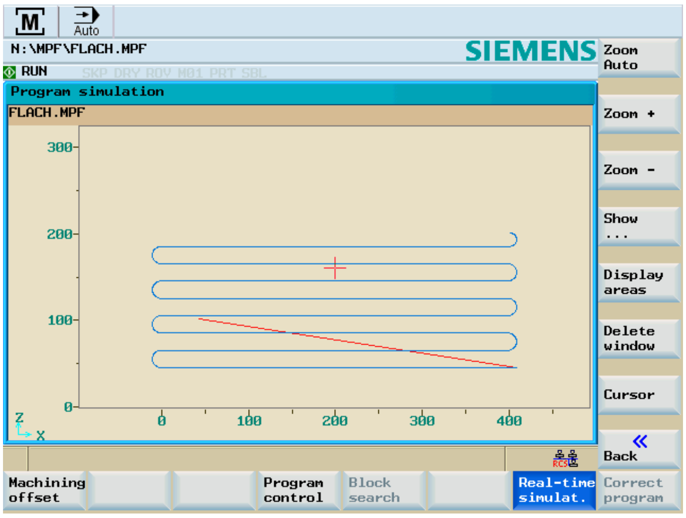


Figure 6-10 "Simultaneous recording" start screen

You can influence how the simultaneous recording function is displayed on the HMI using the following vertical softkeys:

- "Zoom Auto"
- "Zoom +"
- "Zoom -"
- "Show ..."
  - "All G17 blocks"
  - "All G18 blocks"
  - "All G19 blocks"
- "Display areas"

  See the following page for a description.

- "Delete window"
- "Cursor"
  - "Set cursor"
  - "Cursor fine", "Cursor coarse", "Cursor very coarse"

    When the cursor keys are pressed, the cross hair moves in small, average or large steps.

<< Back

Exit the "Simultaneous recording" function.

"Display areas"

Display area

Window, min/max

Using the "Display areas" function, you have the possibility of saving a previously selected area from the simulation display.

The menu for the display area can be selected using the "Window min/max" function.

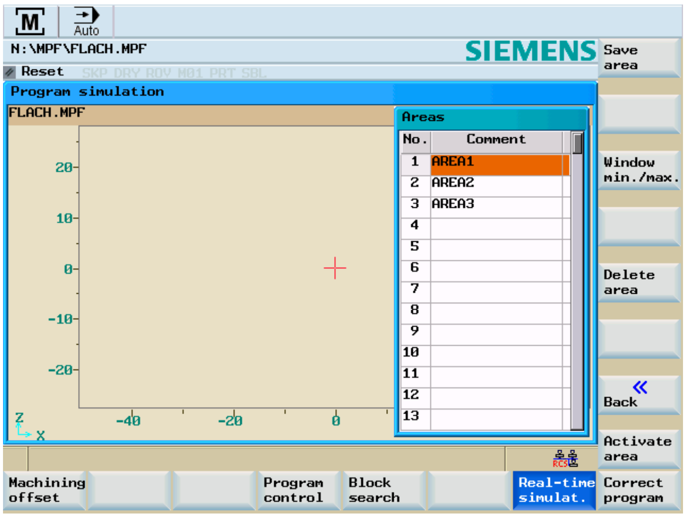


Figure 6-11 Display area "Window min"

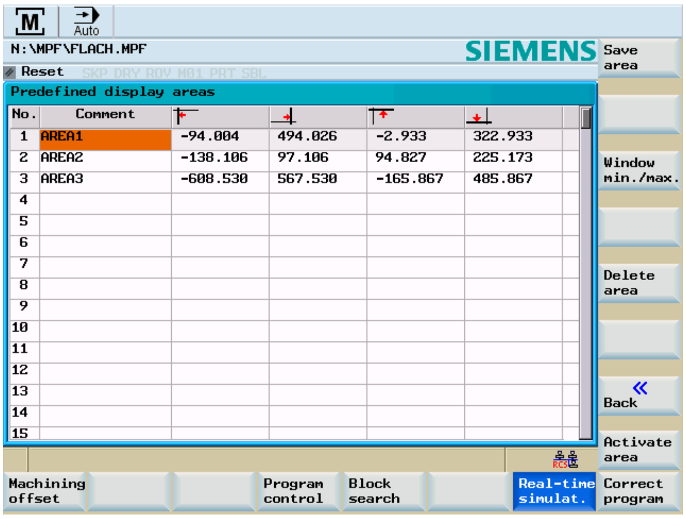


Figure 6-12 Display area "Window max"

Operating sequence to set and save the display area

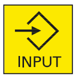
1. You have selected an area in the simulation view.

Display area

2. Press the "Display areas" function.

Window, min/max

3. Press the "Window min/max" so that a maximum display can be see according to the screen "Display areas "Window max".
4. In the "Comment field", you can assign a name to the area.



5. Complete the entry with <Input>.

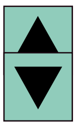
Save area

6. Press "Save area".

Activating or deleting an area

Display area

You have selected a display area.



Using the cursor keys, select the area that you wish to either activate or delete.

Activate area

Press "Activate area" or "Delete area".

Delete area

6.6 Stop / cancel a part program

Operating sequence

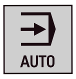
With <NC STOP> the execution of a part program is interrupted.
The interrupted machining can be continued with <NC START>.

Use <RESET> to abort the program currently running.
By pressing <NC START> once again, the aborted program is restarted and executed from the beginning.

6.7 Reapproach after cancellation

After a program cancellation (RESET), you can retract the tool from the contour in manual mode (JOG).

Operating sequence



Select mode <AUTOMATIC> mode.

Block search

Opening the "Block search" window for loading the interruption point.

Interr. point

The interruption point is loaded.

To contour

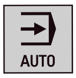
The block search to the interruption point will start. An adjustment to the start position of the interrupted block will be carried out.

Press <NC START> to continue machining.

6.8 Repositioning after interruption

After interrupting the program (<NC STOP>), you can retract the tool from the contour in manual mode (JOG). The control saves the coordinates of the point of interruption. The distances traversed are displayed.

Operating sequence



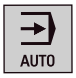
Select <AUTOMATIC> mode.

Press <NC START> to continue machining.

> **CAUTION**
>
> When reapproaching the interruption point, **all axes will traverse at the same time.** Make sure that the traversing area is not obstructed.

6.9 Execute from external

Functionality



In <AUTOMATIC> mode > <PROGRAM MANAGER> operating area, the following interfaces are available for external execution of programs:

Customer CF card	Customer CompactFlash card
RCS connect.	RCS connection for external execution via network (only for SINUMERIK 802D sl pro)
Manufac-turer drive	Manufacturer's drive
USB drive	USB FlashDrive

Start in the following start screen of the Program Manager:

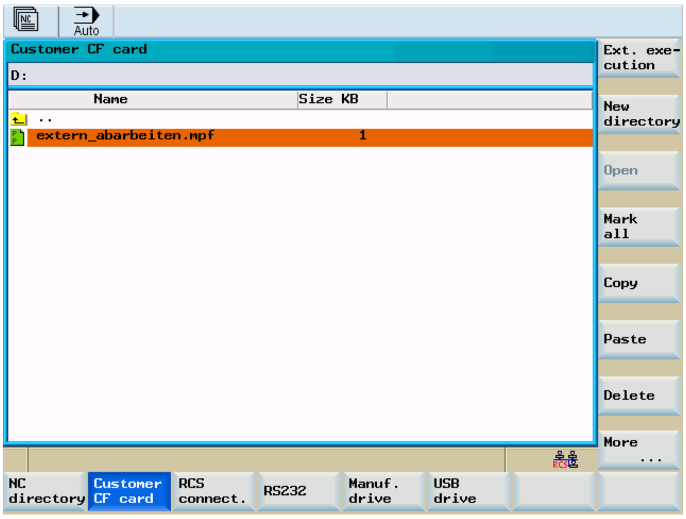


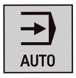
Figure 6-13 The "Program Manager" start screen

Use vertical softkey "Ext. execution" to transmit the selected external program to the control system; to execute this program, press <NC START>.

While the contents of the buffer memory are being processed, the blocks are reloaded automatically.

Operating sequence, execution from customer CompactFlash Card or USB FlashDrive

Requirement: The control system is in the "Reset" state.



Select the <AUTOMATIC> mode key .

Press the <PROGRAM MANAGER> key on the machine control panel.

Customer CF card

USB drive

Press the "Customer CF card" or "USB drive".
You can thus access the directories of the "Customer CF Card / USB FlashDrive".

Place the cursor bar on the desired program.

Ext. execution

Press "Ext. execution".

The program is transferred into the buffer memory and selected and displayed in the program selection automatically.

Press the <NC START> key.

Machining starts. The program is reloaded continuously.

At the end of the program or in case of <RESET>, the program is automatically removed from the control system.

Note

For "Execute externally", **no** block search is possible.

Requirements for external execution via network

- The control system and the external programming device/PC are connected via Ethernet.
- The RCS tool is installed on the programming device/PC.

The following conditions are required on the the devices:

1. Control: (see "User Management")
   - Create an authorization for using the network using the following dialog:

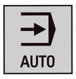
     Operating area <SYSTEM> > "Service Display" > "Service Control" > "Service Network" > "Authorization" > "Create"
2. Control: (see "User log in - RCS log in")
   - Log in for the RCS connection using the following dialog:

     Operating area <SYSTEM> > vertical softkey "RCS log in" > "Log in"
3. Programming device/PC:
   - Start the RCS tool.

4. Programming device/PC:
   – Activate the drive/directory for network operation.
5. Programming device/PC:
   – Establish an Ethernet connection to the control.
6. Control: (see "Connecting / disconnecting a network drive")
   – Connect to the directory activated on the programming device/PC using the following dialog:

     Operating area <SYSTEM> > "Service Display" > "Service Control" > "Service Network" > > "Connect" > "RCS Network" (Select a free drive of the control > Enter the server name and and activated directory of the programming device/PC, for example: "\\123.456.789.0\External Program")

Operating sequences for external execution via network



Select the <AUTOMATIC> mode key .

Press the <PROGRAM MANAGER> key on the machine control panel.

RCS connect.

Press "RCS connect."".
You go to the directories of the PG/PC.

Place the cursor bar on the desired program.

Ext. execution

Press "Ext. execution".

The program is transferred into the buffer memory and selected and displayed in the program selection automatically.

Press the <NC START> key.

Machining starts. The program is reloaded continuously.

At the end of the program or in case of <RESET>, the program is automatically removed from the control system.

Note

The program can only be executed. Program correction is not possible at the control.

Part programming

7

7.1 Part programming overview

Menu tree

Only SINUMERIK 802D sl pro

NC directory	Customer CF card	RCS connection	RS232	Manu. drive	USB drive		
Execute	Execution from external	Execution from external		Execution from external	Execution from external		
New	New directory	New directory		New directory	New directory		
Open	Open	Open	Send	Open	Open		
Select all	Select all	Select all	Receive	Select all	Select all		
Copy	Copy	Copy		Copy	Copy		
Paste	Paste	Paste		Paste	Paste		
Delete	Delete	Delete	Error log	Delete	Delete		
Next ...	Next ...	Next ...	Next	Next ...	Next ...		

Figure 7-1 "Program Manager" menu tree

Functionality

The PROGRAM MANAGER operating area is the management area for workpiece programs in the control system. In this area, programs can be created, opened for modification, selected for execution, copied, and inserted.

Operating sequence

Press the <PROGRAM MANAGER> key to open the program directory.

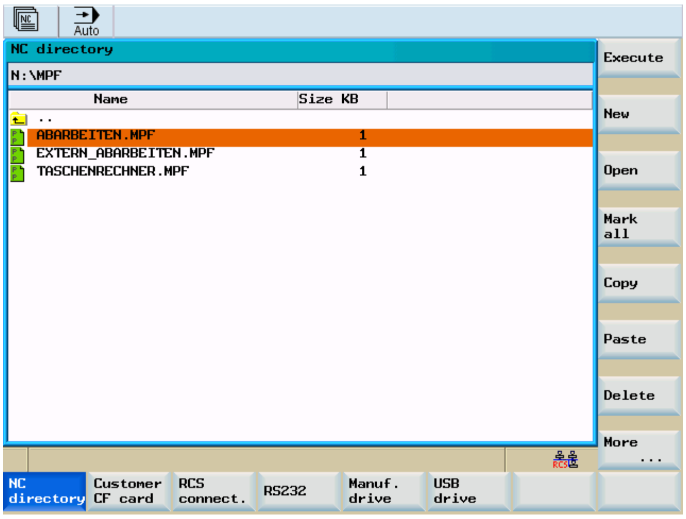


Figure 7-2 The "Program Manager" start screen

Use the cursor keys to navigate in the program directory. To find program names quickly, simply type the initial letter of the program name. The control system will automatically position the cursor on a program with matching characters.

Softkeys

NC directories	Use this softkey to display the directories of the NC.
Exe-cute	Use this softkey to select the program on which the cursor is placed for execution. The control system will switch to the position display. Use <NC START> to start this program.
New	Use the "New" softkey to create a new program.
Open	Use the "Open" softkey to open the file highlighted by the cursor for processing.
Mark all	Use this softkey to select all files for the subsequent operations. The selection can be canceled by pressing the softkey once more.

Note

Selecting individual files:

Position the cursor on the corresponding file and press the <Select> key. The selected line will change its color. If you press the <Select> key once more, the selection is canceled.

Copy

This function will enter one or several files in a list of files (called 'clipboard') to be copied.

Paste

This function will paste files or directories from the clipboard to the current directory.

Delete

When selecting the "Delete" softkey, the file selected by the cursor is deleted after a confirmation warning. If several files have been selected, all these files will be deleted after a confirmation warning.

Use the "OK" softkey to execute the deletion request and "Abort" to discard.

More ...

Use this softkey to branch to further functions.

Renaming

A window opens where you can rename the program you have selected beforehand using the cursor.

After you have entered the new name, either press "OK" to confirm or "Abort" to cancel.

Preview window

This function opens a window displaying the first seven lines of a file if the cursor has been positioned on the program name for a certain time.

Find

A window opens up where you can enter a file name you are looking for.

After you have entered the name, either press "OK" to confirm or "Abort" to cancel.

Enables

A selected directory can be released for network operation.

Split window

The function splits the window on the HMI. You can use the <Tab> key to switch over between windows.

Proper-ties

The function gives information on the properties of the memory of the selected directory and of the selected file.

Error log

The function gives information in a logfile on the executed functions (e.g. copying a file) as well as on wrongly executed functions of the PROGRAM MANAGER. The logfile will be deleted after cold restart of the control.

Customer CF card

Selecting this softkey provides the functions required to read out / read in files via the customer CompactFlash card and the function "Program execution from external". When the function is selected, the directories of the customer CompactFlash card are displayed.

Ext. execution

Use this softkey to select the program on which the cursor is placed for execution. If the CF card is selected, the program is executed by the NC as an external program. This program must not contain any program calls of part programs which are not stored in the directory of the NC.

RCS connect.

This softkey is needed in connection with the work in the network. Additional information is provided in Chapter, network operation (only for SINUMERIK 802D sl pro).

RS232

The functions required for reading out/reading in files are provided via the RS232 interface.

Send

Use this function to transmit files from the clipboard to a PC connected to the RS232.

Receive

Load files via the RS232 interface.

For the settings of the interface, please refer to the "System" operating area. The part programs must be transmitted using the text format.

Error log

Error log

Manufac-turer drive

Selecting this softkey provides the functions required to read out / read in files via the manufacturer drive and the function "Program execution from external". When the function is selected, the directories of the manufacturer's drive are displayed.

USB drive

Selecting this softkey provides the functions required to read out / read in files via USB FlashDrive and the function "Program execution from external". When the function is selected, the directories of the USB FlashDrive are displayed.

7.2 Enter new program

Operating sequences

PROGRAM MANAGER

You have selected the PROGRAM MANAGER operating area.

NC directories

New

Use the "NC directory" softkeys to select the storage location for the new program.

Press "New". You have the choice of the following options:

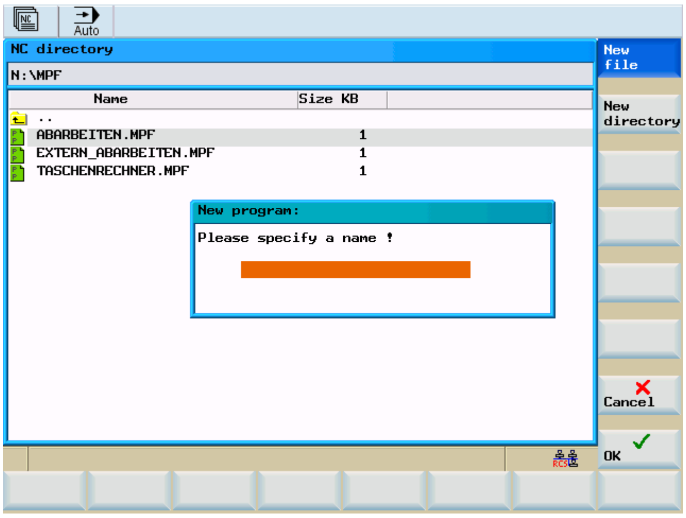


Figure 7-3 New program

New directory

After presssing the softkey "New directory" a dialog window will open up for setting up a new file.

Enter a name and confirm with "OK."

New file

After presssing the softkey "New file" a dialog window will open up for setting up a new program file. in which you can enter the names of the new main programs and subprograms. The .MPF extension for main programs is entered automatically. The .SPF extension for subprograms must be entered along with the program name.

OK

Conclude your entry with "OK". The new part program file will be created, and the editor window is opened automatically.

Abort

Use "Cancel" to cancel the creation of the program. the window is closed.

7.3 Editing the part program

Functionality

A part program can only be edited if it is currently not being executed.

Any modifications to the part program are stored immediately.

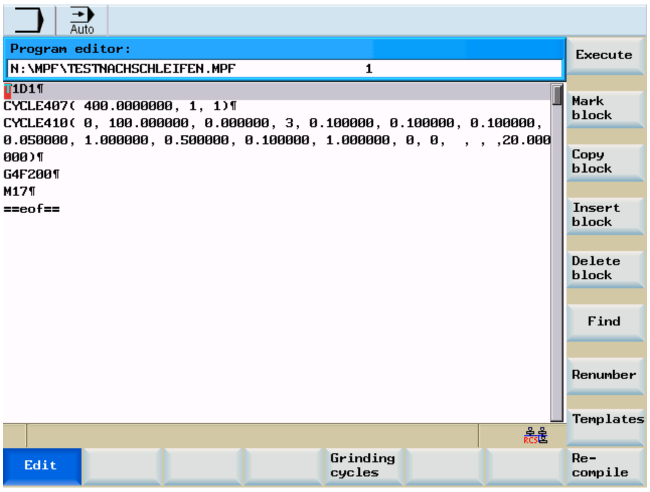


Figure 7-4 Program editor start screen

Menu tree

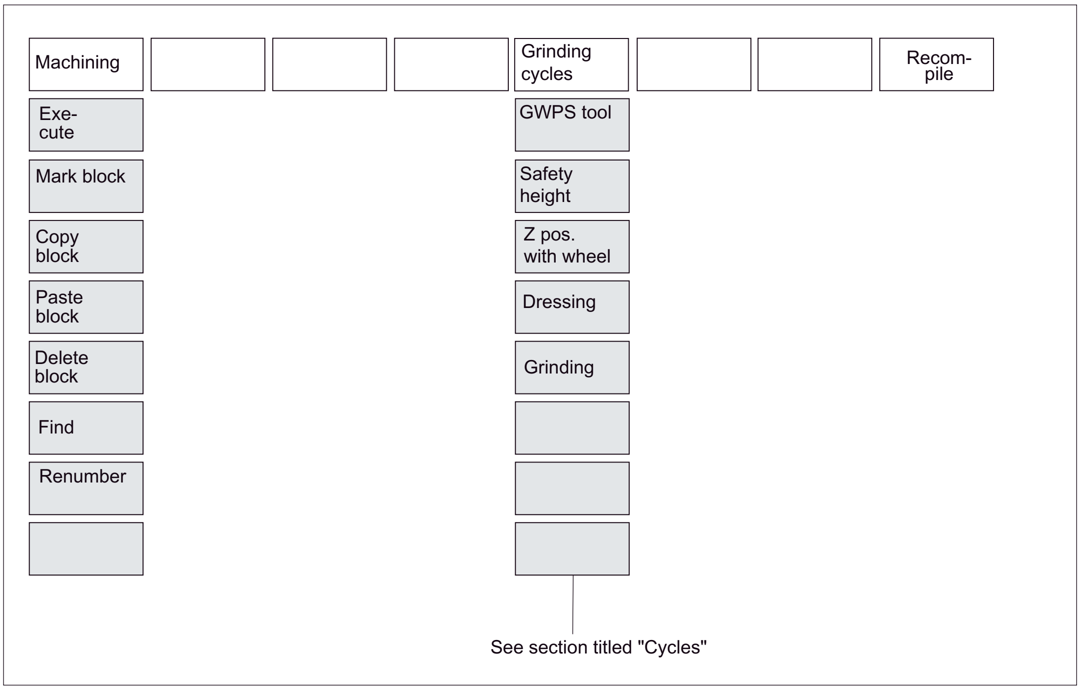


Figure 7-5 "Program" menu tree

Operating sequence

PROGRAM MANAGER	In the PROGRAM MANAGER operating area, select the program to be edited.
Open	Press the <Open> softkey. The selected program will open.

Softkeys

Machining	Use this softkey to edit a file.
Ext. execution	Use this softkey to execute the selected file.
Mark block	Use this softkey to select a text segment up to the current cursor position (alternatively: <CTRL+B>)
Copy block	Use this softkey to copy a selected block to the clipboard (alternatively: <CTRL+C>)
Insert block	Use this softkey to paste a text from the clipboard at the current cursor position (alternatively: <CTRL+V>)
Delete block	Use this softkey to delete a selected text (alternatively: <XTRL+X>)
Find	Use the <Find> softkey to search for a string in the program file displayed. Type the term you are looking for in the input line and use the <OK> softkey to start the search. Use <Abort> to close the dialog box without starting the search process.
Renumber	Use this softkey to replace the block numbers from the current cursor position up to the program end.
Grinding cycles	See section titled "Cycles" (Page 156)
Recomp.	For recompilation, position the cursor on the cycle calling line in the program. This function decodes the cycle name and prepares the screenform with the relevant parameters. If there are any parameters beyond the range of validity, the function will automatically use the default values. After closing the screenform, the original parameter block is replaced by the corrected block.

Note

Only automatically generated blocks can be recompiled.

System 8

8.1 "System" operating area

Functionality

The SYSTEM operating area includes functions required for parameterizing and analyzing the NCK, the PLC and the drive.

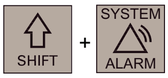
Depending on the functions selected, the horizontal and the vertical softkey bars change. The menu tree shown below **only** includes the horizontal softkeys.

Menu tree

Start-up	Machine data	Service Display	PLC		Start-up Files		
NC	General MD	Service Axes	STEP 7 connect.		802D Data		
PLC	Axis MD	Service Drives	PLC State		Customers CF card		
HMI	Channel MD	Service ext. bus	State List		RCS connect		
	Drives MD	Service Control	PLC Program		RS232		
		Service overview	Program List		Manufacturer's drive		
	Display MD				USB Drive		
	Servo trace	Servo trace			Manu. archive		
		Version	Edit PLC alarm txt				

Figure 8-1 System menu tree

Operating sequence



The full CNC keyboard is used to change to the <SHIFT> and <SYSTEM> operating areas and the start screen is displayed.

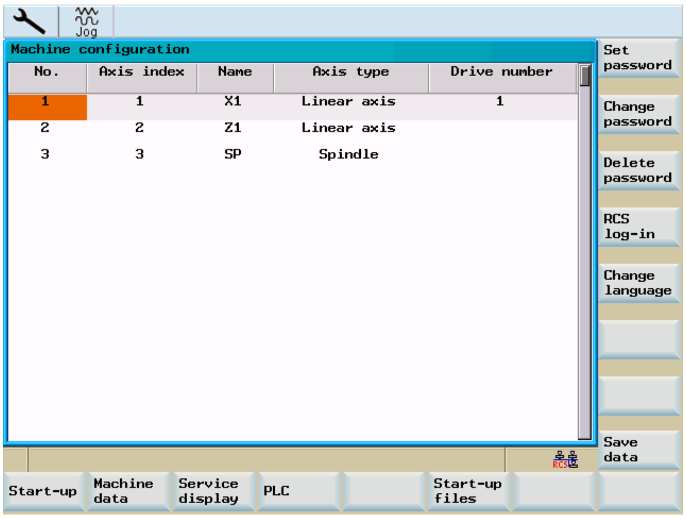
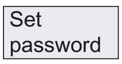


Figure 8-2 "System" operating area start screen

Softkeys

The start screen vertical softkeys are described below.



"Set password"

Three password levels are distinguished in the control system, which provide different access rights:

- System password
- Manufacturer password
- User password

It is possible to change certain data corresponding to the access levels. If you do not know the password, access will be denied.

Note

Also see SINUMERIK 802D sl "Lists".

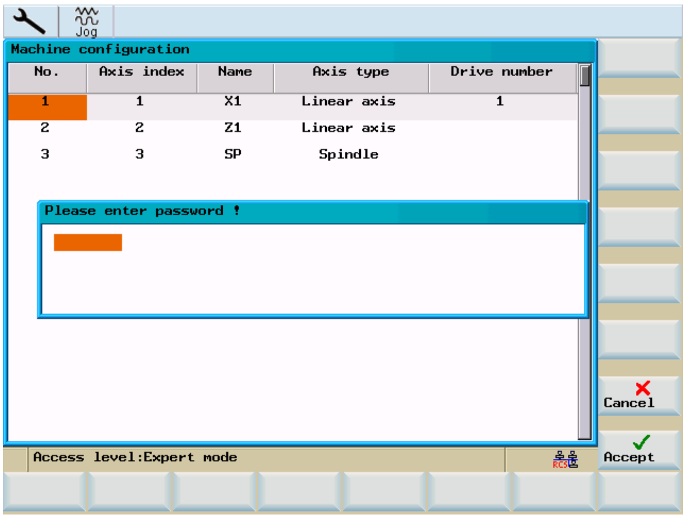


Figure 8-3 Entering the password

After selecting the "Accept" softkey, the password is set.

Use "Abort" to return without any action to the "System" start screen.

Change password

"Change Password"

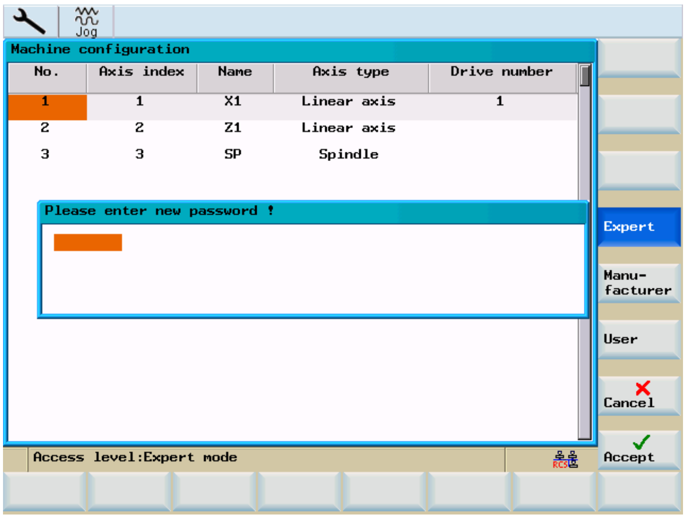


Figure 8-4 Change password

Depending on the access right, various possibilities are offered in the softkey bar to change the password.

Select the password level using the appropriate softkeys. Enter the new password and press "Accept" to complete your input. You will be prompted to enter the new password once more for confirmation.

Press "Accept" to complete the password change.

Use "Abort" to return without any action to the start screen.

Delete password — Resetting the credential

RCS log-in — User network log-in

Change language — Use "Change language" to select the user interface language.

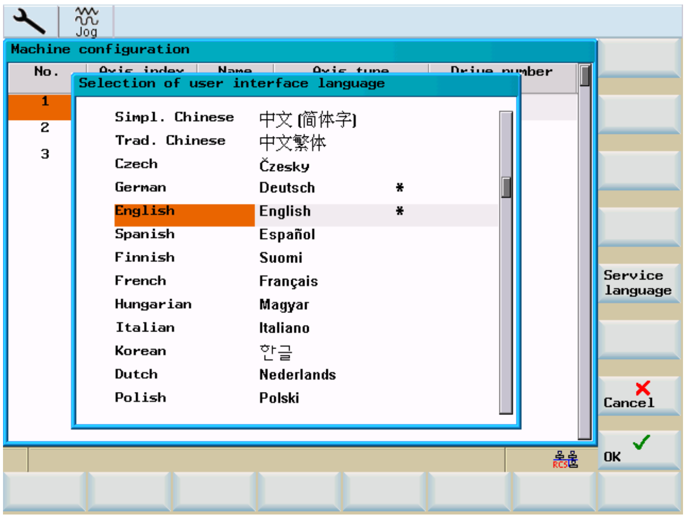


Figure 8-5 User interface language

Use the cursor keys to select the language and confirm it by pressing "OK".

Note

The HMI is automatically restarted when a new language is selected.

Service language — Use "Service language" to always select "English" as the user interface language.

Press the "Service language" softkey again to restore the previously active language (e. g. "Simpl. Chinese").

Note

An asterisk "*" marks the languages you have used.

Save data

"Save data"

This function will save the contents of the volatile memory into a nonvolatile memory area.

Requirement: There is no program currently executed.

Do not carry out any operator actions while the data backup is running!

The NC and PLC data are backed up. The drive data are not backed up.

Note

Saved data can be called via the following operator action:

- Press the <SELECT> key while the control system is booting.
- In the setup menu, select "Reload saved user data".
- Press the <Input> key

Note

Data that have been backed up can be called again from the operating area <SYSTEM> > "Start-up" > "Power up with backed up data"!

8.2 SYSTEM - "Start-up" softkeys

Start-up

Commissioning

NC

Use this softkey to select the NC power-up mode.

Select the desired mode using the cursor.

- Normal power-up
  The system is restarted
- Power-up with default data
  The display machine data are reset to their standard values (restores the initial state when originally supplied)
- Power-up with backed up data
  The system restarts with the data that were last backed up (see Backup data)

PLC

The PLC can be started in the following modes:

- Restart
- Memory reset

Furthermore, it is possible to link the start with a subsequent **debugging mode**.

HMI

Selects the power-up mode of the HMI.

Select the desired mode using the cursor.

- Normal power-up
  The system is restarted
- Power-up with default data
  The system restarts with default values (restores the initial state when originally supplied)

OK

Use "OK" to RESET the control system and to carry out a restart in the mode selected.

Use the <RECALL> key to return to the system start screen without performing any action.

8.3 SYSTEM - "Machine data" softkeys

References

You will find a description of the machine data in the following manufacturers´ documents:

SINUMERIK 802D sl List Manual

SINUMERIK 802D sl Function Manual for turning, milling, nibbling

Machine data

Machine data

Any changes in the machine data have a substantial influence on the machine.

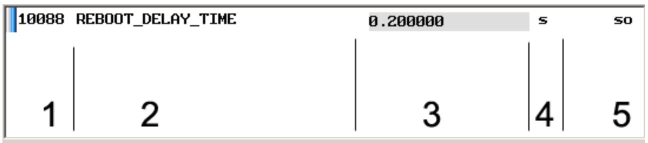


Figure 8-6 Structure of a machine data line

Table 8- 1 Legend

No.	Significance		
1	MD number		
2	Name		
3	Value		
4	Unit		
5	Effective	so	immediately effective
		cf	with confirmation
		re	Reset
		po	Power on

⚠ CAUTION
Incorrect parameterization may result in destruction of the machine!

The machine data are divided into the groups described in the following.

General machine data

General MD

Open the "General machine data" window. Use the Page Up / Page Down keys to browse forward / backward.

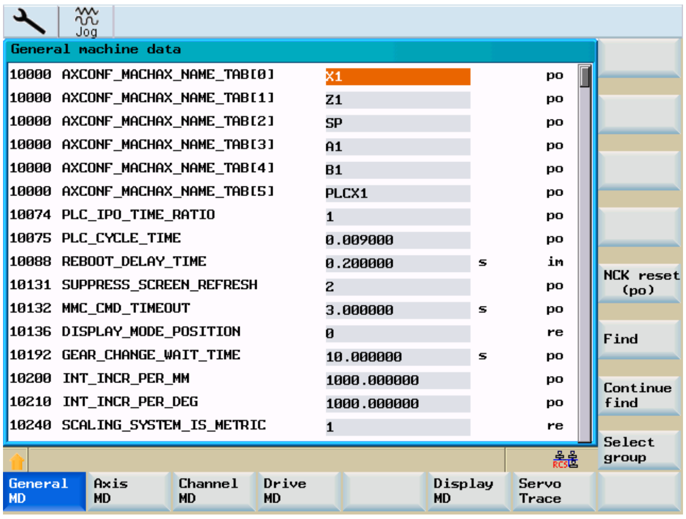


Figure 8-7 General machine data

NCK reset (po)

Executes a warm restart at the control.

Find

"Find"

Type the number or the name (or a part of the name) of the machine data you are looking for and press "OK".

The cursor will jump to the data searched.

Continue find

Use this softkey to continue searching for the next match.

Select group

This function provides various display filters for the active machine data group. Further softkeys are provided:

- "Expert": Use this softkey to select all data groups of the expert mode for display.
- "Filter active": Use this softkey to activate all data groups selected. After you have quit the window, you will only see the selected data on the machine data display.
- "Select all": Use this softkey to select all data groups of the Expert mode for display.
- "Deselect all": Selecting this softkey deselects all data groups.

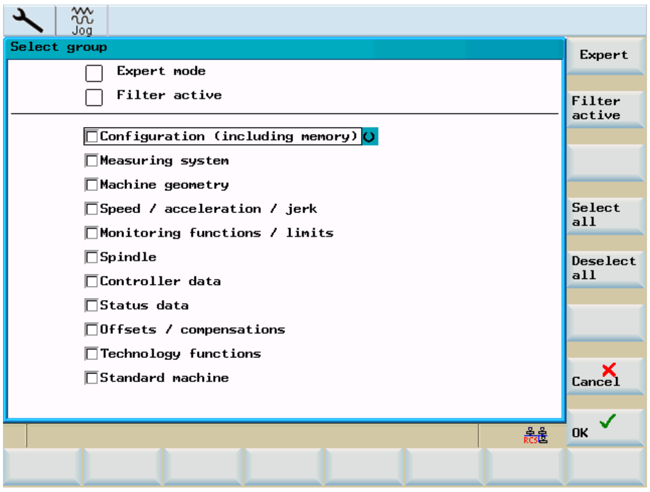


Figure 8-8 Display filter

Axis-specific machine data

axis MD

Open the "Axis-specific machine data" window. The softkey bar will be supplemented by the softkeys "Axis +" and "Axis ".

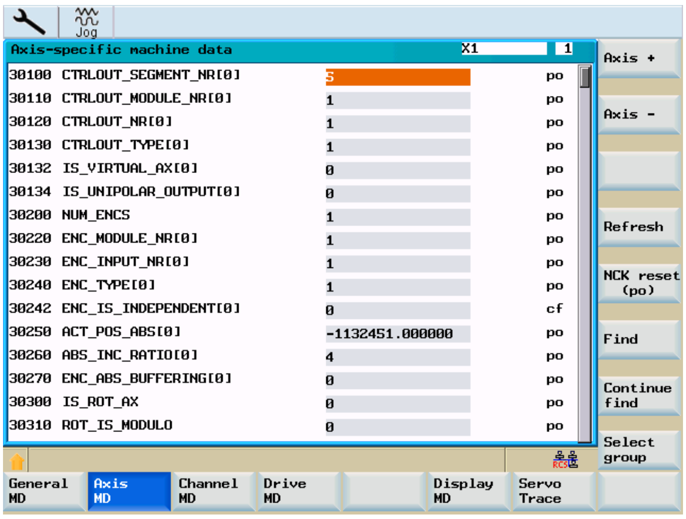


Figure 8-9 Axis-specific machine data

The data of axis 1 are displayed.

Axis +
Update

Use "Axis +" or "Axis " to switch to the machine area of the next or previous axis.

The contents of the machine data are updated.

Channel-specific machine data

chan
MD

Open the "Channel-specific machine data" window. Use the PageUp / PageDown keys to browse forward / backward.

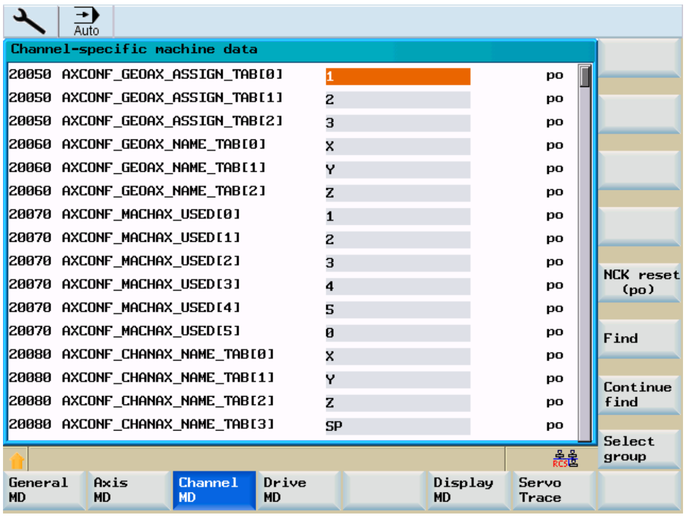


Figure 8-10 Channel-specific machine data

SINAMICS drive machine data

Drive MD

Open the "Drive machine data" dialog box.

The first dialog box displays the current configuration, as well as the states of the control, power supply and drive units.

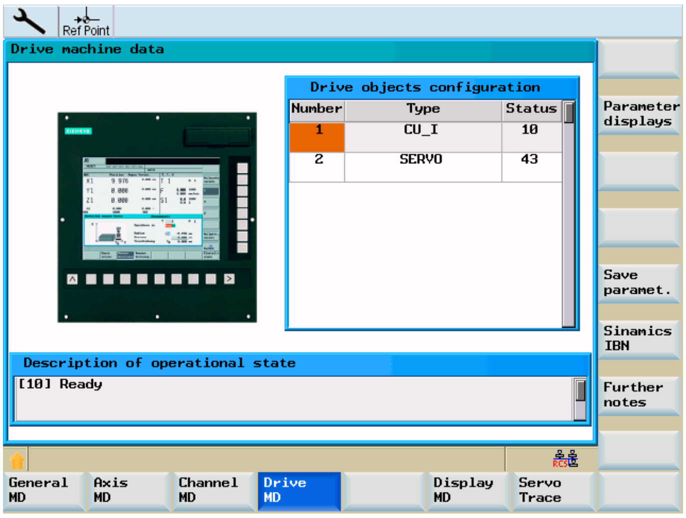


Figure 8-11 Drive machine data

Display parameter

To display the parameters, position the cursor on the appropriate unit and press the "Parameter display" softkey. For a description of the parameters, please refer to the documentation of SINAMICS drives.

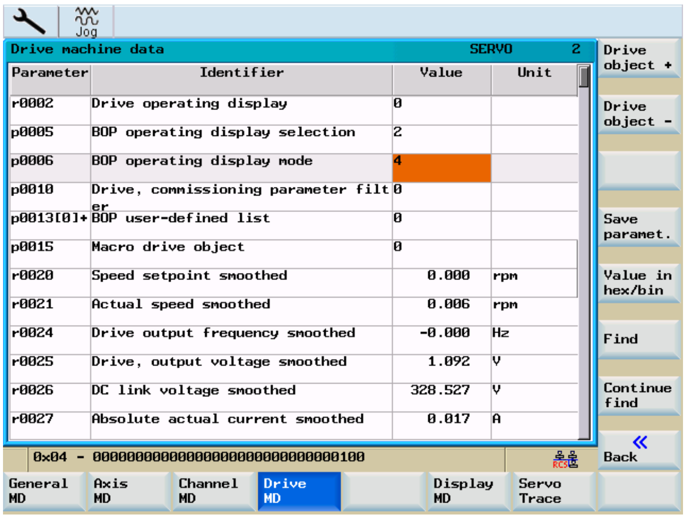


Figure 8-12 Parameter list

Drive object +

Drive object -

Switch to the respective drive objects.

Value in hex/bin

In the note line, the selected value is displayed in hexadecimal and binary values.

Find

Use these functions to search in the parameter list for the term you are looking for.

Continue find

Display of machine data

Display MD

Open the "Display machine data" window. Use the PageUp / PageDown keys to browse forward / backward.

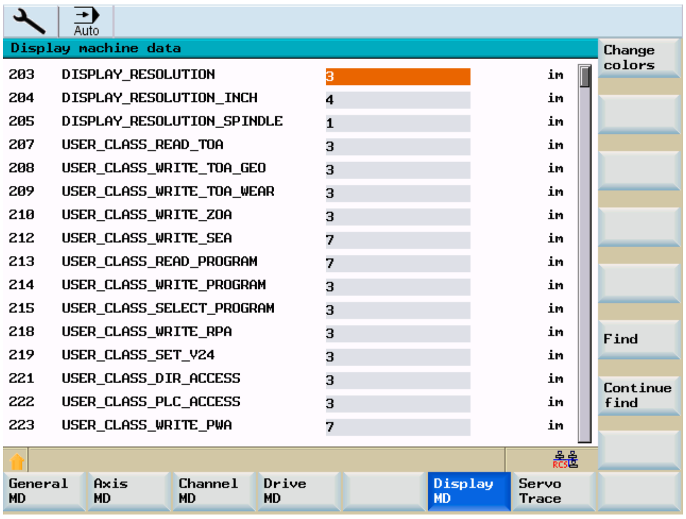


Figure 8-13 Display of machine data

Color changing

Use the "Softkey color" and "Window color" softkeys to specify user-defined color settings. The displayed color consists of the components red, green and blue.
The "Change color" window displays the values currently set in the input fields. The desired color can be produced by changing these values. In addition, the brightness can be changed.

The next mixing ratio is displayed temporarily upon completion of an input. Use the cursor keys to switch between the input fields.

With "OK", the settings are accepted and the dialog box is closed. Selecting the "Abort" softkey will close the dialog box without accepting your changes.

Color Softkey

Use this function to change the colors of the tip and softkey area.

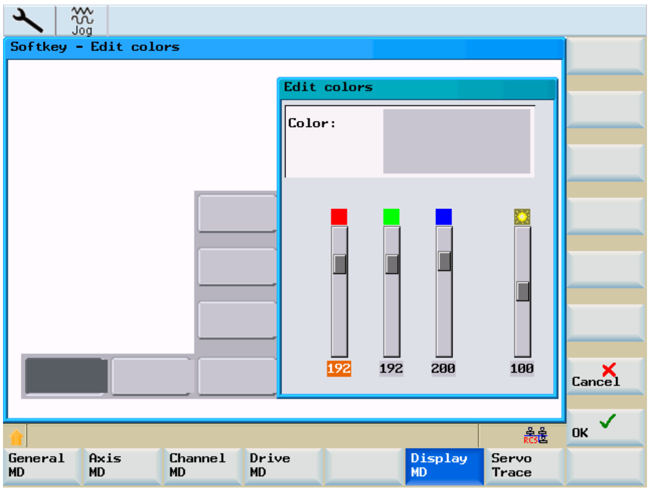


Figure 8-14 Edit softkey color.

Color Window

Use this softkey to change the color of the border of dialog boxes.
The "Active window" softkey function will assign your settings to the focus window, and the "Inactive window" function to the non-active window.

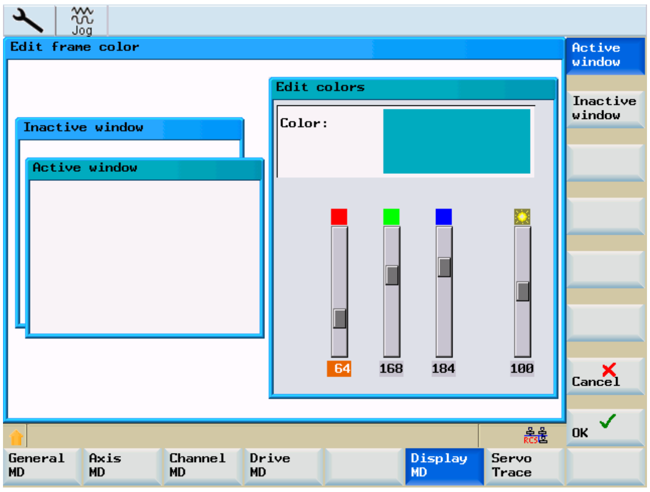


Figure 8-15 Edit frame color.

8.4 SYSTEM - "Service display"

Service display

The "Service display" window appears on the screen.

The start screen for the "Service control" function is shown in the following diagram.

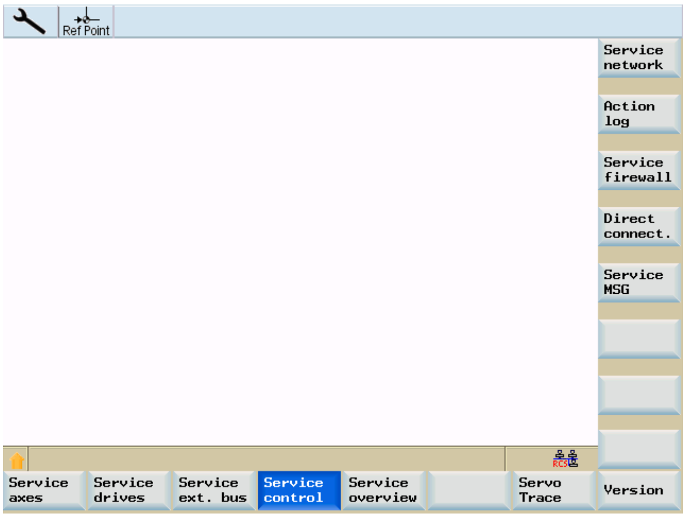


Figure 8-16 The "Service control" start screen

Service axes

This window displays information on the axis drive.

The "Axis +" or "Axis -" softkeys are additionally displayed. These can be used to display the values for the next or previous axis.

Service drives

This window displays information in respect of the digital drive.

Service ext. Bus

The window displays information on external bus settings.

Service control

Use this softkey function to activate the window for the following functions:

- "Service network" (see chapter "Network operation")
- "Action log" (see chapter "Action log")
- "Service Firewall" (see chapter "Network operation")
- "Direct connect." (see chapter "Network operation")
- "Service MSG" (see chapter "Service MSG")

Service Overview

This window contains information about

- Assignment, Machine axis <=> Channel axis <=> Drive number
- The enable status of the NC and drive
- Drive state regarding ready, faults and alarms

Servo trace

An oscilloscope function is available in this window to optimize the drives (see chapter "Servo trace").

Version

This window displays the version numbers and the date of creation of the individual CNC components.

The following functions can be selected from this window (also see chapter "Versions"):

- "HMI details"
- "License key"
- "Options"
- "Save as"

  The displayed versions can be saved in a text file

8.4.1 Action log

Action log

The function "Action log" is provided for service events. The contents of the action log file can only be accessed through a system password on the HMI.

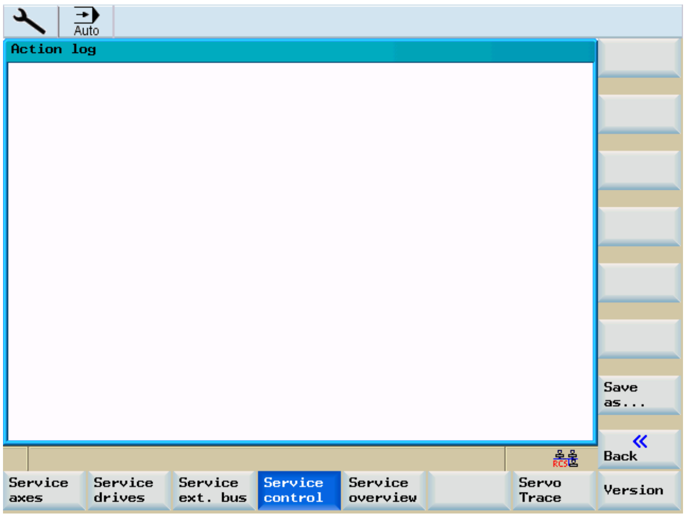


Figure 8-17 Action log

Save under

Irrespective of the system password, it is possible to output the file using softkey "Save under..." on a CF card or on the USB FlashDrive.

In case of difficulties, please contact the hotline (see the "Technical Support" Section in the preface for contacting the hotline).

8.4.2 Servo trace

Servo trace

An oscilloscope function is provided for the purpose of optimizing the drives. This enables graphical representation:

- of the velocity setpoint
- of the contour violation
- of the following error
- of the actual position value
- of the position setpoint
- of exact stop coarse / fine

The start of tracing can be linked to various criteria allowing a synchronous tracing of internal control states. This setting must be made using the "Select signal" function.

To analyze the result, the following functions are provided:

- Changing and scaling of abscissa and ordinate;
- Measuring of a value using the horizontal or vertical marker;
- Measuring of abscissa and ordinate values as a difference between two marker positions;
- Storing of the result as a file in the part program directory. Thereafter, it is possible to export the file using either RCS802 or the CF card and to process the data in MS Excel.

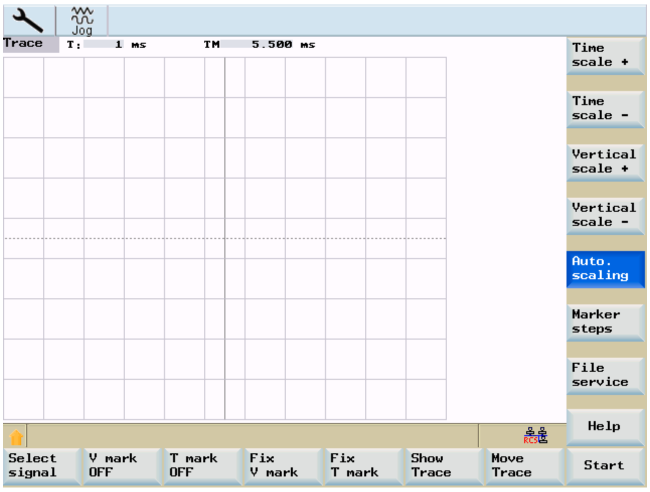


Figure 8-18 Servo trace start screen

The header of the diagram contains the current scaling of the abscissa and the difference value of the markers.

The diagram shown above can be moved within the visible screen area using the cursor keys.

1 Time Base
2 Marker position time
3 Difference in time between marker 1 and current marker position.

Figure 8-19 Meaning of the fields

Select signal

Use this menu to parameterize the measuring channel.

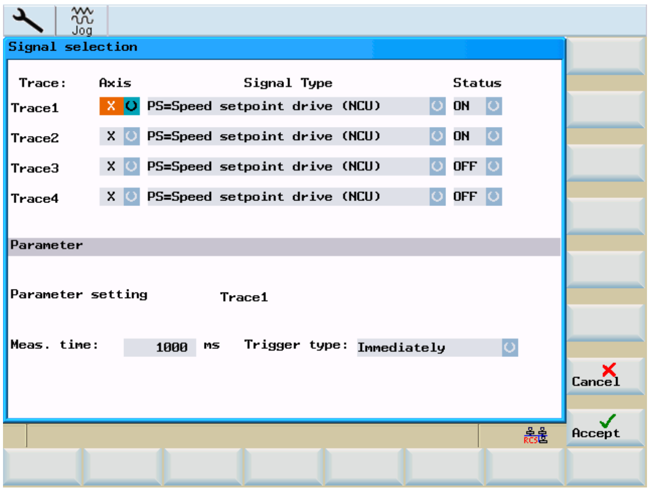


Figure 8-20 Select signal

- **Selecting the axis:** To select the axis, use the "Axis" toggle field.
- **"Signal type":**

  Following error
  Controller difference
  Contour deviation
  Position actual value
  Speed actual value
  Speed setpoint
  Compensation value
  Parameter block
  Position setpoint controller input
  Speed setpoint controller input
  Acceleration setpoint controller input
  Speed feedforward control value
  Exact stop fine signal
  Exact stop coarse signal
- **"Status":**

  On: Tracing is performed in this channel
  Off: Channel inactive

The parameters for the measuring time and for the trigger type for channel 1 can be set in the lower screen half. The remaining channels will accept this setting.

- **Determining the measuring period:** The measuring period in ms is entered directly into the "Measuring period" input field (6,133 ms max.).
- **Selecting the trigger condition:** Position the cursor on the "Trigger condition" field and select the relevant condition using the toggle key.
  - No trigger, i.e. the measurement starts directly after selecting the "Start" softkey;
  - Positive edge;
  - Negative edge
  - Exact stop fine reached;
  - Exact stop coarse reached

V mark off

T mark off

The "V mark ON" / "V mark OFF"softkeys are used to hide/show the vertical gridlines. Using the "Select signal" function you can determine the signal to be displayed in the vertical axis.

The "T mark ON" / "T mark OFF"softkeys are used to hide/show the horizontal gridlines of the time axis.

Fix V mark

Use the markers to determine the differences in the horizontal or vertical directions. To do this, place the marker on the start point and press "Fix V mark" or "Fix T mark". The difference between the starting point and the current marker position is now displayed in the status bar. The softkey labels will change to "Free V mark" or "Free T mark".

Trace display

This function opens another menu level offering softkeys for hiding / displaying the diagrams. If a softkey is displayed on a black background, the diagrams are displayed for the selected trace channel.

Time scale +

Use this function to zoom in / zoom out the time basis.

Vertical scale +

Use this function to increase / reduce the resolution (amplitude).

Marker steps

Use these softkeys to define the step sizes of the markers.

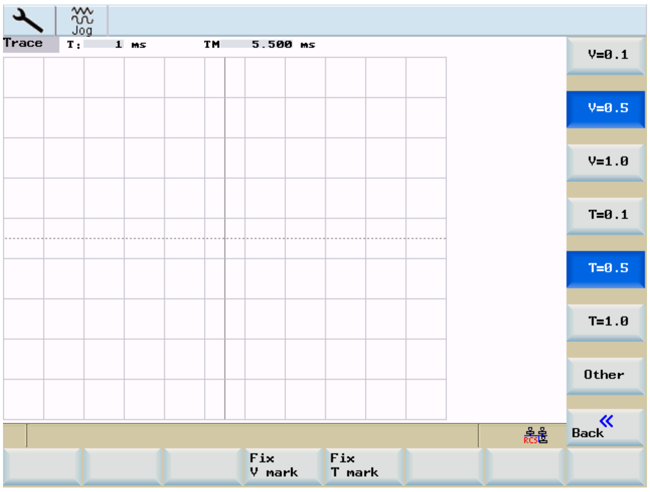


Figure 8-21 Marker steps

The markers are moved using the cursor keys at a step size of one increment. larger step sizes can be set using the input fields. The value specifies how many grid units the marker must be moved per "SHIFT" + cursor movement. When a marker reaches the margin of the diagram, the grid automatically appears in the horizontal or vertical direction.

File service

Use this softkey to save or load trace data.

