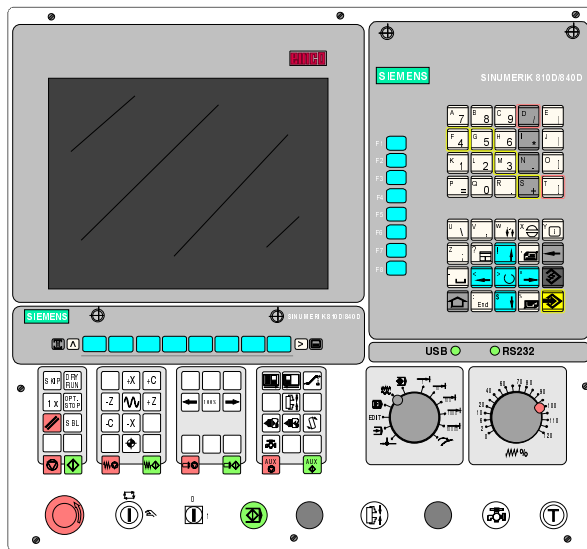


# EMCO WinNC SINUMERIK 810D/840D Milling

## Software Description/ Software version from 21.00



### Software Description

### EMCO WinNC

### SINUMERIK 810D/840D Milling

Ref.No. EN 1814 Edition G2007-06

This manual is electronically available (.pdf) upon request at any time on the EMCO homepage.

EMCO Maier Ges.m.b.H.  
P.O. Box 131  
A-5400 Hallein-Taxach/Austria  
Phone ++43-(0)62 45-891-0  
Fax ++43-(0)62 45-869 65  
Internet: [www.emco.at](http://www.emco.at)  
E-Mail: [service@emco.at](mailto:service@emco.at)

**emco** group  
Designed for your Profit

**Notice**

This software description contains all functions that may be carried out with WinNC. However, the availability of functions is dependent on the machine you operate with WinNC.



## Preface

The EMCO WinNC SINUMERIK 810D/840D Milling Software is part of the EMCO training concept on PC-basis.

This concept aims at learning the operation and programming of a certain machine control on the PC.

The milling machines of the EMCO PC MILL und CONCEPT MILL series can be directly controlled via PC by means of the EMCO WinNC for the EMCO MILL.

The operation is rendered very easy by the use of a digitizer or the control keyboard with TFT flat panel display (optional accessory), and it is didactically especially valuable since it remains very close to the original control.

Apart of this software description and the machine description a teaching software CD-ROM "WinTutorial" (CNC examples, operation, description of instructions and cycles) is in preparation.

This manual does not include the whole functionality of the control software SINUMERIK 810D/840D Milling, however emphasis was laid on the simple and clear illustration of the most important functions so as to achieve a most comprehensive learning success.

In case any questions or proposals for improving this manual should arise, please contact us directly:

EMCO MAIER Gesellschaft m. b. H.  
Department Technical Documentation  
A-5400 Hallein, Austria