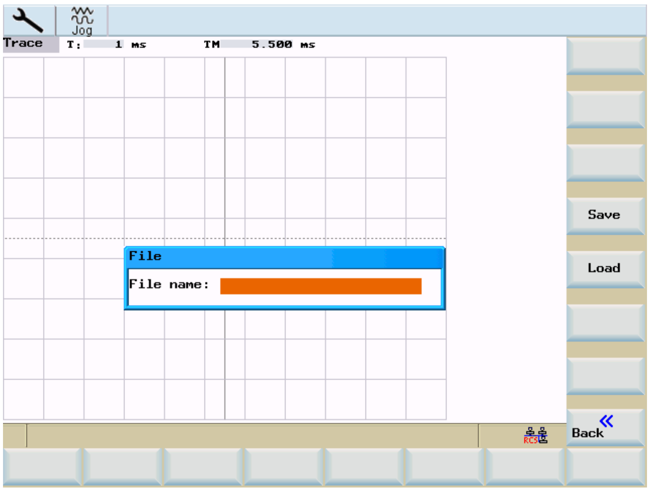


Figure 8-22 Trace data

Type the desired file name without extension in the "File name" field.

Use the "Save" softkey to save the data with the specified name in the part program directory. Thereafter, the file can be exported, and the data can be processed in MS Excel.

"Load" loads the specified file and graphically displays the data.

8.4.3 Version/HMI details

Version

This window displays the version numbers and the date of creation of the individual CNC components.

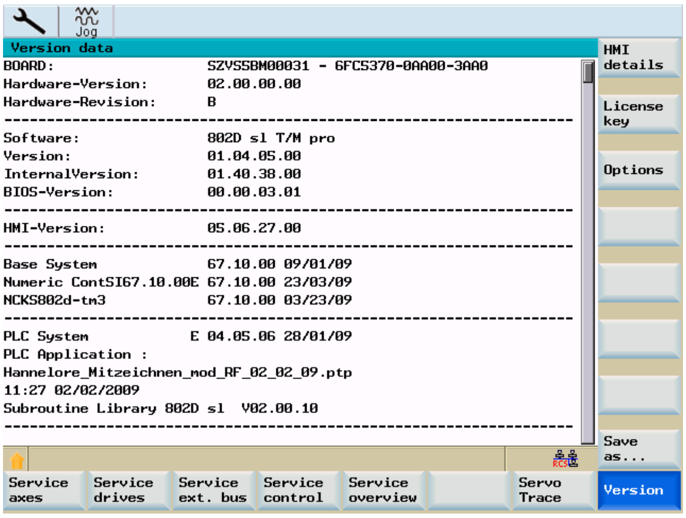


Figure 8-23 Version

Note

The version releases shown in the version screen shot are for example only.

Save under

Saves the contents of the "Version" window in a text file. The target (e.g. "customer CF card") can be selected.

HMI Details

The "HMI details" menu is intended for servicing and can only be accessed via the user password level. All programs provided by the operator unit are displayed with their version numbers. By reloading software components, the version numbers can be differ from each other.

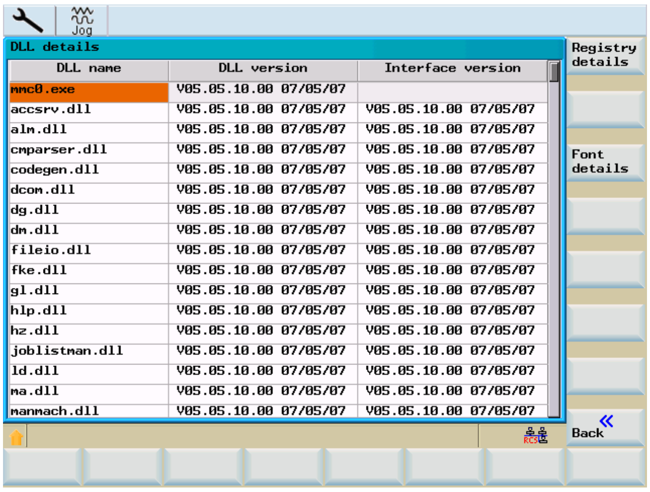


Figure 8-24 The "HMI version" menu area

Registry Details

This "Registry details" function displays the assignment of the hard keys (operating area keys POSITION (machine), OFFSET PARAM (parameter), PROGRAM (program), PROGRAM MANAGER (progman), ...) for the programs to be started in the form of a list. For the meanings of the individual columns, please refer to the table below.

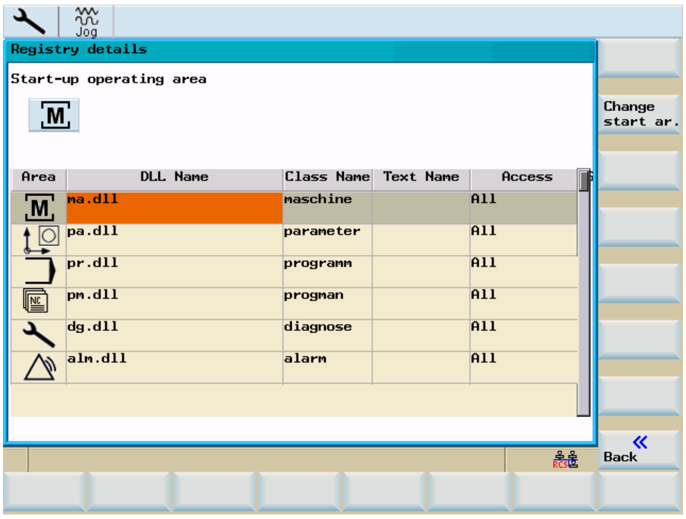


Figure 8-25 Registry details

Note

After the system has booted, the control system automatically starts the <POSITION> operating area. If a start behavior is required, the "Change ready to start" function allows defining another starting program.

The starting operating area is then displayed above the table in the "Registry Details" window.

Font Details

The "Font details" function displays the data of the loaded character sets in the form of a list.

Figure 8-26 Font details

license key

Entering the license key.

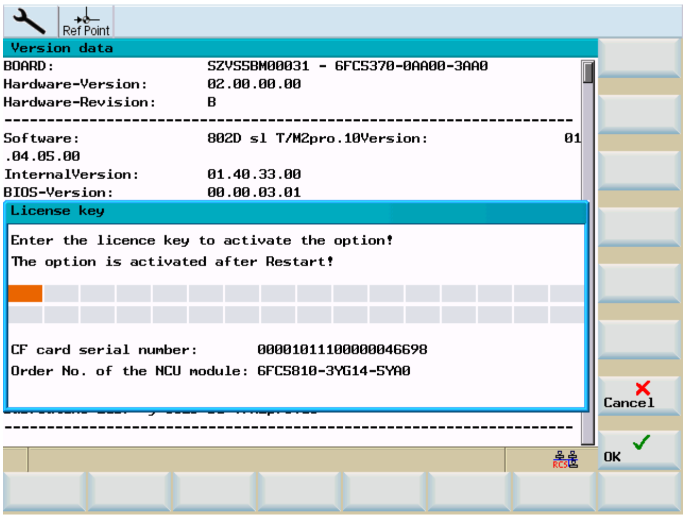


Figure 8-27 License key

References

SINUMERIK 802D sl Operating Instructions for Turning, Milling, Grinding, Nibbling; Licensing in SINUMERIK 802D sl

Options

Setting the licensed options.

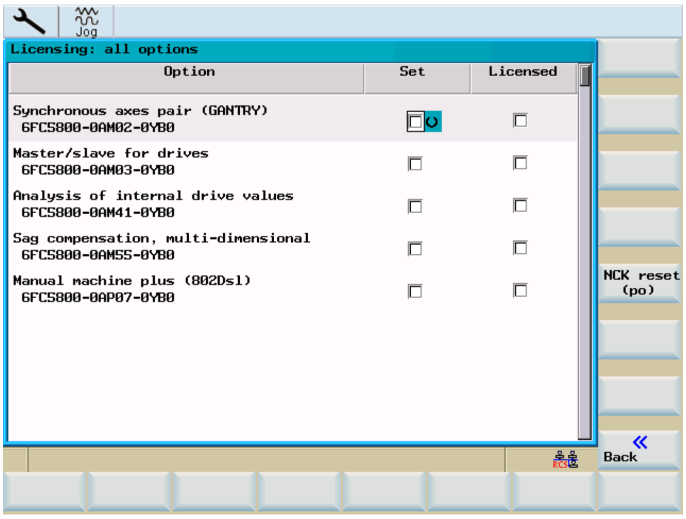


Figure 8-28 Options

References

SINUMERIK 802D sl Operating Instructions for Turning, Milling, Grinding, Nibbling; Licensing in SINUMERIK 802D sl

NCK reset (po)

Executes a warm restart at the control.

8.4.4 Service MSG

Service MSG

The "Service MSG" function allows message texts/messages to be output via the following interfaces:

- Output via the RS232 interface (V24) as data stream without protocol
- Output in a file

Message texts/messages include:

- Alarms
- Texts of MSG commands

The message texts/messages are programmed in the part program using a specified syntax. The particular syntax is described in the following table:

Table 8- 2 Syntax of the message texts/messages

Output	Syntax ("<interface>: Message text")
via RS232 interface (V24)	MSG ("V24: Message text")
in a file	MSG ("File: Message text")
Alarm line at the HMI	MSG ("Alarm text")

The MSG text output is defined using the MSG command as well as by appropriately parameterizing the output interface. For the alarm output, only the output interface has to be taken into consideration.

If the information line "Processing error MSG command occurred" is output, then the error protocol can be evaluated under the operating area <SYSTEM> > "Service display" > "Service control" > "Service MSG" > "Error protocol".

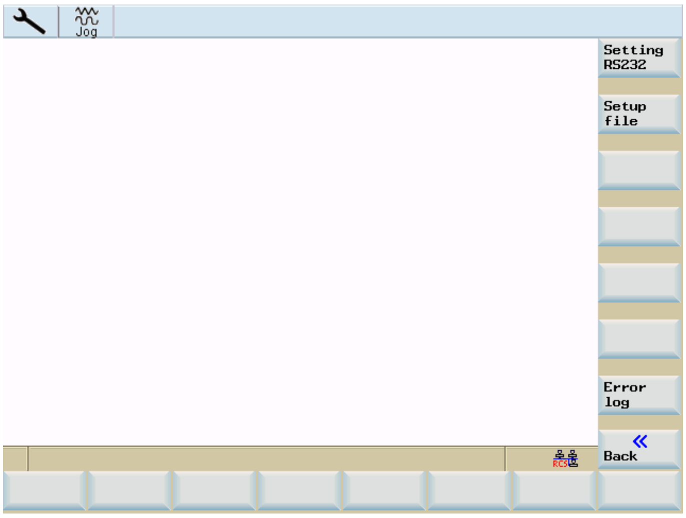


Figure 8-29 Dialog box, Service MSG

Settings for output via the RS232 interface

Setting RS232

Settings of the RS232 output interface.

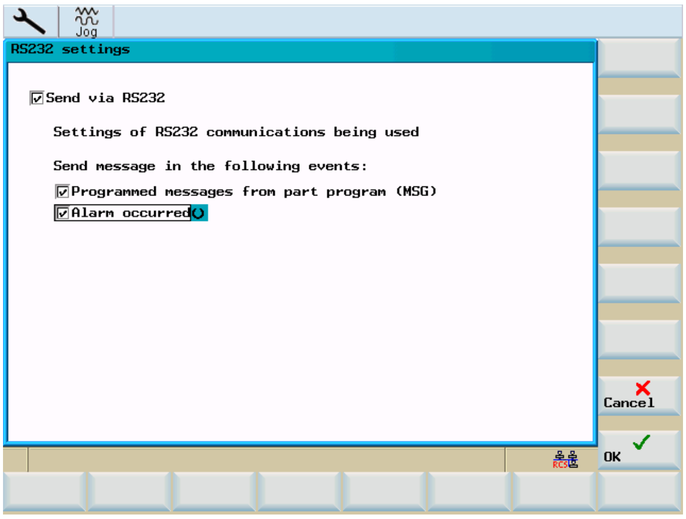


Figure 8-30 Dialog box, RS232 interface settings

"Sending messages via this interface can be activated or deactivated using the "Send via RS232" checkbox. Incoming messages are ignored when the interface is deactivated!

Note

When transferring a file via a serial interface (RS232), please note the end of transmission character for RS232 communication (analog to the communication setting, RS232 on HMI).

Further, when sending via RS232, it can be defined as to which messages are sent for which events:

- Programmed messages from the part program
- An alarm has occurred

The settings are saved and the dialog box exited by pressing the "OK" softkey.

The dialog box is exited without saving by pressing "Cancel".

To transfer messages via the RS232 interface, the communication settings from the operating area <SYSTEM> > "Start-up files" > "RS232" > "Settings" are used.

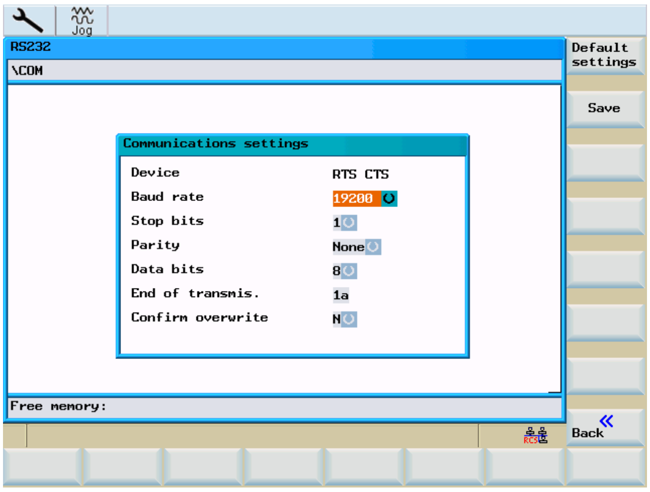


Figure 8-31 Parameters of the RS232 interface

Note

When using the MSG service via RS232, the RS232 interface must not be active for another application.

The means, e.g. the RS232 interface must not be active from the operating area <SYSTEM> "PLC" > "Step7 connect."

Settings to output in a file

Setting File

Settings for the file storage location.

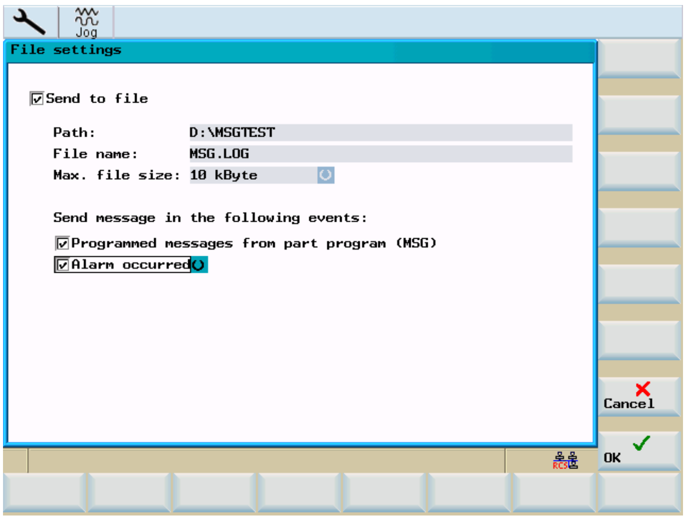


Figure 8-32 Dialog box, file settings

Sending messages to the selected file is activated or deactivated using the "Send to file" checkbox. When the interface is deactivated, messages are not output and the information line "Processing error MSG command occurred".

A path, the file name and the max. size of the file can be selected.

Drive D (customer CF card), F: (USB drive) and the drive connected per RCS connection can be selected in the "Path" input field.

10kByte, 100kByte and 1MByte can be selected as max. file size. When the max. size is reached, the file is written as ring buffer, i.e. at the beginning, as many lines are deleted line-by-line as is required by the new message at the end of the file.

Here, when sending to a file, it can be defined as to which messages are sent for which events:

- Programmed messages from the part program
- An alarm has occurred

The settings are saved and the dialog box exited by pressing the "OK" softkey.

The dialog box is exited without saving by pressing "Cancel".

Error log

Error log

Error log display.

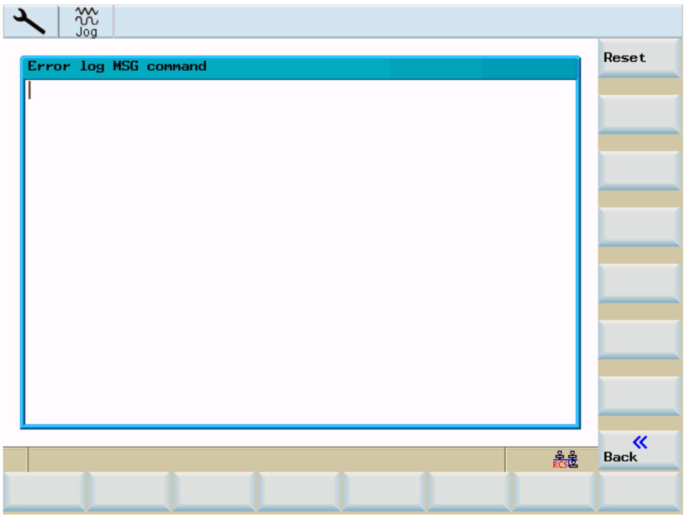


Figure 8-33 Dialog box, error log

All messages with the associated error information, where an error occurred when processing them, are saved in the error log.

The error log can be deleted using the "reset" softkey.

The dialog box is closed by pressing "Back".

Note

The error log can be used for analysis when the information line "Processing error MSG command occurred" is output.

Example of programming using the "MSG" command

For SINUMERIK 802D sl, messages programmed in the NC program are displayed in the alarm display as standard.

Table 8- 3 Activating/deleting messages

```
N10 MSG ("Roughing the contour")             ; The text "Roughing the contour"
                                             is displayed in the alarm display
N20 X… Y… N …
N…
N90 MSG ()                                   ; Delete message from the alarm
                                             display
```

Table 8- 4 Message text contains a variable

```
N10 R12=$AA_IW[X]                            ; Actual position of the X axis in
                                             R12
N20 MSG ("Check position of X axis"<<R12<<)  ; Activate message
N20 X… Y… N …
N…
N90 MSG ()                                   ; Delete message from the alarm
                                             display
```

To output messages to other interfaces, an additional command is located in front of the actual message text that defines the output interface of this message.

Table 8- 5 Messages to the RS232 output interface

```
N20 MSG ("V24: Roughing the contour")        ; The text "Roughing the contour" is
                                             sent in the ASCII format via the
                                             RS232 interface
```

Table 8- 6 Messages to the output interface file

```
N20 MSG ("FILE: Roughing the contour")       ; The text "Roughing the contour" is
                                             sent to the selected file
```

Note

If, in the part program, the text for the messages is repeated unchanged, then after each output, a command for an empty text must be entered.

e.g.

```
...
MSG("<interface>: Sample text")
MSG("<interface>:")
...
...
MSG("<interface>: Sample text")
MSG("<interface>:")
...
...
MSG("<interface>: Sample text")
MSG("<interface>:")
```

8.5 SYSTEM - "PLC" softkeys

PLC

This softkey provides further functions for diagnostics and commissioning of the PLC.

Connect to STEP 7

This softkey opens the configuration dialog for the interface parameters of the STEP 7 connection using the RS232 interface of the control system.

If the RS232 interface is already occupied by the data transfer, you can connect the control system to the PLC802 programming tool on the programming device/PC only if the transmission is completed.

The RS232 interface is initialized with activation of the connection.

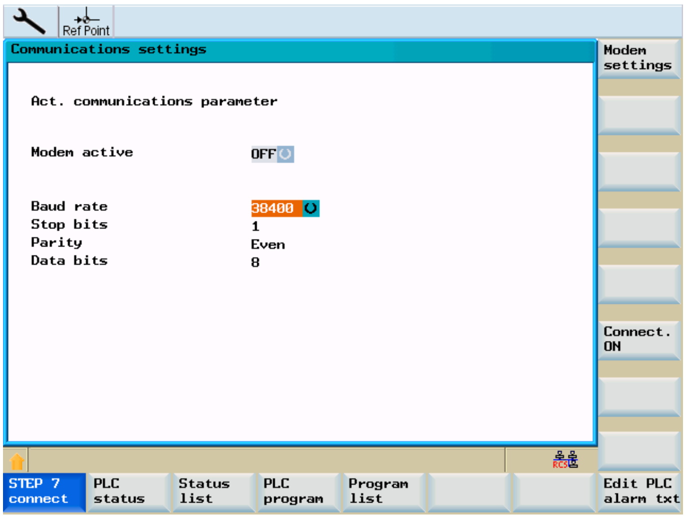
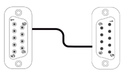


Figure 8-34 Communication settings

The baud rate is set using the toggle field. The following values are possible: 9600 / 19200 / 38400 / 57600 / 115200.



Note

The appropriate connection symbol is displayed at the bottom right after the connection has been established. The communication setting can then no longer be changed.

Modem

If the data transfer is performed on the RS232 interface via modem, start with the following initialization option:

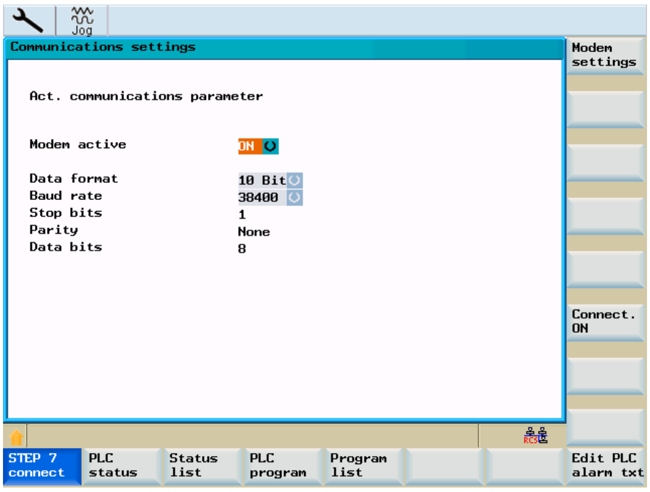


Figure 8-35 Initialize the modem

The following initializations are possible via toggle fields:

- Baud rate

  9600 / 19200 / 38400 / 57600 / 115200.

- Parity:

  "without" for 10 bit

  "odd" for 11 bit

Using the "Modem settings" softkey you can make the following additional settings for a connection that does not yet exist:

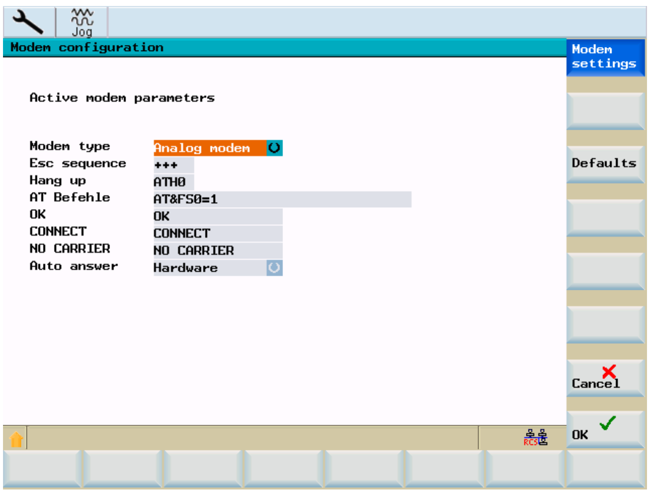


Figure 8-36 Modem settings

You can select the following modem types via toggle field:

- Analog modem
- ISDN box
- Mobile phone

Note

The types of both communication partners must match with each other.

When you want to enter several AT command sets, you have to start with AT only once and simply have to add all other commands, e.g. AT&FS0=1E1X0&W.

Refer to the manufacturers' manuals to look up the commands and their parameters, as they sometimes differ even between the devices of one manufacturer. The default values of the control system are therefore only a real minimum and should be verified in any case before they are used for the first time.

Connect. on

Use this softkey to activate the connection between the control system and the programming device/PC. The program waits for the call of Programming Tool PLC802. No modifications to the settings are possible in this state.

The softkey label changes to "Connection inactive".

By pressing "Connection inactive", the transfer from the control system can be terminated at any point. Now it is possible again to make changes in the settings.

The active or inactive state is kept even after Power On (except power-up with the default data). An active connection is displayed by a symbol in the status bar.

Press "RECALL" to exit the menu.

Additional functions

PLC status

Use this function to display and change the current states of the memory areas listed in the following table.

It is possible to display 16 operands at the same time.

Table 8- 7 Memory areas

Inputs	I	Input byte (IBx), input word (Iwx), input double-word (IDx)
Outputs	Q	Output byte (Qbx), output word (Qwx), output double-word (QDx)
Flags	M	Flag byte (Mx), flag word (Mw), flag double-word (MDx)
Times	T	Time (Tx)
Meters	C	Counter (Cx)
data	V	Data byte (Vbx), data word (Vwx), data double-word (VDx)
Format	B H D	Binary Hexadecimal Decimal
		The binary representation is not possible with double words. Counters and timers are represented decimally.

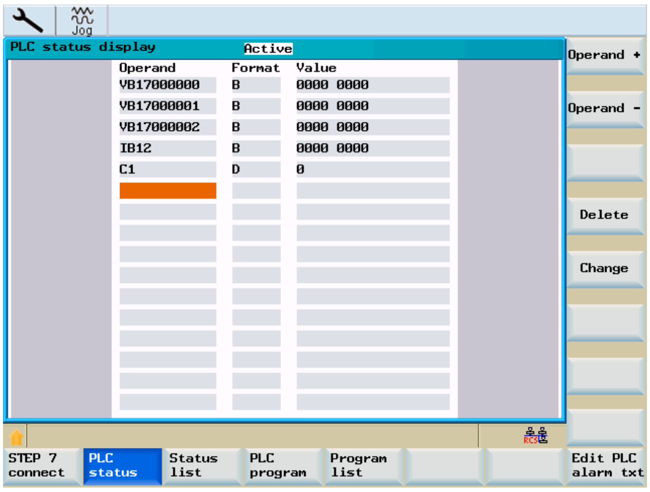


Figure 8-37 PLC status display

Operand +

The operand address displays the value incremented by 1.

Operand -

The operand address displays the value respectively decremented by 1.

Delete

Use this softkey to delete all operands.

Change

Cyclic updating of the values is interrupted. Then you can change the values of the operands.

Status list

Use the "Status list" function to display and modify PLC signals.

There are 3 lists to choose from:

- Inputs (default setting) left-hand list
- Flags (default setting) center list
- Outputs (default setting) right-hand list
- Variable

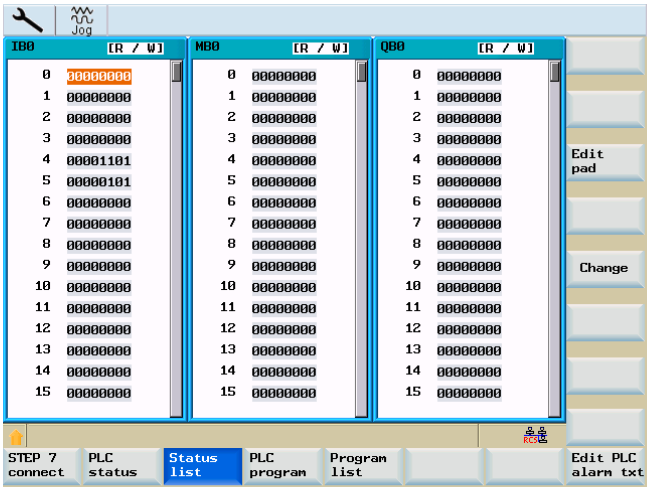


Figure 8-38 PLC status list

Change

Use this softkey to change the value of the highlighted variable. Press the "Accept" softkey to confirm your changes.

Edit pad

Use this softkey to assign the active column a new area. To this end, the interactive screenform offers four areas to choose from. For each column, a start address can be assigned which must be entered in the relevant input field. When you quit the interactive screenform, the control system will save your settings.

Figure 8-39 The "Data type" selection screen

Use the cursor keys and the "Page Up" / "Page Down" keys to navigate in and between the columns.

PLC program

PLC diagnosis using a ladder diagram (see chapter "PLC diagnosis using a ladder diagram").

Program list

Using the PLC, you may select part programs and run them via the PLC. To this end, the PLC user program writes a program number to the PLC interface, which is then converted to a program name using a reference list. It is possible to manage max. 255 programs.

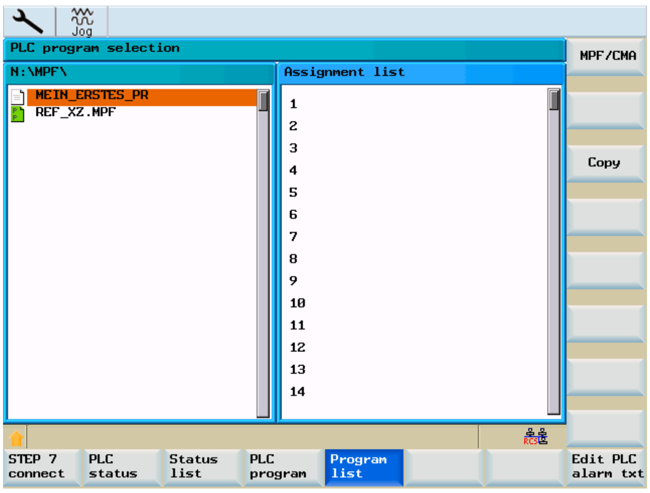


Figure 8-40 PLC program list

This dialog displays all files of the MPF directory and their assignment in the reference list (PLCPROG.LST) in the form of a list. You can use the TAB key to switch between the two columns. The Copy, Insert and Delete softkey functions are displayed with reference to a specific context. If the cursor is placed on the left-hand side, only the Copy function is available. On the right-hand side, the Insert and Delete functions are provided to modify the reference list.

List of references for interface signals

SINUMERIK 802D sl Function Manual; Various Interface Signals (A2)

SINUMERIK 802D sl List Manual

Copy

Writes the selected file name to the clipboard.

Paste

Pastes the file name at the current cursor position.

Delete

Deletes the selected file name from the assignment list.

Structure of the reference list (file PLCPROG.LST)

It is divided into 3 areas:

Number	range	Protection level
1 to 100	User area	User
101 to 200	Machine manufacturer	Machine manufacturer
201 to 255	Siemens	Siemens

The notation is carried out for each program by lines. Two columns are intended per line, which must be separated from each other by TAB, space or the "|" character. In the first column, the PLC reference number must be specified, and in the second column, the file name.

Example:

1 | shaft.mpf

2 | taper.mpf

Edit PLC alarm txt

This function can be used to insert or modify PLC user alarm texts. Select the desired alarm number using the cursor. At the same time, the text currently valid is displayed in the input line.

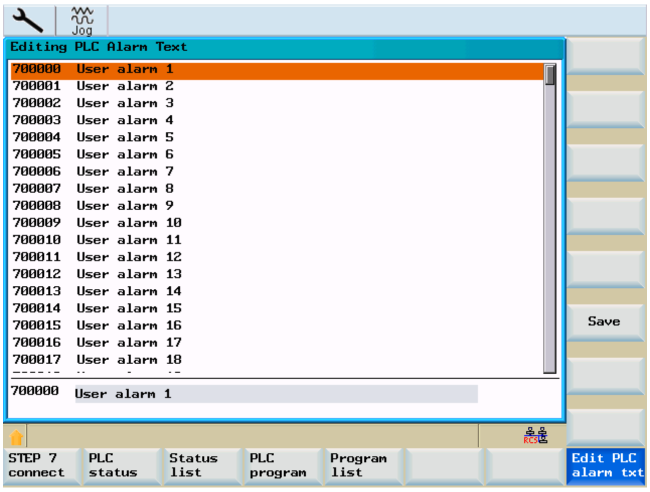


Figure 8-41 Editing the PLC alarm text

Enter the new text in the input line. Press the "Input" key to complete your input and select "Save" to save it.

For the notation of the texts, please refer to the operating instructions.

8.6 SYSTEM - "Start-up files" softkeys

Start-up files

The menu allows general files, commissioning archives and PLC projects to be created, read-out or read-in, copied, deleted etc.

This window displays the contents of the selected drive in a tree structure. The horizontal softkeys display the available drives for selection in the form of a list. The vertical softkeys provide the control functions possible for the drive in question.

There are the following permanently set drive assignments:

- 802D data: Commissioning data
- Customer CF card: Customer data on the CF card
- RCS connection: Data of a drive released on PC/PG via the the RCS tool (only for SINUMERIK 802D sl pro)
- RS232: Serial Interface
- Manufacturer drive: Data that the manufacturer specifically stored
- USB drive: Customer data on a USB FlashDrive
- Manufacturer archive: Commissioning data archived on the system CompactFlash Card

All data is handled using the "Copy & Paste" principle.

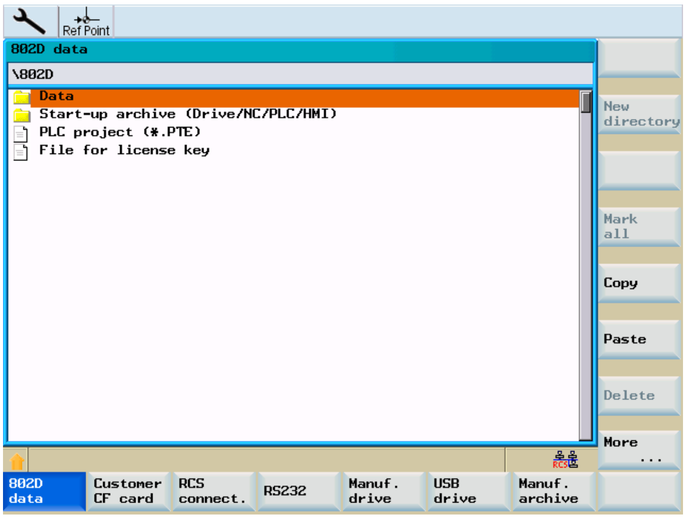


Figure 8-42 Start-up files

802D data

The individual data groups in the "802D data" area have the following significance:

Note

The sag compensation is ONLY listed if the associated function was activated.

- Data (in text format)

  These data are special initialization data and are transferred in an ASCII file.

  – Machine data
  – Setting data
  – Tool data
  – R parameters
  – Work offset
  – Leadscrew error compensation
  – Sag compensation
  – Global user data
- Commissioning archive (drive/NC/PLC/HMI)

  These data constitute a commissioning file for HMI data and are transferred in the binary format using the HMI archive format.

  – Drive machine data
  – NC data
  – NC directories
  – Display machine data
  – Leadscrew error compensation
  – Sag compensation
  – PLC project
  – HMI data and applications
- PLC project (*.PTE)

  A direct exchange between the control system and programming tool is possible without conversion with the support of PLC project handling in the programming tool export format.
- File for license key

Customer CF card

Reading-in and reading-out data on a CompactFlash Card (CF card).

RCS connect.

Reading-in and reading-out data to a PG/PC via a network. The RCS tool must be installed on the PG/PC (only for SINUMERIK 802D sl pro).

Note

The RCS tool provides a detailed online help function. Refer to this help menu for further details e.g. establishing a connection, project management etc.

RS232

Reading-in and reading-out data via the RS232 interface.

More ...

Error log

Note

Using the softkey function "Continue...", you may also inspect the transmission log. The "Error log" function is available for that.

Set-ting.

Use this function to display and change the RS232 interface parameters. Any changes in the settings come into effect immediately.

Selecting the "Save" softkey will save the selected settings even beyond switching off.

The "Default settings" softkey will reset all settings to their default settings.

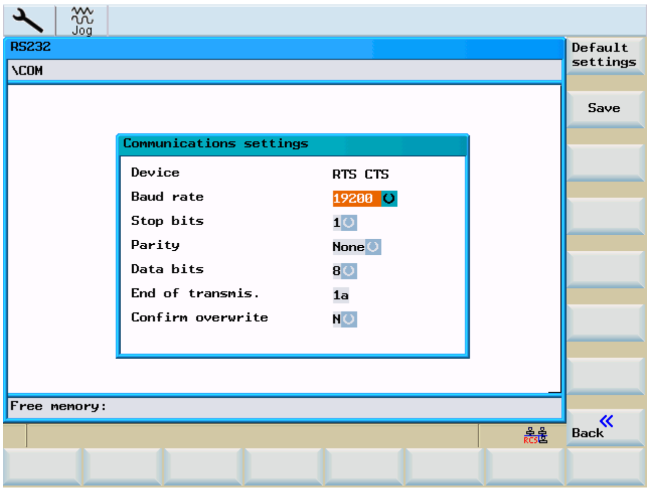


Figure 8-43 Parameters of the RS232 interface

Interface parameters

Table 8- 8 Interface parameters

Parameter	Description
Device type	**RTS CTS** The signal RTS (Request to Send) controls the send mode of the data transfer device. The CTS signal indicates the readiness to transmit data as the acknowledgment signal for RTS.
Baud rate	... used to set the interface transmission rate. 300 baud 600 baud 1200 baud 2400 baud 4800 baud 9600 baud 19200 baud 38400 baud 57600 baud 115200 baud
Stop bits	Number of stop bits with asynchronous transmission Input: 1 stop bit (default setting) 2 stop bits
Parity	Parity bits are used for error detection. These are added to the coded character to convert the number of digits set to "1" into an odd or even number. Input: No parity (default setting) Even parity Odd parity
Data bits	Number of data bits with asynchronous transmission Input: 7 data bits 8 data bits (default)
Overwriting with confirmation	Y: When reading in, it is checked whether the file already exists in the NC. N: The files are overwritten without confirmation warning.

Manufac-
turer drive

Reading-in and reading-out data of the manufacturer's directory "F".

USB
drive

Reading-in and reading-out data of a USB FlashDrive.

manu. archive

Use this function to create/restore a commissioning archive on/from the system CompactFlash Card.

No archive file has been created in the following display. The symbol for the zip archive sends a signal with an exclamation mark.

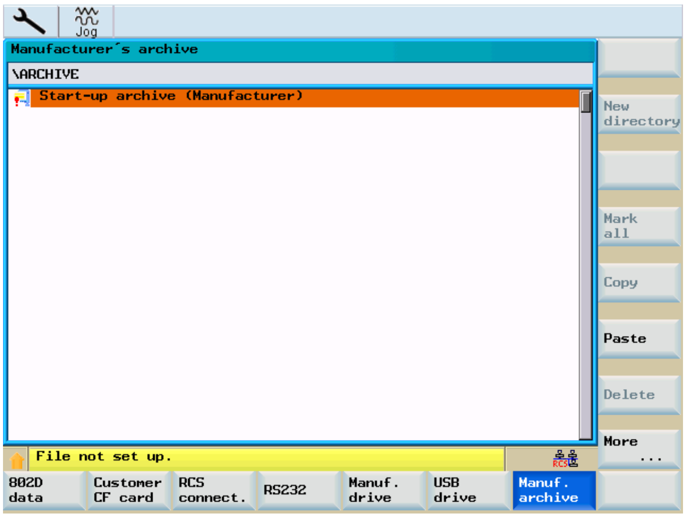


Figure 8-44 Manufacturers` archive, archive file not yet created

Vertical softkeys

The following vertical softkeys are available upon activating the file functions:

- "Rename": Use this function to rename a file selected beforehand using the cursor.
- "New directory": Creates a new directory
- "Copy": Use this softkey to copy one or more files to the clipboard.
- "Paste": Use this softkey to paste files or directories from the clipboard to the current directory.
- "Delete": Deletes the selected file name from the assignment list.
- "Select all": Use this softkey to select all files for subsequent operations.
- "Properties": Display memory capacity.
- "Job list": Displays a list with active file jobs and provides the option to terminate or display a file job.

More ...

Use this function to switch to the respective vertical softkeys.

Note

If individual functions are grayed out, then these functions are not available for the displayed drive/directory.

8.7 Alarm display

Operating sequence

The alarm window is opened. You can sort the NC alarms using softkeys; PLC alarms will not be sorted.

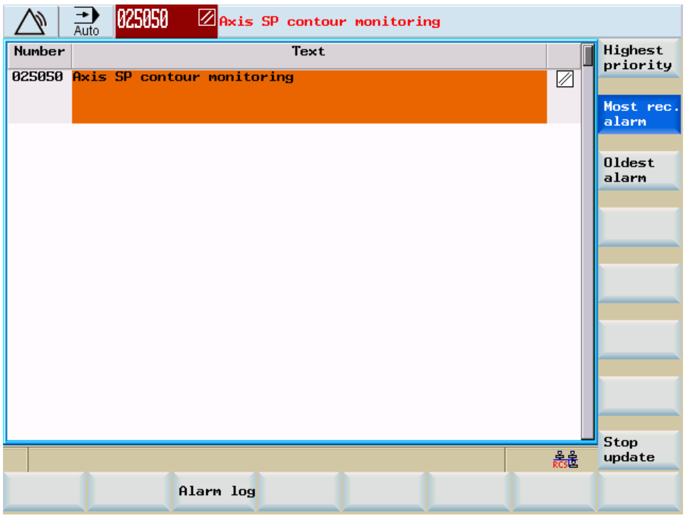


Figure 8-45 Alarm display window

Softkeys

Highest priority	Use this softkey to display all alarms sorted by their priorities. The highest priority alarm is at the beginning of the list.
Most rec. alarm	Use this softkey to display the alarms sorted by the time of their occurrence. The most recent alarm stands at the beginning of the list.
Oldest alarm	Use this softkey to display the alarms sorted by the time of their occurrence. The oldest alarm stands at the beginning of the list.
Stop updating	Updating of pending alarms is stopped / started.

Alarm log

All alarms are logged.

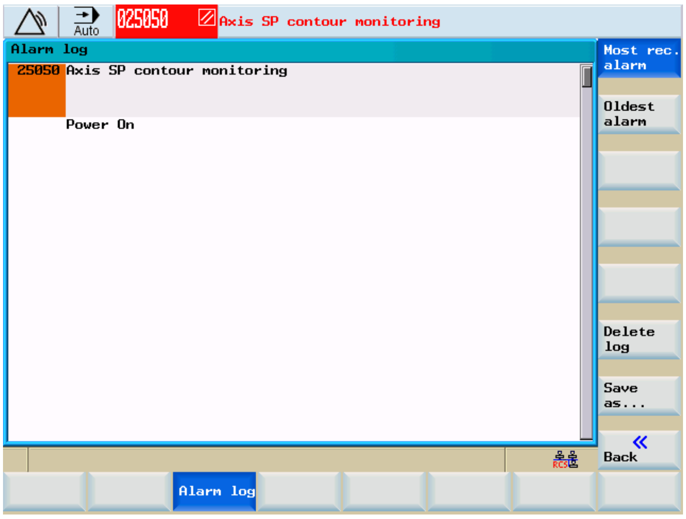


Figure 8-46 Alarm log

The log is deleted using softkey "Delete log".

Save under

The file is output using softkey "Save under..." on a CF card or on the USB FlashDrive.

Cycles 9

9.1 Overview of cycles

Cycles consist of technology subroutines.

You can use cycles for general implementation of a certain machining process, such as plunge-cut grinding, dressing, or longitudinal grinding.

These cycles are adapted to individual tasks by parameter assignment.

In principle, grinding involves two different types of technological sequence:

- Dressing the grinding wheel
- Grinding the workpiece

The dressing and grinding machining operations are achieved by means of NC cycles.

Grinding tools need to be dressed after a certain time in service to compensate for worn wheels and to restore their original profile.

Dressing of a wheel pursues two objectives:

- Profiling is aimed at achieving the required wheel shape.
- Sharpening is aimed at restoring the wheel's ability to cut.

The workpiece grinding cycles enable surface grinding with the Y/Z infeed axes (right-angled).

Grinding cycles

The SINUMERIK 802D sl controller supports the following cycles for surface grinding:

CYCLE406	Z positioning with grinding wheel
CYCLE407	Safety position approaching
CYCLE408	3-stage plunge-cut oscillation (roughing, finishing and fine-finishing)
CYCLE409	3-stage surface grinding (roughing, finishing and fine-finishing)
CYCLE416	Dressing and profiling
CYCLE426	Oscillating plunge cutting
CYCLE427	Surface grinding with continuous infeed
CYCLE428	Surface grinding with intermittent infeed
CYCLE429	Profile grinding
CYCLE430	Dressing with profile roller
CYCLE446	Selection of grinding wheel peripheral speed

The cycles are supplied with the tool box. They are loaded via the RS232 interface into the part program memory during the start-up of the control system.

Operation while a machining cycle is underway

All cycles are compatible with both automatic infeed and handwheel infeed. You have the option of interrupting the machining process while it is underway and forcing intermediate dressing.

Note

The possible operations (e.g. intermediate dressing) are assigned to the keys of the CNC machine control panel by the machine manufacturer.

Further processing then takes place on the basis of a preliminary dimensional value.

If a cycle is parameterized, then the icons above the parameter sketch will indicate the corresponding possible operations that are available (see figure below).

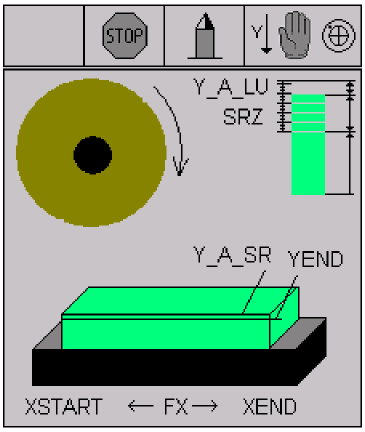
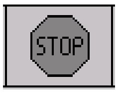
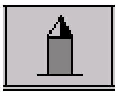


Figure 9-1 Parameter sketch for a cycle

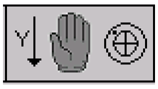
What the icons mean:



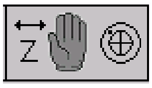
Interrupt cycle with adoption of retraction position and off-loading



Intermediate dressing with off-loading



Handwheel infeed possible in Y direction



Handwheel infeed possible in Z direction

Note

The functions associated with the icons (e.g. "interrupt cycle") remain effective all the time that there is an infeed amount available.

The "interrupt cycle" function also applies during air grinding.

9.2 Cycle requirements

Prerequisites

The following requirements must be met in respect of the individual grinding cycles:

1. The machining plane must be active.

   In principle, all planes are possible with surface grinding.

   However, G19 is selected by default, as it is the option required for dressing (Y/Z).

   This plane is also activated for the purpose of selecting the geometry axes.

   However, other planes (G17, G18) can be active within the context of the process.

   The relationship between the planes depends on the infeed axis and the relevant oscillating axis/axes.
2. A tool number and the associated tool offset must be programmed before the cycle is called.
3. The grinding wheel peripheral speed is calculated from the parameters that contain the peripheral speeds in CYCLE446.
4. The file called SGUD_MA.DEF contains default values for all the machine-specific M commands, such as those that relate to the direction of spindle rotation, coolant and measuring device switching.

   These values can be changed by the machine manufacturer.
5. In addition to assigning values for the grinding wheel and dresser offset data, the tool-specific grinding data must also be defined.
6. For example, in order to determine the initial infeed value, the computational resolution must be at least 10 times higher than the input resolution in the display machine data so that the start position can be calculated for surface grinding.

   The following machine data is used for the computational resolution:

   – Computational resolution

     MD10200 INT_INCR_PER_MM (computational resolution for linear positions) or
     MD10210 INT_INCR_PER_DEG (computational angular resolution)

   – Input resolution

     MD203 DISPLAY_RESOLUTION (display resolution) or
     MD204 DISPLAY_RESOLUTION_INCH (display resolution for inch measuring system)

9.3 Programming cycles

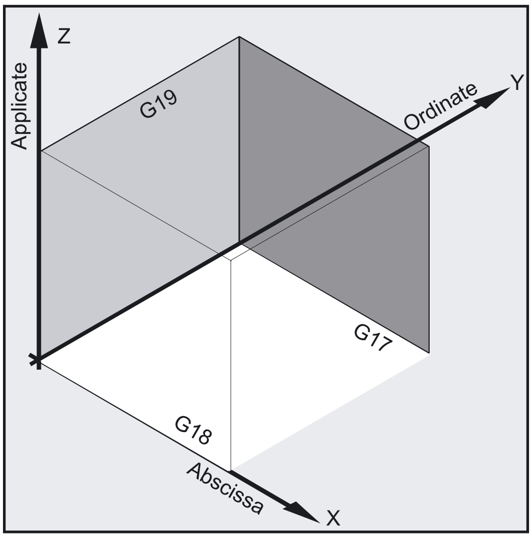
A cycle is defined as a subroutine with a name and parameter list assigned.

9.3.1 Call and return conditions

The G functions effective prior to the cycle call and the programmable offsets remain active beyond the cycle.

The machining level (G17, G18, G19) must be defined before calling the cycle. A cycle operates in the current plane with the following axes:

- 1st axis of the plane (abscissa)
- 2nd axis of the plane (ordinate)
- Tool axis/infeed axis, 3rd axis, standing vertically to the plane (applicate).



Level and axis allocation:

Command	Plane	Vertical infeed axis
`G17`	X/Y	Z
`G18`	Z/X	Y
`G19`	Y/Z	X

9.3.2 Error messages and error handling

9.3.2.1 General information

If error conditions are detected in the cycles, an alarm is generated and the execution of the cycle is aborted.

The cycles continue to output messages in the dialog line of the control. These message will not interrupt the program execution.

Reference

For more information on errors and required responses, as well as messages output in the controller's dialog line, please refer to the SINUMERIK 802D sl Diagnostics Manual.

9.3.2.2 Error handling within cycles

Alarms with numbers between 61000 and 62999 generated in the cycles. This range of numbers, in turn, is divided again with regard to alarm responses and cancel criteria. The error text that is displayed together with the alarm number gives you more detailed information on the error cause.

Alarm number	Clearing criterion	Alarm Response
61000 ... 61999	NC_RESET	Block preparation in the NC is aborted
62000 ... 62999	Clear key	Program execution is not interrupted; display only

9.3.3 Cycle call and parameter list

The cycles use user-defined variables. The defining parameters for the cycles can be transferred via the parameter list when the cycle is called.

Note

Cycle calls must always be programmed in a separate block.

Basic information on assigning parameters to cycles

The Programming Guide describes the parameter list of every cycle with the

- order and the
- type.

It is imperative to observe the order of the defining parameters.

Each defining parameter of a cycle has a certain data type. The parameter being used must be specified when the cycle is called. In this parameter list, the following can be transferred:

- R parameters
- Constants

If R parameters are used in the parameter list, they must first be assigned values in the calling program. Proceed as follows to call the cycles:

- With an incomplete parameter list or
- By leaving out parameters

If you want to exclude the last transfer parameters that have to be written in a call, you can prematurely terminate the parameter list with ")". If any parameters are to be omitted within the list, a comma "..., ,..." must be written as a placeholder.

Note

No plausibility checks are made of parameter values with a discrete or limited value range unless an error response has been specifically described for a cycle.

If a cycle is called the parameter list of which contains more entries than parameters defined in the cycle, the general NC alarm 12340 "Too many parameters" is displayed and the cycle is not executed.

Cycle call

The individual methods for writing a cycle are shown in the programming examples provided for the individual cycles.

Hardware prerequisites

The grinding machine must meet additional hardware requirements before any grinding cycles can be used.

One or two handwheels are required for motion overlay during setup.

There must be connection options for the following external equipment:

- Acoustic emission sensor
- Measurement control
- Touch trigger probe
- 7 rapid inputs via MCPA for:
  - Measurement control (5 inputs)
  - Acoustic emission sensor (2 inputs)

Call and return conditions

The grinding cycles are programmed independently of the actual axis names. The collions-free approach to the grinding position is to be done in the higher-level program before the cycle is called.

The required values for spindle speed and direction of spindle rotation must be programmed in the part program if there are no defining parameters in the grinding cycle.

The G functions active prior to the cycle call remain active beyond the cycle.

Coordinate systems for grinding

In general, CNC grinding machines have separate coordinate systems for grinding and dressing. The zero points of both coordinate systems must be defined once when setting up the machine.

The workpiece zero is defined by the operator when setting up the machine, by scratching the workpiece in all necessary axes. All additional geometric specifications for creating the automatic program refer to this zero point.

The dresser zero is defined during setup by scratching the wheel with the dressing diamonds. It serves as a reference point for the dressing program.

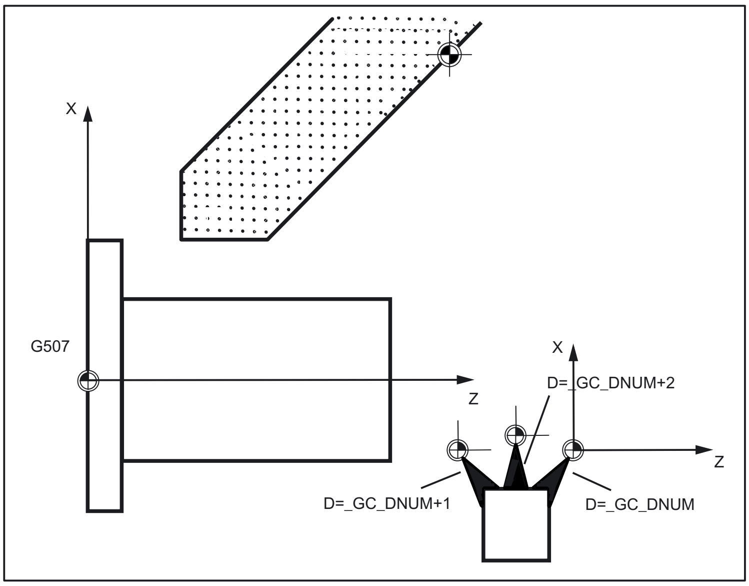


Figure 9-2 Coordinate systems for grinding

Plane definition

Before using the grinding cycles, G507 must be activated. Typically, the infeed axis is the first geometry axis.

A tool length compensation must be selected before the cycle is called. Tool length compensation is always effective in the selected plane and remains active even after the end of the cycle.

Types of grinding wheels

The cycles support two types of grinding wheels: vertical and inclined wheels.

During machining, the wheel feeds only in the Y or Z direction.

The use of measuring devices and sensors

When grinding, the following measuring devices/sensors can be used:

- Measuring probe
- Measurement control
- Acoustic emission sensor

Using a swiveling measuring probe, a longitudinal position in Z is detected. This axis position is stored on a parameter and aids in calculating the errors that occur in the compensation for each workpiece.

Measurement control is performed at the same time as the grinding machining on the workpiece diameter. The measurement control switches over the feedrate or defines the end position at the allowance coordinates in Y for roughing, finishing and fine finishing.

The acoustic emission sensor implements the feedrate stop when the workpiece diameter sparkles. Time-optimized approach conditions are created.

9.4 Zyklenunterstützung im Programmeditor

The program editor provides programming support for adding cycle calls to the program and for entering parameters.

Function

The cycle support offers the following functions:

- Cycle selection via soft keys
- Input screen forms for parameter assignment with help displays

Decompilable program code is generated from the individual screens.

Summary of required files

The following files constitute the basis for cycle support:

- sc.com
- cov.com

Note

These files are loaded during the start-up of the control system and must always remain loaded.

Operating the cycle support

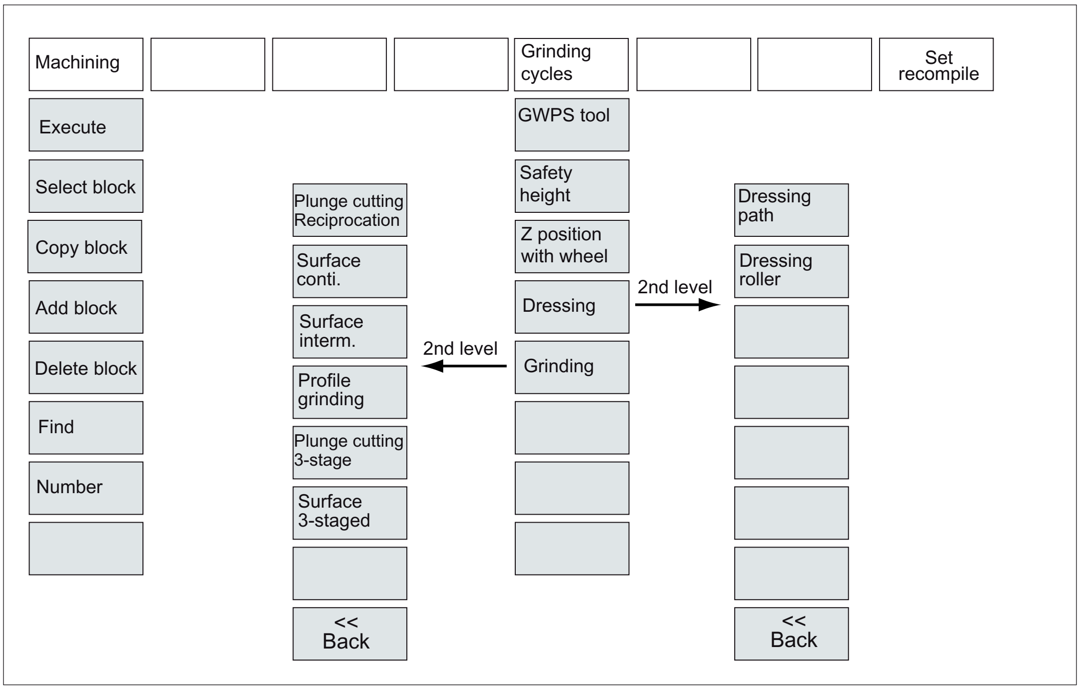


Figure 9-3 Menu tree for cycle support

To add a cycle call to the program, carry out the following steps one after the other:

- From the horizontal softkey bar, selection bars for the individual cycles can be selected using the <Grinding cycles> softkey.
- To select a cycle, keep using the vertical softkey bar until the appropriate input screen plus help display appears on the screen.
- Number values can be directly entered. Entered values are verified for proper range.
- Some parameters that may have only a few values are selected using the toggle key.
- Press <OK> to complete your input (or <Abort> in case of error).

Recompiling

Recompiling of program codes serves to make modifications to an existing program using the cycle support.

Position the cursor on the line to be modified and select the <Recompile> softkey. This will reopen the corresponding input screen from which the program piece has been created, and you can modify and accept the values.

9.5 Z positioning with grinding wheel - CYCLE406

Programming

CYCLE406(N_SITZ, CLEAR, CAL, Z_LPOS, MODE, Y_POS, Z_POS, ZSTW, A_Z, F_LU, F_SR, N_FR, FX, XSTART, XENDE)

Parameters

Table 9- 1 Parameter CYCLE406

Parameters	Data type	Meaning
N_SITZ	INT	Seat number
CLEAR	INT	Deletes the old offset prior to the approach
CAL	INT	Sets axis to Z position at end
Z_LPOS	INT	Retraction direction
MODE	INT	Type of approach: 0 = Sensor+Handwheel 1 = Sensor+ measurement 2 = Handwheel only 3 = Handwheel+measurement
Y_POS	REAL	Height
Z_POS	REAL	Z position for setting value
ZSTW	REAL	Z offset
A_Z	REAL	Z allowance after contact
F_LU	REAL	Feedrate for sparking, per stroke
F_SR	REAL	Grinding feedrate, per stroke
N_FR	REAL	Sparking-out strokes with oscillation
FX	REAL	X axis feedrate
XSTART	REAL	X axis starting position
XENDE	REAL	X axis end position

Function

This cycle is used for approaching and setting a Z position with the grinding wheel.

Sequence

The cycle moves to the Z preliminary position and commences the approach either with an optional acoustic emission sensor or just with the handwheel.

Once contact has been detected, grinding is performed, either in accordance with a handwheel value or in relation to the contact point.

Once the end point is reached, if the CAL parameter is set to "1", the Z axis will be set to the Z position.

If no acoustic emission sensor input has been configured, the cycle will immediately switch to the handwheel at the start of the operation, provided that an air grinding feedrate exists.

Sketch of the geometry parameters

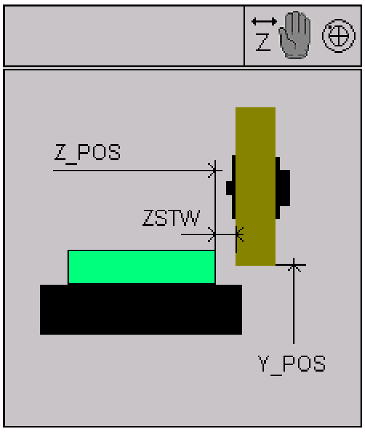


Figure 9-4 Z positioning with grinding wheel - CYCLE406

Programming example

Machining sequence:

- Inching at position -100.0000 mm to Y position -20.0000 mm, old offset is deleted first and axis value is set at the end point.
- The handwheel is used exclusively for the infeed.
- If an automatic infeed is selected, it will be implemented on a per-stroke basis.
- The start position is at -110.000 mm.
- The table feedrate is 30,000 mm/min between points -350 and 100 (450 oscillating path).
- Sparking-out strokes are performed when infeed is aborted.
- In the example below, which illustrates the next step, the sequence continues with a 3-stage oscillating plunge cutting cycle. This will always grind away 50 micrometers, because the axis has been calibrated to -100 mm in advance.

```
N10 T2D2
N20 CYCLE446( 20)
N30 CYCLE406( 0, 1, 1, -1, 2, -20, -100, 10, 0.1, 3, 0.1, 1, 30000, -350, 100)
N40 CYCLE408( 0, 100, -350, -20, -99.95, 0, 0, 0, 0, 0, -0.05, 0.02, 0.005, 0, 0.01,
0.002, 0.001, 1, 1, 5, 0, 1, 0, 0, 0.01, 30000, 30000, 30000, 30000)
N50 M30
```

9.6 Safety position - CYCLE407

Programming

CYCLE407(YS, STORE, KOORD)

Parameters

Table 9- 2 Parameters of CYCLE407

Parameter	Data type	Meaning
YS	REAL	Retraction position mm
STORE	INT	Stores position globally 0/1
KOORD	INT	Position in WCS=1 and in MCS=0

Function

This cycle is used for approaching a safety position during the grinding process or during interruptions such as intermediate dressing.

Sequence

The cycle checks the current position and approaches it if the infeed axis is smaller than the value entered.

The position is approached in accordance with the KOORD parameter in the workpiece or machine coordinate systems (basic coordinates).

Sketch of the geometry parameters

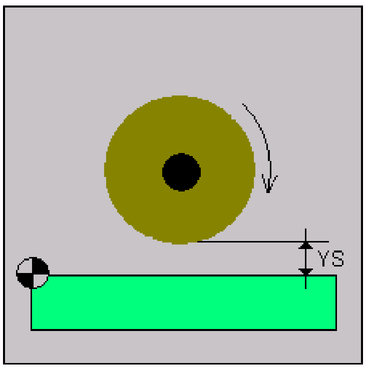


Figure 9-5 Safety position - CYCLE407

9.7 3-stage plunge cut oscillation (roughing, finishing and fine-finishing) - CYCLE408

Programming

CYCLE408(N_SITZ, XSTART, XEND, YEND ZEND, KS, Y_A_LU, Y_A_SR, Y_A_SL, Y_A_FS, Z_A_LU, Z_A_SR, Z_A_SL, Z_A_FS, LUZ, SRZ, SLZ, FSZ, N_SR, N_SL, N_FS, D_SR, D_SL, D_FS, ESL, EFS, FX_SR, FX_SL, FX_FS, FR)

Parameters

Table 9- 3 Parameters of CYCLE408

Parameter	Data type	Meaning
N_SITZ	INT	Seat number
XSTART	REAL	X axis starting position (abs)
XEND	REAL	X axis end position (abs)
YEND	REAL	Y position of wheel, finished dimension (abs)
ZEND	REAL	Z position of wheel (abs), finished dimension
KS	INT	Acoustic emission sensor
Y_A_LU	REAL	Air grinding allowance (incr.)
Y_A_SR	REAL	Roughing allowance (incr.)
Y_A_SL	REAL	Finishing allowance (incr.)
Y_A_FS	REAL	Fine-finishing allowance (incr.)
Z_A_LU	REAL	Air grinding allowance (incr.)
Z_A_SR	REAL	Roughing allowance (incr.)
Z_A_SL	REAL	Finishing allowance (incr.)
Z_A_FS	REAL	Fine-finishing allowance (incr.)
LUZ	REAL	Air grinding feedrate, per stroke
SRZ	REAL	Roughing feedrate, per stroke
SLZ	REAL	Finishing feedrate, per stroke
FSZ	REAL	Fine-finishing feedrate, per stroke
N_SR	INT	Sparking-out strokes following roughing
N_SL	INT	Sparking-out strokes following finishing
N_FS	INT	Sparking-out strokes following fine-finishing
D_SR	INT	Dressing strokes prior to roughing
D_SL	INT	Dressing strokes prior to finishing
D_FS	INT	Dressing strokes prior to fine-finishing
ESL	INT	Off-loading prior to finishing
EFS	INT	Off-loading prior to fine-finishing
FX_SR	REAL	X feedrate, roughing
FX_SL	REAL	X feedrate, finishing
FX_FS	REAL	X feedrate, fine-finishing
FR	REAL	Y feedrate/Z feedrate

Function

The 3-stage plunge-cut cycle is called for the purpose of machining a slot or surface that is smaller than the wheel width.

It can also be used for grinding front faces. The surface concerned is ground using the oscillation method.

Infeed takes place at the reversal points. Intermediate dressing, interruption and use of the handwheel are all supported.

The keys will only ever respond at the end point.

At the end, the tool retracts to the allowance position.

Infeed is possible in the Y or Z directions and the Z value can be either negative or positive.

A higher infeed amount can be used at the start of machining. Once contact has been detected using an acoustic emission sensor, the tool will retract by the acoustic emission sensor infeed amount minus the grinding infeed amount. If no acoustic emission sensor input has been configured, the cycle will immediately switch to the handwheel at the start of the operation.

The cycle automatically divides cutting into roughing, finishing and fine-finishing. Sparking-out strokes can be programmed for the end of each stage.

Off-loading is also possible at the end of the roughing and finishing stages.

Dressing strokes can be programmed to take place prior to each technology step.

Sequence

Traverse to allowance position.

Approach X starting position and Z position.

Start oscillating motion, infeed at reversal points.

The first infeed once oscillating motion has commenced is adjusted to ensure that all additional infeed operations correspond to the infeed amount. This process is also performed following interruptions, intermediate dressing and deselection of the handwheel override function.

Following interruption/dressing, an off-loading value is applied as the tool approaches the machining start point.

At the end, the tool retracts to the allowance position.

Sketch of the geometry parameters

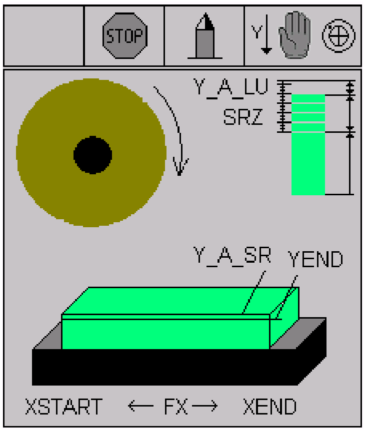


Figure 9-6 3-stage plunge cut workshop cycle (roughing, finishing and fine-finishing) - CYCLE408

Programming example - 1

Machining sequence:

- Plunge cut in Y direction at a peripheral speed of 20 m/s.
- In part program block N30: allowance of 0.1 mm, acoustic emission sensor, and 0.1 mm standard infeed, finishing reversing point 0.05 mm, fine-finishing reversing point 0.01 mm.
- Infeed of 0.03 mm per stroke with acoustic emission sensor and 0.02 mm infeed for roughing, 0.01 mm infeed for finishing, and 0.002 mm infeed for fine-finishing.
- 1 sparking-out stroke following roughing, 2 strokes following finishing, and 3 strokes following fine-finishing.
- Single-stroke dressing for finishing and fine-finishing respectively.
- Off-loading value of 0.03 mm following roughing and 0.015 mm following finishing.

```
N10 T1D1
N20 CYCLE446( 20)
N30 CYCLE408( 0, 100, -100, 0, 0, 0, 0.1, 0.1, 0.05, 0.01, 0.1, 0, 0.05, 0.01, 0.03,
0.02, 0.01, 0.002, 1, 2, 3, 0, 1, 1, 0.03, 0.015, 30000, 29000, 28000, 3000)
N40 M30
```

Programming example - 2

Machining sequence:

- Plunge cut in Z direction at a peripheral speed of 20 m/s.
- In part program block N30: allowance of 0.1 mm, acoustic emission sensor, and 0.1 mm standard infeed, finishing reversing point 0.05 mm, fine-finishing reversing point 0.01 mm.
- Infeed of 0.03 mm per stroke with acoustic emission sensor and 0.02 mm infeed for roughing, 0.01 mm infeed for finishing, and 0.002 mm infeed for fine-finishing.
- 1 sparking-out stroke following roughing, 2 strokes following finishing, and 3 strokes following fine-finishing.
- Single-stroke dressing for finishing and fine-finishing respectively.
- Off-loading value of 0.03 mm following roughing and 0.015 mm following finishing.

```
N10 T1D1
N20 CYCLE446( 20)
N30 CYCLE408( 0, 100, -100, 0, 0, 0, 0.1, 0, 0.05, 0.01, 0.1, -0.1, 0.05, 0.01,
0.03, 0.02, 0.01, 0.002, 1, 2, 3, 0, 1, 1, 0.03, 0.015, 30000, 29000, 28000, 3000)
N40 M30
```

9.8 3-stage surface grinding (roughing, finishing and fine-finishing) - CYCLE409

Programming

CYCLE409(N_SITZ, XSTART, XENDE, ZSTART, ZEND, YEND, B_SR, B_SL, B_FS, KS, Y_A_LU, Y_A_SR, Y_A_SL, Y_A_FS, LUZ, SRZ, SLZ, FSZ, N_SR, N_SL, N_FS, D_SR, D_SL, D_FS, ESL, EFS, FX_SR, FX_SL, FX_FS, FZ_SR, FZ_SL, FZ_FS, FY)

Parameters

Table 9- 4 Parameters of CYCLE409

Parameter	Data type	Meaning
N_SITZ	INT	Seat number
XSTART	REAL	X axis starting position (abs)
XEND	REAL	X axis end position (abs)
ZSTART	REAL	Z axis starting position (abs)
ZEND	REAL	Z axis end position (abs)
YEND	REAL	Finished dimension (abs.)
B_SR	INT	Roughing machining type: 0 - Continuous 1 - Intermittent
B_SL	INT	Finishing machining type: 0 - Continuous 1 - Intermittent
B_FS	INT	Fine-finishing machining type: 0 - Continuous 1 - Intermittent
KS	INT	Acoustic emission sensor
Y_A_LU	REAL	Air grinding allowance (incr.)
Y_A_SR	REAL	Roughing allowance (incr.)
Y_A_SL	REAL	Finishing allowance (incr.)
Y_A_FS	REAL	Fine-finishing allowance (incr.)
LUZ	REAL	Air grinding feedrate, per stroke
SRZ	REAL	Roughing feedrate, per stroke
SLZ	REAL	Finishing feedrate, per stroke
FSZ	REAL	Fine-finishing feedrate, per stroke
N_SR	INT	Sparking-out strokes following roughing
N_SL	INT	Sparking-out strokes following finishing
N_FS	INT	Sparking-out strokes following fine-finishing
D_SR	INT	Dressing strokes prior to roughing
D_SL	INT	Dressing strokes prior to finishing
D_FS	INT	Dressing strokes prior to fine-finishing

Parameter	Data type	Meaning
FX_SR	REAL	X feedrate, roughing
FX_SL	REAL	X feedrate, finishing
FX_FS	REAL	X feedrate, fine-finishing
FZ_SR	REAL	Z feedrate, roughing
FZ_SL	REAL	Z feedrate, finishing
FZ_S	REAL	Z feedrate, fine-finishing
FY	REAL	Y feedrate

Function

The 3-stage surface grinding cycle allows you to machine a surface that is wider than the wheel itself. The surface concerned is ground using the oscillation method/intermittently.

Y infeed takes place at the X and Z reversal points (left/rear, right/rear, left/front or right/front).

Intermediate dressing, interruption and use of the handwheel are all supported.

The keys will only ever respond at the end point.

At the end, the tool retracts to the starting position.

A higher infeed amount can be used at the start of machining.

Once contact has been detected using an acoustic emission sensor, the tool will retract by the acoustic emission sensor infeed amount minus the grinding infeed amount.

If no acoustic emission sensor input has been configured, the cycle will immediately switch to the handwheel at the start of the operation.

The cycle automatically divides cutting into roughing, finishing and fine-finishing.

Sparking-out strokes can be programmed for the end of each stage.

Off-loading is also possible at the end of the roughing and finishing stages.

Dressing strokes can be programmed to take place prior to each technology step.

Sequence

Approach allowance position, approach X starting position and Z starting position.

Start oscillating motion, infeed at reversal points.

The first infeed once oscillating motion has commenced is adjusted to ensure that all additional infeed operations correspond to the infeed amount. This process is also performed following interruptions, intermediate dressing and deselection of the handwheel override function.

The oscillating motion is not performed with the oscillation function, as this would cause problems in terms of residual path deletion, and would overload the PLC.

Following interruption/dressing, an off-loading value is applied as the tool approaches the machining start point.

At the end, the tool retracts to the starting position.

Sketch of the geometry parameters

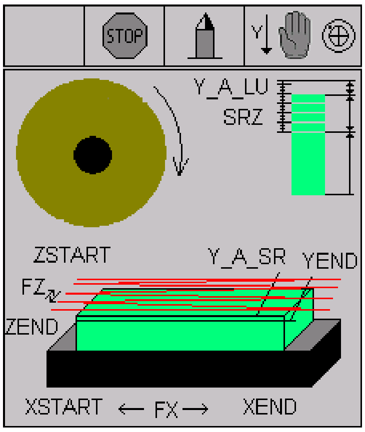


Figure 9-7 3-stage surface grinding workshop cycle (roughing, finishing and fine-finishing) - CYCLE409

Programming example

Machining sequence:

- Grind surface at a grinding wheel peripheral speed of 20 m/s.
- In part program block N30: air allowance of 0.05 mm and roughing allowance of 0.05 mm, finishing allowance of 0.025 mm, and fine-finishing allowance of 0.005 mm.
- Infeed of 0.03 mm per stroke with acoustic emission sensor and 0.02 mm infeed for roughing, 0.01 mm infeed for finishing, and 0.001 mm infeed for fine-finishing.
- 1 sparking-out stroke following roughing, 2 strokes following finishing, and 3 strokes following fine-finishing.
- Single-stroke dressing for finishing and fine-finishing respectively.
- 2 strokes prior to finishing, 1 stroke prior to fine-finishing.
- Off-loading value of 0.03 mm following roughing and 0.015 mm following finishing.
- With roughing, infeed is performed continuously in Z.
- With finishing and fine-finishing, infeed is performed intermittently.

```
N10 T2D1
N20 CYCLE446( 20)
N30 CYCLE409( 0, 100, -350, 0, -150, 10, 0, 1, 1, 0, 0.05, 0.05, 0.025, 0.005, 0.03,
0.02, 0.01, 0.001, 1, 2, 3, 0, 2, 1, 0.03, 0.015, 30000, 29000, 28000, 40, 35, 30,
3000)
N40 M30
```

9.9 Dressing and profiling - CYCLE416

Programming

CYCLE416(Y_AB, Z_AB_L, Z_AB_R, F_DL_AB, F_BL_AB, F_DR_AB, F_BR_AB, F_Z_AB, N_ABR, USCH, N_AWST)

Parameters

Table 9- 5 Parameters of CYCLE416

Parameter	Data type	Meaning
Y_AB	REAL	Dressing amount in Y (incr.)
Z_AB_L	REAL	Dressing amount in Z, left (incr.)
Z_AB_R	REAL	Dressing amount in Z, right (incr.)
F_DL_AB	REAL	Dressing feedrate in Y, left
F_BL_AB	REAL	Dressing feedrate in the path, left
F_DR_AB	REAL	Dressing feedrate in Y, right
F_BR_AB	REAL	Dressing feedrate in the path, right
F_Z_AB	REAL	Dressing feedrate in Z
N_ABR	INT	Number of dressing strokes
USCH	REAL	Grinding wheel peripheral speed
N_AWST	INT	Number of workpieces between two dressing operations

Function

This cycle is used for dressing wheels with a path dresser.

Following each dressing stroke, the dressed amount is taken into account in the wear parameters of the current tool offset.

Dressing is performed in accordance with workpiece counter _GC_WKS.

Sequence

Dressing involving setup cycles for the path dresser dressing method (CYCLE432).

Sketch of the geometry parameters

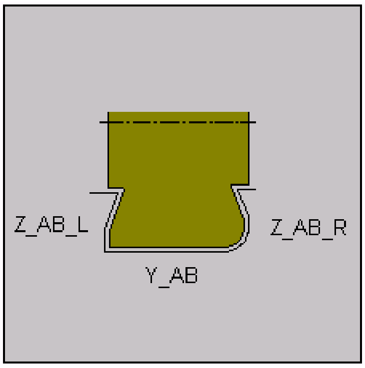


Figure 9-8 Dressing and profiling - CYCLE416

Programming example

Machining sequence:

- Single-stroke dressing with dressing amounts of 0.02 mm (left) and 0.01 mm (right) after every 5 workpieces.
- The number of idle strokes is stored in the tool data, along with the peripheral speed ratios of the spindles in the case of a rotating dresser.

```
N10 T1D1
N20 CYCLE416(0.02,0.01,0.01,0.2,0.2,0.2,0.2,0.2,1,35,5)
N30 M30
```

9.10 Oscillating plunge cutting - CYCLE426

Programming

CYCLE426(N_SITZ, XSTART, XEND, YEND, ZEND, ZU_ART, KS, Y_A_LU, Y_A_SR, Z_A_LU, Z_A_SR, LUZ, SRZ, N_FR, A_HEB, FR, FX)

Parameters

Table 9- 6 Parameters of CYCLE426

Parameter	Data type	Meaning
N_SITZ	INT	Seat number
XSTART	REAL	X axis starting position (abs)
XEND	REAL	X axis end position (abs)
YEND	REAL	Y position of wheel, finished dimension (abs)
ZEND	REAL	Z position of wheel (abs), finished dimension
ZU_ART	INT	Infeed left/right/both sides
KS	INT	Acoustic emission sensor
Y_A_LU	REAL	Air grinding allowance (incr.)
Y_A_SR	REAL	Allowance (incr.)
Z_A_LU	REAL	Air grinding allowance (incr.)
Z_A_SR	REAL	Allowance (incr.)
LUZ	REAL	Air grinding feedrate, per stroke
SRZ	REAL	Infeed per stroke
N_FR	INT	Sparking-out strokes
A_HEB	REAL	Retraction amount (incr.)
FR	REAL	Feedrate at reversal point
FX	REAL	Table feedrate

Function

The oscillating plunge cutting cycle is called for the purpose of machining a slot or surface that is smaller than the wheel width. It can also be used for grinding front faces. The surface concerned is ground using the oscillation method.

Infeed takes place at the reversal points. Intermediate dressing, interruption and use of the handwheel are all supported.

The keys will only ever respond at the end point.

At the end, the tool retracts or grinding continues with a different technology.

Infeed is possible in the Y or Z directions and the Z value can be either negative or positive.

Machining will take place in the Z direction if no roughing allowance has been programmed for Y.

A higher infeed amount can be used at the start of machining. Once contact has been detected using an acoustic emission sensor, the tool will retract by the acoustic emission sensor infeed amount minus the grinding infeed amount.

If no acoustic emission sensor input has been configured, the cycle will immediately switch to the handwheel at the start of the operation.

Sequence

Approach allowance position, approach X starting position and Z position.

Start oscillating motion, infeed at reversal points.

The first infeed once oscillating motion has commenced is adjusted to ensure that all additional infeed operations correspond to the infeed amount. This process is also performed following interruptions, intermediate dressing and deselection of the handwheel override function.

Following interruption/dressing, an off-loading value is applied as the tool approaches the machining start point.

At the end, the tool retracts by the retraction amount or machining continues using a new technology, assuming that the end point of the first machining operation is the same as the starting point for the new machining operation. In other words, if the oscillating axis is at the X axis end or start point and the allowance is correct, machining will continue from this point. As a result, it is possible to combine any number of technology seats.

Sketch of the geometry parameters

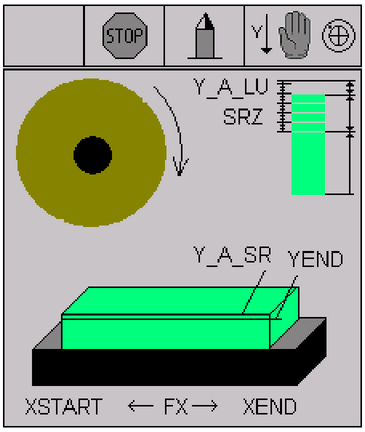


Figure 9-9 Oscillating plunge cutting - CYCLE426

Programming example - 1

Machining sequence:

- Plunge cut in Y direction at a grinding wheel peripheral speed of 20 m/s with two infeed blocks.
- In part program block N30: allowance of 0.1 mm and infeed of 0.005 mm per stroke, no retraction at the end and, therefore, grinding immediately continues in part program block N40 with an allowance of 0.02 mm and infeed of 0.002 mm per stroke.
- Both operations involve 2 sparking-out strokes. These are performed with the same geometry positions and feedrates.
- The fact that the start point of the second machining operation is precisely matched to an end point of the first machining operation means that infeed can take place straight away. In order for this to happen, grinding with an acoustic emission sensor must be deactivated.

```
N10 T1D1
N20 CYCLE446( 20)
N30 CYCLE426( 0, -350, 100, 10.500000, -95, 0, 0, 0, 0.100000, 0, 0, 0.005000,
0.005000, 2, 0, 1000, 30000)
N40 CYCLE426( 0, -350, 100, 10.480000, -95, 0, 0, 0, 0.020000, 0, 0, 0.005000,
0.002000, 2, 1, 1000, 30000)
M30
```

Programming example - 2

Machining sequence:

- Plunge cut in Z direction at a grinding wheel peripheral speed of 20 m/s with two infeed blocks.
- In part program block N30: allowance of 0.01 mm, acoustic emission sensor, and standard infeed of 0.01 mm and infeed of 0.005 mm per stroke with acoustic emission sensor at end point, plus infeed of 0.002 mm with grinding at end point, no retraction at the end and, therefore, grinding immediately continues in part program block N40 with an allowance of 0.02 mm and infeed of 0.002 mm per stroke.
- Both operations involve 2 sparking-out strokes. These are performed with the same geometry positions and feedrates.
- The fact that the start point of the second machining operation is precisely matched to an end point of the first machining operation means that infeed can take place straight away. In order for this to happen, grinding with an acoustic emission sensor must be deactivated.

```
N10 T2D1
N20 CYCLE446( 20)
N30 CYCLE426( 0, 100, -350, 10.5, -95, 1, 0, 0, 0.01, 0.01, 0.005, 0.002, 2, 0,
1000, 30000)
N40 CYCLE426( 0, -350, 100, 10.5, -95.02, 1, 0, 0, 0.01, 0.01, 0, 0.002, 2, 1,
1000, 30000)
M30
```

9.11 Surface grinding with continuous infeed - CYCLE427

Programming

CYCLE427(N_SITZ, XSTART, XEND, ZSTART, ZEND, YEND, ZU_ART, KS, A_LU, A_SR, LUZ, SRZ, N_FR, A_HEB, FY, FX, FZ)

Parameters

Table 9- 7 Parameter CYCLE427

Parameters	Data type	Meaning
N_SITZ	INT	Seat number
XSTART	REAL	X axis starting position (abs)
XEND	REAL	X axis end position (abs)
ZSTART	REAL	Z axis starting position (abs)
ZEND	REAL	Z axis end position (abs)
YEND	REAL	Finished dimension (abs.)
ZU_ART	INT	Infeed left/right/both sides
KS	INT	Acoustic emission sensor
A_LU	REAL	Air grinding allowance (incr.)
A_SR	REAL	Allowance (incr.)
LUZ	REAL	Air grinding feedrate, per stroke
SRZ	REAL	Infeed per stroke
N_FR	INT	Sparking-out strokes
A_HEB	REAL	Retraction amount (incr.)
FY	REAL	Feedrate at reversal point
FX	REAL	Table feedrate
FZ	REAL	Z axis infeed in X, per stroke

Function

The surface grinding cycle with continuous infeed is called for the purpose of machining a surface that is wider than the wheel itself. The surface concerned is ground using the oscillation method. Stroke infeed in X is performed continuously.

Y infeed takes place at the X and Z reversal points (left/rear, right/rear, left/front or right/front).

Intermediate dressing, interruption and use of the handwheel are all supported.

The keys will only ever respond at the end point.

At the end, the tool retracts or grinding continues with a different technology.

A higher infeed amount can be used at the start of machining.

Once contact has been detected using an acoustic emission sensor, the tool will retract by the acoustic emission sensor infeed amount minus the grinding infeed amount.

If no acoustic emission sensor input has been configured, the cycle will immediately switch to the handwheel at the start of the operation.

Sequence

Approach allowance position, approach X starting position and Z starting position.

Start oscillating motion, infeed at reversal points.

The first infeed once oscillating motion has commenced is adjusted to ensure that all additional infeed operations correspond to the infeed amount. This process is also performed following interruptions, intermediate dressing and deselection of the handwheel override function. The oscillating motion is not performed with the oscillation function, as this would cause problems in terms of residual path deletion, and would overload the PLC.

Following interruption/dressing, an off-loading value is applied as the tool approaches the machining start point.

At the end, the tool retracts by the retraction amount or machining continues at the end point using a new technology. In other words, if the oscillating axis is at the X axis end or start point and the allowance is correct, machining will continue from this point. As a result, it is possible to combine any number of technology seats.

Sketch of the geometry parameters

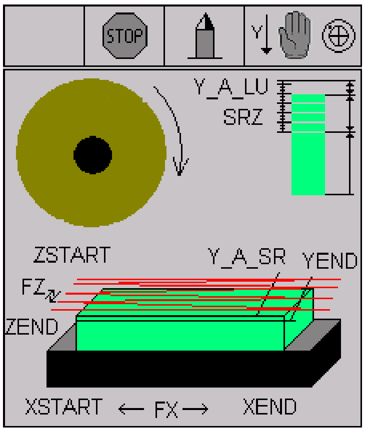


Figure 9-10 Surface grinding with continuous infeed - CYCLE427

Programming example

Machining sequence

- Oscillation with continuous infeed in Z direction at a grinding wheel peripheral speed of 20 m/s with two infeed blocks.
- In part program block N30: allowance of 0.1 mm and infeed of 0.005 mm per stroke, no retraction at the end and, therefore, grinding immediately continues in part program block N40 with an allowance of 0.02 mm and infeed of 0.002 mm per stroke.
- Both operations involve 2 sparking-out strokes. These are performed with the same geometry positions.
- The per-stroke feedrate in X is 40 mm for the 1st operation and 10 mm for the 2nd.
- The fact that the start point of the second machining operation is precisely matched to an end point of the first machining operation means that infeed can take place straight away. In order for this to happen, grinding with an acoustic emission sensor must be deactivated.

```
N10 T2D1
N20 CYCLE446( 20)
N30 CYCLE427( 0, 100, -350, -95, 0, 10.5, 0, 0, 0, 0.1, 0, 0.005, 2, 0, 3000, 30000,
40)
N40 CYCLE427( 0, 100, -350, -95, 0, 10.48, 0, 0, 0, 0.02, 0, 0.002, 2, 1, 3000,
30000, 10)
N50 M30
```

9.12 Surface grinding with intermittent infeed - CYCLE428

Programming

CYCLE428(N_SITZ, XSTART, XEND ZSTART, ZEND, YEND, ZU_ART, ZZU_ART, KS, A_LU, A_SR, LUZ, SRZ, N_FR, A_HEB, FY, FX, FZ)

Parameters

Table 9- 8 Parameter CYCLE428

Parameters	Data type	Meaning
N_SITZ	INT	Seat number
XSTART	REAL	X axis starting position (abs)
XEND	REAL	X axis end position (abs)
ZSTART	REAL	Z axis starting position (abs)
ZEND	REAL	Z axis end position (abs)
YEND	REAL	Finished dimension (abs.)
ZU_ART	INT	Infeed: - left - right - both sides
ZZU_ART	INT	Intermittent infeed: - left - right - both sides
KS	INT	Acoustic emission sensor
A_LU	REAL	Air grinding allowance (incr.)
A_SR	REAL	Allowance (incr.)
LUZ	REAL	Air grinding feedrate, per stroke
SRZ	REAL	Infeed per stroke
N_FR	INT	Sparking-out strokes
A_HEB	REAL	Retraction amount (incr.)
FY	REAL	Feedrate at reversal point
FX	REAL	Table feedrate
FZ	REAL	Z axis infeed in X, per stroke

Function

The surface grinding cycle with intermittent infeed is called for the purpose of machining a surface that is wider than the wheel itself. The surface concerned is ground using the oscillation method. Infeed is performed at the end of the strokes in X.

This infeed is circular so that sudden acceleration can be avoided.

With this method, it is also possible to select the infeed side in the Z direction.

Y infeed takes place at the X and Z reversal points (left/rear, right/rear, left/front or right/front).

Intermediate dressing, interruption and use of the handwheel are all supported.

The keys will only ever respond at the end point.

At the end, the tool retracts or grinding continues with a different technology.

A higher infeed amount can be used at the start of machining. Once contact has been detected using an acoustic emission sensor, the tool will retract by the acoustic emission sensor infeed amount minus the grinding infeed amount. If no acoustic emission sensor input has been configured, the cycle will immediately switch to the handwheel at the start of the operation.

The value of the FZ parameter (infeed per stroke) can be used to determine whether the X values should be used for the start or the end of the circular motion. If the values used are negative, then the X positions will not be overtraveled, even with circular interpolation.

Sequence

Traverse to allowance position. Approach X starting position and Z starting position.

Start oscillating motion, infeed at reversal points.

The first infeed once oscillating motion has commenced is adjusted to ensure that all additional infeed operations correspond to the infeed amount. This process is also performed following interruptions, intermediate dressing and deselection of the handwheel override function.

The oscillating motion is not performed with the oscillation function, as this would cause problems in terms of residual path deletion, and would overload the PLC.

Following interruption/dressing, an off-loading value is applied as the tool approaches the machining start point.

At the end, the tool retracts by the retraction amount or machining continues at the end point using a new technology. In other words, if the oscillating axis is at the X axis end or start point and the allowance is correct, machining will continue from this point. As a result, it is possible to combine any number of technology seats.

Sketch of the geometry parameters

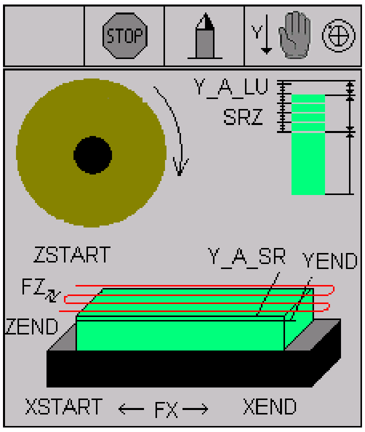


Figure 9-11 Surface grinding with intermittent infeed - CYCLE428

Programming example

Machining sequence:

- Oscillation with continuous infeed in Z direction at a grinding wheel peripheral speed of 20 m/s with two infeed blocks.
- In part program block N30: allowance of 0.1 mm and infeed of 0.005 mm per stroke, no retraction at the end and, therefore, grinding immediately continues in part program block N40 with an allowance of 0.02 mm and infeed of 0.002 mm per stroke.
- Both operations involve 2 sparking-out strokes. These are performed with the same geometry positions.
- The per-stroke feedrate in X is 40 mm for the 1st operation and 10 mm for the 2nd.
- The fact that the start point of the second machining operation is precisely matched to an end point of the first machining operation means that infeed can take place straight away. In order for this to happen, grinding with an acoustic emission sensor must be deactivated.

```
N10 T2D1
N20 CYCLE446( 20)
N30 CYCLE428( 0, 100, -350, -95, 0, 10.500000, 0, 0, 0, 0, 0.1, 0, 0.005000, 2, 0,
3000, 30000, 40)
N40 CYCLE428( 0, 100, -350, -95, 0, 10.48, 0, 0, 0, 0, 0.02, 0, 0.002, 2, 1, 3000,
30000, 10)
N50 M30
```

9.13 Profile grinding - CYCLE429

Programming

CYCLE429(N_SITZ, XSTART, ZSTART, YEND, KS, A_LU, A_SR, LUZ, SRZ, N_FR, A_HEB, FX, KONTUR)

Parameters

Table 9- 9 Parameters of CYCLE429

Parameter	Data type	Meaning
N_SITZ	INT	Seat number
XSTART	REAL	X axis starting position (abs)
ZSTART	REAL	Z axis starting position (abs)
YEND	REAL	Y position at starting position of Z axis (abs) and X axis (abs)
KS	INT	Acoustic emission sensor
A_LU	REAL	Air grinding allowance (incr.)
A_SR	REAL	Allowance (incr.)
LUZ	REAL	Air grinding feedrate, per stroke
SRZ	REAL	Infeed per stroke
N_FR	INT	Sparking-out strokes
A_HEB	REAL	Retraction amount (incr.)
FX	REAL	Table feedrate
KONTUR	String	Contour name

Function

The profile grinding cycle is called for the purpose of machining a surface with a free contour.

The entire contour is contained within the contour subroutine.

The cycle merely handles the infeed process and the interrupt keys, which do not have any effect until the end of the contour is reached.

Both intermediate dressing and interruption are supported.

At the end of each stroke, retraction is performed followed by a return to the starting point.

Machining must be performed using active tool nose radius compensation (G41/G42).

The tool offset is modified by the cycle.

The user has the option of programming both an outward and a return motion in the free contour. In this case, no retraction amount is required.

The user must select the correct G group (G41/G42) in the contour program, because this program also happens to contain the machining direction.

Sequence

Enter allowance in work offset.

Traverse to retraction amount.

Traverse to X starting position and Z starting position.

Call contour program.

Retract and reset work offset.

Approach the start point again until the allowance has been machined.

At the end, the tool retracts by the retraction amount.

Machining continues at the end point using a new technology.

Sketch of the geometry parameters

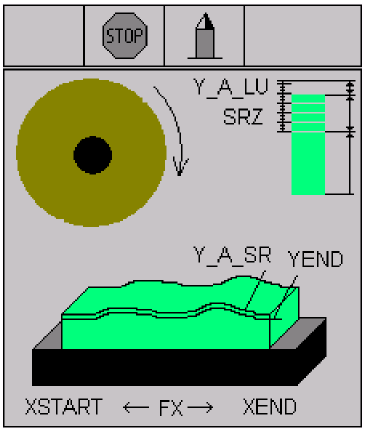


Figure 9-12 Profile grinding - CYCLE429

Programming example

Machining sequence:

- Profile grinding with an allowance of 0.02 mm and an infeed of 0.005 mm per stroke.
- The surface offset is used for the infeed. The contour is always programmed to the finished dimension.
- 2 sparking-out strokes are then performed at the end.
- Retraction should be programmed as follows:
  - It should be possible to perform a return journey in absolute safety.
  - The return motion is contained within the contour.
- In the example below, subsequent machining is carried out with a different infeed but the same contour.
- The contour must be stored as an individual subroutine. The direction of tool nose radius compensation is also activated in the contour.

```
N10 T1D1
N20 CYCLE446( 20)
CYCLE429( 0, 100, 0, 0, 0, 0, 0.02, 0, 0.005, 2, 20, 10000, "K123")
N30 Y0
N40 CYCLE429( 1, 100, 0, 0, 0, 0.02, 0, 0.005, 2, 20, 10000, "K123")
N50 M30
```

Example contour program

```
%_N_K123_SPF
G17
G42
G64 G90 X100 Y0
G64 X0 Y0
G64 X-100 Y10
G64 X-210 Y0
G64 X-260
RET
```

9.14 Dressing with profile roller - CYCLE430

Programming

CYCLE430(Y_AB, F_TVOR, F_VOR, N_AUS, N_ABR, USCH, N_AWST)

Parameters

Table 9- 10 Parameters of CYCLE430

Parameter	Data type	Meaning
Y_AB	REAL	Dressing amount in Y (incr.)
F_TVOR	REAL	Insertion stroke in mm/rev
F_VOR	REAL	Dressing feedrate in mm/rev
N_AUS	REAL	Coasting revolutions
N_ABR	INT	Number of dressing strokes
USCH	REAL	Grinding wheel peripheral speed
N_AWST	INT	Number of workpieces between two dressing operations

Function

This cycle is used for dressing wheels using a profile roller.

Following dressing (each stroke), the dressed amount is taken into account in the wear parameters of the current tool offset.

Dressing is performed in accordance with workpiece counter GC_WKS.

Sequence

If a profiling allowance is specified, this is processed first. This value can also be used to search for a dresser if no sensor system is installed.

When processing the profiling allowance, no dresser wear compensation is currently applied.

The profiling allowance is taken into account in the base dimension of the dresser when selecting the valid coordinate system. This does away with the requirement for a programmable work offset for grinding operations. The coasting revolutions are the number of revolutions taken for the roller to come to a stop against the wheel.

Sketch of the geometry parameters

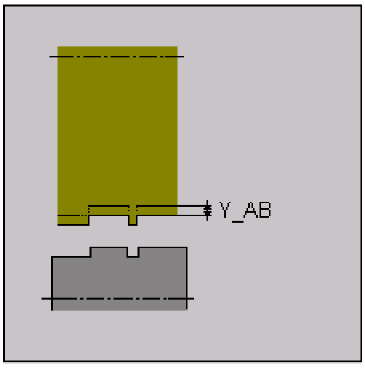


Figure 9-13 Dressing with profile roller - CYCLE430

Programming example

Machining sequence:

- 2-stroke dressing with a dressing amount of 0.02 mm and 2 sparking-out revolutions every 5 workpieces.
- The number of idle strokes is stored in the tool data, along with the peripheral speed ratios of the spindles.

```
N10 T1D1
N20 CYCLE430(0.02,20,0.2,2,2,35,5)
N30 M30
```

9.15 Selection of the grinding wheel peripheral speed - CYCLE446

Programming

CYCLE446(SUG)

Parameter

Table 9- 11 Parameters of CYCLE446

Parameter	Data Type	Meaning
GWPS	REAL	Value of the grinding wheel peripheral speed

Function

This function is used to switch on the grinding wheel at a desired peripheral wheel speed, including the testing of the max. peripheral wheel speed and RPM. If the speed is exceeded, a message is issued (no alarm). The value is limited to the respective maximum value. This is checked for all wheels that are mounted on the spindle (wheels of a set). A setup menu is also required in order to obtain an overview of the wheels used.

Checking and calculation is performed on the currently largest diameter of the wheels. This is a purely calculated monitoring function. Internally, no limitations are set that implement reliable monitoring. This must be ensured by the user.

For machines without NC spindles, it is possible to use a computation of the necessary speed with a spindle number ≤ 0 if the cycle CYCLE425 is available. In this case, the CYCLE425 receives the computed and limited speed. At this point, the user can give this speed to groups or directly to an external actuator (M functions, etc.). The user must then assign the speed set, which may deviate from the required speed, to parameter _GC_PARR[5]. In this way, the dressing cycle can compute, for example, the necessary dressing feedrate in mm/rev using the correct speed.

Programming 10

10.1 Fundamental principles of NC programming

10.1.1 Program names

Each program has its own program name. The name can be freely chosen during program creation, taking the following conventions into account:

- The first two characters must be letters.
- Use only letters, digits or underscore.
- Do not use delimiters (see Section "Character set").
- The decimal point must only be used for separation of the file extension.
- Do not use more than 27 characters.

Example: WORKPIECE

10.1.2 Program structure

Structure and content

The NC program consists of a sequence of blocks (see Table below).

Each block represents a machining step.

Instructions are written in the blocks in the form of words.

The last block in the execution sequence contains a special word for the end of program: e.g. M2.

Table 10- 1 NC program structure

Set	Word	Word	Word	...	; Comment
Set	N10	G0	X20	...	; 1. Set
Set	N20	G2	Z37	...	; 2. Set
Set	N30	G91	...	...	; ...
Set	N40	...	...	...	
Set	N50	M2			; End of program

10.1.3 Word structure and address

Functionality/structure

A word is a block element and mainly constitutes a control command. The word consists of:

- **Address characters**: generally a letter and a
- **Numerical value**: a sequence of digits which with certain addresses can be added by a sign put in front of the address, and a decimal point.

  A positive sign (+) can be omitted.

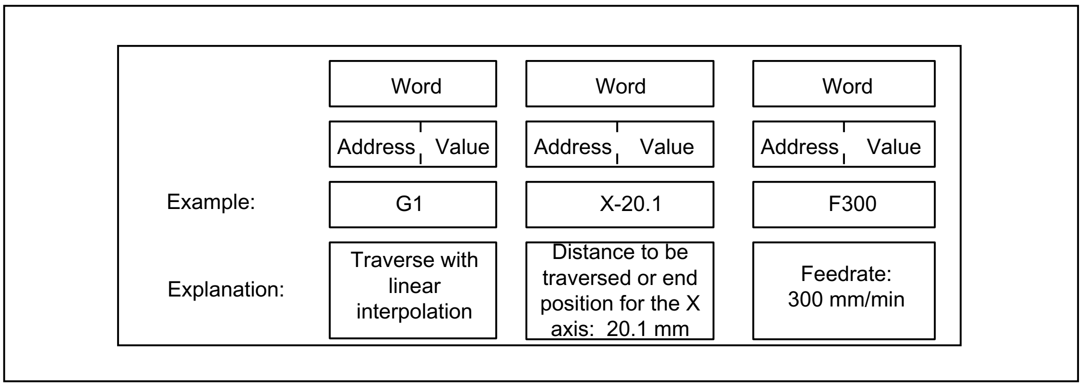


Figure 10-1 Word structure (example)

Several address characters

A word can also contain several address letters. In this case, however, the numerical value must be assigned via the intermediate character **"="**.

Example: **CR=5.23**

Additionally, it is also possible to call G functions using a symbolic name (see also Section "List of instructions").

Example: **SCALE** ; Enable scaling factor

Extended address

For the following addresses, the address is extended by 1 to 4 digits to obtain a higher number of addresses.

R: Arithmetic parameters

H: H function

I, J, K: Interpolation parameters/intermediate point

In this case, the value must be assigned using an equality sign "=" (see also Section "List of instructions").

Example: **R10=6.234 H5=12.1 I1=32.67**

10.1.4 Block format

Functionality

A block should contain all data required to execute a machining step.

Generally, a block consists of several **words** and is always completed with the **end-of-block character** " L_F " (Linefeed). This character is automatically generated when pressing the linefeed or <Input>key when writing.

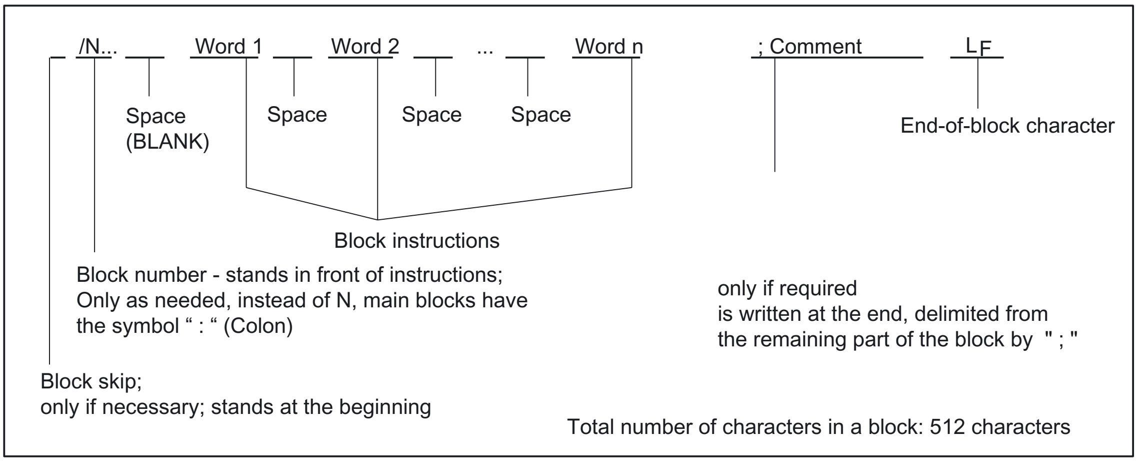


Figure 10-2 Block structure diagram

Word order

If there are several instructions in a block, the following order is recommended:
N... G... X... Z... F... S... T... D... M... H...

Note regarding block numbers

First select the block numbers in steps of 5 or 10. Thus, you can later insert blocks and nevertheless observe the ascending order of block numbers.

Block skip

Blocks of a program, which are to be executed not with each program run, can be **marked** by a slash / in front of the block number.
The block skip itself is activated via **Operation** (program control: "SKP") or by the programmable controller (signal). A section can be skipped by several blocks in succession using " / ".
If a block must be skipped during program execution, all program blocks marked with " / " are not executed. All instructions contained in the blocks concerned will not be considered. The program is continued with the next block without marking.

Comment, remark

The instructions in the blocks of a program can be explained using comments (remarks). A comment always starts with a semicolon " ; " and ends with end-of-block.
Comments are displayed together with the contents of the remaining block in the current block display.

Messages

Messages are programmed in a separate block. A message is displayed in a special field and remains active until a block with a new message is executed or until the end of the program is reached. Max. 65 characters can be displayed in message texts.
A message without message text cancels a previous message.
MSG("THIS IS THE MESSAGE TEXT")

See also chapter "Service MSG".

Programming example

```
N10                                        ; G&S company, order no. 12A71
N20                                        ; Pump part 17, drawing no.: 123 677
N30                                        ; Program created by H. Adam, Dept. TV 4
N40 MSG("DRAWING NO.: 123677")
:50 G54 F4.7 S220 D2 M3                    ;Main block
N60 G0 G90 X100 Z200
N70 G1 Z185.6
N80 X112
/N90 X118 Z180                             ; Block can be suppressed
N100 X118 Z120
N110 G0 G90 X200
N120 M2                                    ; End of program
```

10.1.5 Fonts

The following characters are used for programming; they are interpreted in accordance with the relevant definitions.

Letters, digits

A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W X, Y, Z

0, 1, 2, 3, 4, 5, 6, 7, 8, 9

No distinction is made between upper and lower case letters.

Printable special characters

(	Open parenthesis	"	Inverted commas
)	Close parenthesis	_	Underscore (belongs to letters)
[	Open square bracket	.	Decimal point
]	Close square bracket	,	Comma, separator
<	less than	;	Comment start
>	greater than	%	Reserved; do not use
:	Main block, end of label	&	Reserved; do not use
=	Assignment, part of equation	'	Reserved; do not use
/	Division, block suppression	$	System variable identifiers
*	Multiplication	?	Reserved; do not use
+	Addition and positive sign	!	Reserved; do not use
-	Subtraction, minus sign		

Non-printable special characters

LF: End-of-block character

Blank: Separator between words; blank

Tabulator: Reserved; do not use

10.1.6 List of instructions

Functions available with SINUMERIK 802D sl plus and pro

Address	Meaning	Value assignments	Information	Programming
D	Tool offset number	0 ... 9only integer, no sign	Contains offset data for a certain tool T... ; D0 à offset values= 0, max. 9 D numbers per tool	D...
F	Feed	0.001 ... 99 999.999	Path velocity of a tool/workpiece; unit: mm/min or mm/revolution depending on G94 or G95	F...
F	Dwell time (block with G4)	0.001 ... 99 999.999	Dwell time in seconds	G4 F...; separate block
G	G function (preparatory function)	Only integer, specified values	The G functions are divided into G groups. Only one G function of a group can be programmed in a block. A G function can be either modal (until it is canceled by another function of the same group) or only effective for the block in which it is programmed (non-modal).	G... or symbolic name, e.g.: CIP
			G group:	
G0	Linear interpolation at rapid traverse rate		1: Motion commands	G0 X... Z...
G1 *	Linear interpolation at feedrate		(type of interpolation)	G1 X...Z... F...
G2	Circular interpolation clockwise			G2 X... Z... I... K... F... ; Center and end point G2 X... Z... CR=... F... ; Radius and end point G2 AR=... I... K... F... ; Aperture angle and center point G2 AR=... X... Z... F... ; Aperture angle and end point
G3	Circular interpolation counter-clockwise			G3 ... ; otherwise as for G2
CIP	Circular interpolation through intermediate point			CIP X... Z... I1=... K1=... F... ;I1, K1 is intermediate point
CT	Circular interpolation; tangential transition			N10 ... N20 CT Z... X... F... ; circle; tangential transition to the previous path segment N10
G4	Dwell time		2: Special motions, dwell time non-modal	G4 F...;separate block, F: Time in seconds or G4 S.... ;separate block, S: in spindle revolutions

Address	Meaning	Value assignments	Information	Programming
G74	Reference point approach			G74 X1=0 Z1=0 ;separate block, (machine axis identifier!)
G75	Fixed point approach			G75 X1=0 Z1=0 ;separate block, (machine axis identifier!)
TRANS	translation, programmable		3: Write memory	TRANS X... Z... ;separate block
SCALE	Programmable scaling factor		non-modal	SCALE X... Z... ; scaling factor in the direction of the specified axis, separate block
ROT	rotation, programmable			ROT RPL=... ;rotation in the current plane G17 to G19, separate block
MIRROR	Programmable mirroring			MIRROR X0 ; coordinate axis whose direction is changed, separate block
ATRANS	additive translation, programming			ATRANS X... Z... ; separate block
ASCALE	Additive programmable scaling factor			ASCALE X... Z... ; scaling factor in the direction of the specified axis, separate block
AROT	additive programmable rotation			AROT RPL=... ; rotation in the current plane G17 to G19, separate block
AMIRROR	additive programmable mirroring			AMIRROR X0 ; coordinate axis whose direction is changed, separate block
G25	Lower spindle speed limitation or lower working area limitation			G25 S... ;separate block G25 X... Z... ;separate block
G26	Upper spindle speed limitation or upper working area limitation			G26 S... ; separate block G26 X... Z... ; separate block
G17	X/Y plane		6: Plane selection	
G18 *	Z/X plane			
G19	Y/Z plane			
G40 *	Tool radius compensation OFF		7: Tool radius compensation modally effective	
G41	Tool radius compensation left of contour			
G42	Tool radius compensation right of contour			
G500	Settable work offset OFF		8: Settable zero offset modally effective	
G54	1st Settable zero offset			
G55	2nd Settable zero offset			
G56	3rd Settable zero offset			

Address	Meaning	Value assignments	Information	Programming
G57	4th Settable zero offset			
G58	5th Settable zero offset			
G59	6th Settable zero offset			
G53	Non-modal skipping of the settable work offset		9: Skipping of the settable work offset non-modal	
G153	Non-modal skipping of the settable work offset including base frame			
G60 *	Exact stop		10: Approach behavior	
G64	Continuous-path mode		modally effective	
G9	Non-modal exact stop		11: Non-modal exact stop non-modal	
G601 *	Exact stop window, fine, with G60, G9		12: Exact stop window	
G602	Exact stop window, coarse, with G60, G9		modally effective	
G70	Inch dimension input		13: Inch / metr.dimension input	
G71 *	Metric dimension data input		modally effective	
G700	Inch dimension data input; also for feedrate F			
G710	Metric dimension data input; also for feedrate F			
G90 *	Absolute dimension data input		14: Absolute / incremental dimension	
G91	Incremental dimension		modally effective	
G94 *	Feed F in mm/min		15: Feedrate / spindle	
G95	Feedrate F in mm/spindle revolutions		modally effective	
G96	Constant cutting rate ON (F in mm/rev., S in m/min)			G96 S... LIMS=... F...
G97	Constant cutting speed OFF			
G450 *	Transition circle		18: Behavior at corners when working with tool radius compensation	
G451	Point of intersection		modally effective	
BRISK *	Jerking path acceleration		21: Acceleration profile	
SOFT	Jerk-limited path acceleration		modally effective	
FFWOF *	Feedforward control OFF		24: Precontrol	
FFWON	Feedforward control ON		modally effective	
WALIMON *	Working area limitation ON		28: Working area limitation modally effective	; applies to all axes activated via setting data; values set via G25, G26
WALIMOF	Working area limitation OFF			
G290 *	SIEMENS mode		47: External NC languages	
The functions marked with an asterisk (*) act when starting the program (in the default condition of the control system, unless otherwise programmed and if the machine manufacturer has preserved the default settings for the grinding technology).				