# Contents

<b>A: Basics</b> .....	<b>A 1</b>	<b>D: Programming</b> .....	<b>D 1</b>
Reference Points of the EMCO Milling Machines .....	A 1	Surveys .....	D 2
Zero offset .....	A 2	G- commands .....	D 2
Coordinate System .....	A 2	M- Commands .....	D 4
Coordinate System with Absolute Programming .....	A 2	Cycles .....	D 5
Coordinate System with Incremental Programming .....	A 2	Command shortcuts .....	D 6
Tool Data .....	A 3	Arithmetic functions .....	D 8
<b>B: Key Description</b> .....	<b>B 1</b>	Calculator .....	D 9
Control Keyboard, Digitizer Overlay .....	B 1	System variables .....	D 10
Address and Numeric Keyboard .....	B 2	Working Movements .....	D 11
Double-Shift Function .....	B 2	G0, G1 Linear interpolation (cartesian) .....	D 11
Key Functions .....	B 3	G0, G1 Linear interpolation (polar) .....	D 11
Screen Division .....	B 4	G2, G3, CIP Circular interpolation .....	D 12
Machine Control Keys .....	B 5	G4 Dwell time .....	D 16
PC Keyboard .....	B 7	G9, G60, G601, G602, G603 Exact positioning .....	D 17
<b>C: Operation</b> .....	<b>C 1</b>	G64, G641 Contouring mode .....	D 18
Operation principle .....	C 1	G17, G18, G19 Working plane selection .....	D 18
Call basic menu .....	C 1	G25, G26 Programmable working area limitation .....	D 19
Navigation in the menu window .....	C 1	G25, G26 Programmable spindle speed limitation .....	D 19
Navigation in the directories .....	C 2	G33 Thread cutting .....	D 20
Edit inputs / values .....	C 2	G331/G332 Tapping without compensation chuck .....	D 20
Confirm / cancel input .....	C 3	G63 Thread tapping with compensating chuck .....	D 21
Mouse operation .....	C 3	Tool Radius Compensation G40-G42, G450 .....	D 22
Survey Operating Areas .....	C 4	Zero offsets G53-G57, G500-G599, SUPA .....	D 24
Operating Area Machine .....	C 5	Inch dimensions G70, .....	D 24
Approach reference point .....	C 6	Metric dimensions G71 .....	D 24
Traverse slides manually .....	C 6	Coordinaten, Zero Offset .....	D 25
Traverse slides in increments .....	C 7	G90 Absolute dimensions .....	D 25
MDA mode .....	C 8	G91 Incremental dimensions .....	D 25
Automatic mode .....	C 8	Working plane G17-G19 .....	D 25
Operating Area Parameter .....	C 9	Feed Programming G94, G95 .....	D 26
Tool data .....	C 9	Polar coordinates G110-G112 .....	D 27
R Parameter (arithmetic parameter) .....	C 9	Soft approach and leaving G140 - G341, DISR, DISCL, FAD .....	D 28
Workpiece counter (R90, R91) .....	C 10	Collision Detection NORM, KONT .....	D 30
Setting data .....	C 11	Cycle call .....	D 31
Zero offset .....	C 13	Drilling Cycles .....	D 32
Total effective zero offset .....	C 15	CYCLE81 Drilling, Centering .....	D 33
Operating Area Program .....	C 16	CYCLE82 Drilling, Counterboring .....	D 33
Program administration .....	C 17	CYCLE83 Deep hole drilling .....	D 35
Create workpiece directory .....	C 19	CYCLE84 Rigid tapping .....	D 38
Create / edit program .....	C 19	CYCLE840 Tapping with compensation chuck .....	D 41
Program simulation .....	C 21	CYCLE85 Boring 1, CYCLE89 Boring 5 .....	D 43
Operating Area Services .....	C 23	CYCLE86 Boring 2 .....	D 44
Interface settings .....	C 23	CYCLE87 Boring 3 .....	D 45
Drive settings .....	C 23	CYCLE88 Boring 4 .....	D 45
Read-in data .....	C 24	Row of hole HOLES1, Circle of hole HOLES2 .....	D 46
Send data .....	C 25	Linear pattern CYCLE801(*) .....	D 49
Copying and pasting data from the clipboard .....	C 26	Milling Cycles .....	D 51
Operating Area Diagnosis .....	C 27	CYCLE71 Face milling .....	D 52
Display of software versions .....	C 27	CYCLE72 Contour milling .....	D 54
Operating Area Start-up .....	C 28	CYCLE 90 Thread cutting .....	D 56
		General: Longhole, Slot milling .....	D 58
		Longholes on a circle LONGHOLE .....	D 59
		Slots on a circle SLOT, Circular slot SLOT2 .....	D 61
		Rectangular pocket POCKET1, Circular pocket POCKET2 .....	D 64
		Rectangular pocket milling POCKET3, Circular pocket milling POCKET4 .....	D 67
		Rectangular stud milling CYCLE76(*), Circular stud milling CYCLE77(*) .....	D 71