Address	Meaning	Value assignments	Information	Programming
H H0= to H9999=	H function	± 0.0000001 ... 9999 9999 (8 decimal places) or with specification of an exponent: ± (10-300 ... 10+300)	Value transfer to the PLC; meaning defined by the machine manufacturer	H0=... H9999=... e.g.: H7=23.456
I	Interpolation parameters	±0.001 ... 99 999.999 Thread: 0.001 ... 2000.000	Belongs to the X axis; meaning dependent on G2,G3 ->circle center or G33, G34, G35 G331, G332 à thread pitch	See G2, G3 and G33, G34, G35
K	Interpolation parameters	±0.001 ... 99 999.999 Thread: 0.001 ... 2000.000	Belongs to the Z axis; otherwise, as with I	See G2, G3 and G33, G34, G35
I1=	Intermediate point for circular interpolation	±0.001 ... 99 999.999	Belongs to the X axis; specification for circular interpolation with CIP	See CIP
K1=	Intermediate point for circular interpolation	±0.001 ... 99 999.999	Belongs to the Z axis; specification for circular interpolation with CIP	See CIP
L	Subroutine; name and call	7 decimals; integer only, no sign	Instead of a free name, it is also possible to select L1 ...L9999999; this also calls the subroutine (UP) in a separate block, Please note: L0001 is not always equal to L1. The name "LL6" is reserved for the tool change subroutine.	L.... ;separate block
M	Additional function	0 ... 99 only integer, no sign	For example, for initiating switching actions, such as "Coolant ON"; max. 5 M functions per block	M...
M0	Programmed stop		The machining is stopped at the end of a block containing M0; to continue, press NC START.	
M1	Optional stop		As with M0, but the stop is only performed if a special signal (Program control: "M01") is present.	
M2	End of program		Can be found in the last block of the processing sequence	
M30	-		Reserved; do not use	
M17	-		Reserved; do not use	
M3	CW rotation of spindle (for master spindle)			
M4	CCW rotation of spindle (for master spindle)			
M5	Spindle stop (for master spindle)			

Address	Meaning	Value assignments	Information	Programming
Mn=3	CW rotation of spindle (for spindle n)		n = 1 or = 2	M2=3 ; CW rotation stop for spindle 2
Mn=4	CCW rotation of spindle (for spindle n)		n = 1 or = 2	M2=4 ; CCW rotation stop for spindle 2
Mn=5	Spindle stop (for spindle n)		n = 1 or = 2	M2=5 ; Spindle stop for spindle 2
M6	Tool change		Only if activated with M6 via the machine control panel; otherwise, change directly using the T command	
M40	Automatic gear stage switching (for master spindle)			
Mn=40	Automatic gear stage switching (for spindle n)		n = 1 or = 2	M1=40 ; automatic gear stage ; for spindle 1
M41 to M45	Gear stage 1 to gear stage 5 (for master spindle)			
Mn=41 to Mn=45	Gear stage 1 to gear stage 5 (for spindle n)		n = 1 or = 2	M2=41; 1st gear stage for spindle 2
M70, M19	-		Reserved; do not use	
M...	Remaining M functions		Functionality is not defined by the control system and can therefore be used freely by the machine manufacturer	
N	Block number - subblock	0 ... 9999 9999 only integer, no sign	Can be used to identify blocks with a number; is written in the beginning of a block	N20
:	Block number of a main block	0 ... 9999 9999 only integer, no sign	Special block identification, used instead of N... ; such a block should contain all instructions for a complete subsequent machining step.	:20
P	Number of subroutine passes	1 ... 9999 only integer, no sign	Is used if the subroutine is run several times and is contained in the same block as the call	L781 P... ;separate block N10 L871 P3 ; three cycles
R0 to R299	Arithmetic parameters	± 0.0000001 ... 9999 9999 (8 decimal places) or with specification of an exponent: ± (10-300 ... 10+300)		R1=7.9431 R2=4 with specification of an exponent: R1=-1.9876EX9; R1=-1 987 600 000
Arithmetic functions			In addition to the 4 basic arithmetic functions using the operands + - * /, there are the following arithmetic functions:	
SIN( )	sinusoidal	Degrees		R1=SIN(17.35)
COS()	Cosine	Degrees		R2=COS(R3)
TAN()	Tangent	Degrees		R4=TAN(R5)

Address	Meaning	Value assignments	Information	Programming
ASIN()	Arc sine			R10=ASIN(0.35) ; R10: 20.487 degrees
ACOS()	Arc cosine			R20=ACOS(R2) ; R20: ... Degrees
ATAN2(,)	Arctangent2		The angle of the sum vector is calculated from 2 vectors standing vertically one on another. The 2nd vector specified is always used for angle reference. Result in the range: -180 to +180 degrees	R40=ATAN2(30.5,80.1) ; R40: 20.8455 degrees
SQRT()	Square root			R6=SQRT(R7)
POT()	Square			R12=POT(R13)
ABS()	Absolute value			R8=ABS(R9)
TRUNC()	Truncate to integer			R10=TRUNC(R2)
LN()	Natural logarithm			R12=LN(R9)
EXP()	Exponential function			R13=EXP(R1)
RET	Subroutine end		Used instead of M2 - to maintain the continuous-path control mode	RET ;separate block
S...	Spindle speed (master spindle)	0.001 ... 99 999.999	Unit of measurement of the spindle r.p.m.	S...
S1=...	Spindle speed for spindle 1	0.001 ... 99 999.999	Unit of measurement of the spindle r.p.m.	S1=725 ; speed 725 r.p.m. for spindle 1
S2=...	Spindle speed for spindle 2	0.001 ... 99 999.999	Unit of measurement of the spindle r.p.m.	S2=730 ; speed 730 r.p.m. for spindle 2
S	Cutting rate with G96 active	0.001 ... 99 999.999	Cutting rate unit m/min with G96; for master spindle only	G96 S...
S	Dwell time in block with G4	0.001 ... 99 999.999	Dwell time in spindle revolutions	G4 S... ;separate block
T	Tool number	1 ... 32 000 only integer, no sign	The tool change can be performed either directly using the T command or only with M6. This can be set in the machine data.	T...
X	Axis	±0.001 ... 99 999.999	Positional data	X...
Z	Axis	±0.001 ... 99 999.999	Positional data	Z...
AC	Absolute coordinate	-	The dimension can be specified for the end or center point of a certain axis, irrespective of G91.	N10 G91 X10 Z=AC(20) ;X - incremental dimension, Z - absolute dimension
ACC[*axis*]	Percentage acceleration override	1 ... 200, integer	Acceleration override for an axis or spindle; specified as a percentage	N10 ACC[X]=80 ;for the X axis 80% N20 ACC[S]=50;for the spindle: 50%

Address	Meaning	Value assignments	Information	Programming
ACP	Absolute coordinate; approach position in the positive direction (for rotary axis, spindle)	-	It is also possible to specify the dimensions for the end point of a rotary axis with ACP(...) irrespective of G90/G91; also applies to spindle positioning	N10 A=ACP(45.3) ; Approach absolute position of the A axis in the positive direction N20 SPOS=ACP(33.1); Position spindle
ACN	Absolute coordinate; approach position in the negative direction (for rotary axis, spindle)	-	It is also possible to specify the dimensions for the end point of a rotary axis with ACN(...) irrespective of G90/G91; also applies to spindle positioning	N10 A=ACN(45.3) ; Approach absolute position of the A axis in the negative direction N20 SPOS=ACN(33.1); Position spindle
ANG	Angle for the specification of a straight line for the contour definition	±0.00001 ... 359.99999	Specified in degrees; one possibility of specifying a straight line when using G0 or G1 if only one end-point coordinate of the plane is known or if the complete end point is known with contour ranging over several blocks	N10 G1 X... Z.... N11 X... ANG=... or contour over several blocks: N10 G1 X... Z... N11 ANG=... N12 X... Z... ANG=...
AR	Aperture angle for circular interpolation	0.00001 ... 359.99999	Specified in degrees; one possibility of defining the circle when using G2/G3	See G2, G3
CALL	Indirect cycle call	-	Special form of the cycle call; no parameter transfer; the name of the cycle is stored in a variable; only intended for cycle-internal use	N10 CALL VARNAME ; variable name
CHF	Chamfer; general use	0.001 ... 99 999.999	Inserts a chamfer of the specified chamfer length between two contour blocks	N10 X... Z.... CHF=... N11 X... Z...
CHR	Chamfer; in the contour definition	0.001 ... 99 999.999	Inserts a chamfer of the specified leg length between two contour blocks	N10 X... Z.... CHR=... N11 X... Z...
CR	Radius for circular interpolation	0.010 ... 99 999.999 Negative sign - for selecting the circle: greater than semicircle	One possibility of defining a circle when using G2/G3	See G2, G3
CYCLE...	Machining cycle	Only specified values	The call of the machining cycles requires a separate block; the appropriate transfer parameters must be loaded with values. Special cycle calls are also possible with an additional MCALL or CALL.	
CYCLE406	Z positioning with grinding wheel			N10 CYCLE406(...) ; separate block

Address	Meaning	Value assignments	Information	Programming
CYCLE407	Safety position			N10 CYCLE407(...) ; separate block
CYCLE408	3-stage plunge cut workshop cycle (roughing, finishing and fine-finishing)			N10 CYCLE408(...) ; separate block
CYCLE409	3-stage surface grinding workshop cycle (roughing, finishing and fine-finishing)			N10 CYCLE409(...) ; separate block
CYCLE426	Oscillating plunge cutting			N10 CYCLE426(...) ; separate block
CYCLE427	Surface grinding with continuous infeed			N10 CYCLE427(...) ; separate block
CYCLE428	Surface grinding with intermittent infeed			N10 CYCLE428(...) ; separate block
CYCLE429	Profile grinding			N10 CYCLE429(...) ; separate block
CYCLE430	Dressing with profile roller			N10 CYCLE430(...) ; separate block
DC	Absolute coordinate; approach position directly(for rotary axis, spindle)	-	It is also possible to specify the dimensions for the end point of a rotary axis with DC(...) irrespective of G90/G91; also applies to spindle positioning	N10 A=DC(45.3) ;Approach absolute position of the A axis directly N20 SPOS=DC(33.1); Position spindle
DEF	Definition instruction		Defining a local user variable of the type BOOL, CHAR, INT, REAL, directly at the beginning of the program	DEF INT VARI1=24, VARI2 ; 2 variables of the type INT ; name defined by user
FRC	Non-modal feedrate for chamfer/rounding	0, >0	In case FRC=0: Feedrate Fwill act	For the unit, see F and G94, G95; for chamfer/rounding, see CHF, CHR, RND
FRCM	Modal feedrate for chamfer/rounding	0, >0	In case FRCM=0: Feedrate Fwill act	For the unit, see F and G94, G95; for rounding/modal rounding, see RND, RNDM
FXS [*axis*]	Travel to fixed stop	=1: select =0: Deselection	*Axis:* Use the machine identifier	N20 G1 X10 Z25 FXS[Z1]=1 FXST[Z1]=12.3 FXSW[Z1]=2 F...
FXST [*axis*]	Clamping torque, travel to fixed stop	> 0.0 ... 100.0	in %, max. 100% from the max. torque of the drive, *axis:* Use the machine identifier	N30 FXST[Z1]=12.3
FXSW [*axis*]	Monitoring window, travel to fixed stop	> 0.0	Unit of measurement mm or degrees, axis-specific, *axis:* Use the machine identifier	N40 FXSW[Z1]=2.4
GOTOB	GoBack instruction	-	A GoTo operation is performed to a block marked by a label; the jump destination is in the direction of the program start.	N10 LABEL1: ... ... N100 GOTOB LABEL1

Address	Meaning	Value assignments	Information	Programming
GOTOF	GoForward instruction	-	A GoTo operation is performed to a block marked by a label; the jump destination is in the direction of the end of the program.	N10 GOTOF LABEL2 ... N130 LABEL2: ...
IC	Coordinate specified using incremental dimensions	-	The dimension can be specified for the end or center point of a certain axis irrespective of G90.	N10 G90 X10 Z=IC(20) ;Z - incremental dimension, X - absolute dimension
IF	Jump condition	-	If the jump condition is fulfilled, the GoTo operation to the block with the following *label is performed;* , otherwise, the next instruction/block will follow.In one block, several IF instructions are possible. Relational operators: = = equal, <> not equal > greater than, < less than >= greater than or equal to <= less than or equal to	N10 IF R1>5 GOTOF LABEL3 ... N80 LABEL3: ...
LIMS	Upper limit speed of the spindle with G96, G97	0.001 ... 99 999.999	Limits the spindle speed with the G96 function enabled - constant cutting rate and G97	See G96
MEAS	Measurement with deletion of distance-to-go	+1 -1	=+1: Measuring input 1, rising edge =-1: Measuring input1, falling edge	N10 MEAS=-1 G1 X... Z... F...
MEAW	Measurement without deletion of distance-to-go	+1 -1	=+1: Measuring input 1, rising edge =-1: Measuring input1, falling edge	N10 MEAW=1 G1 X... Z... F...
$A_DBB[n] $A_DBW[n] $A_DBD[n] $A_DBR[n]	Data byte Data word Data double-word Real data		Reading and writing PLC variables	N10 $A_DBR[5]=16.3 ; Write Real variables ; with offset position 5 ; (position, type and meaning are agreed between NC and PLC)
$AA_FXS [*axis*]	Status, travel to fixed stop	-	Values: 0 ... 5 *Axis*: Machine axis identifier	N10 IF $AA_FXS[X1]==1 GOTOF ....
$AA_MM [*axis*]	Measurement result for an axis in the machine coordinate system	-	*Axis*: Identifier of an axis (X, Z) traversing when measuring	N10 R1=$AA_MM[X]
$AA_MW [*axis*]	Measurement result for an axis in the workpiece coordinate system	-	*Axis*: Identifier of an axis (X, Z) traversing when measuring	N10 R2=$AA_MW[X]

Address	Meaning	Value assignments	Information	Programming
$AC_MEA [1]	Measuring job status	-	Default condition: 0: Default condition, probe did not switch 1: Probe switched	N10 IF $AC_MEAS[1]==1 GOTOF .... ; Continue program when probe has switched ...
$A..._..._ TIME	Timer for run time: $AN_SETUP_TIME $AN_POWERON_TIME $AC_OPERATING_TIME $AC_CYCLE_TIME $AC_CUTTING_TIME	0.0 ... 10+300 min (read only value) min (read only value) s s s	System variable: Time since the control system has last booted Time since the control system has last booted normally Total runtime of all NC programs Runtime of the NC program (only of the selected program) Tool action time	N10 IF $AC_CYCLE_TIME==50.5 ....
$AC_..._ PARTS	Workpiece counter: $AC_TOTAL_PARTS $AC_REQUIRED_PARTS $AC_ACTUAL_PARTS $AC_SPECIAL_PARTS	0 ... 999 999 999, integer	System variable: Total actual count Set number of workpiece Current actual count Count of workpieces - specified by the user	N10 IF $AC_ACTUAL_PARTS==15 ....
$AC_ MSNUM	Number of the active master spindle		read-only	
$P_ MSNUM	Number of programmed master spindle		Read-only	
$P_NUM_ SPINDLES	Number of configured spindles		Read-only	
$AA_S[n]	Actual speed of spindle n		Spindle number n =1 or =2, read-only	
$P_S[n]	Last programmed speed of spindle n		Spindle number n =1 or =2, read-only	
$AC_ SDIR[n]	Current direction of rotation of spindle n		Spindle number n =1 or =2, read-only	
$P_ SDIR[n]	Last programmed direction of rotation of spindle n		Spindle number n =1 or =2, read-only	
$P_ TOOLNO	Number of the active tool T	-	read-only	N10 IF $P_TOOLNO==12 GOTOF ....
$P_TOOL	Active D number of the active tool	-	read-only	N10 IF $P_TOOL==1 GOTOF ....
MSG ()	Message	max. 65 characters	Message text in inverted commas	MSG("MESSAGE TEXT") ; separate block ... N150 MSG() ; Clear previous message

Address	Meaning	Value assignments	Information	Programming
RND	Rounding	0.010 ... 99 999.999	Inserts a rounding with the specified radius value tangentially between two contour blocks	N10 X... Z.... RND=... N11 X... Z...
RNDM	Modal rounding	0.010 ... 99 999.999 0	- Inserts roundings with the specified radius value tangentially at the following contour corners; special feedrate possible:FRCM= ... - Modal rounding OFF	N10 X... Y.... RNDM=.7.3 ;modal rounding ON N11 X... Y... .... N100 RNDM=.0 ;modal rounding OFF
RPL	Angle of rotation with ROT, AROT	±0.00001 ... 359.9999	Specification in degrees; angle for a programmable rotation in the current plane G17 to G19	see ROT, AROT
SET(, , ,) REP()	Set values for the variable fields		SET: Various values, from the specified element up to: according to the number of values REP: the same value, from the specified element up to the end of the field	DEF REAL VAR2[12]=REP(4.5) ; all elements value 4.5 N10 R10=SET(1.1,2.3,4.4) ; R10=1.1, R11=2.3, R4=4.4
SETMS(n) SETMS	Define spindle as master spindle	n= 1 or n= 2	n: Number of the spindle, if only SETMS is set, the default master spindle comes into effect	N10 SETMS(2) ; separate block, 2nd spindle = master
SF	Thread starting point when using G33	0.001 ... 359.999	Specified in degrees; the thread starting point with G33 will be offset by the specified value	See G33
SPI(n)	converts the spindle number n into the axis identifier		n =1 or =2, axis identifier: e.g. "SP1" or "C"	
SPOS SPOS(n)	spindle position	0.0000 ... 359.9999	specified in degrees; the spindle stops at the specified position (to achieve this, the spindle must provide the appropriate technical prerequisites: position control) Spindle number n: 1 or 2	N10 SPOS=.... N10 SPOS=ACP(...) N10 SPOS=ACN(...) N10 SPOS=IC(...) N10 SPOS=DC(...)
STOPFIFO	Stops the fast machining step		Special function; filling of the buffer memory until STARTFIFO, "Buffer memory full" or "End of program" is detected.	STOPFIFO; separate block, start of filling N10 X... N20 X...
STARTFIFO	Start of fast machining step		Special function; the buffer memory is filled at the same time.	N30 X... STARTFIFO ;separate block, end of filling
STOPRE	Preprocessing stop		Special function; the next block is only decoded if the block before STOPRE is completed.	STOPRE ; separate block
G05	Activates oblique plunge-cutting		Can only be set with inclined axis (TRAANG)	G05 X...
G07	Approach starting position		Can only be set with inclined axis (TRAANG)	G07 X... Z...

10.2 Positional data

10.2.1 Programming dimensions

In this section you will find descriptions of the commands, with which you can directly program dimensions taken from a drawing. This has the advantage that no extensive calculations have to be made for NC programming.

Note

The commands described in this section stand in most cases at the start of a NC program. The way, in which these functions are combined, is not intended to be a patent remedy. For example, the choice of working plane may be made at another point in the NC program. The real purpose of this and the following sections is to illustrate the conventional structure of an NC program.

Overview of typical dimensions

The basis of most NC programs is a drawing with concrete dimensions.

When implementing in a NC program, it is helpful to take over exactly the dimensions of a workpiece drawing into the machining program. These can be:

- Absolute dimension, G90 modally effective applies for all axes in the block, up to revocation by G91 in a following block.
- Absolute dimension, X=AC(value) only this value applies only for the stated axis and is not influenced by G90/G91. This is possible for all axes and also for SPOS, SPOSA spindle positionings, and interpolation parameters I, J, K.
- Absolute dimension, X=CC(value) directly approaching the position by the shortest route, only this value applies only for the stated rotary axis and is not influenced by G90/G91. Is also possible for SPOS, SPOSA spindle positionings.
- Absolute dimension, X=ACP(value) approaching the position in positive direction, only this value is set only for the rotary axis, the range of which is set to 0... < 360 degrees in the machine data.
- Absolute dimension, X=ACN(value) approaching the position in negative direction, only this value is set only for the rotary axis, the range of which is set to 0... < 360 degrees in the machine data.
- Incremental dimension, G91 modally effective applies for all axes in the block, until it is revoked by G90 in a following block.
- Incremental dimension, X=IC(value) only this value applies exclusively for the stated axis and is not influenced by G90/G91. This is possible for all axes and also for SPOS, SPOSA spindle positionings, and interpolation parameters I, J, K.
- Inch dimension, G70 applies for all linear axes in the block, until revoked by G71 in a following block.
- Metric dimension, G71 applies for all linear axes in the block, until revoked by G70 in a following block.

- Inch dimension as for G70, but applies also for feedrate and length-related setting data.
- Metric dimension as for G71, but applies also for feedrate and length-related setting data.

10.2.2 Plane selection: G17 to G19

Functionality

To assign, for example, tool radius and tool length compensations, a plane with two axes is selected from the three axes X, Y and Z. In this plane, you can activate a tool radius compensation.

For drill and cutter, the length compensation (length1) is assigned to the axis standing vertically on the selected plane (see Section "Tool and tool offsets"). It is also possible to use a 3-dimensional length compensation for special cases.

Another influence of plane selection is described with the appropriate functions (e.g. Section "Support for the contour definition programming").

The individual planes are also used to define the direction of rotation of the circle for the circular interpolation CW or CCW. In the plane in which the circle is traversed, the abscissa and the ordinate are designed and thus also the direction of rotation of the circle. Circles can also be traversed in a plane other than that of the currently active G17 to G19 plane (see Section "Axis Movements").

The following plane and axis assignments are possible:

Table 10- 2 Plane and axis assignments

G function	Plane (abscissa/ordinate)	Vertical axis on plane (length compensation axis when drilling/milling)
G17	X/Y	Z
G18	Z/X	Y
G19	Y/Z	X

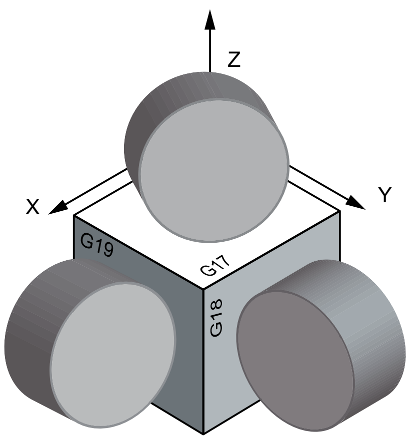


Figure 10-3 Planes and axis assignment

Programming example

```
N10 G17 T... D... M...    ; X/Y plane selected
N20 ... X... Y... Z...    ; tool length compensation (length1) in Z axis
```

10.2.3 Absolute/incremental dimensioning: G90, G91, AC, IC

Functionality

With the instructions G90/G91, the written positional data X, Y, Z, ... are evaluated as a coordinate point (G90) or as an axis position to traverse to (G91). G90/G91 applies to all axes.

Irrespective of G90/G91, certain positional data can be specified for certain blocks in absolute/incremental dimensions using AC/IC.

These instructions do **not determine the path** by which the end points are reached; this is provided by a G group (G0, G1, G2 and G3... see Chapter "Axis Movements").

Programming

G90	; Absolute dimension data
G91	; Incremental dimension data
X=AC(...)	; Absolute dimensioning for a certain axis (here: X axis), non-modal
X=IC(...)	; Absolute dimensioning for a certain axis (here: X axis), non-modal

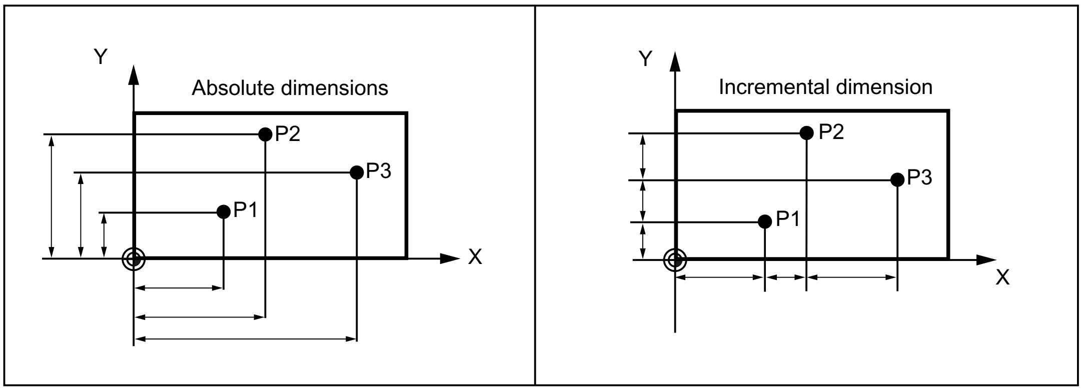


Figure 10-4 Different dimensioning types in the drawing

Absolute dimensioning G90

With absolute dimensioning, the dimensioning data refers to the **zero of the coordinate system currently active** (workpiece or current workpiece coordinate system or machine coordinate system). This is dependent on which offsets are currently active: programmable, settable, or no offsets.

Upon program start, G90 is active for **all axes** and remains active until it is deselected in a subsequent block by G91 (incremental dimensioning data) (modally active).

Incremental dimensioning G91

With incremental dimensioning, the numerical value of the path information corresponds to the **axis path to be traversed**. The leading sign indicates the **traversing direction**.

G91 applies to all axes and can be deselected in a subsequent block by G90 (absolute dimensioning).

Specification with =AC(...), =IC(...)

After the end point coordinate, write an equality sign. The value must be specified in round brackets.

Absolute dimensions are also possible for circle center points using =AC(...). Otherwise, the reference point for the circle center is the circle starting point.

Programming example

```
N10 G90 X20 Z90           ; Absolute dimensions
N20 X75 Z=IC(-32)         ; X-dimensions remain absolute, incremental Z dimension
...
N180 G91 X40 Z20          ; Switch-over to incremental dimensioning
N190 X-12 Z=AC(17)        ; X-remains incremental dimensioning, Z-absolute
```

10.2.4 Dimensions in metric units and inches: G71, G70, G710, G700

Functionality

If workpiece dimensions that deviate from the base system settings of the control are present (inch or mm), the dimensions can be entered directly in the program. The required conversion into the base system is performed by the control system.

Programming

G70	; Inch dimensions
G71	' Metric dimensions
G700	; Inch dimensions, also for feedrate F
G710	; Metric dimensions, also for feedrate F

Programming example

```
N10 G70 X10 Z30          ; Inch dimensions
N20 X40 Z50              ;G70 continues to act
...
N80 G71 X19 Z17.3        ; metric dimensioning from this point on
```

Information

Depending on the **default setting** you have chosen, the control system interprets all geometric values as either metric **or** inch dimensions. Tool offsets and settable work offsets including their display are also to be understood as geometrical values; this also applies to the feedrate F in mm/min or inch/min. The default setting can be set via machine data.

All examples listed in this manual are based on a **metric default setting**.

G70 or G71 evaluates all geometric parameters that directly refer to the **workpiece**, either as inches or metric units, for example:

- Positional data X, Y, Z, ... for G0,G1,G2,G3,G33, CIP, CT
- Interpolation parameters I, J, K (also thread pitch)
- Circle radius CR
- **Programmable** work offset (TRANS, ATRANS)
- Polar radius RP

All remaining geometric parameters that are not direct workpiece parameters, such as feedrates, tool offsets, and **settable** work offsets, are not affected by **G70/G71**.

G700/G710 however, also affects the feedrate F (inch/min, inch/rev. or mm/min, mm/rev.).

10.2.5 Polar coordinates, pole definition: G110, G111, G112

Functionality

In addition to the common specification in Cartesian coordinates (X, Y, Z), the points of a workpiece can also be specified using polar coordinates.

Polar coordinates are also helpful if a workpiece or a part of it is dimensioned from a central point (pole) with specification of the radius and the angle.

Plane

The polar coordinates refer to the plane activated with G17 to G19. In addition, the third axis standing vertically on this plane can be specified. When doing so, spatial specifications can be programmed as cylinder coordinates.

Polar radius RP=...

The polar radius specifies the distance of the point to the pole. It is stored and must only be written in blocks in which it changes, after changing the pole or when switching the plane.

Polar angle AP=...

The angle is always referred to the horizontal axis (abscissa) of the plane (for example, with G17: X axis). Positive or negative angle specifications are possible.

The polar angle remains stored and must only be written in blocks in which it changes, after changing the pole or when switching the plane.

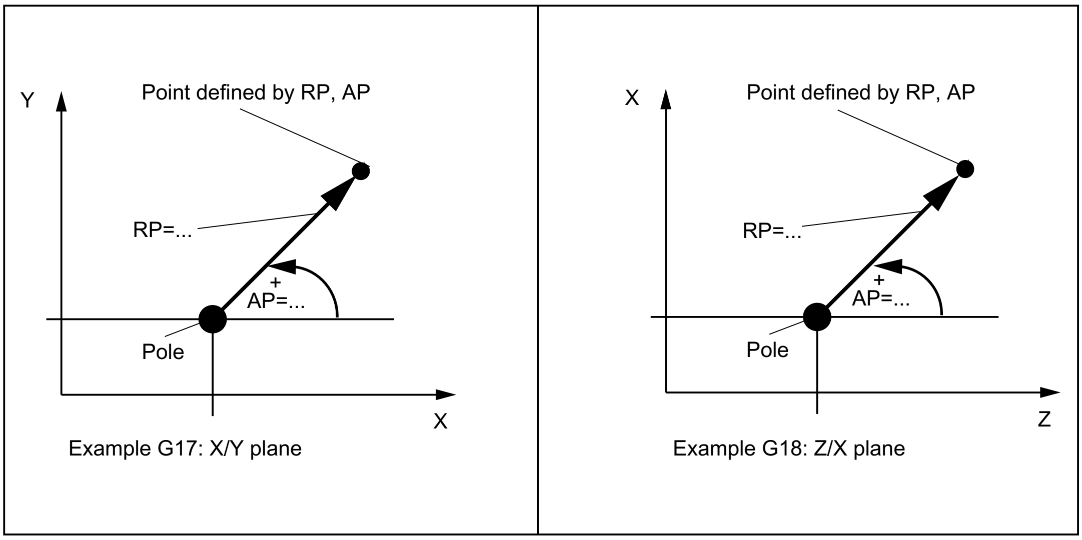


Figure 10-5 Polar radius and polar angle with definition of the positive direction in different planes

Pole definition, programming

G110	Pole specification relative to the setpoint position last programmed (in the plane, e.g. with G17: X/Y)
G111	; Pole specification relative to the origin of the current workpiece coordinate system (in the plane, e.g. with G17: X/Y)
G112	; Pole specification, relative to the last valid pole; preserve plane

Note

Pole specifications

- Pole definitions can also be performed using polar coordinates. This makes sense if a pole already exists.
- If no pole is defined, the origin of the current workpiece coordinate system will act as the pole.

Programming example

```
N10 G17                         ; X/Y plane
N20 G111 X17 Y36                ; Pole coordinates in the current workpiece
                                coordinate system
...
N80 G112 AP=45 RP=27.8          ; New pole, relative to the last pole as a polar
                                coordinate
N90 ... AP=12.5 RP=47.679       ; Polar coordinate
N100 ... AP=26.3 RP=7.344 Z4    ; Polar coordinate and Z axis (= cylinder
                                coordinate)
```

Traversing with polar coordinates

The positions programmed using polar coordinates can also be traversed as positions specified with Cartesian coordinates as follows:

- G0 - linear interpolation with rapid traverse
- G1 - linear interpolation with feedrate
- G2 - circular interpolation CW
- G3 - circular interpolation CCW

(see also Section "Axis movements").

10.2.6 Programmable work offset: TRANS, ATRANS

Functionality

The programmable work offset can be used:

- for recurring shapes/arrangements in various positions on the workpiece
- when selecting a new reference point for the dimensioning
- as a stock allowance when roughing

This results in the current workpiece coordinate system. The rewritten dimensions use this as a reference.

The offset is possible in all axes.

Programming

TRANS X... Y... Z...	; programmable offset, deletes old instructions for offsetting, rotation, scaling factor, mirroring
ATRANS X... Y... Z...	; programmable offset, additive to existing instructions
TRANS	; without values: clears old instructions for offset, rotation, scaling factor, mirroring

The instructions that contain TRANS or ATRANS each require a separate block.

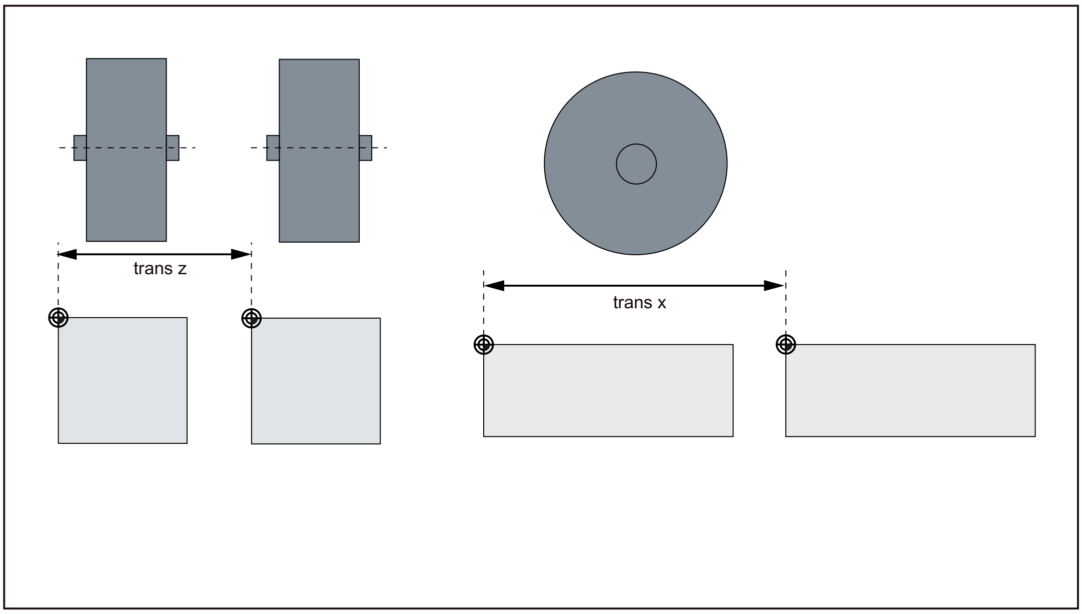


Figure 10-6 Programmable offset (example)

Programming example

```
N20 TRANS X20 Y15          ; Programmable translation
N30 L10                    ; Subroutine call; contains the geometry to be offset
...
N70 TRANS                  ; Offset cleared
```

Subroutine call - see Section "Subroutine technique "

10.2.7 Programmable rotation: ROT, AROT

Functionality

The rotation is performed in the current plane G17 or G18 or G19 using the value of RPL=... specified in degrees.

Programming

ROT RPL=...	; Programmable rotation, deletes old instructions for offsetting, rotation, scaling factor, mirroring
AROT RPL=...	; Programmable rotation, additive to existing instructions
ROT	; without values: Clears old instructions for offset, rotation, scaling factor, mirroring

The instructions that contain ROT or AROT each require a separate block.

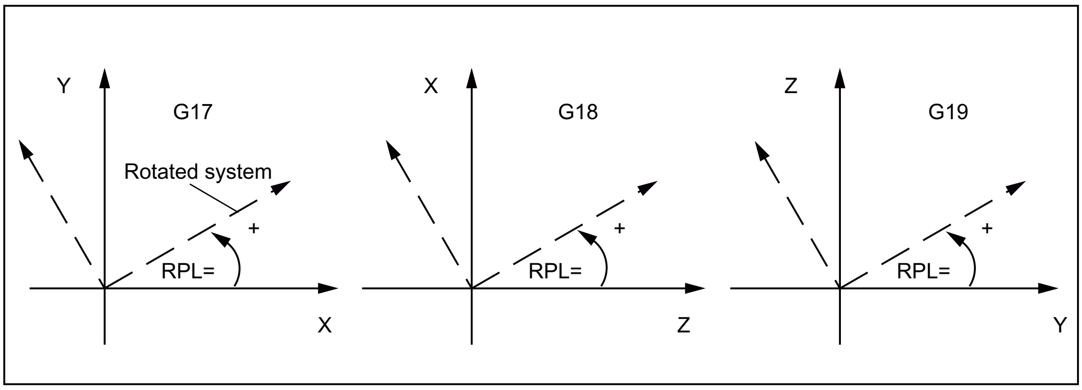


Figure 10-7 Definition of the positive direction of the angle of rotation in the individual planes

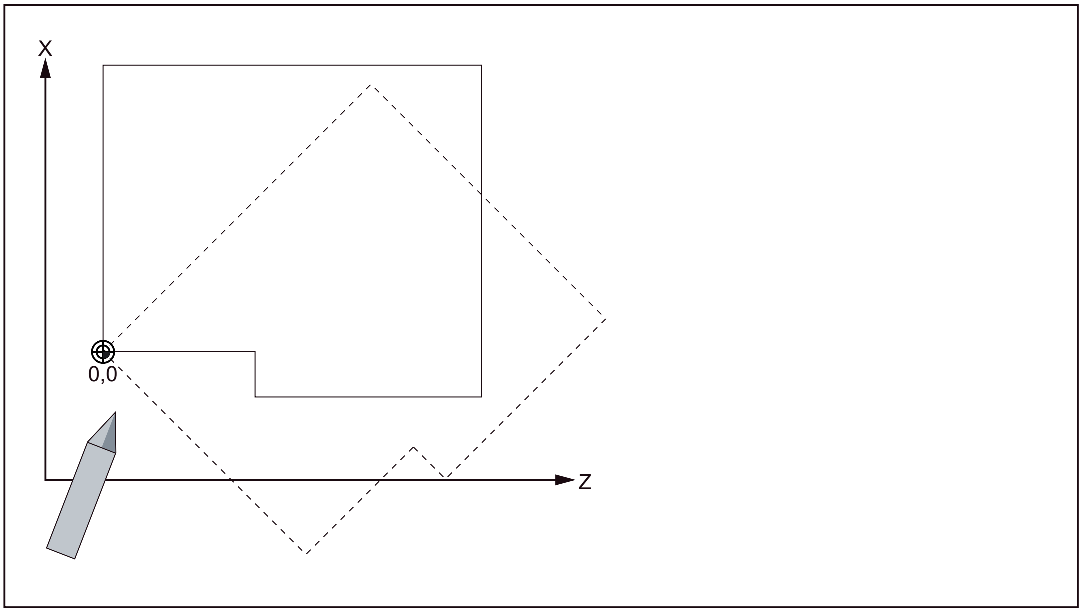


Figure 10-8 Programming example for programmable offset and rotation

Programming example

```
N10 G17 ...              ; X/Y plane
N20 TRANS X20 Y10        ; Programmable translation
N30 L10                  ; Subroutine call; contains the geometry to be offset
N40 TRANS X30 Y26        ; New offset
N50 AROT RPL=45          ; Additive 45 degree rotation
N60 L10                  ; Subroutine call
N70 TRANS                ; Offset and rotation cleared
...
```

Subroutine call - see Section "Subroutine technique "

10.2.8 Programmable scaling factor: SCALE, ASCALE

Functionality

A scale factor can be programmed for all axes with SCALE, ASCALE. The path is enlarged or reduced by this factor in the axis specified. The currently set coordinate system is used as the reference for the scale change.

Programming

SCALE X... Y... Z...	; Programmable scaling factor, clears old instructions for offset, rotation, scaling factor, mirroring
ASCALE X... Y... Z...	; Programmable scaling factor, additive to existing instructions
SCALE	; without values: clears old instructions for offset, rotation, scaling factor, mirroring

The instructions that contain SCALE or ASCALE each require a separate block.

Note

For circles, the same factor should be used in both axes.

If ATRANS is programmed with SCALE/ASCALE active, these offset values are also scaled.

Programming example

```
N10 G17                          ; X/Y plane
N20 L10                          ; Programmed contour original
N30 SCALE X2 Y2                  ; Contour in X and Z enlarged two times
N40 L10
N50 ATRANS X2.5 Y18              ; Values are also scaled!
N60 L10                          ; Contour enlarged and offset
```

Subroutine call - see Section "Subroutine technique "

10.2.9 Programmable mirroring: MIRROR, AMIRROR

Functionality

MIRROR and AMIRROR can be used to mirror workpiece shapes on coordinate axes. All traversing motions of axes for which mirroring is programmed are reversed in their direction.

Programming

MIRROR X0 Y0 Z0	; Programmable mirroring, clears old instructions for offset, rotation, scaling factor, mirroring
AMIRROR X0 Y0 Z0	; Programmable mirroring, additive to existing instructions
MIRROR	; Without values: clears old instructions for offset, rotation, scaling factor, mirroring

The instructions that contain MIRROR or AMIRROR each require a separate block. The axis value has no influence. A value, however, must be specified.

Note

Any active tool radius compensation (G41/G42) is reversed automatically when mirroring.

The direction of rotation of the circle G2/G3 is also reversed automatically when mirroring.

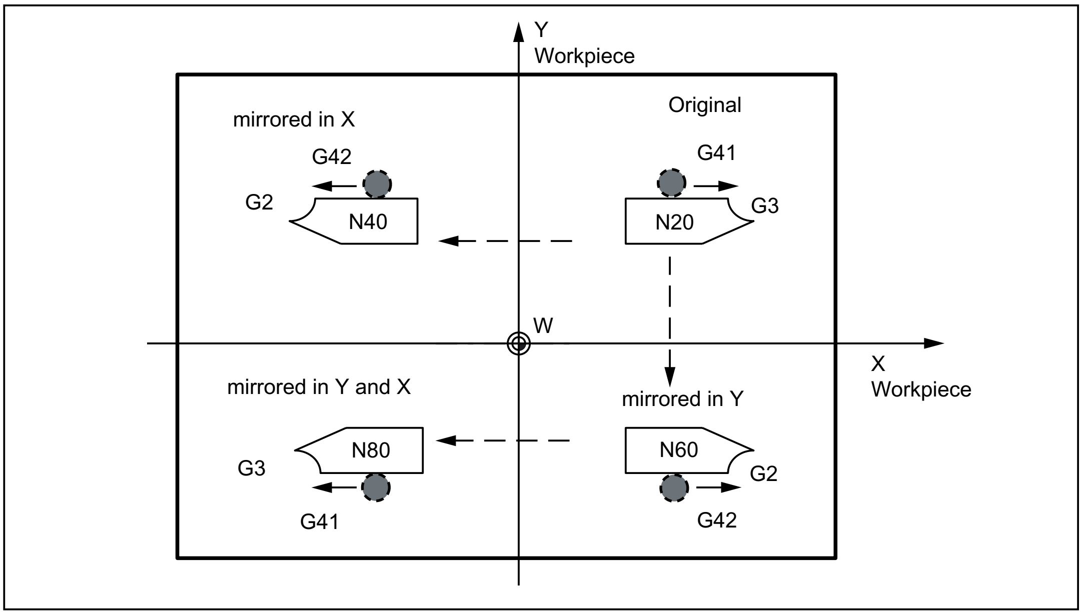


Figure 10-9 Example for mirroring with the tool position shown

Programming example

Mirroring in different coordinate axes with influence on an active tool radius compensation and G2/G3:

```
...
N10 G17                    ; X/Y plane, Z standing vertically on it
N20 L10                    ; Programmed contour with G41
N30 MIRROR X0              ; Direction changed in X
N40 L10                    ; Mirrored contour
N50 MIRROR Y0              ; Direction changed in Y
N60 L10
N70 AMIRROR X0             ; Mirroring once more, but now in X
N80 L10                    ; Twice-mirrored contour
N90 MIRROR                 ; Mirroring off
...
```

Subroutine call - see Section "Subroutine technique "

10.2.10 Workpiece clamping - settable work offset: G54 to G59, G500, G53, G153

Functionality

The settable work offset specifies the position of the **workpiece zero** on the machine (offset of the workpiece zero with respect to the machine zero). This offset is determined upon clamping of the workpiece into the machine and must be entered in the corresponding data field by the operator. The value is activated by the program by selection from six possible groupings: G54 to G59.

Note

Workpiece clamping at an angle is possible by entering the angles of rotation around the machine axes. These rotation portions are activated with the offset G54 to G59.

For information on operation, see Section "Setting/changing the work offset"

Programming

G54	; 1st Settable zero offset
G55	; 2nd Settable zero offset
G56	; 3rd Settable zero offset
G57	; 4th Settable zero offset
G58	; 5th Settable zero offset
G59	; 6th Settable zero offset
G500	; Settable zero offset OFF - modal
G53	; settable zero offset OFF non-modal, also suppresses programmable offset
G153	; As with G53; additionally suppresses base frame

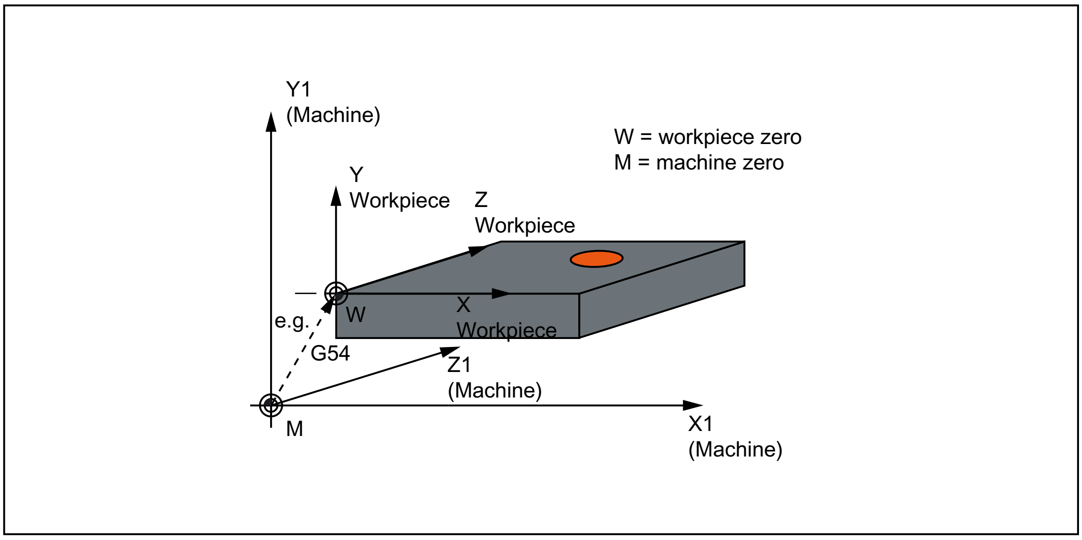


Figure 10-10 Settable zero offset

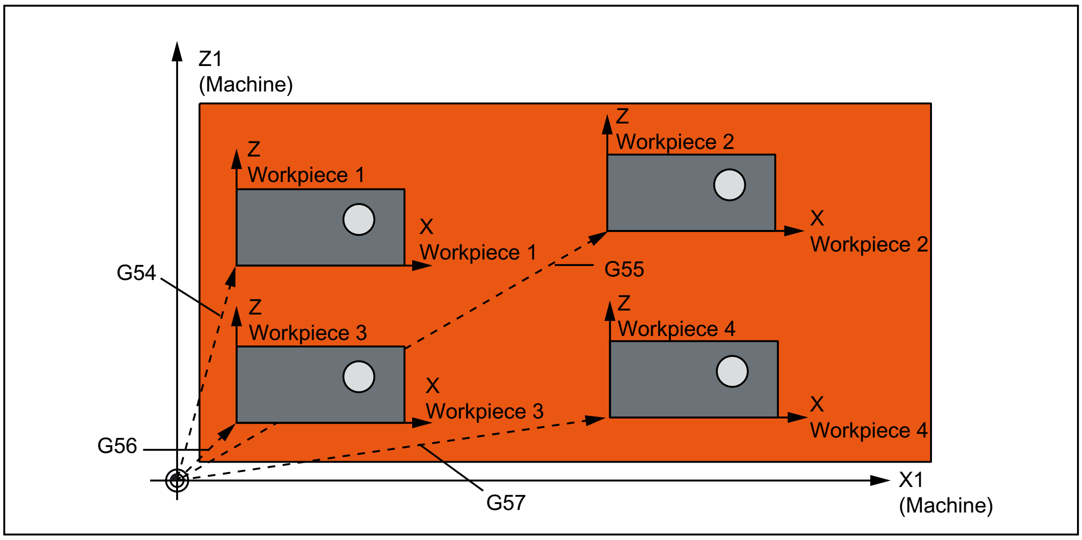


Figure 10-11 Various workpiece clamping positions when drilling/milling

Programming example

```
N10 G54 ...              ; Call first settable zero offset
N20 L47                  ; Machining of workpiece 1, here using L47
N30 G55 ...              ; Call second settable zero offset
N40 L47                  ; Machining of workpiece 2, here using L47
N50 G56 ...              ; Call third settable zero offset
N60 L47                  ; Machining of workpiece 3, here using L47
N70 G57 ...              ; Call fourth settable zero offset
N80 L47                  ; Machining of workpiece 4, here using L47
N90 G500 G0 X...         ; Deactivate settable zero offset
```

Subroutine call - see Section "Subroutine technique "

10.2.11 Programmable working area limitation: G25, G26, WALIMON, WALIMOF

Functionality

With G25, G26, a working area can be defined for all axes in which it is possible to traverse, with no traversing allowed outside this area. With the tool length compensation active, the tool tip is decisive; The coordinate parameters are machine-based.

In order to be able to use the working area limitation, it must be activated for the respective axis. This is done via the input screen under "Offset param" > "Setting data" > "Working area limit".

There are two options for defining the working area:

- Entering values via the input screen of the control system under "Offset param" > "Setting data" > "Working area limit".

  This makes the working area limitation effective in JOG mode as well.
- Programing with G25/G26

  The values for the individual axes can be changed in the part program. The values that were input in the input screen ("Offset param" > "Setting data" > "Working area limit") are overwritten.

The working area limitation is enabled/disabled in the program by WALIMON/WALIMOF.

Programming

G25 X... Y... Z...	; Lower working area limitation
G26 X... Y... Z...	; Upper working area limitation
WALIMON	; Working area limitation ON
WALIMOF	; Working area limitation OFF

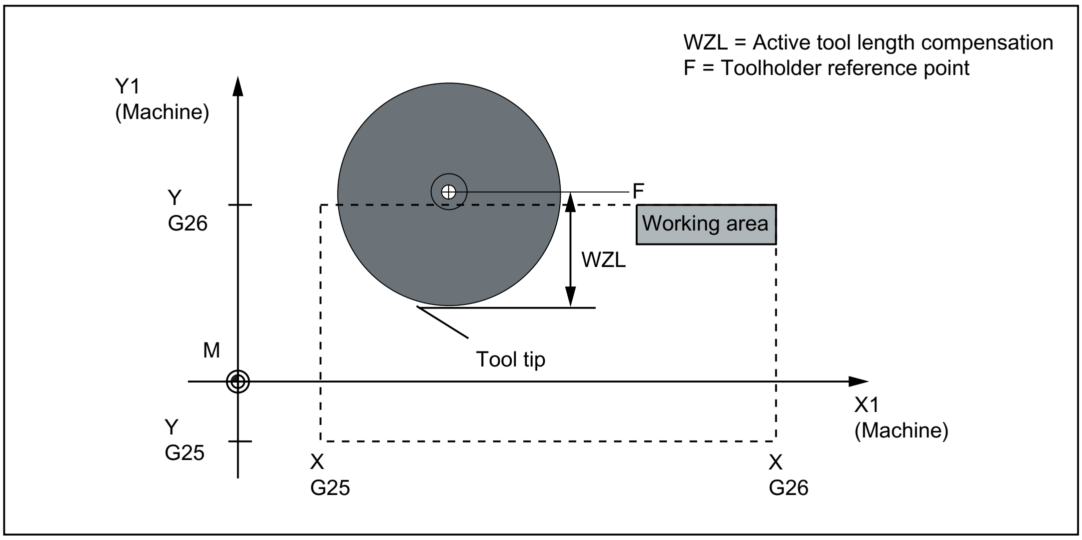


Figure 10-12 Programmable working area limitation (example: 2-dimensional)

Note

- The channel axis designation from MD20080 AXCONF_CHANAX_NAME_TAB should be used for G25, G26.

  With SINUMERIK 802D sl, kinematic transformations (TRANSMIT, TRACYL) are possible. It is possible that different axis designations for MD20080 and the MD20060 AXCONF_GEOAX_NAME_TAB geometry axis designation are specified here.
- G25, G26 is also used in connection with the address S for the spindle speed limitation (see also Section "Spindle speed limitation").
- A working area limitation can only be activated if the reference point for the relevant axes has been approached.

Programming example

```
N10 G25 X10 Y-20 Z30        ; Values of the lower working area limitation
N20 G26 X100 Y110 Z300      ; Values of the upper working area limitation
N30 T1 M6
N40 G0 X90 Y100 Z180
N50 WALIMON                 ; Working area limitation ON
...                         ; Work only within working range limits
N90 WALIMOF                 ; Working area limitation OFF
```

10.3 Axis movements

10.3.1 Linear interpolation with rapid traverse: G0

Functionality

The rapid traverse movement G0 is used for rapid positioning of the tool, but not for **direct workpiece machining.**

All the axes can be traversed simultaneously - on a straight path.

For each axis, the maximum speed (rapid traverse) is defined in machine data. If only one axis traverses, it uses its rapid traverse. If two or three axes are traversed simultaneously, the path velocity (e.g. the resulting velocity at the tool tip) must be selected such that the **maximum possible path velocity** with consideration of all axes involved results.

A programmed feedrate (F word) has no meaning for G0. G0 remains active until canceled by another instruction from this G group (G1, G2, G3, ...).

Programming

G0 X... Y... Z...	; Cartesian coordinates
G0 AP=... RP=...	; Polar coordinates
G0 AP=... RP=... Z...	; Cylindrical coordinates (3dimensional)

Note

Another option for linear programming is available with the angle specification ANG=... (see Section "Blueprint programming").

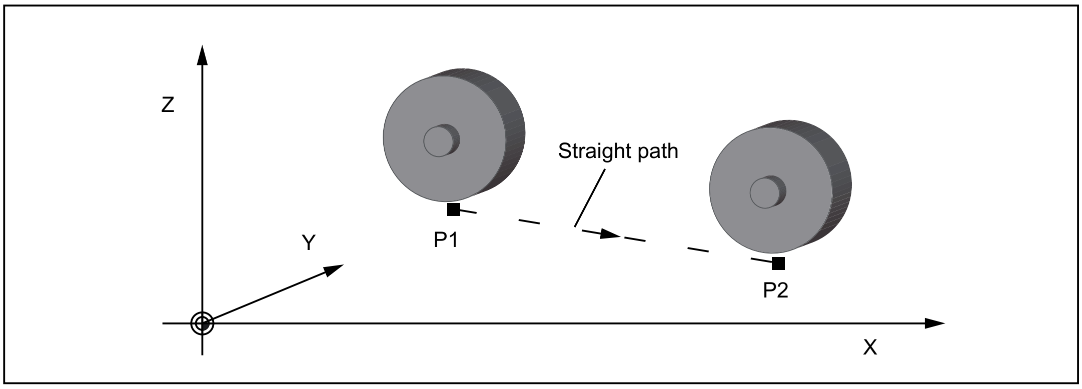


Figure 10-13 Linear interpolation with rapid traverse from point P1 to P2

Programming example

```
N10 G0 X100 Y150 Z65                    ; Cartesian coordinate
...
N50 G0 RP=16.78 AP=45                   ; Polar coordinate
```

Information

Another group of G functions exists for movement to the position (see Section "Exact stop / continuous-path control mode: G9, G60, G64").

For G60 exact stop, a window with various precision values can be selected with another G group. For exact stop, an alternative instruction with non-modal effectiveness exists: G9.

You should consider these options for adaptation to your positioning tasks.

10.3.2 Linear interpolation with feedrate: G1

Functionality

The tool moves from the starting point to the end point along a straight path. The **path velocity** is determined by the programmed **F word**.

All axes can be traversed simultaneously.

G1 remains active until canceled by another instruction from this G group (G0, G2, G3, ...).

Programming

G1 X... Y... Z... F...	; Cartesian coordinates
G1 AP=... RP=... F...	; Polar coordinates
G1 AP=... RP=... Z... F...	; cylindrical coordinates (3dimensional)

Note

Another option for linear programming is available with the angle specification ANG=... (see Section "Blueprint programming").

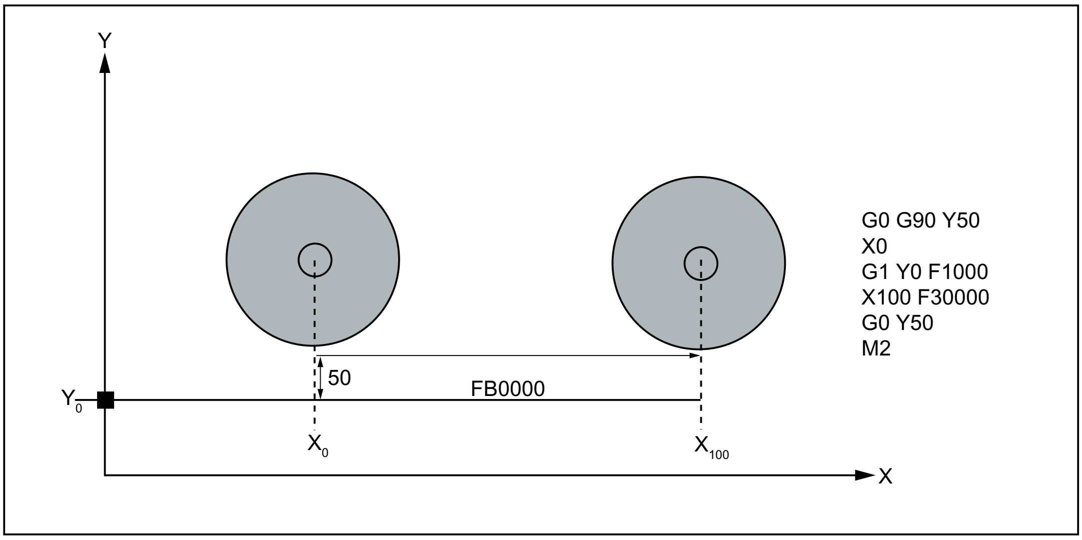


Figure 10-14 Linear interpolation in three axes using the example of a slot

Programming example

```
N05 G0 G90 X40 Y48 Z2 S500 M3     ; The tool traverses in rapid traverse on
                                  P1, three axes concurrently, spindle
                                  speed = 500 rpm, clockwise
N10 G1 Z-12 F100                  ; Infeed on Z-12, feed 100 mm/min
N15 X20 Y18 Z-10                  ; Tool travels on a straight line in
                                  space on P2
N20 G0 Z100                       ; Retraction in rapid traverse
N25 X-20 Y80
N30 M2                            ; End of program
```

To machine a workpiece, spindle speed S ... and direction M3/M4 are required (see Section "Spindle movement").

10.3.3 Circular interpolation: G2, G3

Functionality

The tool moves from the starting point to the end point along a circular path. The direction is determined by the G function:

G2: clockwise

G3: counter-clockwise

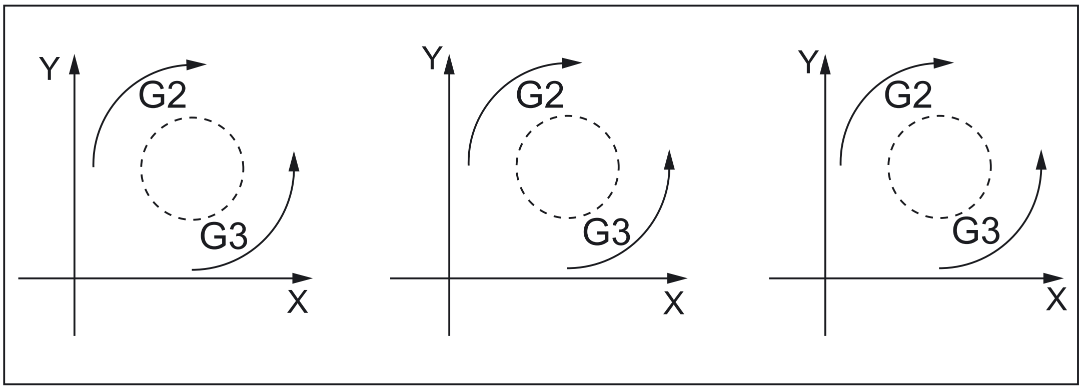


Figure 10-15 Definition of the direction of rotation of the circle G2/G3 in the three possible planes

The description of the desired circle can be given in various ways:

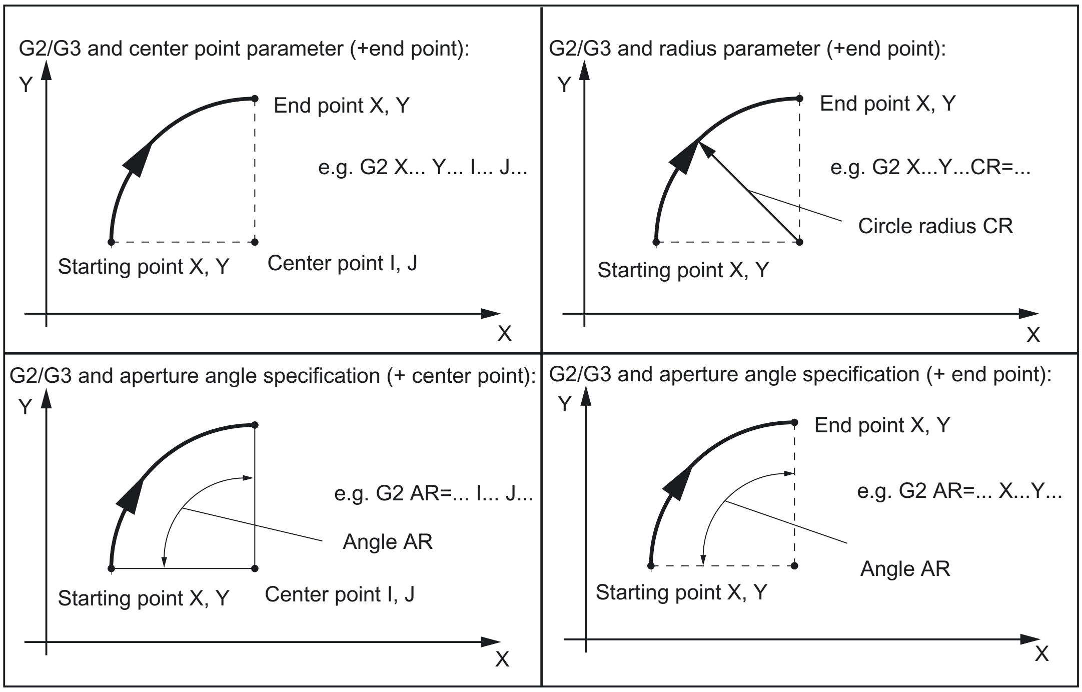


Figure 10-16 Possibilities of circle programming with G2/G3 using the example of the axes X/Y and G2

G2/G3 remains active until canceled by another instruction from this G group (G0, G1, ...).

The **path velocity** is determined by the programmed **F word**.

Programming

G2/G3 X... Y... I... J...	; Center and end point
G2/G3 CR=... X... Y...	; Circle radius and end point
G2/G3 AR=... I... J...	; Opening angle and center point
G2/G3 AR=... X... Y...	; Opening angle and end point
G2/G3 AP=... RP=...	; Polar coordinates, circle around the pole

---

Note

Further possibilities for circle programming result from:

CT - circle with tangential connection and

CIP - circle via intermediate point (see next sections).

Input tolerances for the circle

Circles are only accepted by the control system with a certain dimensional tolerance. The circle radius at the starting and end points are compared here. If the difference is within the tolerance, the center point is exactly set internally. Otherwise, an alarm message is issued.

The tolerance value can be set via machine data (see "Operating Instructions" 802Dsl).

Information

Full circles in a block are only possible if the center point and the end point are specified.

For circles with radius specification, the arithmetic sign of CR=... is used to select the correct circle. It is possible to program two circles with the same starting and end points, as well as with the same radius and the same direction. The negative sign in front of CR=-... determines the circle whose circle segment is greater than a semi-circle; otherwise, the circle with the circle segment is less than or equal to the semi-circle and determined as follows:

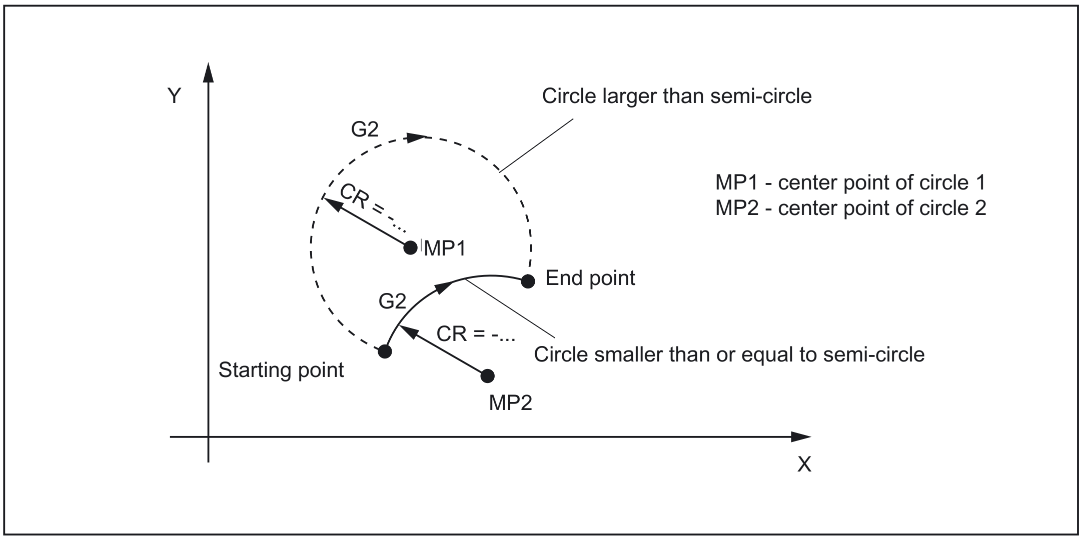


Figure 10-17 Selection of the circle from two possible circles with radius specification using the sign of CR=

Programming example: Definition of center point and end point

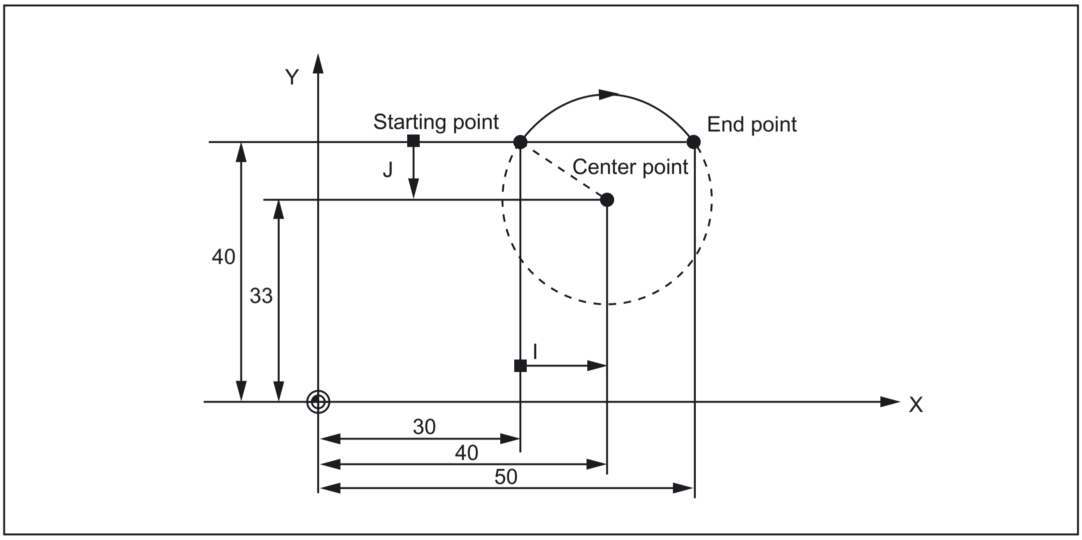


Figure 10-18 Example for center point and end point specification

```
N5 G90 X30 Y40                              ; Starting point circle for N10
N10 G2 X50 Y40 I10 J-7                      ; End point and center point
```

Note

Center point values refer to the circle starting point!

Programming example: End point and radius specification

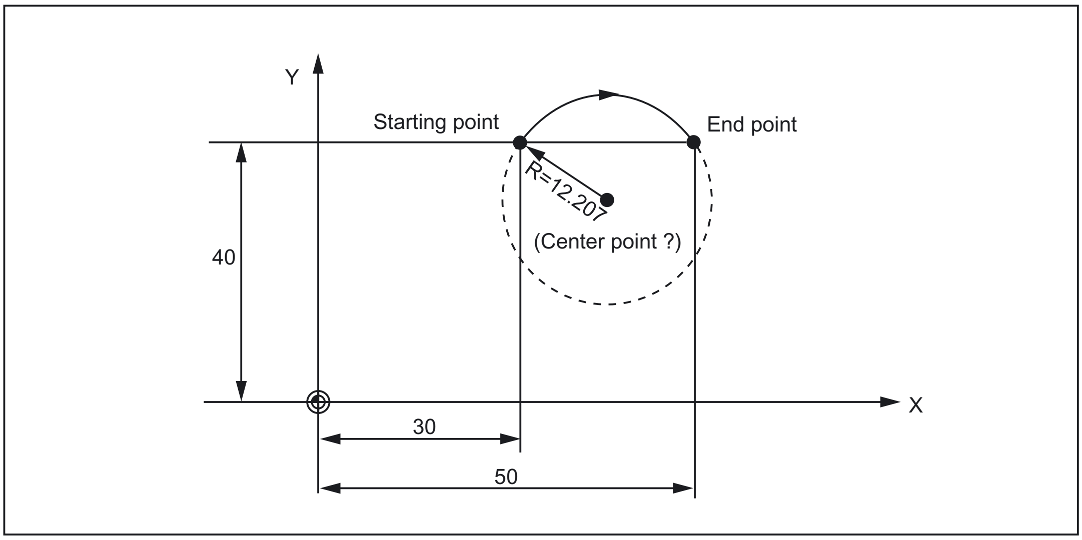


Figure 10-19 Example for end point and radius specification

```
N5 G90 X30 Y40                      ; Starting point circle for N10
N10 G2 X50 Y40 CR=12.207            ; End point and radius
```

Note

With a negative leading sign for the value with CR=-..., a circular segment larger than a semi-circle is selected.

Programming example: Definition of end point and aperture angle

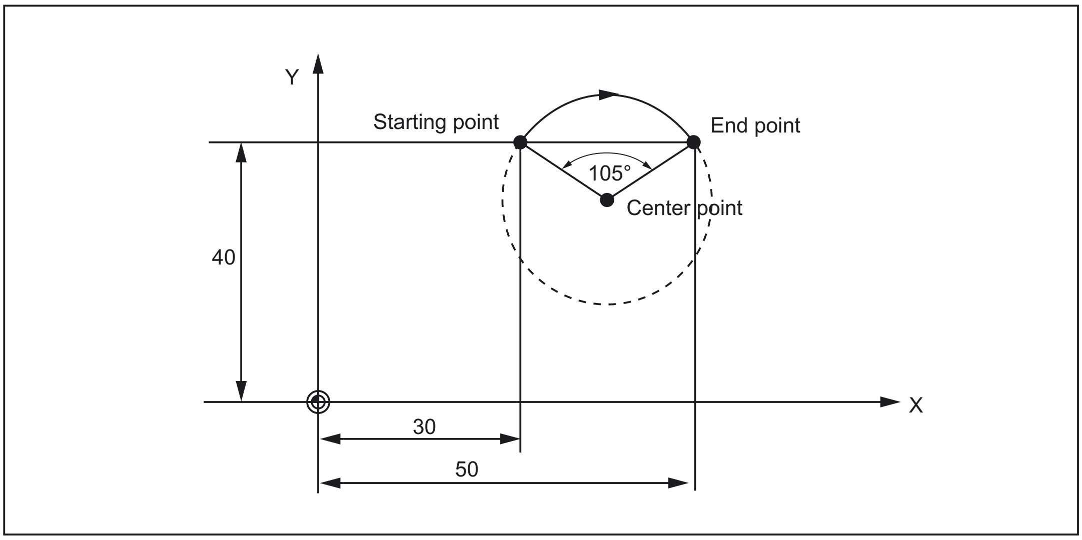


Figure 10-20 Example for end point and aperture angle specification

```
N5 G90 X30 Y40                          ; Starting point circle for N10
N10 G2 X50 Y40 AR=105                   ; End point and aperture angle
```

Programming example: Definition of center point and aperture angle

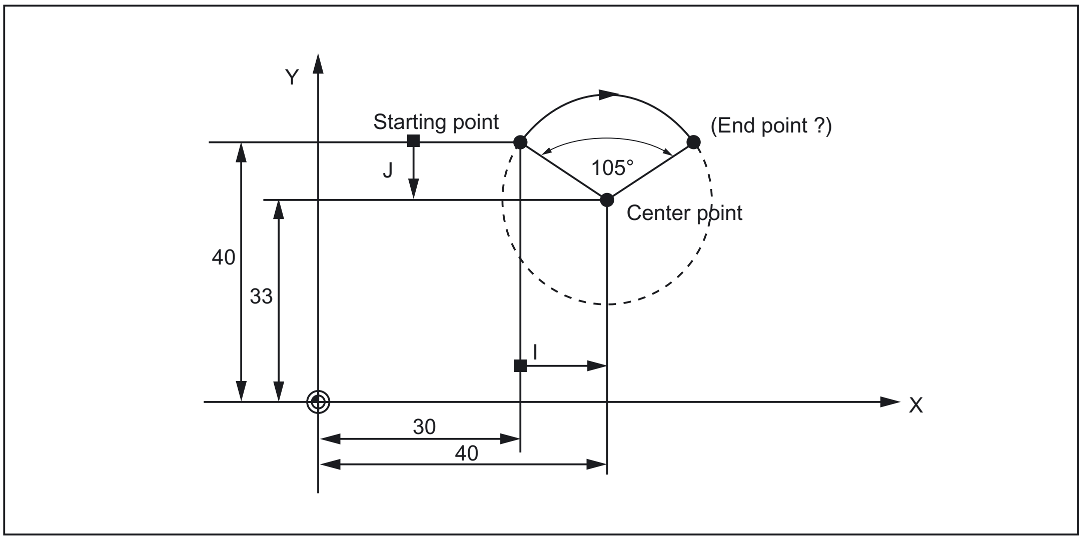


Figure 10-21 Example for center point and aperture angle specification

```
N5 G90 X30 Y40                          ; Starting point circle for N10
N10 G2 I10 J-7 AR=105                   ; Center point and aperture angle
```

Note

Center point values refer to the circle starting point!

Programming example: Polar coordinates

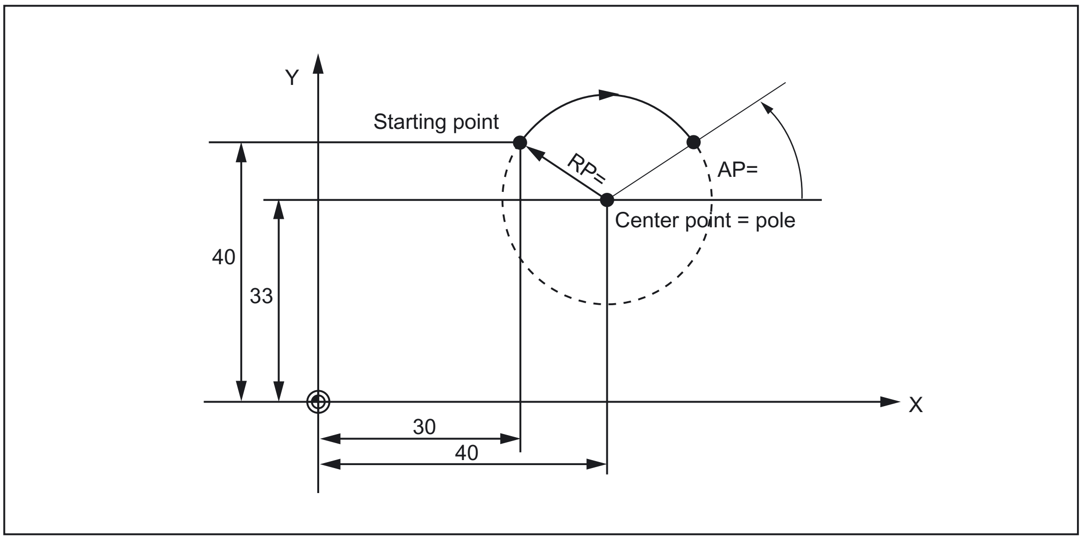


Figure 10-22 Example for circle with polar coordinates

```
N1 G17                                  ; X/Y plane
N5 G90 G0 X30 Y40                       ; Starting point circle for N10
N10 G111 X40 Y33                        ; Pole = circle center
N20 G2 RP=12.207 AP=21                  ; Polar specifications
```

10.3.4 Circular interpolation via intermediate point: CIP

Functionality

If you know **three contour points** of the circle, instead of center point or radius or aperture angle, then it is advantageous to use the CIP function.

The direction of the circle results here from the position of the intermediate point (between starting and end points). The intermediate point is written according to the following axis assignment:

I1=... for the X axis,

J1=... for the Y axis,

K1=... for the Z axis.

CIP remains active until canceled by another instruction from this G group (G0, G1, G2, ...).

Note

The configured dimensional data G90 or G91 applies to the end point **and** the intermediate point.

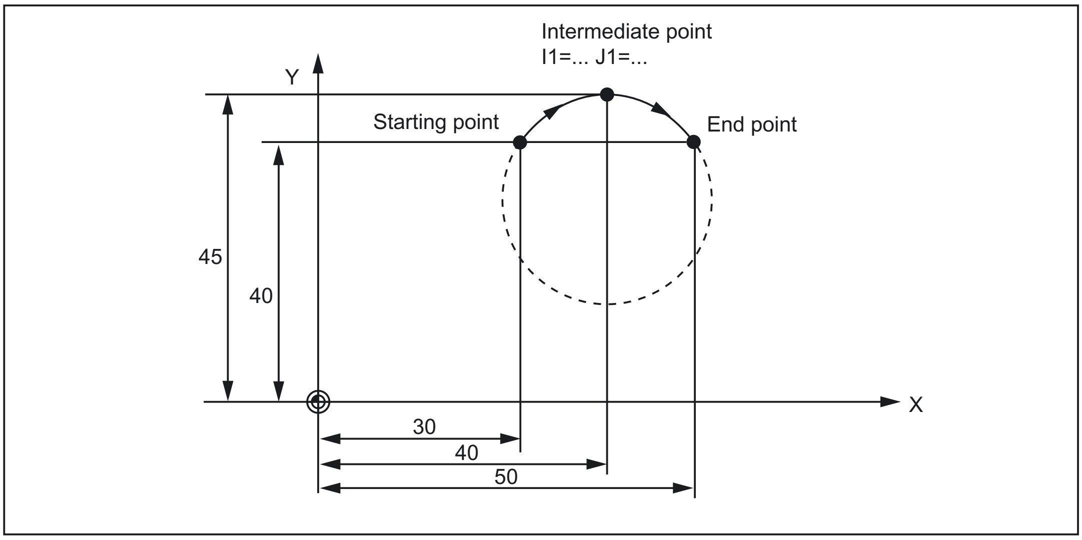


Figure 10-23 Circle with end point and intermediate point specification using the example of G90

Programming example

```
N5 G90 X30 Y40                          ;Starting point circle for N10
N10 CIP X50 Y40 I1=40 J1=45             ; End point and intermediate point
```

10.3.5 Circle with tangential transition: CT

Functionality

With CT and the programmed end point in the current plane G17 through G19, a circle is generated which is connected tangentially to the previous path segment (circle or straight line) in this plane.

This defines the radius and center point of the circle from the geometric relationships of the previous path section and the programmed circle end point.

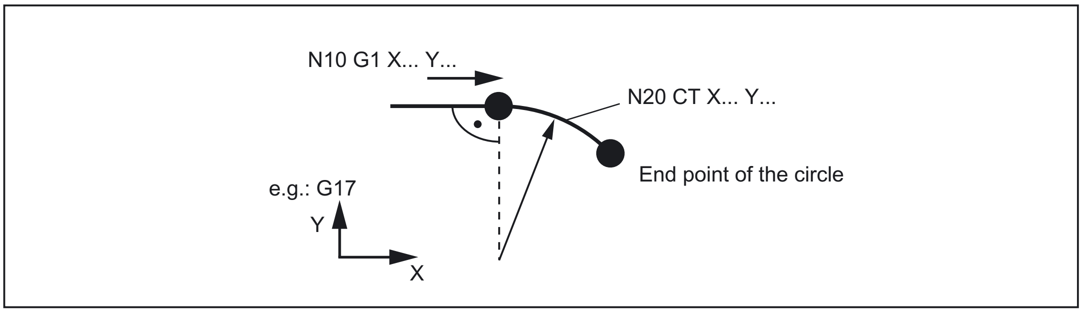


Figure 10-24 Circle with tangential transition to the previous path section

Programming example

```
N10 G1 X20 F300                            ; Straight line
N20 CT X... Y...                           ; Circle with tangential connection
```

10.3.6 Fixed point approach: G75

Functionality

By using G75, a fixed point on the machine, e.g. tool change point, can be approached. The position is stored permanently in the machine data for all axes. A maximum of four fixed points can be defined for each axis.

No offset is effective. The speed of each axis is its rapid traverse.

G75 requires a separate block and is non-modal. The machine axis identifier must be programmed!

In the block after G75, the previous G command of the "Interpolation type" group (G0, G1,G2, ...) is active again.

Programming

```
G75 FP=<n> X1=0 Y1=0 Z1=0
```

Note

FPn references with axis machine date MD30600 $MA_FIX_POINT_POS[n-1]. If no FP has been programmed, then the first fixed point will be selected.

Table 10- 3 Explanation

Command	Significance
G75	Fixed point approach
FP=<n>	Fixed point that is to be approached. The fixed point number is specified: <n> Value range of <n>: 1, 2, 3, 4 If no fixed point is specified, fixed point 1 is approached automatically.
X1=0 Y1=0 Z1=0	Machine axes to be traversed to the fixed point. Here, specify the axes with value "0" with which the fixed point is to be approached simultaneously. Each axis is traversed with the maximum axial velocity.

Programming example

```
N05 G75 FP=1 Z1=0          ; Approach fixed point 1 in Z
N10 G75 FP=2 X1=0 Y1=0     ; Approach fixed point 2 in X and Y,
                           e. g. to change a tool
N30 M30                    ; End of program
```

Note

The programmed position values for X1, Y1, Z1 (any value, here = 0) are ignored, but must still be written.

10.3.7 Reference point approach: G74

Functionality

The reference point can be approached in the NC program with G74. The direction and speed of each axis are stored in machine data.

G74 requires a separate block and is non-modal. The machine axis identifier must be programmed!

In the block after G74, the previous G command of the "Interpolation type" group (G0, G1,G2, ...) is active again.

Programming example

```
N10 G74 X1=0 Y1=0 Z1=0
```

Note

The programmed position values for X1, Y1, Z1 (any value, here = 0) are ignored, but must still be written.

10.3.8 Measuring with touch-trigger probe: MEAS, MEAW

Functionality

The function is available for SINUMERIK 802D sl plus and pro.

If the instruction MEAS=... or MEAW=... is in a block with traversing movements of axes, the positions of the traversed axes for the switching edge of a connected measuring probe are registered and stored. The measurement result can be read for each axis in the program.

For MEAS, the movement of the axes is halted when the selected switching edge of the probe appears and the remaining distance to go is deleted.

Programming

MEAS=1	G1 X... Y... Z... F...	; Measuring with rising edge of the probe, clearing the distance to go
MEAS=-1	G1 X... Y... Z... F...	; Measuring with falling edge of the probe, clearing the distance to go
MEAW=1	G1 X... Y... Z... F...	; Measuring with rising edge of the probe, **without** clearing the distance to go
MEAW=-1	G1 X... Y... Z... F...	; Measuring with falling edge of the probe, **without** clearing the distance to go

CAUTION

For MEAW: Measuring probe travels to the programmed position even after is has triggered. Risk of destruction!

Measuring job status

If the probe has switched, the variable $AC_MEA[1] after the measuring block has the value =1; otherwise, value = 0.

At the start of a measuring block, the variable is set to the value=0.

Measuring result

When the probe is successfully activated, the result of the measurement is available for the axes traversed in the measuring block with the following variables after the measuring block:

in the machine coordinate system: $AA_MM[*axis*]

in the workpiece coordinate system: $AA_MW[*axis*]

Programming example

```
N10 MEAS=1 G1 X300 Z-40 F4000           ; Measurement with deletion of distance-
                                         to-go,
                                         ; Probe with rising edge
N20 IF $AC_MEA[1]==0 GOTOF MEASERR       ; Measuring error?
N30 R5=$AA_MW[X] R6=$AA_MW[Z]            ; Processing of the measured values
..
N100 MEASERR: M0                         ; Measuring error
```

Note

IF instruction - see Section "Conditional program jumps"

10.3.9 Feedrate F

Functionality

The feed F is the **path velocity** and represents the value of the geometric sum of the velocity components of all axes involved. The individual axis velocities therefore result from the portion of the axis path in the overall distance to be traversed.

The feedrate F is effective for the interpolation types G1, G2, G3, CIP, and CT and is retained until a new F word is written.

Programming

F...

Note

For integer values, the decimal point is not required, e.g. F300.

Unit of measure for F with G94, G95

The dimension unit for the F word is determined by G functions:

- G94: F as the feedrate in mm/min
- G95: Feedrate F in mm/spindle revolutions

  (only meaningful when the spindle is running)

  Note

  This unit of measure applies to metric dimensions. According to Section "Metric and inch dimensioning", settings with inch dimensioning are also possible.

Programming example

```
N10 G94 F310                          ; Feedrate in mm/min
...
N110 S200 M3                          ; Spindle rotation
N120 G95 F15.5                        ; Feedrate in mm/revolution
```

Note

Write a new F word if you change G94 - G95.

10.3.10 Exact stop / continuous-path control mode: G9, G60, G64

Functionality

G functions are provided for optimum adaptation to different requirements to set the traversing behavior at the block boundaries and for block advancing. Example: For example, you would like to quickly position with the axes or you would like to machine path contours over multiple blocks.

Programming

G60	; Exact stop, modal
G64	; Continuous-path mode
G9	; Exact stop, non-modal
G601	; Exact stop window fine
G602	; Exact stop window coarse

Exact stop G60, G9

If the exact stop function (G60 or G9) is active, the velocity for reaching the exact end position at the end of a block is decelerated to zero.

Another modal G group can be used here to set when the traversing movement of this block is considered ended and the next block is started.

- G601; Exact stop window fine

  Block advance takes place when all axes have reached the "Exact stop window fine" (value in the machine data).

- G602: Exact stop window coarse

  Block advance takes place when all axes have reached the "Exact stop window coarse" (value in the machine data).

The selection of the exact stop window has a significant influence on the total time if many positioning operations are executed. Fine adjustments require more time.

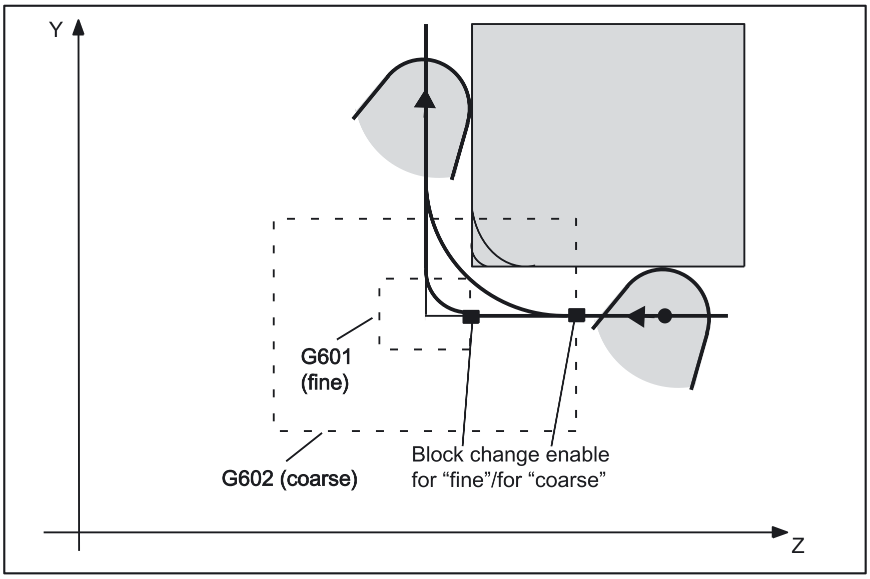


Figure 10-25 Exact stop window coarse or fine, in effect for G60/G9; enlarged display of the windows

Programming example

```
N5 G602                        ; Exact stop window coarse
N10 G0 G60 Z...                ; Exact stop modal
N20 Y... Z...                  ; G60 continues to act
...
N50 G1 G601 ...                ; Exact stop window fine
N80 G64 Z...                   ; Switching over to continuous-path mode
...
N100 G0 G9 Z...                ; Exact stop acts only in this block
N111 ...                       ; Again continuous-path mode
```

Note

The G9 command only generates exact stop for the block in which it is programmed; G60, however, is effective until it is canceled by G64.

Continuous-path control mode G64

The objective of the continuous-path control mode is to avoid deceleration at the block boundaries and to switch **to the next block** with **a path velocity as constant as possible** (in the case of tangential transitions). The function works with **look-ahead velocity control** over several blocks.

For non-tangential transitions (corners), the velocity can be reduced rapidly enough so that the axes are subject to a relatively high velocity change over a short period of time. This may lead to a significant jerk (acceleration change). The size of the jerk can be limited by activating the SOFT function.

Programming example

```
N10 G64 G1 Z... F...          ; Continuous-path mode
N20 Y..                       ; Continuous-path control mode continues to be active
...
N180 G60 ...                  ; Switching over to exact stop
```

Look-ahead velocity control:

In the continuous-path control mode with G64, the control system automatically determines the velocity control for several NC blocks in advance. This enables acceleration and deceleration across multiple blocks with approximately tangential transitions. For paths that consist of short travels in the NC blocks, higher velocities can be achieved than without look ahead.

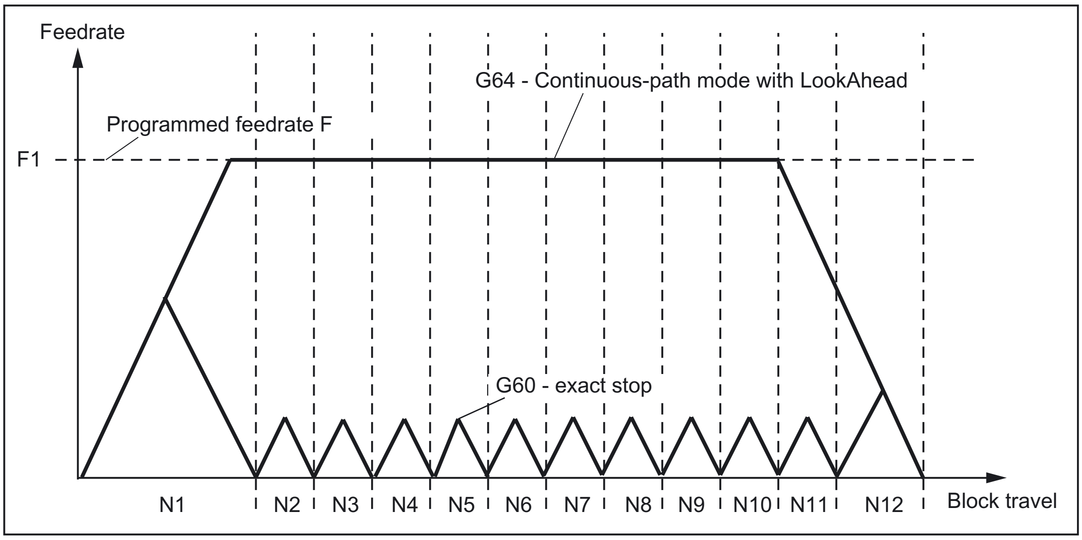


Figure 10-26 Comparison of the G60 and G64 velocity behavior with short travels in the blocks

10.3.11 Acceleration pattern: BRISK, SOFT

BRISK

The axes of the machine change their velocities using the maximum permissible acceleration value until reaching the final velocity. BRISK allows time-optimized working. The set velocity is reached in a short time. However, jumps are present in the acceleration pattern.

SOFT

The axes of the machine accelerate along a non-linear, constant characteristic until reaching the final velocity. With this jerk-free acceleration, SOFT allows for reduced machine load. The same behavior can also be applied to braking procedures.

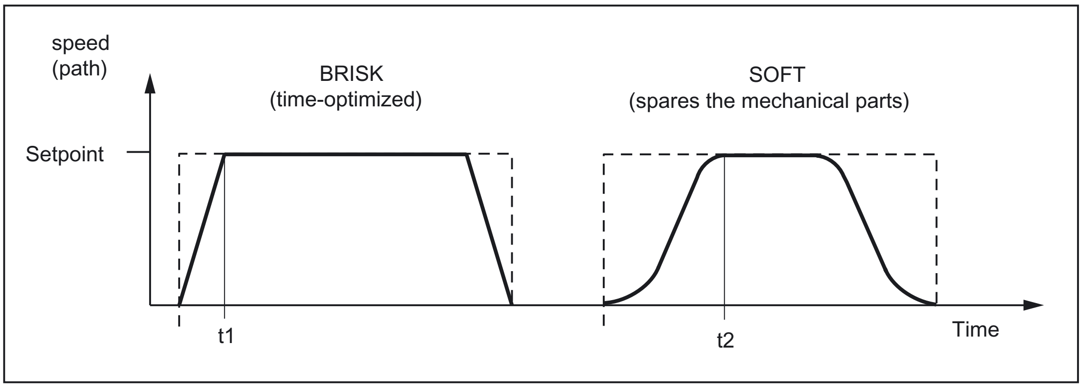


Figure 10-27 Basic course of the path velocity when using BRISK/SOFT

Programming

BRISK	; Jerking path acceleration
SOFT	; Jerk-limited path acceleration

Programming example

```
N10 SOFT G1 X30 Z84 F650       ; Jerk-limited path acceleration
...
N90 BRISK X87 Z104             ; Continuing with jerking path acceleration
...
```

10.3.12 Percentage acceleration override: ACC

Functionality

Certain program sections can require the axis and spindle acceleration set via the machine data to be changed using the program. This programmable acceleration is a percentage acceleration override.

For each axis (e.g. X) or spindle (S), a percentage value >0% and ≤200% can be programmed. The axis interpolation is then carried out with this proportional acceleration. The reference value (100%) is the valid machine data value for the acceleration (depending on whether it is the axis or spindle; for the spindle it depends further on the gear stage and whether it is in positioning mode or speed mode).

Programming

ACC[*axis name*] = percentage	; for axis
ACC[S] = percentage	; for spindle

Programming example

```
N10 ACC[X]=80                  ; 80% acceleration for the x axis
N20 ACC[S]=50                  ; 50% acceleration for the spindle
...
N100 ACC[X]=100                ; Deactivate the override for the X-axis
```

Effectivity

The limitation is effective in all interpolation types of the AUTOMATIC and MDA modes. The limitation is not active in JOG mode and during reference point approach.

The value assignment ACC[...] = 100 deactivates the override; likewise, as do RESET and End of program.

The programmed override value is also active with dry run feedrate.

> **CAUTION**
>
> A value greater than 100% may only be programmed if this load is permissible for the machine mechanics and the drives have the corresponding reserves. Failure to adhere to the limits can lead to damage to the mechanical parts and/or error messages.

10.3.13 Fourth axis

Functionality

Depending on the machine variant, a fourth axis may be necessary, e.g. dividing unit, swiveling dresser, head dresser, etc. This axis can take the form of either a linear or a rotary axis. The designation of this axis must be configured accordingly, e.g. U, C, or A etc. For rotary axes, the traversing range can be configured between 0 ...360 degrees (modulo-behavior).

With an appropriate machine design, the fourth axis can be traversed linear simultaneously with the remaining axes. If the axis is traversed together with the remaining axes (X, Z) in a block that contains G1 or G2/G3, no component of the feedrate F is assigned to this axis; its velocity will depend on the time for traveling along the X and Z axes, and its motion starts and ends with the remaining path axes. The velocity, however, must not be greater than the defined limit value.

If in one block only this fourth axis is programmed, the axis will traverse using the active feedrate F if G1 is active. If the axis is a rotary axis, the unit for F is degrees/min with G94 or degrees/rev. of the spindle with G95.

For this axis, it is also possible to specify (G54 ... G59) and program (TRANS, ATRANS) offsets.

Programming example

The fourth axis is a swivel table (rotary axis) and has the axis identifier A:

```
N5 G94                          ; Feedrate F in mm/min or degrees/min
N10 G0 X10 Y20 Z30 A45          ; Traverse X-Y-Z path with rapid traverse, A at the
                                same time
N20 G1 X12 Y21 Z33 A60 F400     ; Traverse X-Y-Z path at 400 mm/min, A at the same
                                time
N30 G1 A90 F3000                ; Axis A traverses alone to position 90 degrees at a
                                speed of 3000 degrees/min
```

Special instructions for rotary axes: DC, ACP, ACN

For example, for rotary axis A:

```
A=DC(...)            ; Absolute dimensions, approach position directly (on the
                     shortest path)
A=ACP(...)           ; Absolute dimensions, approach position in positive direction
A=ACN(...)           ; Absolute dimensions, approach position in negative direction
```

Example:

```
N10 A=ACP(55.7)      ; Approach absolute position 55.7 degrees in positive direction
```

10.3.14 Dwell time: G4

Functionality

Between two NC blocks, you can interrupt the machining for a defined time by inserting a **separate block** with G4; e.g. for relief cutting.

The words with F... or S... are only used in this block for the specified time. Any previously programmed feedrate F or a spindle speed S remain valid.

Programming

G4 F...	; Dwell time in seconds
G4 S...	; Dwell time in spindle revolutions

Programming example

```
N5 G1 F200 Z-50 S300 M3        ; Feed F; spindle speed S
N10 G4 F2.5                    ; Dwell time 2.5 seconds
N20 Z70
N30 G4 S30                     ; Dwelling 30 revolutions of the spindle, corresponds
                               at S=300 rpm and 100% speed override to: t=0.1 min
N40 X...                       ; Feed and spindle speed remain effective
```

Note

G4 S.. is only possible if a controlled spindle is available (if the speed specifications are also programmed via S...).

10.3.15 Travel to fixed stop

Functionality

This function is available for 802D sl plus and 802D sl pro.

The travel to fixed stop (FXS = Fixed Stop) function can be used to establish defined forces for clamping workpieces, such as those required for sleeves and grippers. The function can also be used for the approach of mechanical reference points. With sufficiently reduced torque, it is also possible to perform simple measurement operations without connecting a probe.

Programming

FXS[axis]=1	; Select travel to fixed stop
FXS[axis]=0	; Deselect travel to fixed stop
FXST[axis]=...	; Clamping torque, specified in % of the max. torque of the drive
FXSW[axis]=…	; Width of the window for fixed-stop monitoring in mm/degrees

---

Note

The **machine axis identifier** should be used as the axis identifier, e.g. X1. The channel axis identifier (e.g. X) is permitted only if e.g. no coordinate rotation is active and this axis is directly assigned to a machine axis.

The commands are modal. The traversing path and the selection of the function FXS[axis]=1 must be programmed **in one block**.

Programming example - selection

```
N10 G1 G94 ...
N100 X250 Z100 F100 FXS[Z1]=1 FXST[Z1]=12.3 FXSW[Z1]=2
                 ; for Z1 machine axis FXS function selected,
                 ; clamping torque 12.3%,
                 ; window width 2 mm
```

Note

When selected, the fixed stop must be located between the start and end positions.

The parameters for torque FXST[ ]= and window width FXSW[ ]= are optional. If these are not written, the values from existing setting data (SD) are in effect. Programmed values are imported to the setting data. At the start, the setting data are loaded with values from machine data. FXST[ ]=... or FXSW[ ]=... can be changed in the program at any time. The changes are applied before traversing movements in the block.

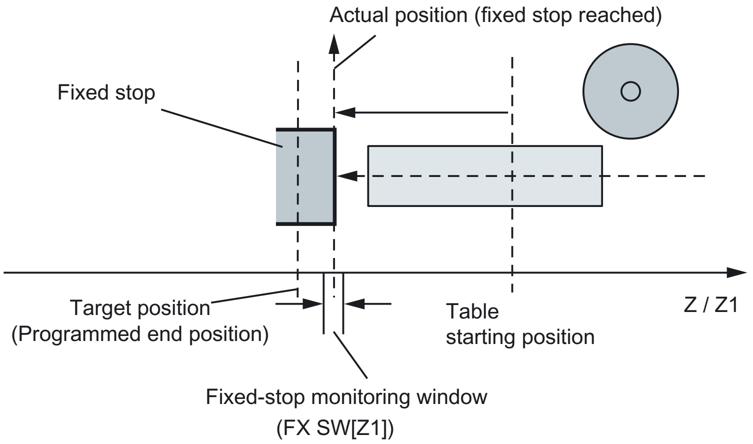


Figure 10-28 Referencing to stop

Other programming examples

`N10 G1 G94 ...`	
`N20 X250 Z100 F100 FXS[X1]=1`	`; Selected for machine axis X1 FXS, clamping torque and window width from SDs`
`N20 X250 Z100 F100 FXS[X1]=1 FXST[X1]=12.3`	`; Selected for machine axis X1 FXS, 12.3% clamping torque and window width from SDs`
`N20 X250 Z100 F100 FXS[X1]=1 FXST[X1]=12.3 FXSW[X1]=2`	`; Selected for machine axis X1 FXS, 12.3% clamping torque and window width 2 mm`
`N20 X250 Z100 F100 FXS[X1]=1 FXSW[X1]=2`	`; Selected for machine axis X1 FXS, clamping torque from SD, window width 2 mm`

Fixed stop reached

When the fixed stop has been reached:

- The distance-to-go is deleted and the position setpoint is manipulated.
- The drive torque increases to the programmed limit value FXST[ ]=... or the value from SD and then remains constant.
- The monitoring of the fixed stop is active within the specified window width (FXSW[ ]=... or value from SD).

Deselecting the function

Deselection of the function triggers a preprocessing stop. The block with FXS[X1]=0 must contain traversing movements.

Example:

`N200 G1 G94 X200 Y400 F200 FXS[X1] = 0`	`; Axis X1 is retracted from the fixed stop to position X= 200 mm.`

Note

The traversing movement to the retraction position must lead away from the fixed stop; otherwise, damage to the fixed stop or to the machine may result.

The block change takes place when the retraction position has been reached. If no retraction position is specified, the block change takes place immediately once the torque limit has been deactivated.

Note

"Measuring with deletion of distance-to-go" ("MEAS" command) and "Travel to fixed stop" cannot be programmed in the same block.

Contour monitoring is not performed while "Travel to fixed stop" is active.

If the torque limit is reduced too far, the axis will not be able to follow the specified setpoint; the position controller then goes to the limit and the contour deviation increases. In this operating state, an increase in the torque limit may result in sudden, jerky movements. Ensure that the axis can still follow. For this reason, it must be verified that the contour deviation is not larger than that with unlimited torque.

A rate of rise ramp for the new torque limit can be defined in MD to prevent any abrupt changes to the torque limit setting (e.g. insertion of a spindle sleeve or quill).

System variable for status: $AA_FXS[*axis*]

This system variable provides the "Travel to fixed stop" status for the axis specified:

- Value =
  - 0: Axis not at fixed stop
  - 1: Fixed stop successfully approached

    (axis is in fixed-stop monitoring window)
  - 2: Approach to fixed stop has failed (axis is not at fixed stop)
  - 3: Travel to fixed stop activated
  - 4: Fixed stop detected
  - 5: Travel to fixed stop is deselected. The deselection is not yet completed.

Query of the system variable in the part program initiates a preprocessing stop.

For SINUMERIK 802D sl, only the static states can be detected before and after selection/deselection.

Alarm suppression

The issuing of the following alarms can be suppressed with machine data:

- 20091 "Fixed stop not reached"
- 20094 "Fixed stop aborted"

References: "Description of Functions", Section "Travel to fixed stop"

10.4 Spindle movement

10.4.1 Spindle speed S, directions of rotation

Functionality

The spindle speed is programmed in revolutions per minute under the address S provided that the machine possesses a controlled spindle.

The direction of rotation and the start or end of the movement are specified via M commands (also see Section "Miscellaneous function M").

M3: Spindle clockwise

M4: Spindle counter-clockwise

M5: Spindle stop

Note

For integer S values, the decimal point can be omitted, e.g. S270.

Information

If you write M3 or M4 in a **block with axis movements**, the M commands become active **before** the axis movements.

Default setting: The axis movements will only start once the spindle has accelerated to speed (M3, M4). M5 is also issued before the axis movement. However, there is no waiting for spindle standstill. The axis movements begin before spindle standstill.

The spindle is stopped at program end or with RESET.

At program start, spindle speed zero (S0) is in effect.

Note

Other settings can be configured via machine data.

Programming example

```
N10 G1 X70 Z20 F300 S270 M3    ; Before the axis traversing X, Z the spindle
                               accelerates to 270 rpm, clockwise
...
N80 S450 ...                   ; Speed change
...
N170 G0 Z180 M5                ; Z movement, spindle comes to a stop
```

10.4.2 Spindle speed limitation: G25, G26

Functionality

In the program, you can limit the limit values that would otherwise apply for a controlled spindle by writing G25 or G26 and the spindle address S with the speed limit value. This overwrites the values entered in the setting data at the same time.

G25 and G26 each require a separate block. A previously programmed speed S is maintained.

Programming

G25 S...	; Lower spindle speed limitation
G26 S...	; Upper spindle speed limitation

Information

The outmost limits of the spindle speed are set in machine data. Appropriate inputs via the operator panel can activate various setting data for further limiting.

Programming example

```
N10 G25 S12            ; Lower spindle speed limitation : 12 rpm
N20 G26 S700           ; Upper spindle speed limitation : 700 rpm
```

Note

G25/G26 are used in conjunction with axis addresses for a working area limitation (see Section "Working area limitation").

10.4.3 Spindle positioning: SPOS

Functionality

Requirement: The spindle must be technically designed for position control.

With the function SPOS= you can position the spindle in a specific **angular position**. The spindle is held in the position by position control.

The **speed** of the positioning procedure is defined in machine data.

With SPOS=*value* from the M3/M4 movement, the respective **direction of rotation** is maintained until the end of the positioning. When positioning from standstill, the position is approached via the shortest path. The direction results from the respective start and end position.

Exception: First movement of the spindle, i.e. if the measuring system is not yet synchronized. In this case, the direction is specified in machine data.

Other movement specifications for the spindle are possible with SPOS=ACP(...), SPOS=ACN(...), ... as for rotary axes (see Section "4. Axis").

The spindle movement takes place parallel to any other axis movements in the same block. This block is ended when both movements are finished.

Programming

SPOS=...	; Absolute position: 0 ... <360 degrees
SPOS=ACP(...)	; Absolute dimensions, approach position in positive direction
SPOS=ACN(...)	; Absolute dimensions, approach position in negative direction
SPOS=IC(...)	; Incremental dimensions, leading sign determines the traversal direction
SPOS=DC(...)	; Absolute dimensions, approach position directly (on the shortest path)

Programming example

```
N10 SPOS=14.3                ; Spindle position 14.3 degrees
...
N80 G0 X89 Z300 SPOS=25.6    ; Positioning spindle with axis movements
                             ; This block is ended when all movements have
                             finished
N81 X200 Z300                ; The N81 block only begins once the spindle position
                             from N80 is reached
```

10.4.4 Gear stages

Function

Up to 5 gear stages can be configured for a spindle for speed / torque adaptation. The selection of a gear stage takes place in the program via M commands (see Section "Miscellaneous function M"):

- M40: Automatic gear stage selection
- M41 to M45: Gear stage 1 to 5

10.4.5 2. Spindle

Function

With SINUMERIK 802D sl plus and 802D sl pro, a 2nd spindle is provided.