Frames .....	D 75	G: Flexible NC- Programming .....	G1
Programmable zero offset TRANS, ATRANS .....	D 76	Variable and arithmetic parameters .....	G1
Programmable rotation ROT, AROT .....	D 77	Variable types .....	G1
Programmable scale factor SCALE, ASCALE .....	D 78	System variable .....	G1
Programmable mirroring MIRROR, AMIRROR .....	D 79	Variable definition .....	G2
Subprograms .....	D 81	User defined variables .....	G2
Subprogram Call in Part Program .....	D 81	Array definition .....	G3
Subprogram End with M17 .....	D 81	Array index .....	G3
Subprogram Nesting .....	D 81	Initialization of arrays .....	G3
Subprogram with SAVE- mechanism .....	D 82	Initialization of value lists, SET .....	G4
Subprograms with passing parameters .....	D 82	Initialization with identical values, REP .....	G4
Beginning of program, PROC .....	D 82	Indirect programming .....	G6
End of program M17, RET .....	D 82	Assignments .....	G6
Subprogram with program repeating, P .....	D 82	Assignment to string variables .....	G6
Modal subprogram MCALL .....	D 83	Arithmetic operations/functions .....	G7
Program jumps .....	D 85	Comparison and logic operations .....	G8
Unconditional program jumps .....	D 85	Priority of operators .....	G9
Conditional program jumps .....	D 85	Type conversion .....	G9
Programming messages, MSG .....	D 86	Length of strings, STRLEN .....	G10
Spindle ON M3 / M4, Speed S, Spindle STOP M5, Spindle positioning SPOS .....	D 87	CASE statement .....	G11
A axis (Dividing device) .....	D 89	Check structures .....	G12
TRACYL .....	D 90	IF-ELSE-ENDIF .....	G12
Feed optimizing CFTCP, CFC, CFIN .....	D 92	Endless- Program loop, LOOP .....	G12
Command Description M Commands .....	D 93	Count loop, FOR .....	G12
Free contour programming .....	D 95	Program loop with condition at beginning of loop, WHILE G13	
		Program loop with condition at the end of loop, REPEAT G13	
		Nesting depth .....	G13
		Runtime response .....	G13
		Supplementary conditions .....	G14
		Suppress current block display, DISPLOF, DISPLON ..	G15
		Single set suppression .....	G15
		SBLOF, SBLON .....	G15
		Single set suppression program specific .....	G15
		Single set suppression at the program .....	G15
		Frames .....	G16
		Predefined frame variables .....	G17
		Frame variable/ frame relationship .....	G17
		Axis function AXNAME, ISAXIS, AX .....	G19
E: Tool Correction / Tool Measuring .....	E 1		
Tool Correction .....	E 1		
Tool programming .....	E 1		
Tool change .....	E 1		
Example .....	E 1		
Direction of length correction .....	E 4		
Required correction values for milling and drilling tools ..	E 4		
Required correction values for angular gear head tools ..	E 5		
Tool Measuring .....	E 6		
F: Program Run .....	F 1		
Preconditions .....	F 1		
Program Selection .....	F 2		
Program Start, Program Stop .....	F 3		
Messages while program run .....	F 3		
Program Control .....	F 4		
Block Search .....	F 5		

**H: Alarms and Messages ..... H1**

Starting Information  
see attachment

**I: Control Alarms ..... I1**

Control Alarms 10000 - 59999 .....	I1
Cycle Alarms 60000 - 63000 .....	I56

## A: Basics

### Reference Points of the EMCO Milling Machines

#### M = Machine zero point

An unchangeable reference point established by the machine manufacturer.

Proceeding from this point the entire machine is measured.

At the same time "M" is the origin of the coordinate system.

#### R = Reference point

A position in the machine working area which is determined exactly by limit switches. The slide positions are reported to the control by the slides approaching the „R“.

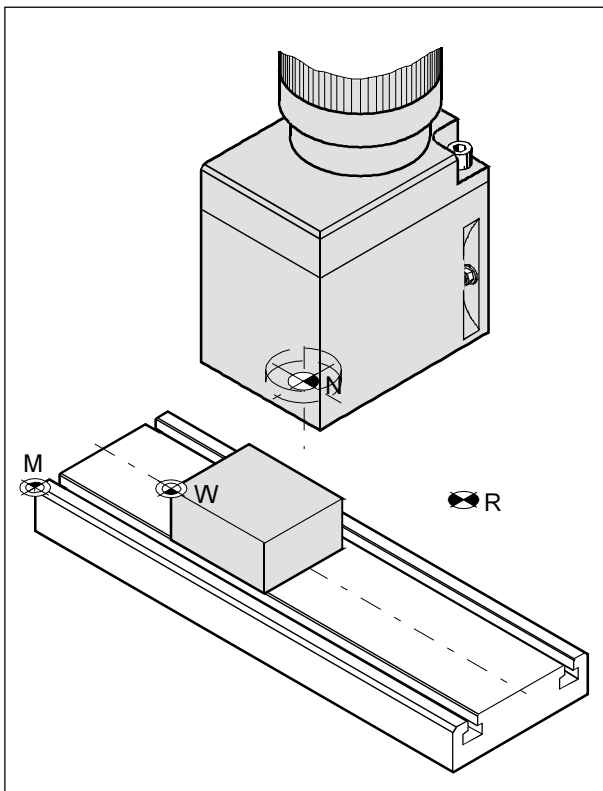
Required after every power failure.

#### N = Tool mount reference point

Starting point for the measurement of the tools. „N“ lies at a suitable point on the tool holder system and is established by the machine manufacturer.

#### W = Workpiece zero point

Starting point for the dimensions in the part program. Can be freely established by the programmer and moved as desired within the part program.