For these control systems, the kinematic transformation functions for grinding are possible. These functions require a second spindle for the driven workpiece.
The main spindle is operated as a rotary axis in these functions.

Master spindle

A series of functions is associated with the master spindle that can only be used with this spindle:

- G95 ; Rev. feedrate
- G96, G97 ; Constant cutting rate
- LIMS ; upper speed limit for G96, G97
- M3, M4, M5, S... ; simple specifications for direction of rotation, stop and speed

The master spindle is defined via configuration (machine data). Generally it is the main spindle (spindle 1). A different spindle can be defined as master spindle in the program:

- SETMS(n) ; spindle n (= 1 or 2) is the master spindle as of now.

Switching back can also be performed via:

- SETMS ; configured master spindle is now master spindle again
- SETMS (1) ; Spindle 1 is now master spindle again.

The definition of the master spindle changed in the program is only valid until End of program/program abort. Thereafter, the configured master spindle is again active.

Programming via spindle number

Some spindle functions can also be selected via the spindle number:

```
S1=..., S2=...              ; Spindle speed for spindle 1 or 2
M1=3, M1=4, M1=5            ; Specifications for direction of rotation, stop for
                            spindle 1
M2=3, M2=4, M2=5            ; Specifications for direction of rotation, stop for
                            spindle 2
M1=40, ..., M1=45           ; gear stages for spindle 1 (if available)
M2=40, ..., M2=45           ; gear stages for spindle 2 (if available)
SPOS [n]                    ; Position spindle n
SPI(n)                      ; Converts spindle number n to axis identifier,
                            ; e.g. "SP1" or "CC"
                            ; n must be a valid spindle number (1 or 2)
                            ; The functions of spindle identifiers SPI(n) and Sn are
                            identical.
$P_S[n]                     ; Last programmed speed of spindle n
$AA_S[n]                    ; Actual speed of spindle n
$P_SDIR[ n ]                ; Last programmed direction of rotation of spindle n
$AC_SDIR[ n ]               ; Current direction of rotation of spindle n
```

2 spindles installed

The following can be interrogated in the program via the system variable:

```
$P_NUM_SPINDLES             ; Number of configured spindles (in the channel)
$P_MSNUM                    ; Number of programmed master spindle
$AC_MSNUM                   ; Number of the active master spindle
```

10.5 Special functions

10.5.1 Constant cutting rate: G96, G97

Requirement

A controlled spindle must be present.

Functionality

With activated G96 function, the spindle speed is adapted to the currently machined workpiece diameter (transverse axis) such that a programmed cutting rate S remains constant on the tool edge:
Spindle speed times diameter = constant.

The S word is evaluated as the cutting rate as of the block with G96. G96 is modally effective until cancellation by another G function of the group (G94, G95, G97).

Programming

```
G96 S... LIMS=... F...      ; Constant cutting speed ON
G97                         ; Constant cutting speed OFF

S                           ; Cutting rate in m/min.
LIMS=                       ; Upper limit speed of the spindle with G96, G97
                            effective
F                           ; feedrate in mm/revolution –as for G95
```

Remark:
If G94 instead of G95 was active before, a new appropriate F value must be written!

Rapid traverse

With rapid traverse G0, there is no change in speed.

Exception: If the contour is approached at rapid traverse and the next block contains an interpolation type G1 or G2, G3, CIP, CT (contour block), then the speed for the contour block is applied already in the approach block with G0.

Upper speed limit LIMS=

During machining from large to small diameters, the spindle speed can increase significantly. In this case, it is recommended the upper spindle speed limitation LIMS=... . LIMS is only effective with G96 and G97.
By programming LIMS=..., the value entered into the setting data (SD 43230: SPIND_MAX_VELO_LIMS) is overwritten. This SD takes effect when LIMS is not written.
The upper limit speed programmed with G26 or defined via machine data cannot be overwritten with LIMS=.

Deactivate constant cutting rate: G97

The function "Constant cutting rate" is deactivated with G97. If G97 is active, a programmed **S word** is given in RPM as the **spindle speed** .
If no new S word is programmed, the spindle turns at the last defined speed with G96 function active.

Programming example

```
N10 ... M3                 ; Spindle's direction of rotation
N20 G96 S120 LIMS=2500     ; Activate constant cutting speed, 120 m/min, speed
                           limit 2,500 r.p.m.
N30 G0 X150                ; no change in speed, because block N31 with G0
N31 X50 Z...               ; no change in speed, because block N32 with G0
N32 X40                    ; Approach on contour, new speed is automatically set as
                           is required for the beginning of block N40
N40 G1 F0.2 X32 Z...       ; Feedrate 0.2 mm/revolution
...
N180 G97 X... Z...         ; Deactivating constant cutting rate
N190 S...                  ; new spindle speed, r.p.m.
```

Information

The G96 function can also be deactivated with G94 or G95 (same G group). In this case, the last **programmed** spindle speed S is active for the remaining machining sequence if no new S word is programmed.

The programmable offset TRANS or ATRANS (see section of that name) should not be used on the transverse axis X or used only with low values. The workpiece zero point should be located at the turning center. Only then is the exact function of G96 guaranteed.

10.5.2 Rounding, chamfer

Functionality

You can insert the chamfer (CHF or CHR) or rounding (RND) elements into a contour corner. If you wish to round several contour corners sequentially in the same manner, use the "Modal rounding" (RNDM) command.
You can program the feedrate for the chamfer/rounding with FRC=... (blockwise) or FRCM= ... (modal). If FRC/FRCM is not programmed, the normal feedrate F is applied.

Programming

```
CHF=…                     ; Insert chamfer, value: Length of chamfer
CHR=...                   ; Insert chamfer, value: Side length of the chamfer
RND=...                   ; Insert rounding, value: Radius of chamfer
RNDM=...                  ; Modal rounding:
                          ; Value >0: Radius of rounding, modal rounding ON
                          : This rounding is inserted in all contour corners.
                          : Value = 0: Modal rounding OFF...
FRC=…                     ; Non-modal feedrate for chamfer/rounding ,
                          Value = 0, feedrate in mm/min (G94) or mm/rev (G95)
FRCM=…                    ; Modal feedrate for chamfer/rounding:
                          ; value >0: Feedrate in mm/min (G94) or mm/rev. (G95),
                          ; modal feedrate for chamfer/rounding ON
                          ; value =0: Modal feedrate for chamfer/rounding OFF
                          ; The feedrate F applies for the chamfer/rounding.
```

Information

The appropriate instruction CHF= ... or CHR=... or RND=... or RNDM=... is written in the block with axis movements leading to the corner.

The programmed value for chamfer and rounding is automatically reduced if the contour length of an involved block is insufficient.
No chamfer/rounding is inserted if

- more than three blocks in the connection are programmed that do not contain any information for traversing in the plane,
- or a plane change is carried out.

F, FRC, FRCM are not active when a chamfer is traversed with G0.

If the feedrate F is active for chamfer/rounding, it is by default the value from the block which leads away from the corner. Other settings can be configured via machine data.

Chamfer CHF or CHR

A linear contour element is inserted between **linear and circle contours** in any combination. The edge is broken.

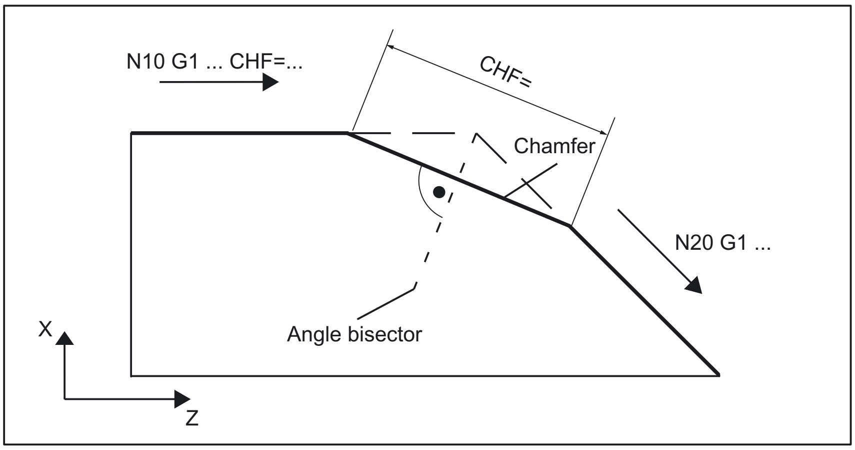


Figure 10-29 Inserting a chamfer with CHF using the example "Between two straight lines"

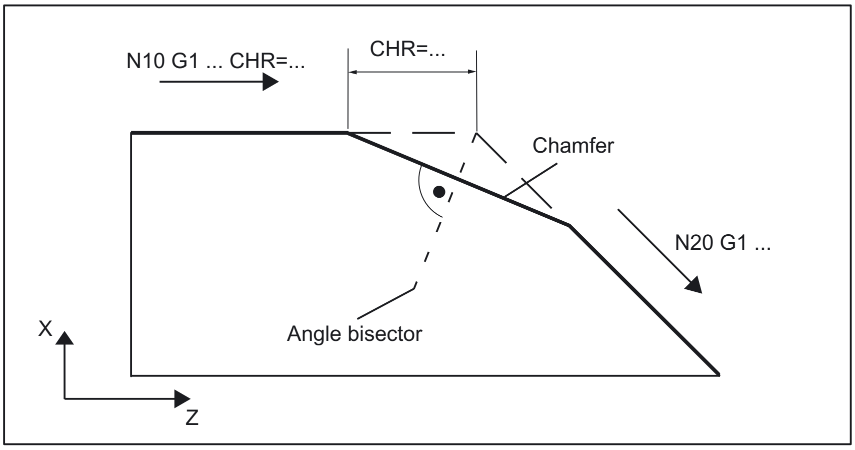


Figure 10-30 Inserting a chamfer with CHR using the example "Between two straight lines"

Programming examples of chamfer

```
N5 F...
N10 G1 X... CHF=5             ; Insert chamfer with chamfer length of 5 mm
N20 X... Z...
...
N100 G1 X... CHR=2            ; Insert chamfer with leg length of 2 mm
N110 X... Z...
...
N200 G1 FRC=200 X... CHR=4    ; Insert chamfer with feedrate FRC
N210 X... Z...
```

Rounding RND or RNDM

A circle contour element can be inserted with tangential connection between the **linear and circle contours** in any combination.

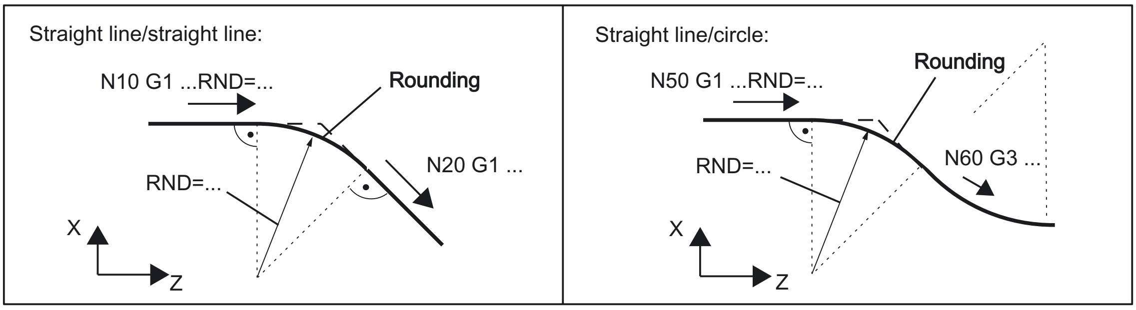


Figure 10-31 Examples for inserting roundings

Programming examples for rounding

```
N5 F...
N10 G1 X... RND=4                ; Insert 1 rounding with radius 4 mm, feedrate F
N20 X... Z...
...
N50 G1 X... FRCM= ... RNDM=2.5   ; Modal rounding, radius 2.5 mm with
                                 ; special feedrate FRCM (modal)
N60 G3 X... Z...                 ; continue inserting this rounding - to N70
N70 G1 X... Z... RNDM=0          ; Modal rounding OFF
...
```

10.5.3 Contour definition programming

Functionality

If the end points for the contour are not directly specified in the machining drawing, it is also possible to use an angle specification ANG=... to determine the straight line. In a contour corner, you can insert the elements chamfer or rounding. The respective instruction CHR= ... or RND=... is written in the block, which leads to the corner.

The blueprint programming can be used in blocks with **G0 or G1** (linear contours).

Theoretically, any number of straight line blocks can be connected and a rounding or a chamfer can be inserted between them. Every straight line must be clearly identified by point values and/or angle values.

Programming

ANG=...	; Angle specification for defining a straight line
RND=...	; Insert rounding, value: Radius of chamfer
CHR=...	; Insert chamfer, value: Side length of the chamfer

Information

The blueprint programming function is executed in the current plane G17 to G19. It is not possible to change the plane during blueprint programming.

If radius and chamfer are programmed in one block, only the radius is inserted regardless of the programming sequence.

Angle ANG

If only one end point coordinate of the plane is known for a straight line or for contours across multiple blocks the cumulative end point, an angle parameter can be used for uniquely definiting the straight line path. The angle is always referred to the abscissa of the current plane G17 to G19, e.g. for G17 on the X axis. Positive angles are aligned counter-clockwise.

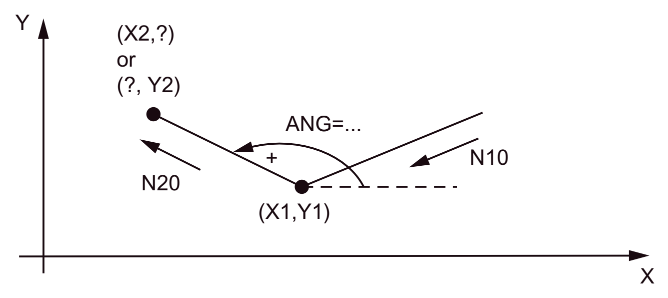
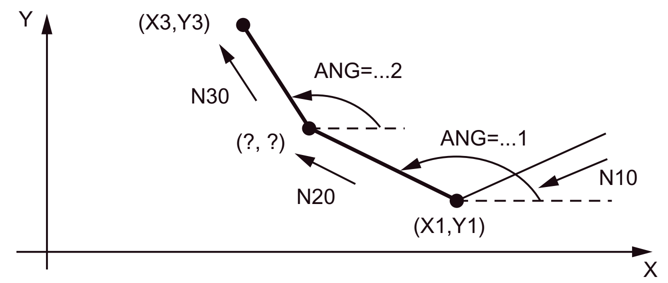
Contour	Programming
	End point in N20 not always known N10 G1 X1 Y1 N20 X2 ANG=... or: N10 G1 X1 Y1 N20 Y2 ANG=... The values are only examples.

Figure 10-32 Specification of an angle for determination of a straight line using the example of the G17 plane

<table>
<tr><th>Contour</th><th>Programming</th></tr>
<tr><td>
</td><td>End point in N20 unknown
N10 G1 X1 Y1
N20 ANG=...1
N30 X3 Y3 ANG=...2

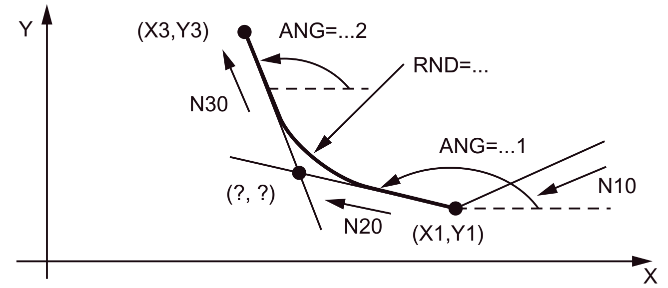
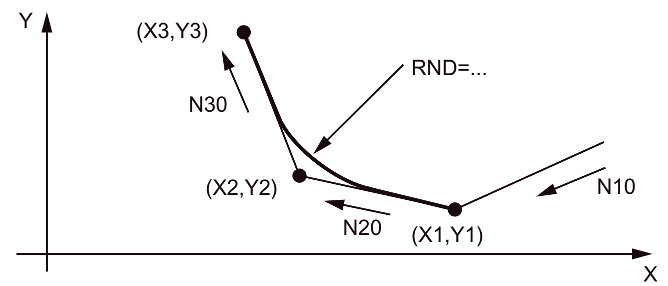
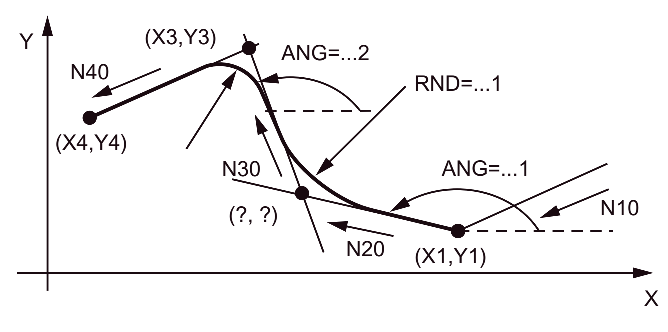
The values are only examples.</td></tr>
<tr><td>
</td><td>End point in N20 unknown,
insert rounding:
N10 G1 X1 Y1
N20 ANG=...1 RND=...
N30 X3 Y3 ANG=...2
analog
Inserting a chamfer:
N10 G1 X1 Y1
N20 ANG=...1 CHR=...
N30 X3 Y3 ANG=...2</td></tr>
<tr><td>
</td><td>End point in N20 known
Inserting a rounding:
N10 G1 X1 Y1
N20 X2 Y2 RND=...
N30 X3 Y3
analog
Inserting a chamfer:
N10 G1 X1 Y1
N20 X2 Y2 CHR=...
N30 X3 Y3</td></tr>
<tr><td>
</td><td>End point in N20 unknown
Inserting a rounding:
N10 G1 X1 Y1
N20 ANG=...1 RND=...1
N30 X3 Y3 ANG=...2 RND=...2
N40 X4 Y4
analog
Inserting a chamfer:
N10 G1 X1 Y1
N20 ANG=...1 CHR=...1
N30 X3 Y3 ANG=...2
N40 X4 Y4
CHR=...2</td></tr>
</table>

Figure 10-33 Multiple block contours using the example of the G17 plane

10.6 Tool and tool offset

10.6.1 General Information

Functionality

During program creation for the workpiece machining, you do not have to take tool lengths or cutting radius into consideration. You program the workpiece dimensions directly, for example following the drawing.

The tool data must be entered separately in a special data area.
In the program, you will merely call the required tool with its offset data. The control system performs the required path compensations based on this data to create the described workpiece. In doing this, automatic compensation of the swivel angle of the grinding wheel takes place via the basic dimension of the tool such that the geometry of the wheel is always entered at under 0 degrees. This also applies to fixed, inclined wheels. The maximum diameter and the wheel width are input into the wheel data image here.

10.6.2 Tool T

Functionality

The tool selection takes place when the T word is programmed. Whether this is a **tool change** or only a **preselection**, is defined in the machine data. For grinding, the tool change (tool call) takes place directly with the T word.

Note:
If a specific tool has been activated, it remains stored as an active tool beyond the end of the program and switching off/on of the control.
If you change a tool manually, enter the change also in the control to make sure the control knows the correct tools. For example, you can start a block with the new T word in MDA mode.

Programming

```
T...                    ; Tool number: 1 ... 32 000
```

Note

In the control system, you can simultaneously store the following maximum values:

- SINUMERIK 802D sl plus: 7 tools with 9 cutting edges each
- SINUMERIK 802D sl pro: 14 tools with 9 cutting edges each.

Programming example

```
N10 T1 D1               ; Tool 1 cutting edge 1
...
N70 T588                ; Tool 588
```

10.6.3 Tool offset number D

Functionality

It is possible to assign 1 to 9 data fields with different tool offset blocks (for multiple cutting edges) to a specific tool. If a special cutting tool is required, it can be programmed with D and the corresponding number.
If no D word is written, D1 is automatically effective.
If D0 is programmed, the offsets for the tool are ineffective.

The tool radius offset numbers are automatically generated when a tool is created (all 9 cutting edges). The cutting edges of the tool have a fixed meaning (geometric position on the grinding wheel). Cutting edges 1, 3 and 5 describe the left wheel edge, cutting edges 2, 4 and 6 describe the right wheel edge for standard contours.

The same applies for all contours (including free contours) when compensating the dressing amount, which means that the odd numbers are left (negative wear value) and the even numbers are right (positive wear value). The wear in direction X (diameter) is the same for all points (negative for grinding direction in negative direction). Cutting edges 7 to 9 are the three possible dressers of a wheel. They are permanently assigned to the sections of the wheel.

Dresser 1 (D7)	Left wheel edge
Dresser 2 (D8)	Right wheel edge
Dresser 3 (D9)	Optional for the diameter and if dresser 1 or 2 is not used.

Option: If the dresser is a diamond roller dresser, which only performs immersion dressing, dresser 1 is always significant here. The other dressers are not used.

Programming

```
D...        ; Tool offset number: 1 ... 9, D0: No compensations active!
```

Information

The tool offsets of the T/D fields have permanent meanings that are entered in the tool management. A list of the parameters appears later in this Section.

Tool length compensations become effective immediately when the tool is active; when no D number was programmed with the values of D1.
The compensation is retracted with the first programmed traversing of the associated length compensation axis.

A tool radius compensation must also be activated by G41/G42.

Programming example

Table 10- 4 Tool change:

```
N10 T1                        ; Tool 1 is activated with the associated D1
N11 G0 X... Z...              ; the length offset compensation is overlaid here
```

```
N50 T4 D2                        ; Load tool 4, D2 from T4 is active
…
N70 G0 Z... D1                   ; D1 for tool 4 active, only cutting edge changed
```

Contents of an compensation memory

- Geometrical dimensions: Length, radius
  These consist of several components (geometry, wear). The control systems computes the components to a certain dimension (e.g. overall length 1, total radius). The respective overall dimension becomes active when the offset memory is activated.
  The way in which these values are computed in the axes is determined by the tool type and the current plane G17, G18, G19.
- The tool type
  The tool type determines which geometry data are required and how they will be computed (wheel types).
- Cutting edge position
  For dressers, you must also enter the cutting edge position.

The following figures provide information on the required tool parameters for the respective tool type.

Entries in the tool parameters	
DP1	403
DP2	Position *
DP3	Length 1
DP4	Length 2
DP6	Radius

* Tool nose position

Wear values correspond to the requirement

Other values should be set to 0

Effect	
G17:	Length 1 in Y Length 2 in X Radius in X/Y
G18:	Length 1 in X Length 2 in Z Radius in Z/X
G19:	Length 1 in Z Length 2 in Y Radius in Y/Z

TPG1	Spindle number
TPG2	Chaining rule
TPG3	Minimum wheel radius
TPG4	Min. wheel width
TPG5	Actual wheel width
TPG6	Maximum speed
TPG7	Max. peripheral speed
TPG8	Angle of the inclined wheel
TPG9	Parameter No. for radius calculation

F: Tool carrier reference point

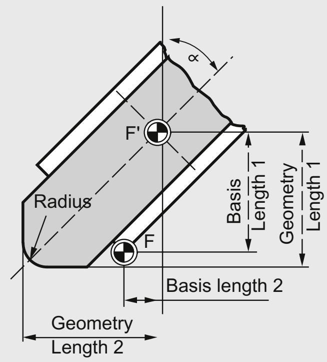


Figure 10-34 Tool types for grinding

10.6.4 Selecting the tool radius compensation: G41, G42

Functionality

A tool with a corresponding D number must be active. The tool radius offset (cutting edge radius offset) is activated by G41/G42. The controller automatically calculates the required equidistant tool paths for the programmed contour for the respective current tool radius. G18 must be active.

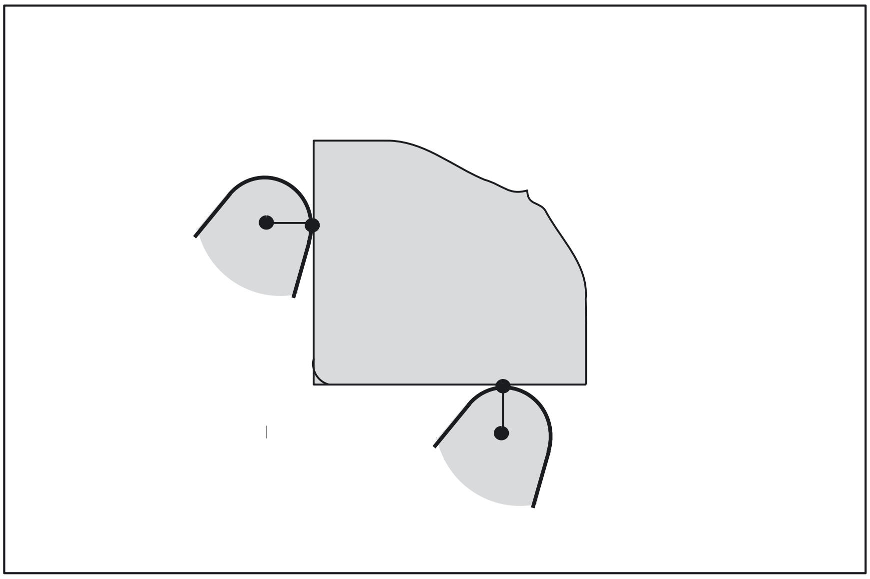


Figure 10-35 Tool radius compensation (cutter radius compensation)

Programming

```
G41 Y... Z...    ; Tool radius compensation left of contour
G42 Y... Z...    ; Tool radius compensation right of contour
```

Remark: The selection can only be made for linear interpolation (G0, G1).

Program both axes. If you only specify one axis, the second axis is automatically completed with the last programmed value.

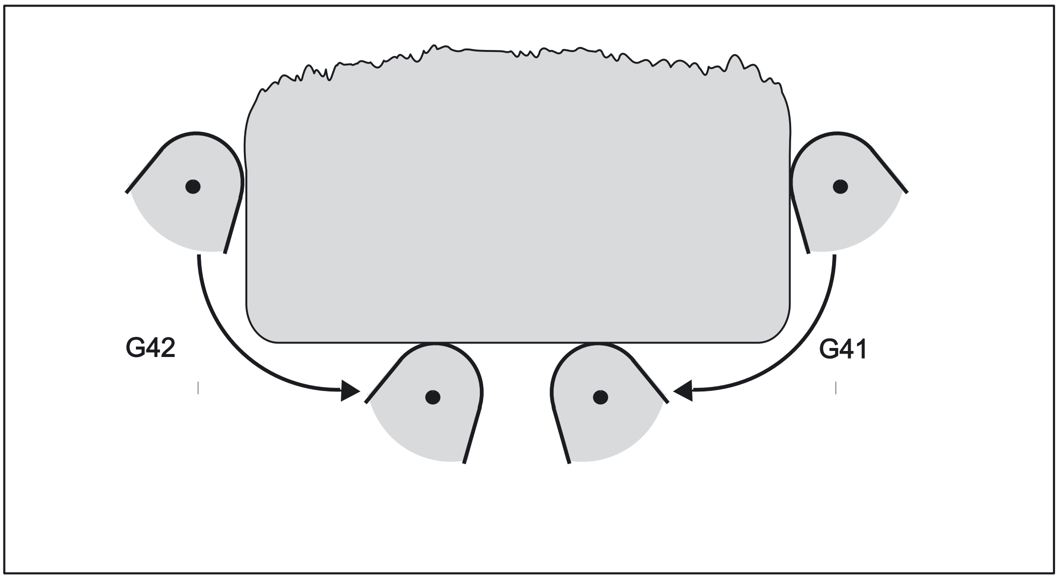


Figure 10-36 Compensation to the right/left of the contour

Starting the compensation

The tool approaches the contour on a straight line and positions itself vertically to the path tangent in the starting point of the contour.
Select the start point so as to ensure collision-free traversing.

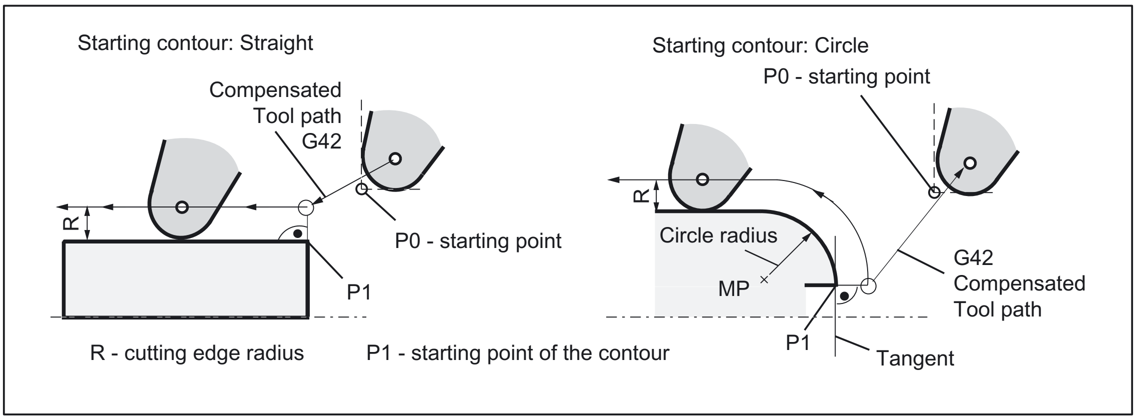


Figure 10-37 Start of the tool radius compensation with the example G42, tool point direction =3

Information

As a rule, the block with G41/G42 is followed by the block with the workpiece contour. However, the contour description may be interrupted by an intervening block that does not contain information for the contour path, e.g. only M command.

Programming example

```
N10 T... F...
N15 Y... Z...                ; P0 - starting point
N20 G1 G42 Y... Z...         ; Selection right of contour, P1
N30 Y... Z... ;              ; Starting contour, circle or straight line
```

10.6.5 Corner behavior: G450, G451

Functionality

Using the functions G450 and G451, you can set the behavior for non-continuous transition from one contour element to another contour element (corner behavior) when G41/G42 is active.
Internal and external corners are detected by the control system automatically. For internal corners, the intersection of the equidistant paths is always approached.

Programming

G450	;Transition circle
G451	;Point of intersection

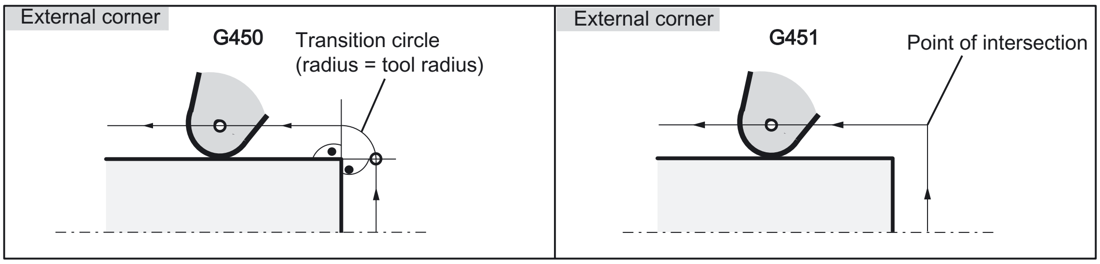


Figure 10-38 Corner behavior at an external corner

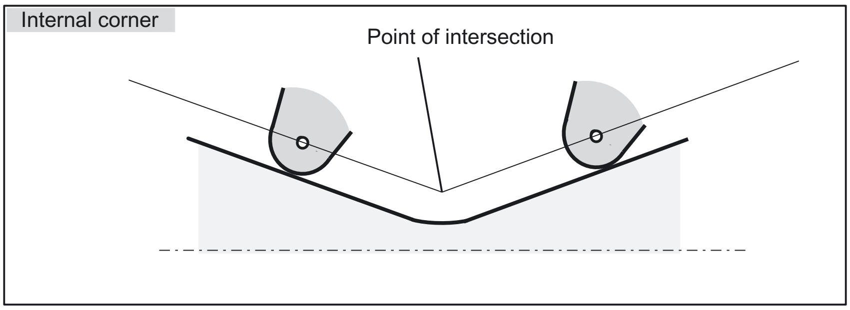


Figure 10-39 Corner behavior at an internal corner

Transition circle G450

The tool center point travels around the workpiece external corner in an arc with the tool radius. In view of the data, for example, as far as the feedrate value is concerned, the transition circle belongs to the next block containing traversing movements.

Point of intersection G451

For a G451 intersection of the equidistant paths, the point (intersection) that results from the center point paths of the tool (circle or straight line) is approached.

10.6.6 Tool radius compensation OFF: G40

Functionality

The compensation mode (G41/G42) is deselected with G40. G40 is also the switch-on position at the beginning of the program.
The tool ends the **block before G40** in the normal end position (compensation vector vertical to the tangent in the end point); independently of the start angle.
If G40 is active, the reference point is the tool tip. The tool tip then travels to the programmed point upon deselection.

Always select the end point of the G40 block such that collision-free traversing is guaranteed!

Programming

```
G40 Y... Z...                ; Tool radius compensation OFF
```

Remark: The compensation mode can only be deselected with linear interpolation (G0, G1).

Program both axes. If you only specify one axis, the second axis is automatically completed with the last programmed value.

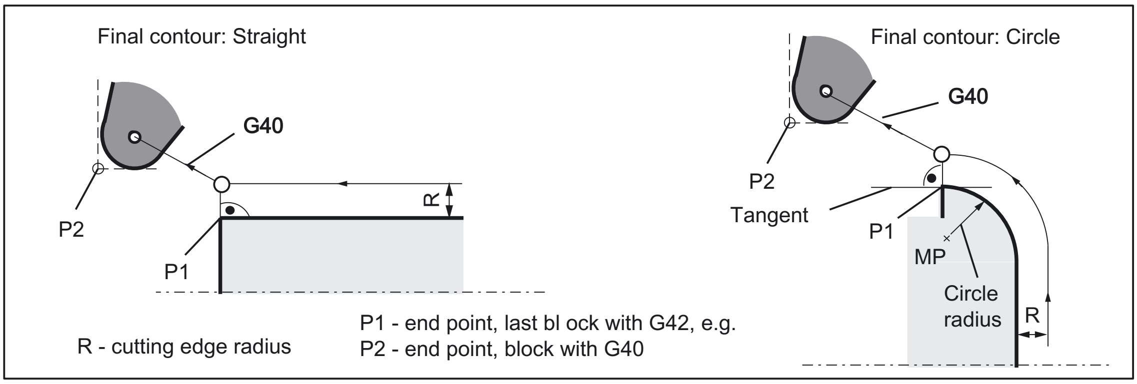


Figure 10-40 Ending the tool radius compensation with G40, with the example of G42, cutting edge position =3

Programming example

```
...
N100 Y... Z...              ;Last block on the contour, circle or straight line, P1
N110 G40 G1 Y... Z...       ;Switch off tool radius compensation,P2
```

10.6.7 Special cases of the tool radius compensation

Change of the compensation direction

The G41 ⇄ G42 compensation direction can be changed without writing G40 in between. The last block that uses the old compensation direction will end at the normal end position of the compensation vector in the end point. The new compensation direction is executed as a compensation start (default setting at starting point).

Repetition of G41, G41 or G42, G42

The same compensation can again be programmed without writing G40 in between. The last block before the new compensation call will end at the normal positon of the compensation vector in the end point. The new compensation is carried out as a compensation start (behavior as described for change in compensation direction).

Changing the offset number D

The offset number D can be changed in the compensation mode. A modified tool radius is active with effect from the block in which the new D number is programmed. Its complete modification is only achieved at the end of the block. In other words: The modification is traversed continuously over the entire block, also for circular interpolation.

Cancellation of compensation by M2

If the offset mode is canceled with M2 (program end) without writing the command G40, the last block with coordinates ends in the normal offset vector setting. No compensating movement is executed. The program ends with this tool position.

Critical machining cases

When programming, pay special attention to cases where the contour path for inner corners is smaller than the tool radius; and smaller than the diameter for two successive inner corners.

Such cases should be avoided.

Also check over multiple blocks that the contour contains no "bottlenecks".

When carrying out a test/dry run, use the largest tool radius you are offered.

Acute contour angles

If very sharp outside corners occur in the contour with active G451 intersection, the control system automatically switches to transition circle. This avoids long idle motions.

10.6.8 Example of tool radius compensation

The wheel should have the contour shown in the figure. Dressing takes place from left to right using MIRROR and G41

Caution: The workpiece zero (XWP) in wheel data must be -110 to be able to program the contour in workpiece coordinates.

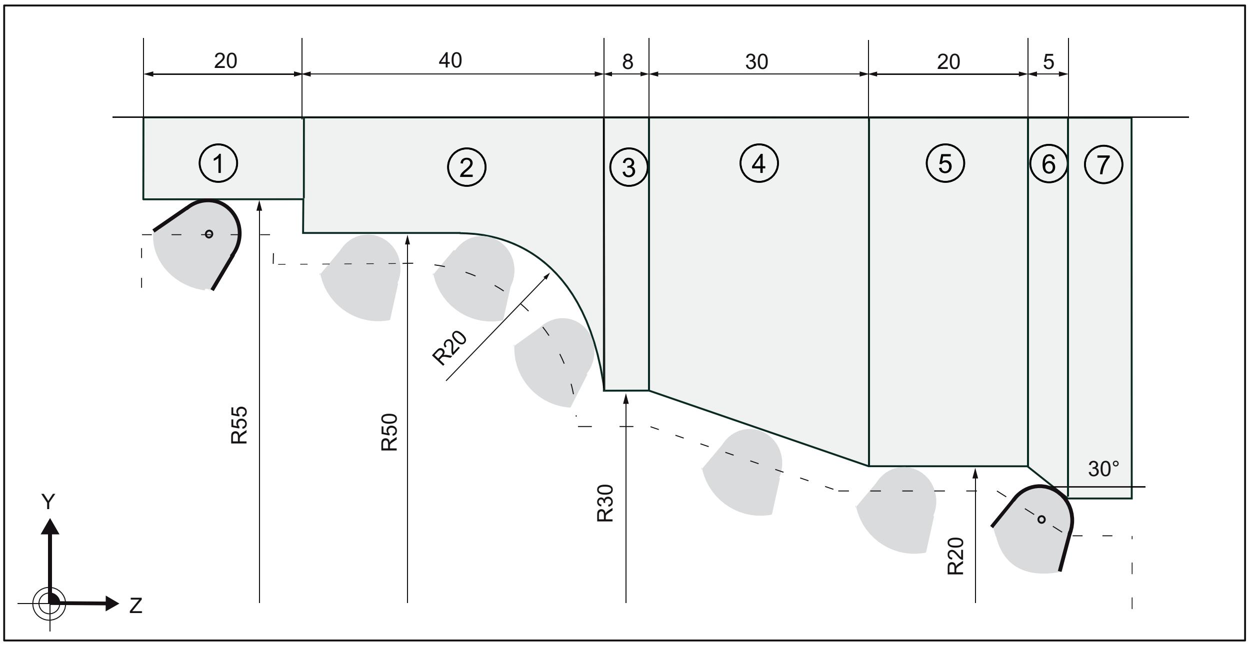


Figure 10-41 Example for contour dressing

```
N1                                  ; Contour cut
N10 F... S... M...                  ; Radius dimension, technological values
N15 G500                            ; Work offset "OFF"
N20 MIRROR X0 Z0                    ; Begin compensation mode
N30 G90 G0 X-90
N40 Z-10
N50 Y110                            ; Approach R55
N60 G41 G64 G1 Z20 F500             ; Dressing contour section ①
N70 Y100
N80 Z60 RND=20                      ; Dressing contour section ①
N90 Y60
N100 Z68                            ; Dressing contour section ①
N110 Y40 Z98                        ; Dressing contour section ①
N120 Z118                           ; Dressing contour section ①
N130 Y30 Z123                       ; Dressing contour section ①
N140 Z123                           ; Dressing contour section ①
N150 G0 Y-90                        ;Move clear
N160 MIRROR                         ; End compensation mode
M17
```

10.7 Miscellaneous function M

Functionality

The miscellaneous function M initiates switching operations, such as "Coolant ON/OFF" and other functionalities.

Various M functions have already been assigned a fixed functionality by the CNC manufacturer. The functions not yet assigned fixed functions are reserved for free use of the machine manufacturer.

Note

An overview of the M miscellaneous functions used and reserved in the control system can be found in section "Overview of instructions".

Programming

```
M...                    ;Max. 5 M functions per block
```

Effect

Activation in blocks with axis movements:

If the functions **M0, M1, M2** are contained in a block with traversing movements of the axes, these M functions become effective **after the traversing movements.**

The functions M3, M4, M5 are output to the internal interface (PLC) before the traversing movements. The axis movements only begin once the controlled spindle has ramped up for M3, M4. For M5, however, the spindle standstill is not waited for. The axis movements already begin before the spindle stops (default setting).

The remaining M functions are output to the PLC with the traversing movements.

If you would like to program an M function directly before or after an axis movement, insert a separate block with this M function. Please note: This block interrupts the G64 continuous path mode and generates exact stop.

Programming example

```
N10 S...
N20 X... M3                   ;M function in the block with axis movement, spindle
                              accelerates before the X axis movement
N180 M78 M67 M10 M12 M37      ;Max. 5 M functions in the block
```

Note

In addition to M and H functions, T, D, and S functions can also be transferred to the PLC (programmable logic controller). In all, a maximum of 10 such function outputs are possible in a block.

Information

With the SINUMERIK 802D sl plus and 802D sl pro, two spindles are possible. This results in an expanded programming capability for the M commands - only for the spindles:

```
M1=3, M1=4, M1=5, M1=40, ...   ; M3, M4, M5, M40, ... for spindle 1
M2=3, M2=4, M2=5, M2=40, ...   ; M3, M4, M5, M40, ... for spindle 2
```

10.8 H function

Functionality

With H functions, floating point data (REAL data type - as with arithmetic parameters, see Section "Arithmetic Parameters R") can be transferred from the program to the PLC.

The meaning of the values for a given H function is defined by the machine manufacturer.

Programming

H0=... to H9999=...	;Max. 3 H functions per block

Programming example

```
N10 H1=1.987 H2=978.123 H3=4      ;3 H functions in block
N20 G0 X71.3 H99=-8978.234        ;With axis movements in block
N30 H5                            ;Corresponds to H0=5.0
```

Note

In addition to M and H functions, T, D, and S functions can also be transferred to the PLC (programmable logic controller). In all, a maximum of 10 function outputs of this type are possible in a part program block.

10.9 Arithmetic parameters (R variables), LUD and PLC variables

10.9.1 Arithmetic parameter R

Functionality

The arithmetic parameters are used if an NC program is not only to be valid for values assigned once, or if you must calculate values. The required values can be set or calculated by the control system during program execution.

Another possibility consists of setting the arithmetic parameter values by operator inputs. If values have been assigned to the arithmetic parameters, they can be assigned to other variable-setting NC addresses in the program.

Programming

R0=... to R299=...	;Assign values to the arithmetic parameters
R[R0]=...	;Indirect programming: Assign a value to the arithmetic parameter R, whose number can be found, e.g. in R0
X=R0	;Assign arithmetic parameters to the NC addresses, e.g. for the X axis

Value assignments

You can assign values in the following range to the R parameters:

±(0.000 0001 ... 9999 9999)
(8 decimal places, arithmetic sign, and decimal point)

The decimal point can be omitted for integer values. A plus sign can always be omitted.

Example:

R0=3.5678 R1=-37.3 R2=2 R3=-7 R4=-45678.123

Use the **exponential notation** to assign an extended range of numbers:

± (10^{-300} ... 10^{+300})

The value of the exponent is written after the **EX** characters; maximum total number of characters: 10 (including leading signs and decimal point)

Range of values for EX: -300 to +300

Example:

```
R0=-0.1EX-5           ;Meaning: R0 = -0.000 001
R1=1.874EX8           ;Meaning: R1 = 187 400 000
```

Note

There can be several assignments in one block incl. assignments of arithmetic expressions.

Assignments to other addresses

The flexibility of an NC program lies in assigning these arithmetic parameters or expressions with arithmetic parameters to other NC addresses. Values, arithmetic expressions and arithmetic parameters can be assigned to all addresses; **Exception: addresses N, G, and L**.

When assigning, write the " = " sign after the address character. It is also possible to have an assignment with a minus sign.

A separate block is required for assignments to axis addresses (traversing instructions).

Example:

```
N10 G0 X=R2            ;Assignment to X axis
```

Arithmetic operations/arithmetic functions

When operators/arithmetic functions are used, it is imperative to use conventional mathematical notation. Machining priorities are set using round brackets. Otherwise, multiplication and division take precedence over addition and subtraction.

Degrees are used for the trigonometrical functions.

Permitted arithmetic functions: see Section "List of instructions"

Programming example: Calculating with R parameters

```
N10 R1= R1+1                       ;The new R1 is calculated from the old R1
                                   plus 1
N20 R1=R2+R3 R4=R5-R6 R7=R8*R9 R10=R11/R12
N30 R13=SIN(25.3)                  ;R13 equals sine of 25.3 degrees
N40 R14=R1*R2+R3                   ; Multiplication and division take precedence
                                   over addition or subtraction R14=(R1*R2)+R3
N50 R14=R3+R2*R1                   ;Result, the same as block N40
N60 R15=SQRT(R1*R1+R2*R2)          ;Meaning:
N70 R1= -R1                        ;The new R1 is the negative old R1
```

Programming example: Assign R parameters to the axes

```
N10 G1 G91 X=R1 Z=R2 F300            ;Separate blocks (traversing blocks)
N20 Z=R3
N30 X=-R4
N40 Z= SIN(25.3)-R5                  ;With arithmetic operations
...
```

Programming example: Indirect programming

```
N10 R1=5                             ;Assigning R1 directly value 5 (integer)
...
N100 R[R1]=27.123                    ;Indirectly assign R5 the value 27.123
```

10.9.2 Local User Data (LUD)

Functionality

The operator/programmer (user) can define his/her own variable in the program from various data types (LUD = Local User Data). These variables are only available in the program in which they were defined. The definition takes place immediately at the start of the program and can also be associated with a value assignment at the same time. Otherwise the starting value is zero.

The name of a variable can be defined by the programmer. The naming is subject to the following rules:

- A maximum of 32 characters can be used.
- It is imperative to use letters for the first two characters; the remaining characters can be either letters, underscore or digits.
- Do not use a name already used in the control system (NC addresses, keywords, names of programs, subroutines, etc.).

Programming / data types

DEF BOOL varname1	;Boolean typ, values: TRUE (=1), FALSE (=0)
DEF CHAR varname2	;Char type, 1 ASCII code character: "a", "b", ...
	;Numerical code value: 0 ... 255
DEF INT varname3	;Integer type, integer values, 32 bit value range:
	;-2 147 483 648 through +2 147 483 647 (decimal)
DEF REAL varname4	;Real type, natural number (like arithmetic parameter R),
	;Value range: ±(0.000 0001 ... 9999 9999)
	;(8 decimal places, arithmetic sign and decimal point) or
	;Exponential notation: ± (10 to power of -300 ... 10 to power of +300)
DEF STRING[string length] varname41	; STRING type, [string length]: Maximum number of characters

Each data type requires its own program line. However, several variables of the same type can be defined in one line.

Example:

```
DEF INT PVAR1, PVAR2, PVAR3=12, PVAR4      ;4 type INT variables
```

Example for STRING type with assignment:

```
DEF STRING[12] PVAR="Hello"                ; Define variable PVAR with a maximum of
                                           12 characters and assign string "Hello"
```

Fields

In addition to the individual variables, one or two-dimensional fields of variables of these data types can also be defined:

```
DEF INT PVAR5[n]                ;One-dimensional field, type INT, n: integer
DEF INT PVAR6[n,m]              ;Two-dimensional field, type INT, n, m: integer
```

Example:

```
DEF INT PVAR7[3]                ;Field with 3 elements of the type INT
```

Within the program, the individual field elements can be reached via the field index and can be treated like individual variables. The field index runs from 0 to a small number of the elements.

Example:

```
N10 PVAR7[2]=24                 ;The third field element (with index 2) is assigned
                                the value 24.
```

Value assignment for field with SET instruction:

```
N20 PVAR5[2]=SET(1,2,3)         ;After the 3rd field element, different values are
                                assigned.
```

Value assignment for field with REP instruction:

```
N20 PVAR7[4]=REP(2)             ;After field element [4] - all are assigned the same
                                value, here 2.
```

10.9.3 Reading and writing PLC variables

Functionality

To allow rapid data exchange between NC and PLC, a special data area exists in the PLC user interface with a length of 512 bytes. In this area, PLC data are compatible in data type and position offset. In the NC program, these compatible PLC variables can be read or written.

To this end, special system variables are provided:

$A_DBB[n]	;Data byte (8-bit value)
$A_DBW[n]	;Data word (16-bit value)
$A_DBD[n]	;Data double-word (32-bit value)
$A_DBR[n]	;REAL data (32-bit value)

"n" stands here for the position offset (start of data area to start of variable) in bytes

Programming example

```
R1=$A_DBR[5]             ;Reading a REAL value, offset 5 (starts at byte 5 of range)
```

Note

The reading of variables generates a preprocessing stop (internal STOPRE).

NOTICE

Writing of PLC tags is generally limited to a maximum of three tags (elements).

Where PLC tags are to be written in rapid succession, one element will be required per write operation.

If more write operations are to be executed than there are elements available, then block transfer will be required (a preprocessing stop may need to be triggered).

Example:

```
$A_DBB[1]=1 $A_DBB[2]=2 $A_DBB[3]=3
STOPRE
$A_DBB[4]=4
```

10.10 Program jumps

10.10.1 Jump destination for program jumps

Functionality

A **label** or a **block number** serve to mark blocks as jump destinations for program jumps. Program jumps can be used to branch to the program sequence.

Labels can be freely selected, but must contain a minimum of 2 and a maximum of 8 letters or numbers of which the **first two characters must be letters** or underscore characters.

Labels that are in the block that serves as the jump destination are **ended by a colon**. They are always at the start of a block. If a block number is also present, the label is located **after the block number**.

Labels must be unique within a program.

Programming example

```
N10 LABEL1: G1 X20            ;LABEL1 is the label, jump destination
...
TR789: G0 X10 Z20             ;TR789 is the label, jump destination
                              - No block number existing
N100 ...                      ;Block number can be jump target
...
```

10.10.2 Unconditional program jumps

Functionality

NC programs process their blocks in the sequence in which they were arranged when they were written.

The processing sequence can be changed by introducing program jumps.

The jump destination can be a block with a **label** or with a **block number**. This block must be located within the program.

The unconditional jump instruction requires a separate block.

Programming

GOTOF label	;Jump forward (in the direction of the last block of the program)
GOTOB label	;Jump backwards (in the direction of the first block of the program)
Label	;Selected string for the label (jump label) or block number

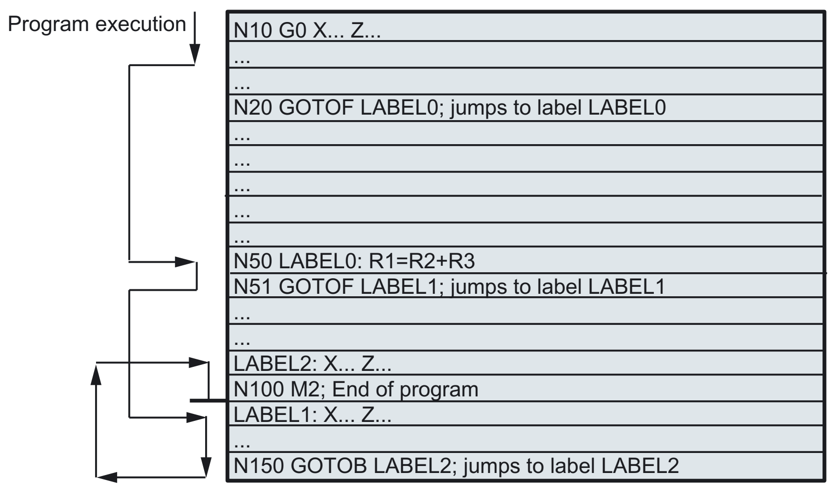


Figure 10-42 Unconditional jumps using an example

10.10.3 Conditional program jumps

Functionality

Jump conditions are formulated after the **IF instruction**. If the jump condition (**value not zero**) is satisfied, the jump takes place.

The jump destination can be a block with a **label** or with a **block number**. This block must be located within the program.

Conditional jump instructions require a separate block. Several conditional jump instructions can be located in the same block.

By using conditional program jumps, you can also considerably shorten the program, if necessary.

Programming

IF condition GOTOF label	;Jump forward
IF condition GOTOB label	;Jump backwards
GOTOF	;Jump direction forward (in the direction of the last block of the program)
GOTOB	;Jump direction backwards (in the direction of the first block of the program)
Label	;Selected string for the label (jump label) or block number
IF	;Introduction of the jump condition
Condition	;Arithmetic parameter, arithmetic expression for formulating the condition

Comparison operations

Operators	Meaning
= =	Equal to
< >	Not equal to
>	greater than
<	less than
> =	greater than or equal to
< =	less than or equal to

The comparison operations support formulating of a jump condition. Arithmetic expressions can also be compared.

The result of comparison operations is "satisfied" or "not satisfied." "Not satisfied" sets the value to zero.

Programming example for comparison operators

```
R1>1                          ;R1 greater than 1
1 < R1                        ;1 less than R1
R1<R2+R3                      ;R1 less than R2 plus R3
R6>=SIN( R7*R7)               ; R6 greater than or equal to SIN (R7) squared
```

Programming example

```
N10 IF R1 GOTOF LABEL1                 ;If R1 is not null then go to the block
                                       having LABEL1
...
N90 LABEL1: ...
N100 IF R1>1 GOTOF LABEL2              ;If R1 is greater than 1 then go to the
                                       block having LABEL2
...
N150 LABEL2: ...
...
N800 LABEL3: ...
...
N1000 IF R45==R7+1 GOTOB LABEL3        ;If R45 is equal to R7 plus 1 then go to the
                                       block having LABEL3
...
Several conditional jumps in the
block:
N10 MA1: ...
...
N20 IF R1==1 GOTOB MA1 IF R1==2 GOTOF MA2 ...
...
N50 MA2: ...
```

Note

The jump is executed for the first fulfilled condition.

10.10.4 Program example for jumps

Task

Approaching points on a circle segment:

Existing conditions:

Start angle: 30° in R1

Circle radius: 32 mm in R2

Position spacing: 10° in R3

Number of points: 11 in R4

Position of circle center in Z: 50 mm in R5

Position of circle center in X: 20 mm in R6

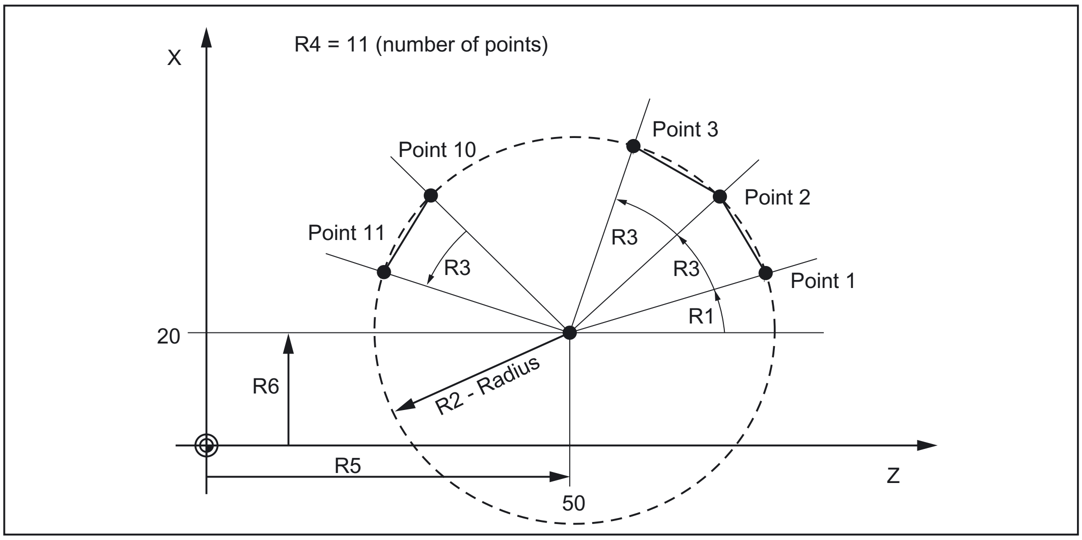


Figure 10-43 Linear approach of points on a circle segment

Programming example

```
N10 R1=30 R2=32 R3=10 R4=11 R5=50 R6=20    ;Assignment of initial values
N20 MA1: G0 Z=R2*COS (R1)+R5               ;Calculation and assignment to axis
X=R2*SIN(R1)+R6                            addresses
N30 R1=R1+R3 R4= R4-1
N40 IF R4 > 0 GOTOB MA1
N50 M2
```

Explanation

In block N10, the starting conditions are assigned to the corresponding arithmetic parameters. The calculation of the coordinates in X and Z and the processing takes place in N20.

In block N30, R1 is incremented by the clearance angle R3, and R4 is decremented by 1.

If R4 > 0, N20 is executed again; otherwise, N50 with End of program.

10.11 Subroutine technique

10.11.1 General information

Usage

Basically, there is no difference between a main program and a subroutine.

Frequently recurring machining sequences are stored in subroutines, e.g certain contour shapes. These subroutines are called at the appropriate locations in the main program and then executed.

One form of a subroutine is the **machining cycle**. Machining cycles contain universally valid machining scenarios. By assigning values via included transfer parameters, you can adapt the subroutine to your specific application.

Layout

The structure of a subroutine is identical to that of a main program (see Section "Program structure"). Like main programs, subroutines contain **M2 - end of program** in the last block of the program sequence. This means a return to the program level where the subroutine was called from.

End of program

The end instruction **RET** can also be used instead of the M2 program end in the subroutine.

RET must be programmed in a separate block.

The RET instruction is used when G64 continuous-path mode is not to be interrupted by a return. With M2, G64 is interrupted and exact stop is initiated.

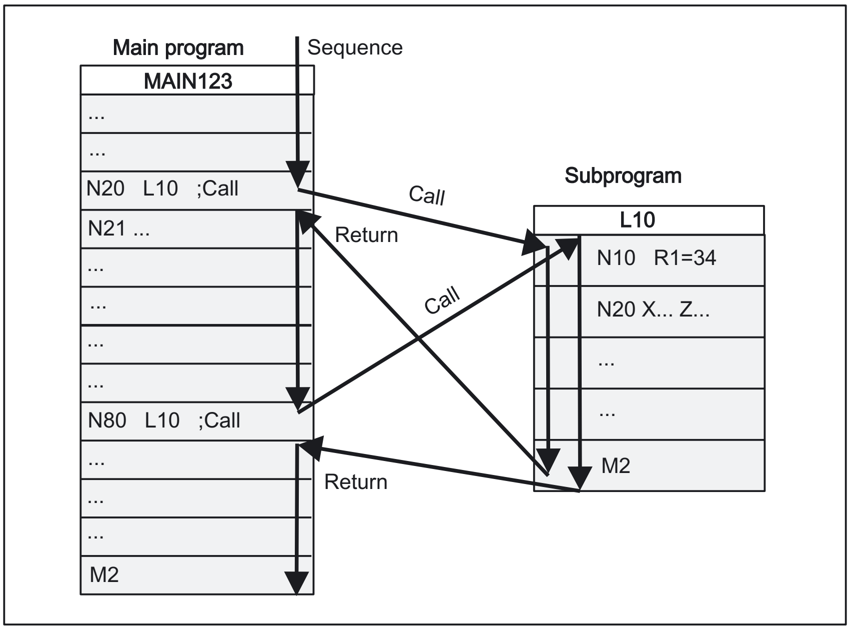


Figure 10-44 Example of a sequence when a subroutine is called in a two-channel manner.

Subroutine name

The subprogram is given a unique name allowing it to be selected from several subroutines. When you create the program, the program name may be freely selected provided the following conventions are observed:

The same rules apply as for the names of main programs.

Example: **BUCHSE7**

It is also possible to use the address word **L...** in subroutines. The value can have 7 decimal places (integers only).

Please observe: With address L, leading zeros are meaningful for differentiation.

Example: **L128** ist nicht **L0128** oder **L00128** !
Dies sind 3 verschiedene Unterprogramme.

Note: The subroutine name **LL6** is reserved for tool change.

Subroutine call

Subroutines are called in a program (main or subprogram) with their names. To do this, a separate block is required.
Example:

```
N10 L785                    ; Subprogram call L785
N20 SHAFT7                  ; Subprogram call SHAFT7
```

Program repetition P...

If a subroutine is to be executed several times in succession, write the number of times it is to be executed in the block of the call after the subroutine name under the **address P**. A maximum of **9,999 cycles** are possible (P1 ... P9999).
Example:

```
N10 L785 P3                 ; Subprogram call L785, 3 cycles
```

Nesting depth

Subroutines can also be called from a subroutine, not only from a main program. In total, up to **8 program levels** are available for this type of nested call, including the main program level.

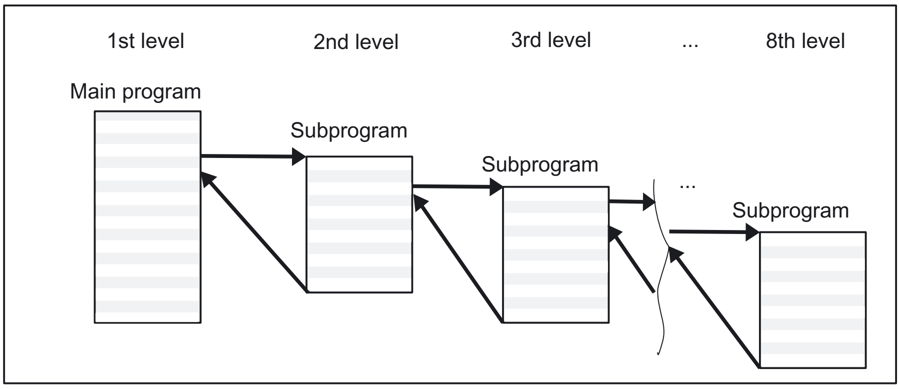


Figure 10-45 Execution with 8 program levels

Information

Modal G functions can be changed in the subroutine, e.g. G90 -> G91. When returning to the calling program, ensure that all modal functions are set the way you need them to be.

Please make sure that the values of your arithmetic parameters used in upper program levels are not inadvertently changed in lower program levels.

When working with SIEMENS cycles, up to 7 program levels are needed.

10.11.2 Calling machining cycles

Functionality

Cycles are technology subroutines that implement a certain machining process in a universally valid way. Adaptation to the particular problem is performaed directly via supply parameters/values when calling the respective cycle.

Programming example

```
N10 CYCLE83(110, 90, ...)          ; Call of cycle 83, transfer values directly,
                                   ; separate block
…
N40 RTP=100 RFP= 95.5 ...          ; Set transfer parameters for cycle 82
N50 CYCLE82(RTP, RFP, ...)         ; Call of cycle 82, separate block
```

10.12 Timers and workpiece counters

10.12.1 Runtime timer

Functionality

The timers are prepared as system variables ($A...) that can be used for monitoring the technological processes in the program or only in the display.

These timers are read-only. There are timers that are always active. Others can be deactivated via machine data.

Timers - always active

- $AN_SETUP_TIME

  Time since the last control powerup with default values (in minutes)

  It is automatically reset in the case of a "Control power-up with default values".

- $AN_POWERON_TIME

  Time since the last control powerup (in minutes)

  It is reset to zero automatically with each power-up of the control system.

Timers that can be deactivated

The following timers are activated via machine data (default setting).

The start is timer-specific. Each active run-time measurement is automatically interrupted in the stopped program state or for feedrate-override-zero.

The behavior of the activated timers for active dry run feedrate and program testing can be specified using machine data.

- $AC_OPERATING_TIME

  Total execution time in seconds of NC programs in the automatic mode

  In the AUTOMATIC mode, the runtimes of all programs between NC START and end of program / RESET are summed up. The timer is zeroed with each power-up of the control system.
- $AC_CYCLE_TIME

  Runtime of the selected NC program (in seconds)

  The runtime between NC Start and End of program / Reset is measured in the selected NC program. The timer is reset with the start of a new NC program.
- $AC_CUTTING_TIME

  Tool action time (in seconds)

  The runtime of the path axes is measured in all NC programs between NC START and end of program / RESET without rapid traverse active and with the tool active (default setting).

  The measurement is interrupted when a dwell time is active.

  The timer is automatically set to zero with each power-up of the control system.

Programming example

```
N10 IF $AC_CUTTING_TIME>=R10 GOTOF WZZEIT            ; Tool operation time limit
                                                     value?
...
N80 WZZEIT:
N90 MSG("Tool action time: Limit value reached")
N100 M0
```

Display

The contents of the active system variables are visible on the screen under
<OFFSET PARAM> -> "Setting data" ">" "Times/counters":

Total run time = $AC_OPERATING_TIME

Program run time = $AC_CYCLE_TIME

Feedrate run time = $AC_CUTTING_TIME

Time since cold restart = $AN_SETUP_TIME

Time since warm restart= $AN_POWERON_TIME

"Program run time" is also visible in the AUTOMATIC mode in the "Position" operating area in the information line.

10.12.2 Workpiece counter

Functionality

The "Workpiece counter" function provides counters for counting workpieces.

These counters exist as system variables with write and read access from the program or via operator input (observe the protection level for writing!).

Machine data can be used to control counter activation, counter reset timing and the counting algorithm.

Counters

- $AC_REQUIRED_PARTS

  Number of workpieces required (workpiece setpoint)

  In this counter you can define the number of workpieces at which the actual workpiece counter $AC_ACTUAL_PARTS is reset to zero.

  The generation of the display alarm 21800 "Workpiece setpoint reached" can be activated via machine data.
- $AC_TOTAL_PARTS

  Total number of workpieces produced (total actual)

  The counter specifies the total number of all workpieces produced since the start time.

  The counter is automatically set to zero upon every booting of the control system.
- $AC_ACTUAL_PARTS

  Number of actual workpieces (actual)

  This counter registers the number of all workpieces produced since the starting time. When the workpiece setpoint is reached ($AC_REQUIRED_PARTS, value greater than zero), the counter is automatically zeroed.
- $AC_SPECIAL_PARTS

  Number of workpieces specified by the user

  This counter allows users to make a workpiece counting in accordance with their own definition. Alarm output can be defined for the case of identity with $AC_REQUIRED_PARTS (workpiece target). Users must reset the counter themselves.

Programming example

```
N10 IF $AC_TOTAL_PARTS==R15 GOTOF SIST                  ; Count reached?
...
N80 SIST:
N90 MSG("Workpiece setpoint reached")
N100 M0
```

Display

The contents of the active system variables are visible on the screen under
<OFFSET PARAM> -> "Setting data" ">" "Times/counters":

Total parts= $AC_TOTAL_PARTS

Required parts= $AC_REQUIRED_PARTS

Number of parts =$AC_ACTUAL_PARTS, $AC_SPECIAL_PARTS not available for display

"Number of parts" is also visible in the AUTOMATIC mode in the "Position" operating area in the information line.

10.13 Multiple feedrate values in one block

Function

The "Several feedrates in one block" function can be used independent of external analog and/or digital inputs to activate

- Different feedrates of an NC block,
- dwell time, and
- Retraction

in synchronism with the movement.

The hardware input signals are combined in one input byte.

Programming

`F2=... F3=...`	In addition to the path feed, you can program up to 2 further feedrates in the block; non-modal
`ST=...`	Dwell time (for grinding technology: sparking-out time); non-modal
`SR=...`	Return path; non modal. The unit for the retraction path refers to the current valid unit of measurement (mm or inch).
`FMA [2,x] =...` `FMA[3,x]=...`	In addition to the path feed, you can program up to 2 further feedrates per axis in the block; non modal
`STA=...`	Axial dwell time (for grinding technology: sparking-out time); non-modal
`SRA=...`	Axial return path; non-modal

FMA and F value

The axial feedrate (FMA value) or path feedrate (F value) corresponds to 100% feedrate. You can use this function to realize feedrates that are smaller than or equal to the axial feedrate or the path feedrate.

Note

If feedrates, dwell time or return path are programmed for an axis on account of an external input, this axis must in this block must not be programmed as POSA axis (positioning axis over multiple blocks).

Look Ahead is also active for multiple feedrates in one block. In this way, the current feedrate is restricted by the Look Ahead value.

Example of programming path motion

The path feed is programmed under the address F and remains valid until an input signal is present. The numerical expansion indicates the bit number of the input that activates the feedrate when changed:

```
F3=20                       ;3 corresponds to input bit 3
F2=5                        ;2 corresponds to input bit 2
ST=1                        ;Dwell time (s) input bit 1
SR=0.5                      ;Return path (mm) input bit 0
```

Example of programming axial motion

The axial path feed is programmed under the address FA and remains valid until an input signal is present.

FMA[3,x]= to FMA[2,x]= can be used to program up to 2 further feeds per axis in the block. The first expression in the square brackets indicates the bit number of the input; the second the axis for which the feedrate is to apply

```
FMA[3, x]=1000              ;Axial feedrate with the value 1000 for X axis, 3
                            ;corresponds to input bit 3
```

Example of axial dwell time and return path

Dwell time and return path are programmed under the following additional addresses:

```
STA[x]=...                  ;Axial dwell time (s) input bit 1
SRA[x]=...                  ;Axial return path (mm) input bit 0
```

If input bit 1 is activated for the dwell time or bit 0 for the return path, the distance to go for the path axes or the relevant single axes is deleted and the dwell time or return started.

Example of several operations in one block

```
N20 T1 D1 F500 G0 X100      ;Initial setting
N25 G1 X105 F=20 F3=5       ; Roughing with F, finishing withF3,
F2=0.5 ST=1.5               ; smooth-finishing with F2, dwell time 1.5 s
SR= 0.5                     ;return path 0.5 mm
N30 ...
…
```

10.14 Oscillation

Function

An oscillating axis travels back and forth between two reversal points 1 and 2 at a defined feedrate, until the oscillating motion is deactivated.

Other axes can be interpolated as desired during the oscillating motion. A continuous infeed can be achieved via a path movement or with a positioning axis, however, there is **no relationship** between the oscillating movement and the infeed movement.

Properties of asynchronized oscillation

- Asynchronous oscillation is active on an axis-specific basis beyond block limits.
- Block-oriented activation of the oscillation movement is ensured by the parts program.
- Combined interpolation of several axes and superimposing of oscillation paths are not possible.

Programming

The following addresses allow asynchronized oscillation to be activated and controlled from the part program.

The programmed values are entered in the corresponding setting data with block synchronization during the main run and remain active until changed again.

Activate, deactivate oscillation: OS

`OS[axis] = 1:` resistor

`OS[axis] = 0:` switch off

Parameter

`OSP1 [axis]=`	`Position of reversal point 1 (oscillating: left reversal point)`
`OSP2 [axis]=`	`Position of reversal point 2 (oscillating: right reversal point)`
`OST1 [axis]=`	`Stopping time at reversal points in seconds`
`OST2 [axis]=`	
`FA[axis]=`	`Feed for oscillating axis`
`OSCTRL [axis]=`	`(Set, reset options)`
`OSNSC [axis]=`	`Number of sparking-out strokes`
`OSE [axis]=`	`End position`
`OS [axis]=`	`1 = activate oscillation; 0 = deactivate oscillation`

Stopping times at reversal points: OST1, OST2

Hold time	Movement in exact stop area at reversal point
-2	Interpolation continues without wait for exact stop
-1	Wait for exact stop coarse

Hold time	Movement in exact stop area at reversal point
0	Wait for exact stop fine
>0	Wait for exact stop fine and then wait for stopping time

The unit for the stopping time is identical to the stopping time programmed with `G4`.

Example of an oscillating axis that should oscillate between two reversal points

The oscillation axis Z must oscillate between 10 and 100. Approach reversal point 1 with exact stop fine, reversal point 2 with exact stop coarse. Machining is performed with feedrate 250 for the oscillating axis. Three sparking-out strokes must be executed at the end of the machining operation followed by approach by oscillation axis to end position 200. The feedrate for the infeed axis is 1, end of the infeed in X direction is at 15.

```
N20 WAITP(X,Y,Z)                    ;Initial setting
N30 G0 X100 Y100 Z100               ;Switch over in positioning axis
                                    operation
N40 WAITP(X,Z)
N50 OSP1[Z]=10 OSP2[Z]=100 ->       ;Reversal point 1, reversal point 2
-> OSE[Z]=200 ->                    ;End position
-> OST1[Z]=0 OST2[Z]=-1 ->          ;Stopping time at U1: Exact stop fine;
                                    ; stopping time at U2: Exact stop coarse
-> FA[Z]=250 FA[X]=1 ->             ;Feed for oscillating axis, infeed axis
-> OSCTRL[Z]=(4,0) ->               ;Setting options
-> OSNSC[Z]=3                       ;Three spark-out strokes
N60 OS[Z]=1                         ;Start oscillation
N70 POS[X]=15                       ;Starting position X axis
N80 POS[X]=50
N90 OS[Z]=0                         ;Stop oscillation
N100 M30
```

-> can be programmed in a single block.

Description

The following apply to the oscillating axis:

- Every axis may be used as an oscillation axis.
- Several oscillation axes can be active at the same time (maximum: the number of the positioning axes).
- Linear interpolation `G1`is always active for the oscillating axis – irrespective of the G command currently valid in the program.

The oscillating axis can

- act as an input axis for a dynamic transformation
- act as a guide axis for gantry and combined-motion axes
- be traversed
  - without jerk limitation (`BRISK`) or
  - with jerk limitation (`SOFT`) or
  - with acceleration curve with a knee (as positioning axes).

Oscillation reversal points

The current offsets must be taken into account when oscillation positions are defined:

- Absolute specification

```
OSP1[Z]=value 1
```

Position of reversal point = sum of offsets + programmed value

- Relative specification

```
OSP1[Z]=IC(value)
```

Position of reversal point = reversal point 1 + programmed value

Example:

```
N10 OSP1[Z]=100 OSP2[Z]=110
.
.
N40 OSP1[Z]=IC(3)
```

Note

`WAITP (axis):`

- If oscillation is to be performed with a geometry axis, you must enable this axis for oscillation with WAITP.
- When oscillation has finished, this command is used to enter the oscillating axis as a positioning axis again for normal use.

Setting feed, FA

The feedrate is the defined feedrate of the positioning axis. If no feedrate is defined, the value stored in the machine data applies.

Defining the sequence of motions, OSCTRL

The control settings for the movement are set with enable and reset options.

```
OSCTRL[oscillating axis] = (set-option, reset-option)
```

The set options are defined as follows (the reset options deselect the settings):

Reset options

These options are deactivated (only if they have previously been activated as setting options).

Setting options

These options are switched over. When `OSE` (end position) is programmed, option 4 is implicitly activated.

Option value	Meaning
0	When the oscillation is deactivated, stop at the next reversal point (default) only possible by resetting values 1 and 2
1	When the oscillation is deactivated, stop at reversal point 1
2	When the oscillation is deactivated, stop at reversal point 2
3	When the oscillation is deactivated, do not approach reversal point if no spark-out strokes are programmed
4	Approach end position after spark-out
8	If the oscillation movement is canceled by deletion of the distance-to-go: then execute spark-out strokes and approach end position if appropriate
16	If the oscillation movement is canceled by deletion of the distance-to-go: reversal position is approached as with deactivation
32	New feed is only active after the next reversal point
64	FA equal to 0, FA = 0: Path overlay is active FA not equal to 0, FA <> 0: Speed overlay is active
128	For rotary axis DC (shortest path)
256	0=The sparking out stroke is a dual stroke (default). 1=Single stroke.

Several options are appended with plus characters.

Example:

The oscillating motion for the Z axis should stop at the reversal point 1 when switched off. Where

- an end position is approached,
- a changed feed acts immediately and should immediately stop the axis after the deletion of distance-to-go.

```
OSCTRL[Z] = (1+4,16+32+64)
```

11 Network operation

11.1 Network operation prerequisites

Introduction

A network function is available for communicating between the control system and a PG/PC.

Prerequisites

The RCS802 tool is required on the PG/PC for communication.

For connecting the control system via the network, various options are available.

These options are described in the chapters "RCS tool" and "Network operation".

The connections are enabled via the following control system interfaces:

- RS232 interface
- Ethernet peer-to-peer interface
- Interface Ethernet network (available only for SINUMERIK 802D sl)

11.2 RCS802 tool

With the RCS802 tool (Remote Control System), you have a tool for your PG/PC that will support you in your daily work with SINUMERIK 802D sl.

The RCS802 tool is part of the SINUMERIK802Dsl and is supplied as CD with each control.

You can connect the control system and the PG/PC using the following interfaces:

Table 11- 1 Interfaces

Interfaces	SINUMERIK 802D sl	RCS802 on PG/PC
RS232	Is available for value, plus and pro.	Are available.
Peer-to-peer Ethernet	Is available for value, plus and pro.	Are available.
Ethernet network	Only available for SINUMERIK 802D sl pro.	Function that requires a license

Functions of the RCS802 tool with license key

NOTICE
You will only obtain the full functionality of the RCS802 tool after importing the license key RCS802.

Table 11- 2 Functions of the RCS802 tool that require a license

Function	RCS802 tool without license key	RCS802 tool with license key
Managing projects	Yes	Yes
Data exchange with SINUMERIK 802D sl	Yes	Yes
Commissioning SINUMERIK 802D sl	Yes	Yes
Setting-up a share drive	No	Yes
Remote control	No	Yes
Screen shot	No	Yes

RCS802 tool

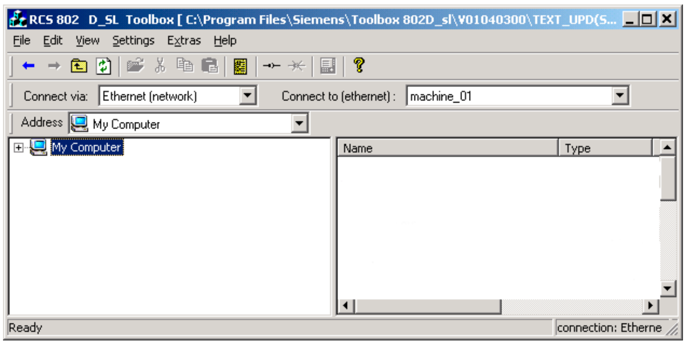


Figure 11-1 Explorer window of the RCS802 tool

After starting the RCS802 tool, you will be in OFFLINE mode. In this mode you only manage files on your PC.

In the ONLINE mode, the directory **Control 802** is also available. This directory makes data exchange with the control system possible. In addition, a remote control function is provided for process monitoring.

The ONLINE connections from the PG/PC to the control are parameterized/activated via the "Setting" > "Connection" menu items in the "Connection Settings" dialog box.

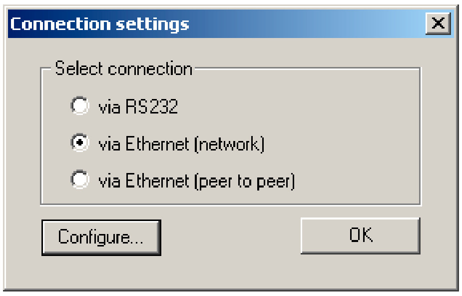
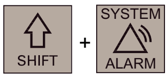


Figure 11-2 Connection Settings

Note

The RCS802 tool includes a detailed online help function. Refer to this help menu for further details e.g. establishing a connection, project management etc.

Operating sequence to make an RS232 connection to the control



- You are now in the <SYSTEM> operating area.

PLC

- Press the "PLC" softkey.

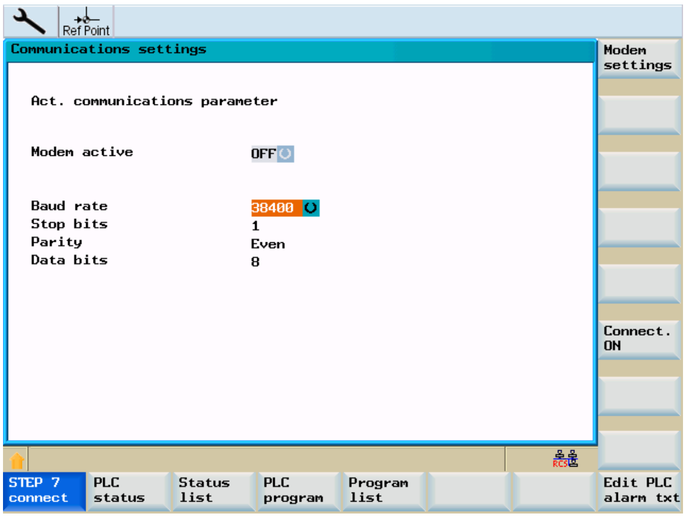


Figure 11-3 Communication settings RS232

Connect to STEP 7

- Set the parameters for communication in the "STEP 7 Connect" dialog.

Connect. on

- Activate the RS232 connection with the "Connect. ON" softkey.

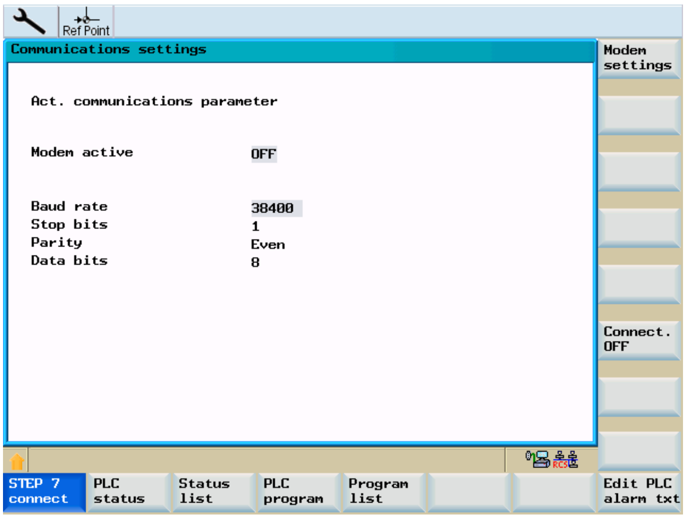


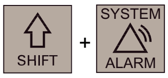
Figure 11-4 RS232 connection active

No modifications to the settings are possible in this state.

The softkey label changes to "Connect. OFF".

In the lower right corner of the screen, the icon shows that the connection to PG/PC via the RS232 interface is active.

Operating sequence to make an Ethernet peer-to-peer connection to the control



- You are now in the <SYSTEM> operating area.

Service display

- Press the softkeys "Service display" >"Service control".

Service control

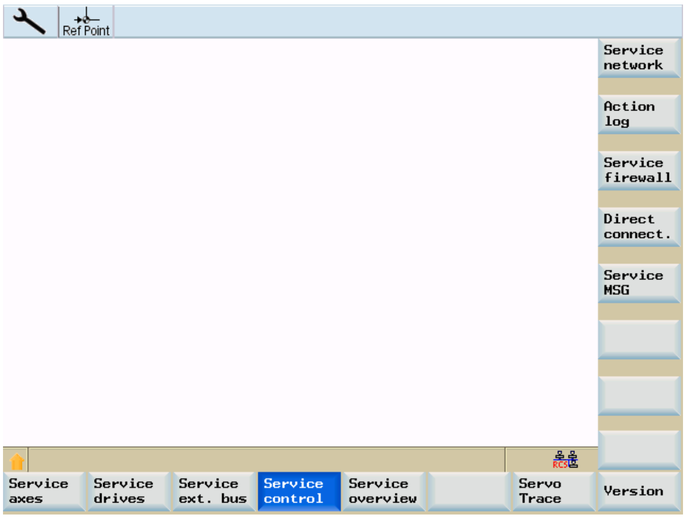


Figure 11-5 "Service control"

Direct connection

- Press the "Direct connect." softkey.

  The following message is shown on the HMI:

  "Connection is set up"

  – IP Address: 169.254.11.22

  – Subnet mask: 255.255.0.0

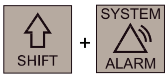
    The IP address and subnet mask shown are fixed values.

    These values cannot be changed.

Direct connection

- You can cancel the Ethernet peer-to-peer connection once more using the "Direct connect." softkey.

Operating sequence to make an Ethernet network connection to the control



- You are now in the <SYSTEM> operating area.

Service display

Service control

- Press the softkeys "Service display" >"Service control".

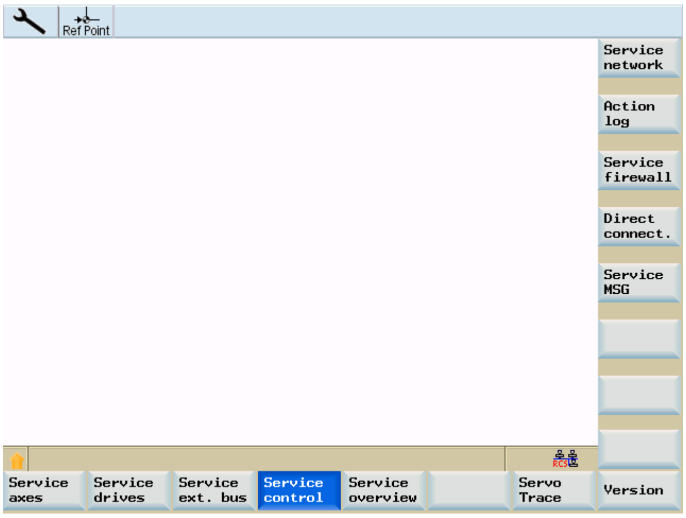


Figure 11-6 "Service control"

Service network

- Press the softkey "Service network" (only available for SINUMERIK 802D sl pro).

Reference

SINUMERIK 802D sl Programming and Operating Manual; Network Operation

11.3 Network operation

Note

The network function is only available for SINUMERIK 802D sl.

Thanks to the integrated network adapter, the control system is network-capable. The following connections are possible:

- Ethernet peer-to-peer: Direct connection between control system and PC using a cross-over cable
- Ethernet network: Integrating the control system into an existing Ethernet network using a patch cable.

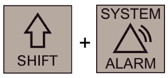
Screened network operation with encrypted data transfer is possible using an 802D specific transmission protocol. This protocol is used, e.g. for transmitting and executing part programs in conjunction with the RCS tool.

11.3.1 Configuring the network connection

Prerequisite

The control system is connected to the PC or the local network via the X5 interface.

Entering network parameters



Switch to the the <SYSTEM> operating area.

Service display

Service control

Press the "Service display" "Service control system" softkeys.

Service network

Select the "Service network" softkey to display the network configuration window.

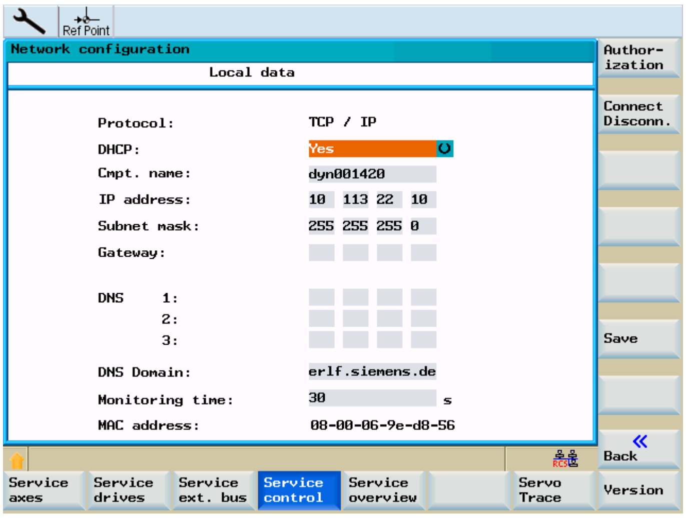


Figure 11-7 "Network configuration" start screen

Table 11- 3 Network configuration required

Parameter	Explanation
DHCP	DHCP log: A DHCP server is needed in the network which dynamically distributes the IP addresses. When **No** is selected fixed network addresses will be assigned. When **Yes** is selected the network addresses are assigned dynamically. Input fields that are no longer needed will be hidden. If you selected "yes", the following steps are necessary to activate the fields for the computer name, IP address and Subnet mask: 1. Press the vertical softkey "Save". 2. Switch the control system off and on again.
Computer name	Name of the control system in the network
IP address	Network address of the control system (e.g. 192.168.1.1)
Subnet mask	Network identification (e.g. 255.255.252.0)

Enabling the communication ports

Service Firewall

Use the "Service Firewall" softkey to enable or disable communication ports.

To ensure maximum possible safety, all ports not needed should be closed.

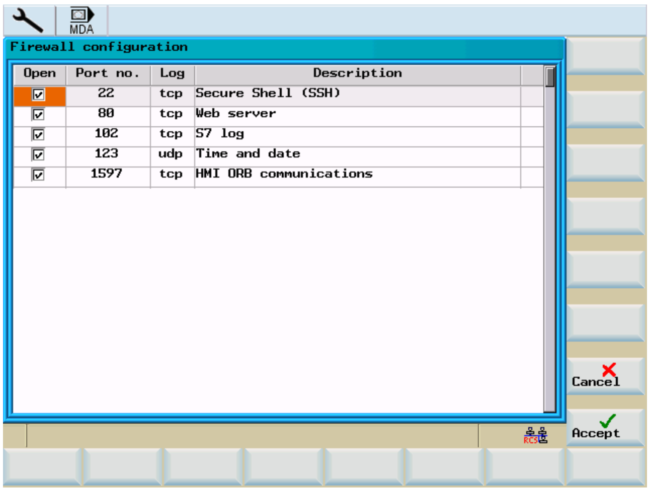


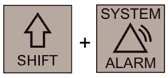
Figure 11-8 Firewall configuration

The RCS network requires the ports 80 and 1597 for communication.

To change the port status, select the relevant port using the cursor. Pressing the <Input> key changes the port status.

Open ports are shown with the checkbox enabled.

11.3.2 User management



Press the "Service display" "Service control system" softkeys in the <SYSTEM> operating area.

Service display

Service control

Service network

Author-ization

Select the "Service network" "Authorization" softkey to display the user account input screen.

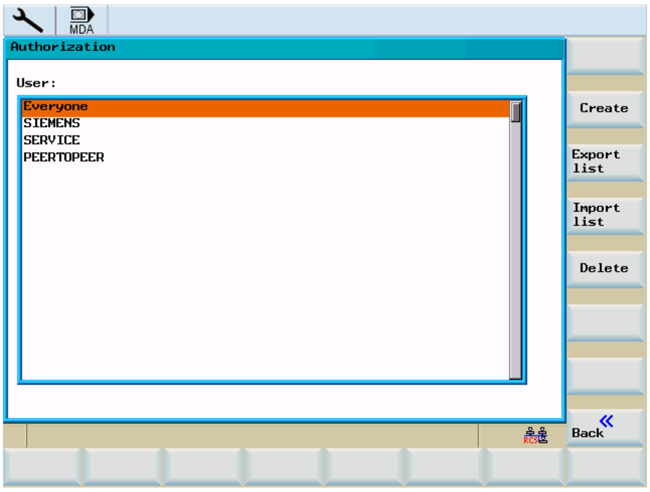


Figure 11-9 User accounts

The user accounts serve for saving personal settings of the users. To create a new account, type the user name and the log-in password in the input fields.

A user account is required for communication between HMI and the RCS tool on the programming device/PC.

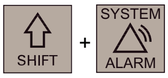
For this purpose the user has to enter this password on the HMI during RCS log-in via network.

This password is required also, if the user want to communicate with the control system from the RCS tool.

Use the "Create" softkey to insert a new user into the user management.

Use the "Delete" softkey to delete the selected user from the user management.

11.3.3 User log in - RCS log in



RCS
log-in

In the <SYSTEM> operating area, select the "RCS Connect" softkey. The user log-in input screen will appear.

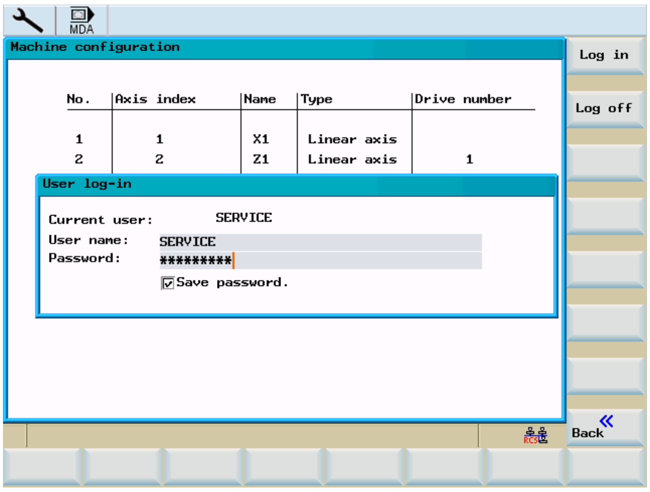


Figure 11-10 User log-in

Logon

Type user name and password into the appropriate input fields and select the "Log in" softkey to confirm your input.

After successful log-in, the user name is displayed in the **Current user** line.

Select the "Back" softkey to close the dialog box.

Note

This log-in simultaneously serves for user identification for remote connections.

Logoff

Press the "Log off" softkey. This will log out the current user, all user-specific settings are saved, and any enables already granted are canceled.

11.3.4 Working on the basis of a network connection

The remote access (access to the control system from a PC or from a network) to the control system is disabled by default.

After log-in of a local user, the following functions are offered to the RCS tool:

- Commissioning functions
- Data transfer (transfer of part programs)
- Remote control for the control system

To grant access to a part of the file system, first share the relevant directories with other users.

Note

If you share directories with other users, the authorized network nodes are granted access to the shared files in the control system. Depending on the sharting option, the user can modify or delete files.

11.3.5 Sharing directories

This function defines the rights for access of remote users to the file system of the control system.

Use the **Program manager** to select the directory you want to share.

Use the "Next..." > "Share" softkeys to open the input screen for sharing the selected directory.

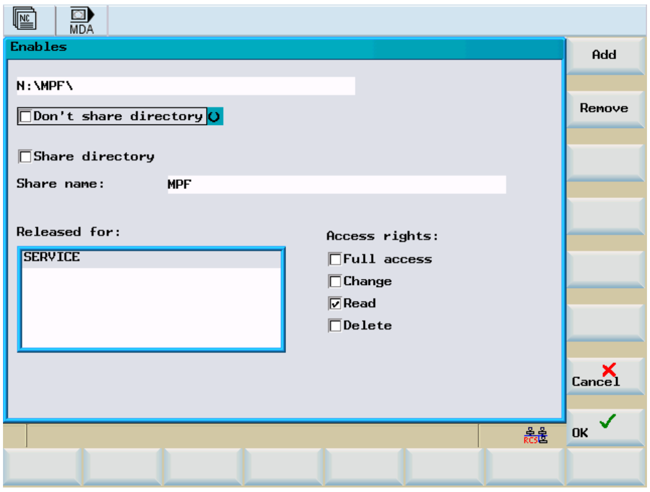
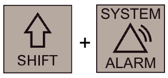


Figure 11-11 Sharing status

- Select the sharing status for the selected directory:
  - **Do not share this directory** Directory will not be shared
  - **Share this directory** The directory will be shared and a share name must be entered.
- Type an identifier into the **Share name** field through which authorized users can access the files in the directory.
- By pressing the "Add" softkey, you arrive at the user list. Select the user. With "Add" you can make any entries in the "Shared" field.
- Define the user rights (**Authorizations**).
  - **Full access** User has full access
  - **Change** User may modify files.
  - **Read** User may read files.
  - **Delete** User may delete files.

By pressing the "OK" softkey the set properties are confirmed. As in Windows, shared directories are marked with a "hand".

11.3.6 Connecting / disconnecting network drives



Press the "Service display" "Service control system" "Service network" softkeys in the <SYSTEM> operating area.

Service display

Service control

Service network

Connect Disconn.

Use "Connect/Disconnect" to enter the network drive configuration area.

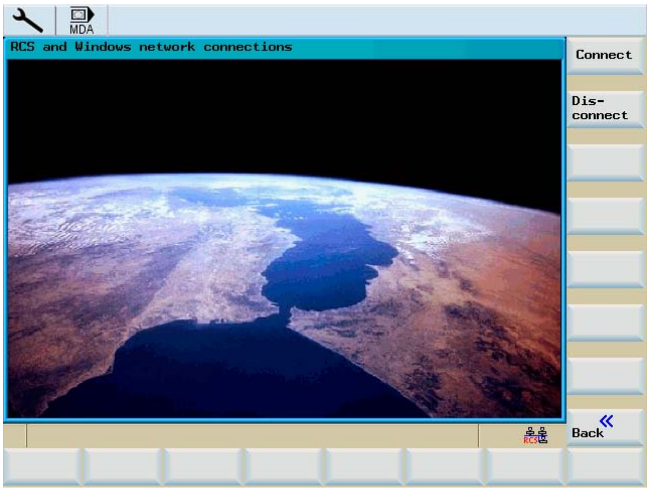


Figure 11-12 Network connections

Connecting network drives

Connect

The "Connect" function is used to assign a local drive to a network drive.

Note

You have shared a directory for a network connection with a certain user on a programming device/PC.

The RCS802 tool includes a detailed online help function. The procedure for using this help function is described in Chapter "RCS802 share drive".

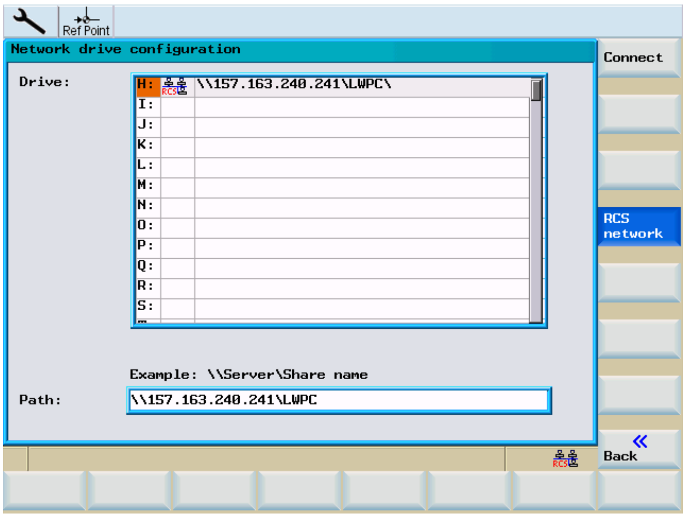


Figure 11-13 Connecting network drives

Sequences of operation for connecting network drives

1. Place the cursor on a free drive.
2. Change to the "Path" input field using the TAB key.

   Specify the IP address of the server and the sharing name.

   Example: \\157.163.240.241\

Connect

Press "Connect".

The server connection is connected with the drive of the control system.

Note

For example, for executing an external subprogram, please see Chapter "Automatic Mode" -> "Execution from external".

Disconnecting network drives

Dis-
connect

By selecting the ">>Back" softkey and the "Disconnect" function you can disconnect an existing network connection.

1. Place the cursor on the relevant drive.
2. Press the "Disconnect" softkey.

   The selected network drive is disconnected from the control.

Data Backup

12

12.1 Data transfer via RS232 interface

Functionality

The RS232 interface of the control system can be used to output data (e.g. part programs) to an external data backup device or to read in data from there. The RS232 interface and your data backup device must be matched with each other.

Operating sequence

PROGRAM MANAGER

You have selected the <PROGRAM MANAGER> operating area and you are in the overview of the NC programs already created.

Select the data to be transmitted using either the cursor or the "Select all" softkey,

Copy

and copy the data to the clipboard.

RS232

Press the "RS232" softkey and select the desired transfer mode.

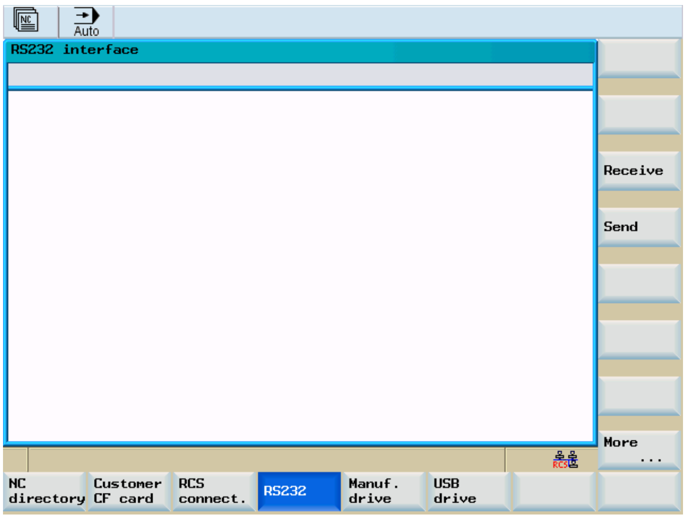


Figure 12-1 Reading out a program

Send

Press "Send" to start the data transfer. All data copied to the clipboard will be transmitted.

Further softkeys

Receive

Load files via the RS232 interface.

More ...

The following function is provided at this level:

Error log

Transmission protocol
This log contains all transmitted files including status information:

- For files to be output
  : name of file
  error log
- For files to be input
  : name of file and path
  error log

Table 12- 1 Transmission messages

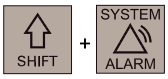
OK	Transmission completed successfully
ERR EOF	End-of-text character received, but archive file incomplete
Time Out	The time monitoring is reporting an interruption of the data transfer
User Abort	Data transfer aborted by the **<Stop>** softkey
Error Com	Error at the COM 1 port
NC / PLC Error	Error message from the NC
Error Data	Data error 1. Files read in with / without header or 2. Files transmitted without file names in the punched-tape format
Error File Name	The file name does not correspond to the name convention of the NC.

12.2 Creating / reading in / reading out a start-up archive

References

SINUMERIK 802D sl Operating Instructions for Turning, Milling, Grinding, Nibbling; Data Backup and Series Start-Up

Operating sequence



Start-up files

Press the "Start-up files" softkey in the <SYSTEM> operating area.

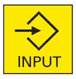
Creating a start-up archive

A start-up archive can be created either with all components or with some selected components.

To create an archive with selected components, the following operator actions are required:

802D data

Press "802D data". Please select the line "Start-up archive (drive/NC/PLC/HMI)" using the direction keys.



Press the "Input" key to open the directory and select the desired lines using the "Select" key.

Copy

Press the "Copy" softkey. The files are copied to the clipboard.

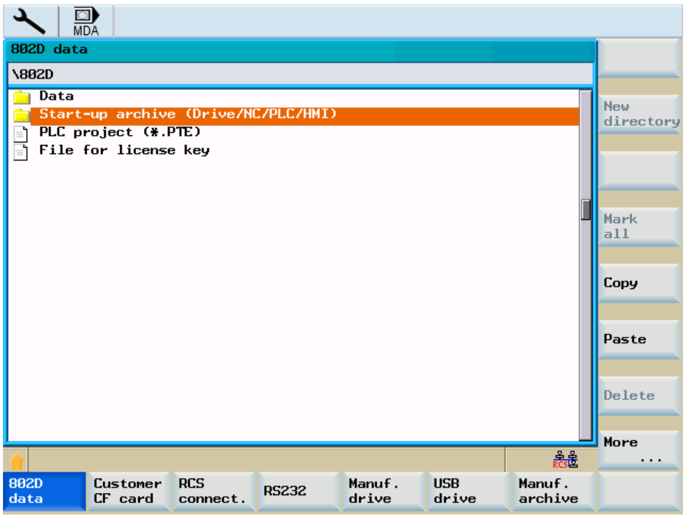


Figure 12-2 Copy entire start-up archive

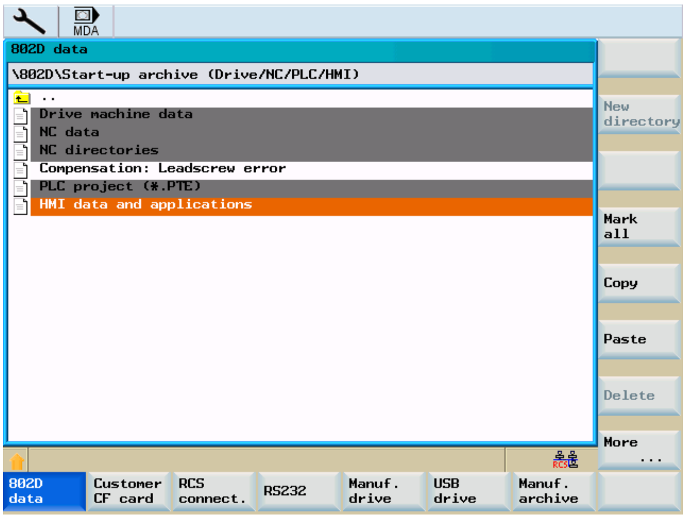


Figure 12-3 Contents of the start-up archive

By pressing the <Select> key, the respective files can be individually selected/deselected in the start-up archive.

Writing the start-up archive to a customer CompactFlash card/USB FlashDrive

Requirement: The CompactFlash Card/USB FlashDrive is inserted, and the start-up archive has been copied to the clipboard.

Operating sequence:

Customer CF card	or
USB drive	Press the "Customer CF card" or "USB drive" softkey. In the directory, select the saving location (directory).
Paste	Use the "Insert" softkey to start writing of the start-up archive.

In the dialog that follows, confirm the name that is specified or enter a new name. Close the dialog box by pressing "OK".

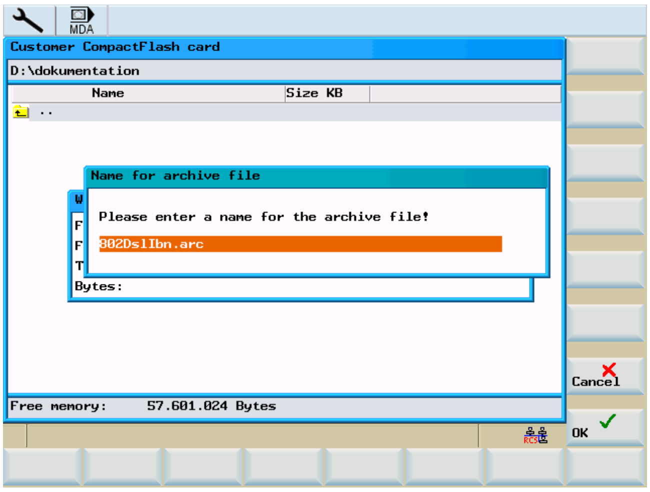


Figure 12-4 Insert files

Reading in start-up archive from customer CompactFlash card/USB FlashDrive

To import a start-up archive, perform the following operator actions:

1. CompactFlash card/USB FlashDrive are inserted
2. Press the "Customer CF card"/"USB drive" softkey and select the line with the desired archive file.
3. Press "Copy" to copy the file to the clipboard.
4. Press the "802D data" softkey and position the cursor on the start-up archive (drive/NC/PLC/HMI) line.
5. Press the "Paste" softkey; commissioning starts.
6. Acknowledge the start dialog on the control system.

12.3 Reading in / reading out PLC projects

When reading in a project, this will be transferred to the file system of the PLC and then activated. To complete the activation, the control system is restarted (warm start).

Reading in project from CompactFlash card/USB FlashDrive

To read in a PLC project, perform the following operator actions:

1. CompactFlash card/USB FlashDrive are inserted
2. Press the "Customer CF card"/"USB drive" softkey and select the line with the desired project file in PTE format.
3. Press "Copy" to copy the file to the clipboard.
4. Press the "802D data" softkey and position the cursor on the **PLC Project (PT802D *.PTE**) line.
5. Press the "Paste" softkey; reading in and activation starts.

Writing project to CompactFlash card/USB FlashDrive

Perform the following operator actions:

1. CompactFlash card/USB FlashDrive are inserted
2. Select the "802D data" softkey and position the direction keys on the **PLC project (PT802D *.PTE)** line.
3. Press "Copy" to copy the file to the clipboard.
4. Press the "Customer CF card"/"USB drive" softkey and select the saving location for the file.
5. Press the "Paste" softkey; the writing process starts.

12.4 Copying and pasting files

In the <PROGRAM MANAGER> operating area and in the "Start-up files" function, files or directories can be copied into another directory or onto a different drive using the softkey functions "Copy" and "Paste". When doing so, the "Copy" function enters the references to the files or directories in a list which is subsequently executed by the "Paste" function. This function will perform the actual copying process.

The list is kept until a new copying process overwrites this list.

Special situation:
If the RS232 interface has been selected as the data target, "Paste" will be replaced by the "Send" softkey function.. When reading in files ("Receive" softkey), it is not necessary to specify a target, since the name of the target directory is not contained in the data flow.

PLC diagnostics

13

Functionality

A PLC user program consists to a large degree of logical operations to realize safety functions and to support process sequences. These logical operations include the linking of various contacts and relays. As a rule, the failure of a single contact or relay results in a failure of the whole system/installation.

To locate causes of faults/failures or of a program error, various diagnostic functions are offered in the "System" operating area.

Operating sequence

SHIFT + SYSTEM ALARM

PLC

Press the "PLC" softkey in the <SYSTEM> operating area.

PLC program

Press "PLC program".

The project stored in the residual memory is opened.

13.1 Screen layout

The screen layout with its division into the main areas corresponds to the layout already described in section "Software Interface".

Any deviations and supplements pertaining to the PLC diagnostics are shown in the following screen.

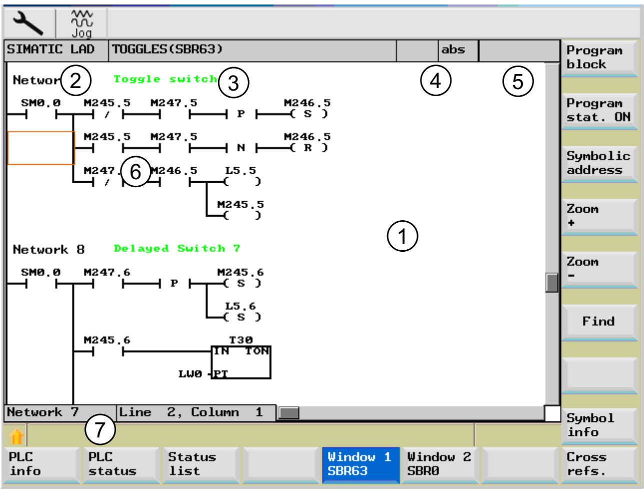


Figure 13-1 Screen layout

Table 13- 1 Key to screen layout

Screen item	Display	Meaning
①	Application area	
②	Supported PLC program language	
③	Name of the active program block Representation: Symbolic name (absolute name)	
④	Program status	
	RUN	Program is running
	STOP	Program stopped
	Status of the application area	
	Sym	Symbolic representation
	abs	Absolute representation
⑤		Display of the active keys
⑥	Focus Performs the tasks of the cursor	
⑦	Tip line contains notes for searching	

13.2 Operating options

In addition to the softkeys and the navigation keys, this area provides still further key combinations.

Hotkeys

The cursor keys move the focus over the PLC user program. When reaching the window borders, it is scrolled automatically.

Table 13- 2 Hotkeys

Keystroke combination	Action
NEXT WINDOW or CTRL ◀	To the first line of the row
END or CTRL ▶	To the last line of the row
PAGE UP	Up a screen
PAGE DOWN	Down a screen
◀	One field to the left
▶	One field to the right
▲	Up a field
▼	Down a field

Keystroke combination	Action
CTRL NEXT WINDOW or CTRL ▲	To the first field of the first network
CTRL END or CTRL ▼	To the last field of the first network
CTRL PAGE UP	Opens the next program block in the same window
CTRL and PAGE DOWN	Opens the previous program block in the same window
SELECT	The function of the Select key depends on the position of the input focus. • Table line: Displays the complete text line • Network title: Displays the network comment • Command: Displays the complete operands
INPUT	If the input focus is positioned on a command, all operands including the comments are displayed.

Softkeys

PLC info

The following PLC properties are shown with this softkey:

- Mode
- Name of the PLC project
- PLC system version
- Cycle time
- Machining time of the PLC user program

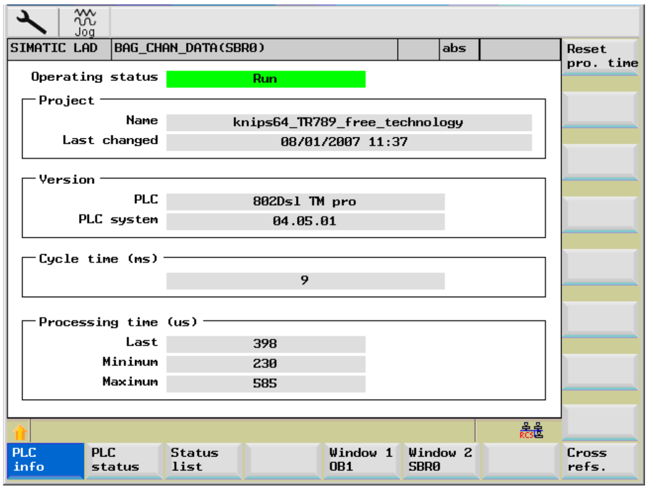


Figure 13-2 PLC info

By pressing the "Reset machining time" softkey, machining time data is reset.

PLC status

The values of the operands can be monitored and changed during program execution using the "PLC status display" window.

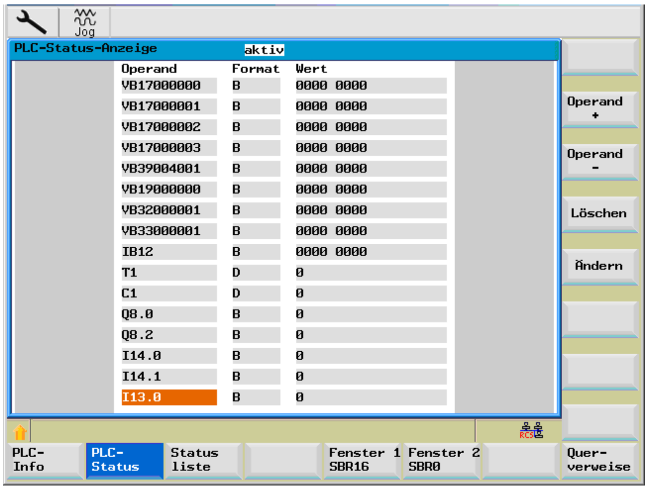


Figure 13-3 PLC status display

Status list

Use the "Status list" softkey to display and modify PLC signals.

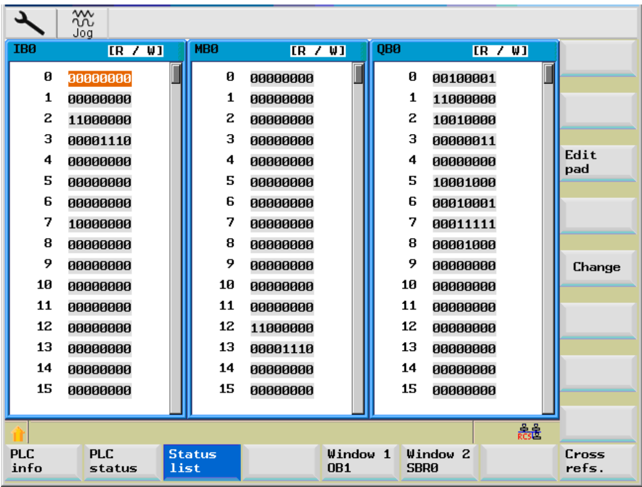


Figure 13-4 Status list

Window 1 OB1

Using the "Window 1 ..." and "Window 2 ..." softkeys you can display any logical and graphical information of a program block. The program block is one of the components of the PLC user program.

The program block can be selected in the "Program list" using the "Open" softkey. The name of the program block will be displayed on the softkey (for "..." e.g. "Window 1 SBR16").

The logics in the ladder diagram (LAD) display the following:

- Networks with program parts and current paths
- Electrical current flow through a number of logical operations

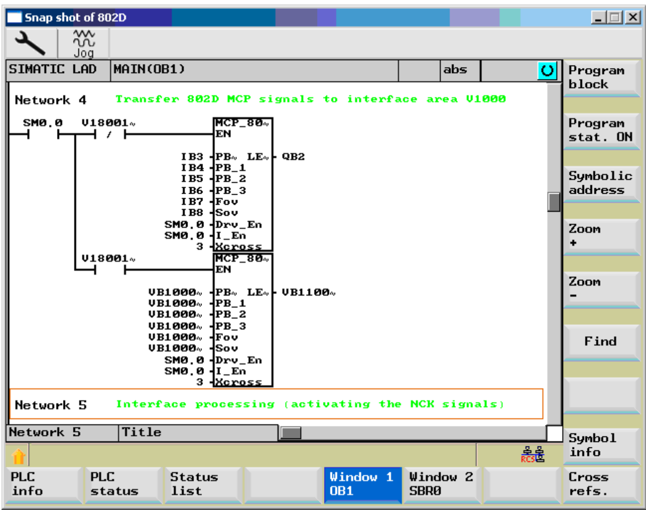


Figure 13-5 Window 1, OB1

Program block

This softkey can be used to select the list of the PLC program blocks.

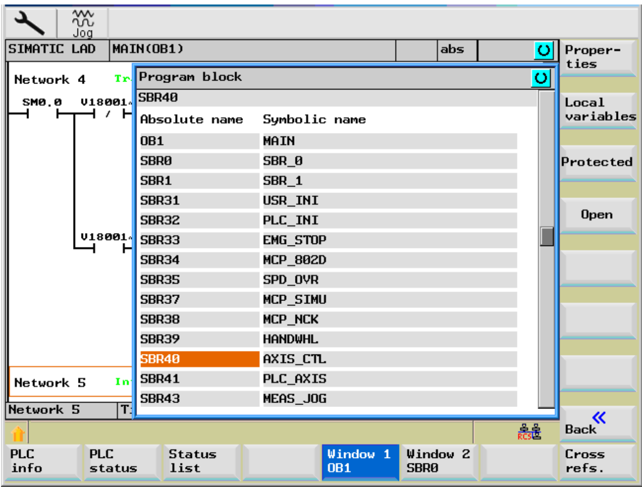


Figure 13-6 Select the PLC program block

Proper-ties

Using this softkey will display the following properties of the selected program block:

- Symbolic name
- Author
- Comments

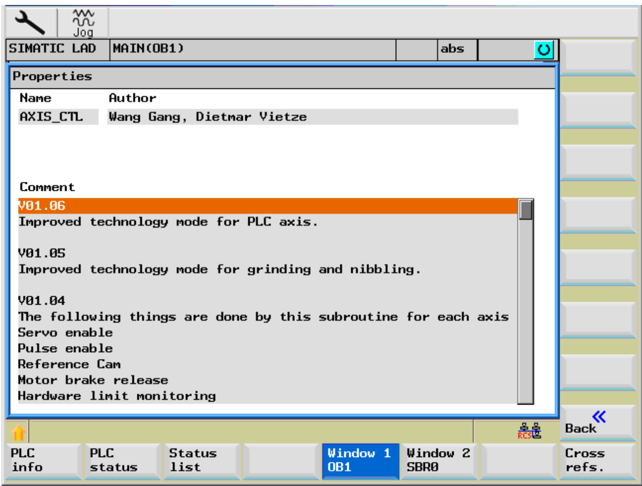


Figure 13-7 Properties of the selected PLC program block

Local variables

Selecting this softkey displays the table of local variables of the selected program block.

There are two types of program blocks.

- OB1 only temporary local variable
- SBRxx temporary local variable

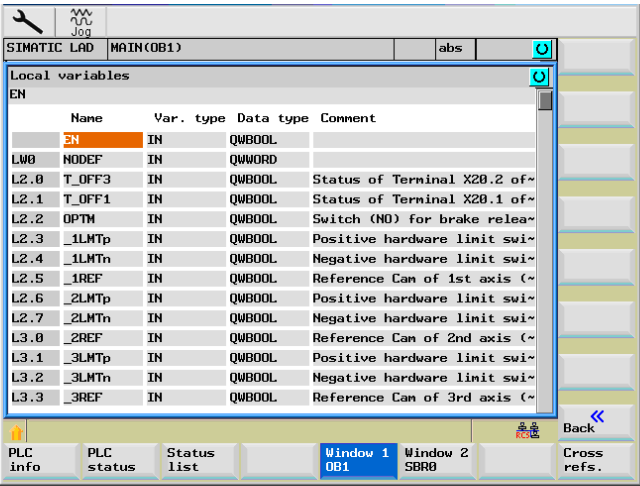


Figure 13-8 Table of local variables for the selected PLC program block

The text of the current cursor position is additionally displayed in a text field above the table.

With longer texts, it is possible to display the whole text by pressing the SELECT key.

Cover

When a program block is protected by a password, this softkey can be used to enable the display of the ladder diagram.

A password is required for this. The password can be allocated during creation of a program block in Programming Tool PLC802.

Open

The selected program block is opened.

The name (absolute) of the program block will then be displayed on "Window 1..." softkey (for "..." e.g. "Window 1 OB1").

Program stat. OFF

Selecting this softkey activates or deactivates the program status display.

You can monitor the current status of the networks from the PLC cycle end.

The states of all operands are displayed in the "Program status" ladder diagram (top right in the window). This LAD acquires the values for the status display in several PLC cycles and then refreshes the status display.

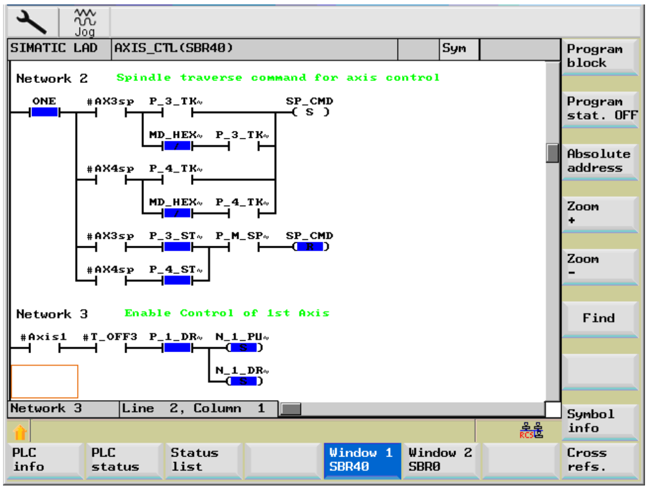


Figure 13-9 "Program status" ON – symbolic representation

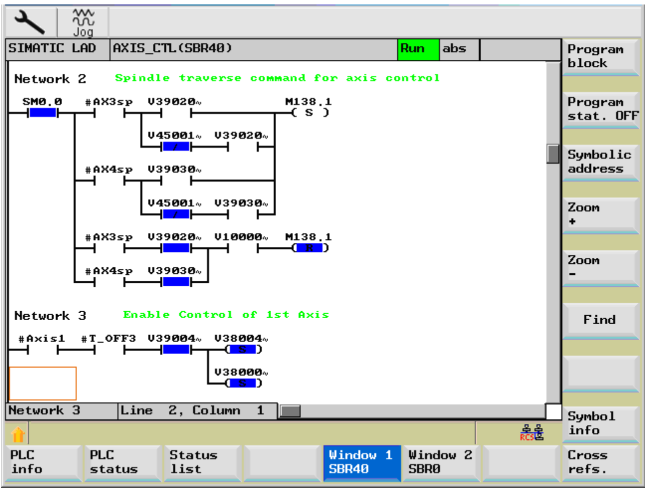


Figure 13-10 "Program status" ON – absolute representation

Symbolic address

Use this softkey to switch between the absolute and symbolic representation of the operands. The softkey labelling changes accordingly.

Depending on the selected type of representation, the operands are displayed either with absolute or symbolic identifiers.

If no symbol exists for a variable, this is automatically displayed absolutely.

Zoom +

Zoom -

The representation in the application area can be zoomed in or zoomed out step by step. The following zoom stages are provided:

20% (default), 60%, 100% and 300%

Find

Can be used to search for operands in the symbolic or absolute representation (see following screen).

A dialog box is displayed from which various search criteria can be selected. Use the "Absolute/symbol. address" softkey to search for a certain operand matching this criterion in both PLC windows (see the following screen). When searching, uppercase and lowercase letters are ignored.

Selection in the upper toggle field:

- Search for absolute and symbolic operands
- Go to network number
- Find SBR command

Further search criteria:

- Search direction down (from the current cursor position)
- Whole program block (from the beginning)
- In one program block
- Over all program blocks

You can search for the operands and constants as whole words (identifiers).

Depending on the display settings, you can search for symbolic or absolute operands.

"OK" starts the search. The found search element is highlighted by the focus. If nothing is found, an appropriate error message will appear in the notes line.

Use the "Abort" softkey to exit the dialog box. no search is carried out.

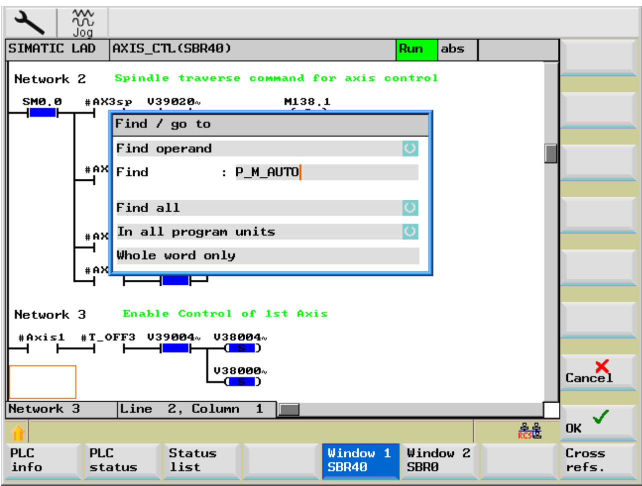


Figure 13-11 Search for symbolic operands

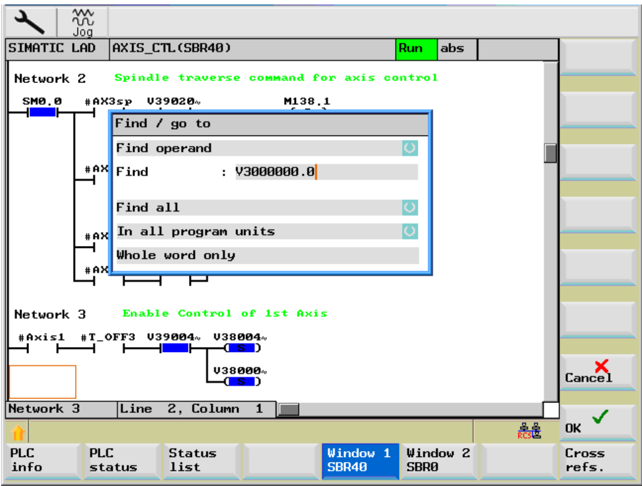


Figure 13-12 Search for absolute operands

If the search object is found, use the "Continue search" softkey to continue the search.

Symbol info

Selecting this softkey displays all symbolic identifiers used in the highlighted network.

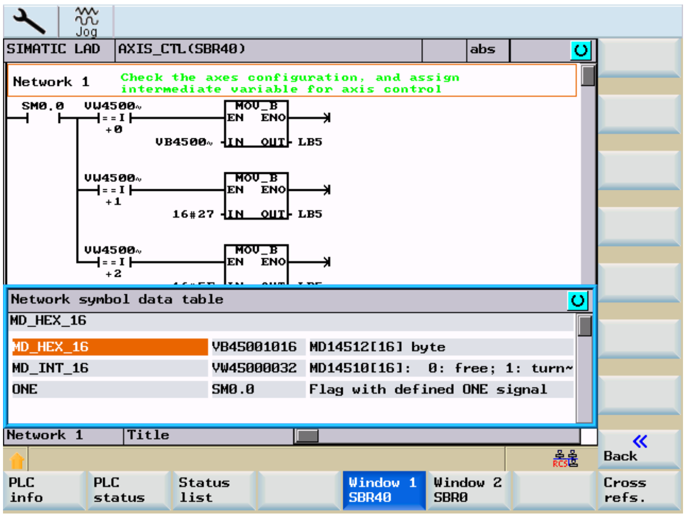


Figure 13-13 Network symbol information table

Cross refs.

Use this softkey to display the list of cross references. All operands used in the PLC project are displayed.

This list indicates in which networks an input, output, flag etc. is used.

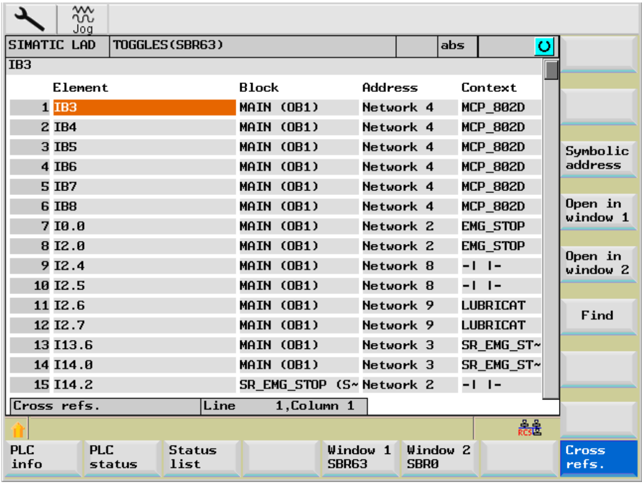


Figure 13-14 Cross references main menu (absolute)

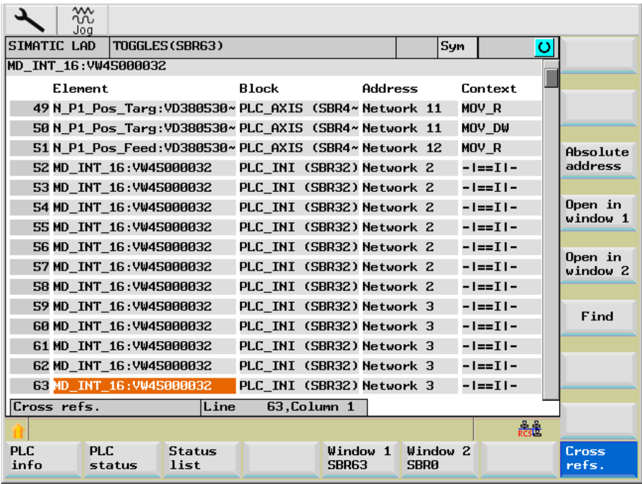


Figure 13-15 Cross references main menu (symbolic))

Open in window 1

You can open the appropriate program segment directly in the 1/2 window using the "Open in Window 1" or "Open in Window 2" function.

Symbolic address

Use this softkey to switch between the absolute and symbolic representation of the components. The softkey labelling changes accordingly.

Depending on the selected type of representation, the components are displayed either with absolute or symbolic identifiers.

If no symbol exists for an identifier, the description is automatically absolute.

The type of representation is displayed in the status line at the top right of the window (e.g. "Abs"). The absolute representation is set by default.

Example:

You want to view the logic interrelation of the absolute operand M251.0 in network 2 in program block OB1.

After the operand has been selected from the cross-reference list and the "Open in Window 1" softkey has been pressed, the corresponding program section is displayed in window 1.

Cross refs.

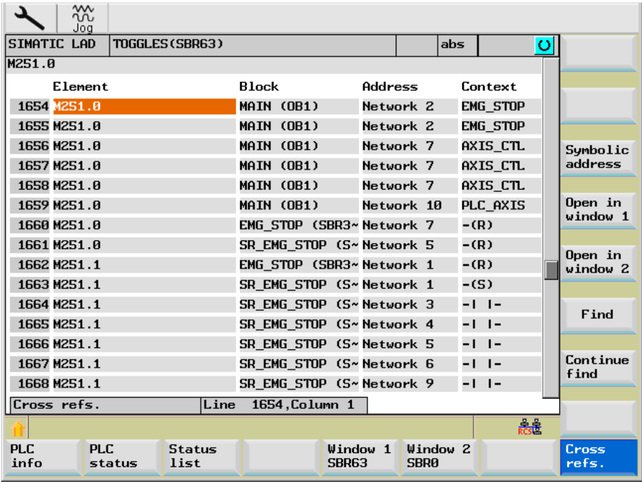


Figure 13-16 Cursor M251.0 in OB1 network 2

Open in window 1

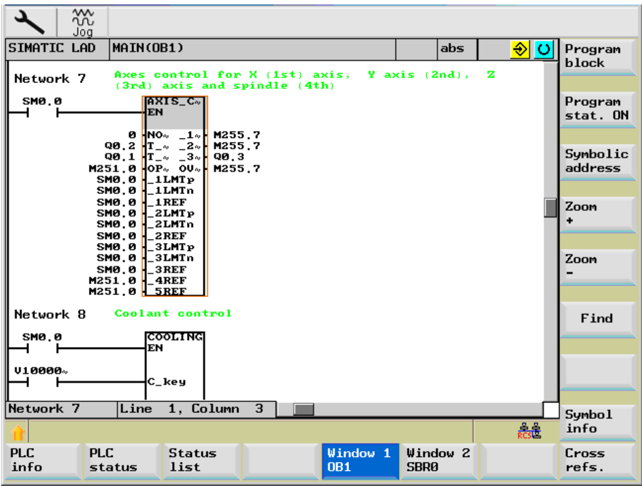


Figure 13-17 M251.0 in OB1 network 2 in window 1

Find

Searching operands in the cross-reference list (see following screen).

You can search for the operands as whole words (identifiers). When searching, uppercase and lowercase letters are ignored.

Search options:

- Search for absolute and symbolic operands
- Go to line

Search criteria:

- Down (from the current cursor position)
- Whole program block (from the beginning)

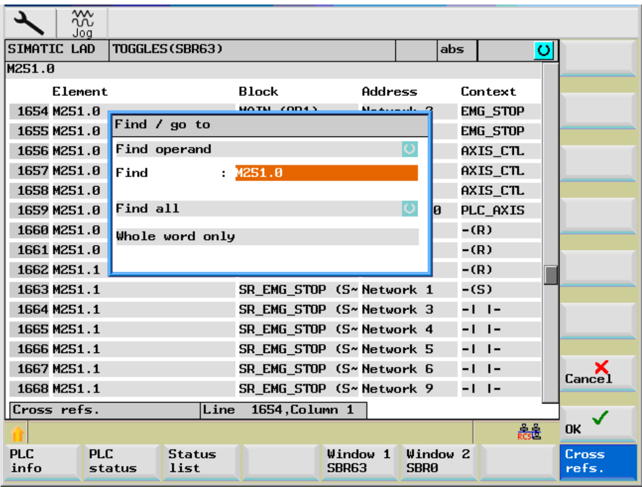


Figure 13-18 Searching for operands in cross references

The text you are looking for is displayed in the notes line. If the text is not found, a corresponding error message is displayed which must be confirmed with "OK".

Appendix A

A.1 User data

The user data is internally processed in the grinding cycles. They are stored in the program manager of the control system (in the directory \DEF) as a definition file and remain stored even when the control is switched off and on.

Description of the user data

The parameters included in the definition files are described as follows:

Name	Type	Default Value	Description
_GC_LERF	REAL		Detected longitudinal position when setting up
_GC_LVER	REAL		Offset during longitudinal position sensing
_GC_LNPVZ	REAL		Initial Z zero shift during calibration
_GC_LXPOS	REAL		X position while longiitudinal position is sensed
_GC_PARR[20]	REAL		REAL type parameters for inter cycle as well as cycle HMI communication
_GC_PAR[0]	INT	0/1	Selection of the type of plunging feedrate in mm/min / specific cutting volumes
_GC_PAR[1]	INT	0/1	Selection of the longitudinal grinding feedrate in mm/min or mm/rev
_GC_PARI[20]	INT		INTEGER type parameters for inter cycle as well as cycle HMI communication
_GC_SYNC	INT	0	HMI synchronisation parameters
_GC_SYNC INIRE	INT	0	Delete synchronisation parameters on reset
_GC_WPC	INT	0	Workpiece counter for dressing interval
_GC_BAXIS	STRING[10]		Name of the swivel axis
_GC_DNUM	INT	7	D number for the 1st data block of dressing data in the tool compensation
_GC_KNVX	INT	0	There it is defined how the detected offset will be taken into account in X: 0 ... Through work offset (NV) 1 ... as wheel diameter offset
_GC_KORR	INT	0	Selection of measurement control compensation computation 0 ... Compensation of the setpoint-actual value difference in the wear of the wheel / dresser 1 ... Compensation of the setpoint-actual value difference in WO in X 2 ... No compensation of the setpoint-actual value difference
_GC_MF[20]	INT		M command number

Name	Type	Default Value	Description
_GC_MF[0] _GC_MF[1] _GC_MF[2] _GC_MF[3] _GC_MF[4] _GC_MF[5] _GC_MF[6] _GC_MF[7] _GC_MF[8] _GC_MF[9] _GC_MF[10] _GC_MF[11] _GC_MF[12] . _GC_MF[13] _GC_MF[14] _GC_MF[15] _GC_MF[16] _GC_MF[17]	INT	3 21 22 33 34 41 42 65 66 80 81 4 7 9	Grinding spindle direction of rotation (M3) Swing in measurement control (M21) Swing out measurement control (M22) Structure-borne noise ON (M33) Structure-borne noise OFF (M34) Advance dresser (M41) Retract dresser (M42) Swing out caliper (M65) Swing in caliper (M66) Enable handwheel (M80) Disable handwheel (M81) Workpiece spindle direction of rotation (M4) Coolant ON (M7) Coolant OFF (M9) Swing in measurement control, program control (M23) Swing out measurement control, program control (M24) Disable stroke reversal if no longitudinal stroke (M27) Enable stroke reversal if longitudinal stroke (M28)
			Number of inputs IN:
_GC_IN_KS	INT	16	Acoustic emission sensor
_GC_IN_MZ0	INT	9	Retract measurement control
_GC_IN_MZ1	INT	10	Time measurement control
_GC_IN_MZ2	INT	11	Switch-over fine finishing measurement control
_GC_IN_MZ3	INT	12	Switch-over finishing measurement control
_GC_IN_MZ4	INT	13	Reserved for inputs/outputs
_GC_IN_ABR	INT	14	Intermediate dressing upon key
_GC_IN_HAND	INT	15	Handwheel key
_GC_IN_BREAK	INT	13	Program interrupt key
_GC_IN_HUB	INT	12	Stroke reversal key
_GC_IN_FEEDSTOP	INT	11	Infeed stop key
_GC_WEARTYP	INT	0	Selection of wear compensation, comparison or nominal dimensions
_GC_SSTAT	INT		Selection ... with/without grinding spindle monitoring
_GC_FEIN[2]	REAL		Global fine compensation
_GC_FEIN[0] _GC_FEIN[1]	REAL		Incremental X fine compensation Incremental Z fine compensation
_GC_SFEIN[10,2]	REAL		Fine compensation seat-specific 1st index … seat number 2nd index ... axis
_GC_RLZTYP	INT	0	Do not approach the return position of the Z-axis in -1-, MCS=0 WCS=1
_GC_RLXTYP	INT	0	Type of return position in
_GC_RLX	REAL		X return position; dresser or workpiece can be collision-free approached using a machine specific return position
_GC_RLZ	REAL		Z return position; dresser or workpiece can be approached without collision using a machine-specific return position.
_GC_BT	REAL		Measurement control tolerance in which a measurement control signal is expected

Name	Type	Default Value	Description
_GC_FWEG	REAL		Free wheel travel path (measurement control)
_GC_SEARCHS			Tag for seat regrinding is evaluated by the cycles so that the individual seat can be identified via a block search.
_GC_SEARCH			Tag for seat regrinding is evaluated by the cycles so that the individual seat can be identified via a block search.
_GC_SEARCHSET			Tag for seat regrinding is evaluated by the cycles so that the axes can be recalibrated.
_GC_SEACRHVALUE[0..2]			Regrinding calibration values
_GC_SUGFEED			Independent of basic system 0 = GWPS in m/s 1 = GWPS in feed/min
_GC_MF[18]			Enable program level abort of CYCLE448
_GC_MF[19]			Blocking and resetting of last program level abort

NOTICE
The values stored as the default must be checked by the machine manufacturer and adapted to the realities of the machine.

A.2 Parameter tables of the tool data

The following parameters, operated from the HMI, are available for the tool offsets.

Table A- 1 Grinding wheel data, x=[1...n] y=[1...6]

Tx	TPG1	INT	Spindle number
Tx	TPG2	INT	Concatenation rule = 0
Tx	TPG3	REAL	Min. wheel diameter
Tx	TPG4	REAL	Min. wheel width
Tx	TPG5	REAL	Current grinding wheel width
Tx	TPG6	REAL	Maximum speed
Tx	TPG7	REAL	Maximum GWPS
Tx	TPG8	REAL	Angle of inclined wheel
Tx	TPG9	INT	Parameter no. for radius calculation
Tx	TPC1	REAL	Wheel type (vertical, inclined, free)
Tx	TPC2	REAL	Amount of crown
Tx	TPC3	REAL	Dressing amount
Tx	TPC4	REAL	Cylindric compensation
Tx	TPC5	REAL	GWPS
Tx	TPC6	REAL	GWPS ratio
Tx	TPC7	REAL	Bypassing strategy (obstacle diameter)
Tx	TPC8	REAL	Basic cutting edge for dressing contour
Tx	TPC9	REAL	X shift
Tx	TPC10	REAL	Z shift

Table A- 2 1. Cutting edge 2. Cutting edge for left/right wheel edge for grinding wheel

Tx Dy	DP1	INT	Tool type=403
Tx Dy	DP2	INT	Cutting edge position (1...9)
Tx Dy	DP3	REAL	D - Diameter of the new wheel
Tx Dy	DP4	REAL	L - Distance of the wheel reference point
Tx Dy	DP5	REAL	(reserved, length 3)
Tx Dy	DP6	REAL	R - Tool nose radius
Tx Dy	DP7	REAL	Dressing amount (μm) left/right
Tx Dy	DP8	REAL	Dresser wear X (μm) left/right
Tx Dy	DP9	REAL	Dresser wear Z (μm) left/right
Tx Dy	DP10	REAL	Path feedrate (mm/rev), left/right
Tx Dy	DP11	REAL	Path feedrate X (mm/rev), left/right
Tx Dy	DP12	REAL	dD - Change in diameter (dressing amount X)
Tx Dy	DP13	REAL	dL - Change in distance (dressing amount Z)
Tx Dy	DP14	REAL	(Length 3)
Tx Dy	DP15	REAL	dR - Change in tool nose radius (radius wear)

Tx Dy	DP16	REAL	Diameter dressing amount (µm)
Tx Dy	DP17	REAL	Dresser wear X (µm) diameter
Tx Dy	DP18	REAL	Dresser wear Z (µm) diameter
Tx Dy	DP19	REAL	Dressing direction (drawing/plunging) diameter
Tx Dy	DP20	REAL	Path feedrate (mm/rev) diameter
Tx Dy	DP21	REAL	Additional compens. in X, diameter, basic dimension
Tx Dy	DP22	REAL	Additional compens. in Z, length in Z, basic dimension
Tx Dy	DP23	REAL	(reserved, length 3)
Tx Dy	DP24	REAL	Diameter compensation of measurement control or cutting edge 1-6 initial dimension
Tx Dy	DP25	REAL	Z compensation of measurement control or initial dimension of each cutting edge
Tx Dy	DPC1	REAL	Left/right overrun
Tx Dy	DPC2	REAL	Left/right radius
Tx Dy	DPC3	REAL	Left/right X chamfer
Tx Dy	DPC4	REAL	Left/right Z chamfer
Tx Dy	DPC5	REAL	Left/right shoulder height
Tx Dy	DPC6	REAL	Left/right back-slope angle
Tx Dy	DPC7	REAL	Left/right back-slope height
Tx Dy	DPC8	REAL	X overrun
Tx Dy	DPC9	REAL	Usable wheel width
Tx Dy	DPC10	REAL	No. of contour program

Table A- 3 3. Cutting edge for grinding wheel

Tx Dy	DP1	INT	Tool type=403
Tx Dy	DP2	INT	Cutting edge position (1...9)
Tx Dy	DP3	REAL	D - Diameter of the new wheel
Tx Dy	DP4	REAL	L - Distance of the wheel reference point
Tx Dy	DP5	REAL	(reserved, length 3)
Tx Dy	DP6	REAL	R - Tool nose radius
Tx Dy	DP7	REAL	Coasting revolutions
Tx Dy	DP8	REAL	Profile roller plunge feed (wheel types 5 and 6)
Tx Dy	DP9	REAL	Profile roller dressing feed (wheel types 5 and 6)
Tx Dy	DP10	REAL	GWPS profile roller (wheel types 5 and 6)
Tx Dy	DP11	REAL	Profile roller GWPS ratio (wheel types 5 and 6)
Tx Dy	DP12	REAL	dD - Change in diameter (dressing amount X)
Tx Dy	DP13	REAL	dL - Change in distance (dressing amount Z)
Tx Dy	DP14	REAL	(Length 3)
Tx Dy	DP15	REAL	dR - Change in tool nose radius (radius wear)
Tx Dy	DP16	REAL	Profile roller dressing number (wheel types 5 and 6)
Tx Dy	DP17	REAL	Reserved

Tx Dy	DP18	REAL	Reserved
Tx Dy	DP19	REAL	Reserved
Tx Dy	DP20	REAL	Reserved
Tx Dy	DP21	REAL	Additional compens. in X, diameter, basic dimension
Tx Dy	DP22	REAL	Additional compens. in Z, length in Z, basic dimension
Tx Dy	DP23	REAL	(reserved, length 3)
Tx Dy	DP24	REAL	Reserved
Tx Dy	DP25	REAL	Reserved
Tx Dy	DPC1	REAL	Idle strokes when dressing a path
Tx Dy	DPC2	REAL	Reserved
Tx Dy	DPC3	REAL	Reserved
Tx Dy	DPC4	REAL	Reserved
Tx Dy	DPC5	REAL	Reserved
Tx Dy	DPC6	REAL	Reserved
Tx Dy	DPC7	REAL	Reserved
Tx Dy	DPC8	REAL	Reserved
Tx Dy	DPC9	REAL	Reserved
Tx Dy	DPC10	REAL	Reserved

Table A- 4 4. to 6th cutting edge for grinding wheels

Tx Dy	DP1	INT	Tool type=403
Tx Dy	DP2	INT	Cutting edge position (1...9)
Tx Dy	DP3	REAL	D - Diameter of the new wheel
Tx Dy	DP4	REAL	L - Distance to the wheel reference point
Tx Dy	DP5	REAL	(reserved, length 3)
Tx Dy	DP6	REAL	R - Tool nose radius
Tx Dy	DP7	REAL	Reserved
Tx Dy	DP8	REAL	Reserved
Tx Dy	DP9	REAL	Reserved
Tx Dy	DP10	REAL	Reserved
Tx Dy	DP11	REAL	Reserved
Tx Dy	DP12	REAL	dD - Change in diameter (dressing amount X)
Tx Dy	DP13	REAL	dL - Change in distance (dressing amount Z)
Tx Dy	DP14	REAL	(Length 3)
Tx Dy	DP15	REAL	dR - Change in tool nose radius (radius wear)
Tx Dy	DP16	REAL	Reserved
Tx Dy	DP17	REAL	Reserved
Tx Dy	DP18	REAL	Reserved
Tx Dy	DP19	REAL	Reserved
Tx Dy	DP20	REAL	Reserved
Tx Dy	DP21	REAL	Additional compens. in X, diameter, basic dimension

Tx Dy	DP22	REAL	Additional compens. in Z, length in Z, basic dimension
Tx Dy	DP23	REAL	(reserved, length 3)
Tx Dy	DP24	REAL	Reserved
Tx Dy	DP25	REAL	Reserved
Tx Dy	DPC1	REAL	Reserved
Tx Dy	DPC2	REAL	Reserved
Tx Dy	DPC3	REAL	Reserved
Tx Dy	DPC4	REAL	Reserved
Tx Dy	DPC5	REAL	Reserved
Tx Dy	DPC6	REAL	Reserved
Tx Dy	DPC7	REAL	Reserved
Tx Dy	DPC8	REAL	Reserved
Tx Dy	DPC9	REAL	Reserved
Tx Dy	DPC10	REAL	Reserved

Table A- 5 7. to 9th cutting edge for dressers

Tx Dy	DP1	INT	Tool type=403
Tx Dy	DP2	INT	Cutting edge position (1...9)
Tx Dy	DP3	REAL	Position
Tx Dy	DP4	REAL	Position
Tx Dy	DP5	REAL	Position
Tx Dy	DP6	REAL	R - Tool nose radius
Tx Dy	DP7	REAL	Diameter
Tx Dy	DP8	REAL	Width
Tx Dy	DP9	REAL	Maximum peripheral speed
Tx Dy	DP10	REAL	Maximum speed
Tx Dy	DP11	REAL	Probing data block
Tx Dy	DP12	REAL	dD - Change in diameter (dressing amount X)
Tx Dy	DP13	REAL	dL - Change in distance (dressing amount Z)
Tx Dy	DP14	REAL	(Length 3)
Tx Dy	DP15	REAL	dR - Change in tool nose radius (radius wear)
Tx Dy	DP16	REAL	Roller circumference speed
Tx Dy	DP17	REAL	Maximum length 1 wear
Tx Dy	DP18	REAL	Maximum length 2 wear
Tx Dy	DP19	REAL	Maximum length 3 wear
Tx Dy	DP20	REAL	Roller direction of rotation optional
Tx Dy	DP21	REAL	Additional compens. in X, diameter, basic dimension
Tx Dy	DP22	REAL	Additional compens. in Z, length in Z, basic dimension
Tx Dy	DP23	REAL	(reserved, length 3)
Tx Dy	DP24	REAL	Z oscillating path
Tx Dy	DP25	REAL	Infeed amount per stroke

Tx Dy	DPC1	REAL	Reciprocation speed
Tx Dy	DPC2	REAL	Dressing amount
Tx Dy	DPC3	REAL	Approaching distance
Tx Dy	DPC4	REAL	X start
Tx Dy	DPC5	REAL	Z start
Tx Dy	DPC6	REAL	Dresser type (0 – X/Z, >0 rear, rotating,...)
Tx Dy	DPC7	REAL	Profile depth
Tx Dy	DPC8	REAL	Safety speed
Tx Dy	DPC9	REAL	X oscillating path
Tx Dy	DPC10	REAL	Reserved

In addition to the default coding of the tool data (tool type, cutting edge position, etc.), the following coded parameters are used.

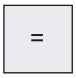
Encoding	Wheel type $TC_TPC1[T]
0	free contour
1	Standard contour straight without rear definitions
2	Standard contour straight with rear definitions
3	Standard contour inclined left
4	Standard contour inclined right
5	Standard contour straight profile roller with geo axes

Encoding	Dressing mode at the diameter $TC_DP19[T,1]
0	neither drawing nor plunging (3rd dresser)
1	drawing (last active dresser)
2	plunging (last active dresser)
11	drawing (1st dresser)
12	plunging (1st dresser)
21	drawing (2nd dresser)
22	plunging (2nd dresser)

Encoding	Dresser type $TC_DPC6[T,_GC_DNUM+dresser-1]
0	Dresser geometry axes (diamond) non-rotating
1	Dresser geometry axes (diamond) non-rotating
...	
11	Dresser, geometry axes (form roll) rotating
12	Dresser, geometry axes (form roll) rotating
21	Dresser, geometry axes (diamond) rotating

A.3 Miscellaneous

A.3.1 Pocket calculator



The calculator function can be activated from any operating area using <SHIFT> and <=> or <CTRL> and <A>.

For calculating, the four basic arithmetic operations are available, as well as the functions "sine", "cosine", "squaring" and "square root". A bracket function is provided to calculate nested terms. The bracket depth is unlimited.

If the input field is already occupied by a value, the function will accept this value into the input line of the pocket calculator.

<Input> starts the calculation. The result is displayed in the pocket calculator.

Selecting the "Accept" softkey enters the result in the input field at the current cursor position of the part program editor and closes the pocket calculator automatically.

Note

If an input field is in the editing mode, it is possible to restore the original status using the "Toggle" key.

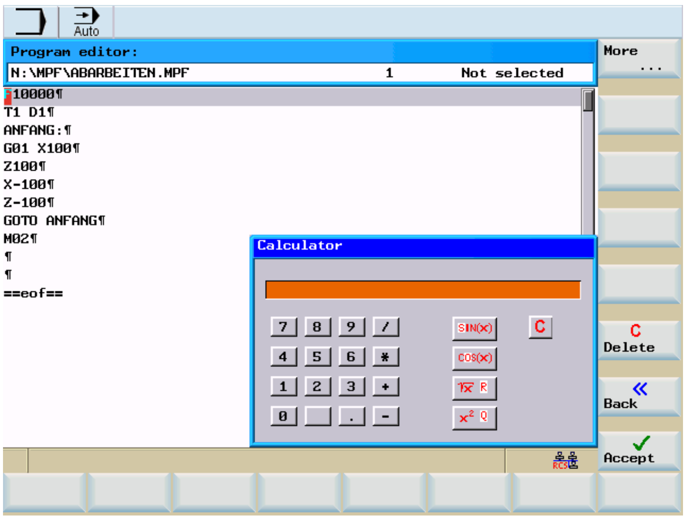


Figure A-1 Pocket calculator

Characters that may be entered

+, -, *, /	Basic arithmetic operations
S	Sine function The X value (in degrees) in front of the input cursor is replaced by the sin(X) value.
O	Cosine function The X value (in degrees) in front of the input cursor is replaced by the cos(X) value.
Q	Square root function The X value in front of the input cursor is replaced by the X^2 value.
R	Square root function The X value in front of the input cursor is replaced by the $\sqrt{X}$ value.
( )	Bracket function (X+Y)*Z

Calculation examples

Task	Input -> Result
100 + (67*3)	100+67*3 -> 301
sin(45_)	45 S -> 0.707107
cos(45_)	45 O -> 0.707107
4^2	4 Q -> 16
$\sqrt{4}$	4 R -> 2
(34+3*2)*10	(34+3*2)*10 -> 400

To calculate auxiliary points on a contour, the pocket calculator offers the following functions:

- Calculating the tangential transition between a circle sector and a straight line
- Moving a point in the plane
- Converting polar coordinates to Cartesian coordinates
- Adding the second end point of a straight line/straight line contour section given from an angular relation

A.3.2 Editing Asian characters

The program editor and PLC alarm text editor both allow you to edit Asian characters.

This function is available in the following Asian language versions:

- Simplified Chinese
- Traditional Chinese (as used in Taiwan)
- Korean

Press <Alt+S> to switch the editor on or off.

Simplified/Traditional Chinese

Characters can be selected according to the pinyin input method, which involves combining letters of the Roman alphabet in order to reproduce the sound of the character.

The editor will then show a list of characters that correspond to that particular sound.

You can then select the character you need.

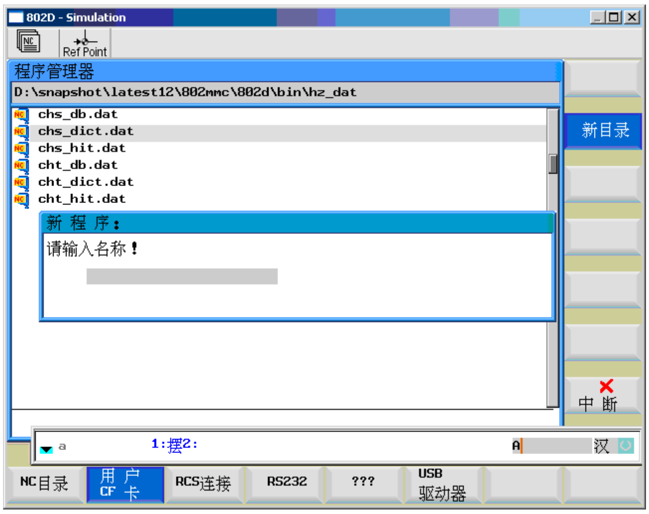


Figure A-2 Example of editing Simplified Chinese

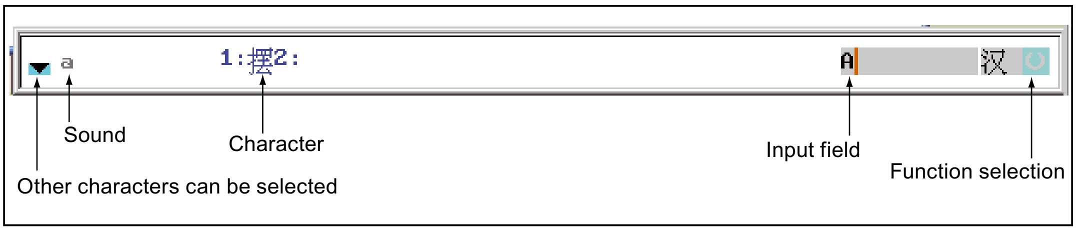


Figure A-3 Structure of editor

The "Function selection" toggle field enables switching between the PinYin-entry method and entering Latin graphic characters as well as activating the function for editing the dictionary.

When a character is selected, the editor records the frequency with which it is selected for a specific phonetic notation and when the editor is again opened, it offers the most frequently used characters.

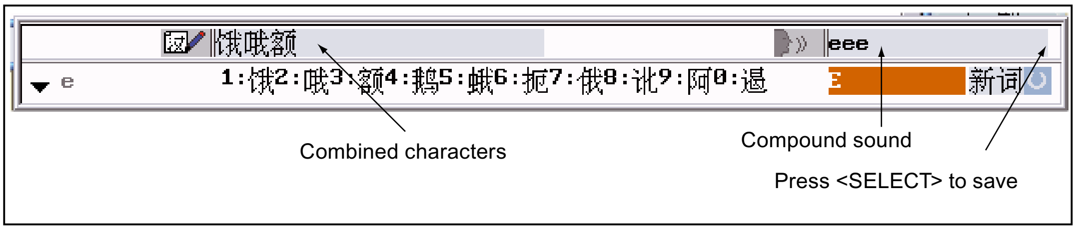


Figure A-4 Structure of editor when learning function is active

- Editing the dictionary

  When this function is activated, another line showing the combined characters and sounds will appear.

  The editor will then offer various characters for this sound, from which you can choose the desired one by entering either of the digits (1 to 9).

  You can toggle the input cursor between the compound phonetic notations field and the phonetic input field by pressing the <TAB> key.

  When the cursor is positioned in the upper field, you can undo the combination by pressing the <backspace> key.

  Press <select> to save the characters currently being displayed.

  Press the <delete> key if you want to delete the currently displayed group of characters from the dictionary.

Korean

To enter Korean characters, you will need a keyboard with the keyboard assignment shown below.

In terms of key layout, this keyboard is the equivalent of an English QWERTY keyboard and individual characters must be grouped together to form syllabic blocks.

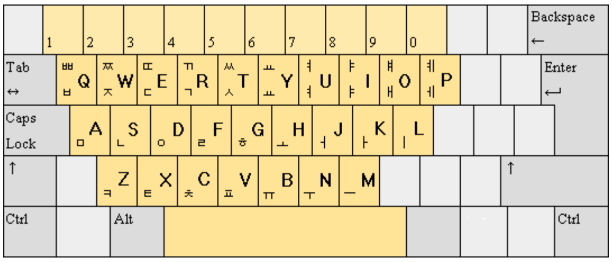


Figure A-5 Korean keyboard assignment

The Korean alphabet (Hangeul) consists of 24 letters: 14 consonants and 10 vowels. The syllable blocks are created by combining consonants and vowels.

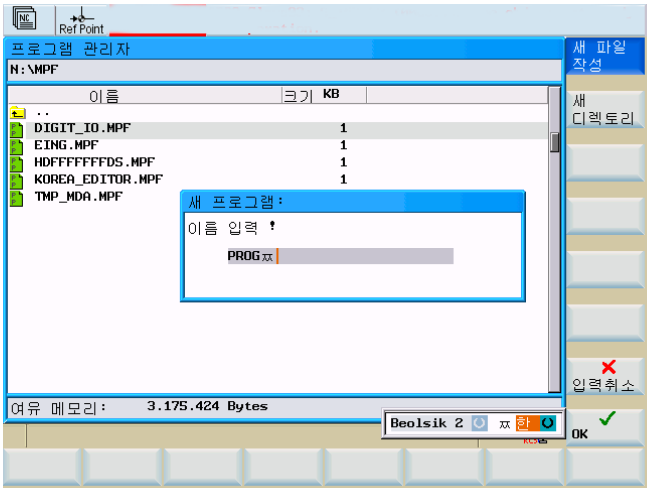


Figure A-6 Korean editor with standard keyboard assignment

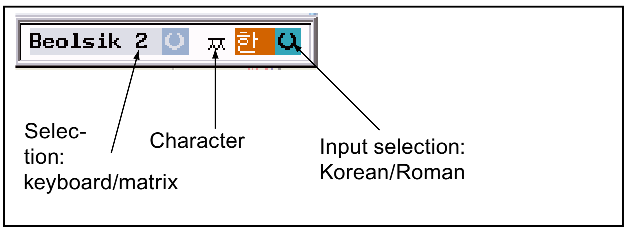


Figure A-7 Structure of Korean editor

- Input via matrix

  If you only have access to a control keyboard, then you can use a matrix input method as an alternative to the keyboard assignment shown above. All you will need for this is the numeric keypad.

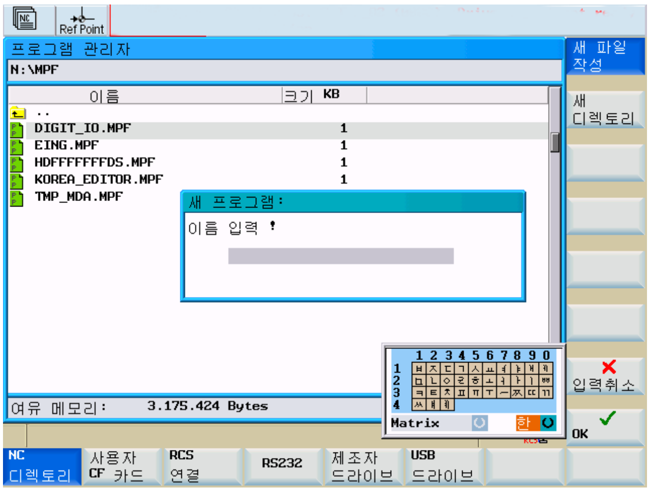


Figure A-8 Korean editor with selection matrix

To select characters, proceed as follows:

- Select a row (the row will be color-highlighted)
- Select a column (the character will briefly be color-highlighted and then transferred to the "Character" field).
- Press the <input> key to transfer the character into the edit field.

A.4 Feedback on the documentation

This document will be continuously improved with regard to its quality and ease of use. Please help us with this task by sending your comments and suggestions for improvement via e-mail or fax to:

E-mail: mailto:docu.motioncontrol@siemens.com

Fax: +49 9131 - 98 2176

Please use the fax form on the back of this page.

To	From
SIEMENS AG I DT MC MS1 P.O. Box 3180 D-91050 Erlangen / Germany	Name:
	Address of your company/department
	Street:
	Zip code: City:
	Phone: /
Fax: +49 9131 - 98 2176 (Documentation)	Fax: /

Suggestions and/or corrections

A.5 Overview

SINUMERIK 802D sl documentation overview

General documentation / catalogs

SINUMERIK	SINUMERIK 802D sl	SINAMICS S120
Advertising brochure	Catalog NC 61	Catalog D21.1 Drive converter chassis units

User documentation

SINUMERIK 802D sl	SINUMERIK 802D sl	SINUMERIK 802D sl 840D sl
Programming and Operating Manual – Turning – Milling – Cylindrical grinding – Surface grinding – Nibbling – Manual machine plus	Diagnostics Manual	Programming Manual ISO turning / milling

Manufacturer / service documentation

SINUMERIK 802D sl	SINUMERIK 802D sl	SINUMERIK 802D sl 840D sl	SINUMERIK 802D sl	SINUMERIK
Operating instructions	Function Manual	Function Manual ISO dialects	Parameter Manual	EMV directives

Electronic documentation

SINUMERIK SINAMICS Motors
DOCONCD DOCONWEB

Glossary

Effective wheel width

Wheel width of the inclined grinding wheel which is used to machine the diameter. It is dependent upon:

- the physical width
- the evading height
- the angle of the wheel

Evasion/evasion angle

Tapering of the left or right side of the grinding wheel for face-grinding operations in which a so-called cross-grinding is produced.

GAP/structure-borne noise/air grinding

Bridging the air gap between the workpiece and grinding wheel with a structure-borne noise microphone, which is built into the machine.

GWPS

Grinding wheel peripheral speed in m/s

MCPA

Input card for rapid I/O to the control system

MD

Machine data; machine data are predefined variables (system variables), with which the NCK, as per the requirements of the machine manufacturer, is adapted to the machine-tool.

SD

Setting data are system variables that indicate the current machine properties to the NCK. Unlike machine data, changes to setting data always become effective immediately.

Seat

Machining operation

Shoulder

Left or right side of the grinding wheel or of the tool

TPS

Workpiece peripheral speed in m/min

XWP/ZWP when dressing a free contour

Workpiece offset for offsetting the programmed contour to the current cutting edge of the grinding wheel; which is necessary so that workpiece coordinates can be programmed in the free contour.

Index