Reference points in the working area

## Zero offset

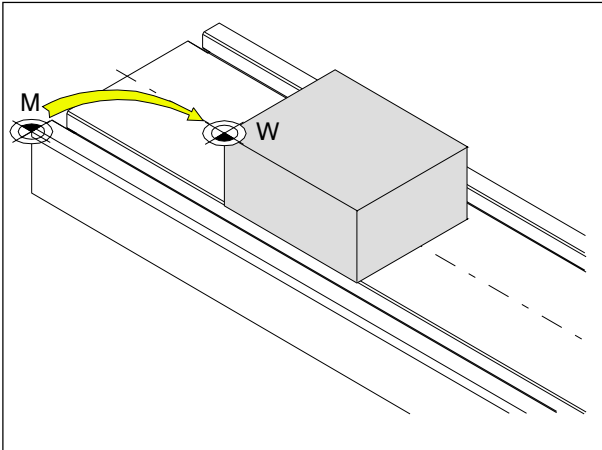
With EMCO milling machines the machine zero point "M" lies on the left front edge of the machine table. This position is unsuitable as a starting point for dimensioning. With the so-called zero offset the coordinate system can be moved to a suitable point in the working area of the machine.

In the Operating Area Parameter - Zero Offsets are four adjustable zero offsets available.

When you define a value in the offset register, this value will be considered with call up in program (G54 - G57) and the coordinate zero point will be shifted from the machine zero M to the workpiece zero W.

The workpiece zero point can be shifted within a program in any number.

More informations see in the command description.



Zero offset from machine zero point M to workpiece zero point W

## Coordinate System

The X coordinate lies parallel to the front edge of the machine table, the Y coordinate lies parallel to the side edge of the machine table, the Z coordinate is vertical to the machine table.

Z coordinate values in minus direction describe movements of the tool system towards the workpiece, values in plus direction away from the work piece.

### Coordinate System with Absolute Programming

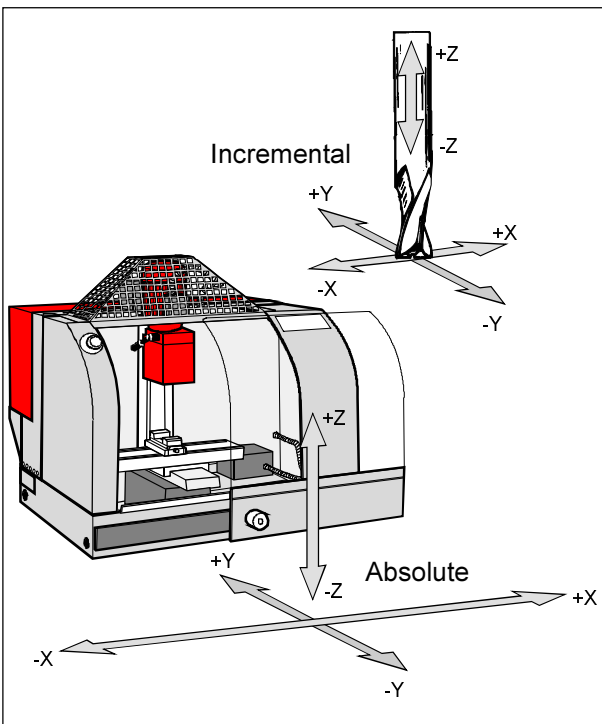
The origin of the coordinate system lies in the machine zero point "M" or after a zero offset in the work piece zero point "W".

All target points are described from the origin of the coordinate system by indication of the respective X, Y and Z distances.

### Coordinate System with Incremental Programming

The origin of the coordinate system lies at the tool mount reference point "N" or at the tool tip after a tool call-up.

With incremental programming the actual paths of the tool (from point to point) are described.



Absolute coordinates refer to a fixed point, incremental coordinates to the tool position



## Tool Data

Aim of the tool data calculation: The control should use the tool tip or the tool centre point for positioning, not the tool mount reference point.

Every tool used for machining must be measured. Important is to measure the distance from the tool tip to the tool mount reference point "N".

In the so-called tool data register the measured tool length data and tool radii can be stored.

Indicating the cutter radius is necessary **only** when a **cutter radius compensation** is used for this tool!

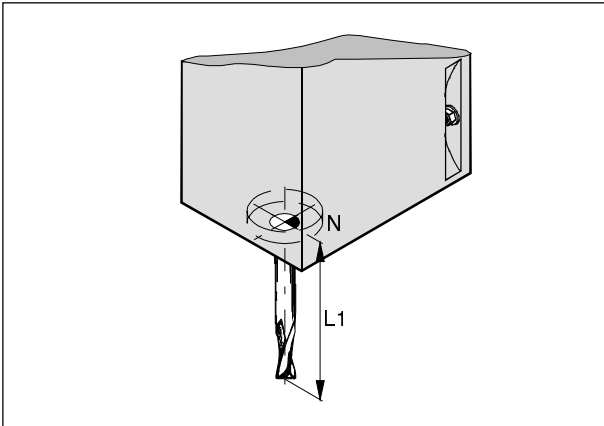
For G17 (XY plane active) applies:

The tool data measuring occurs for:

L1: in Z direction absolute from point "N"

R: cutter radius

For all other active planes, L1 is always vertical to the active plane. In this manual the normal application G17 is described.

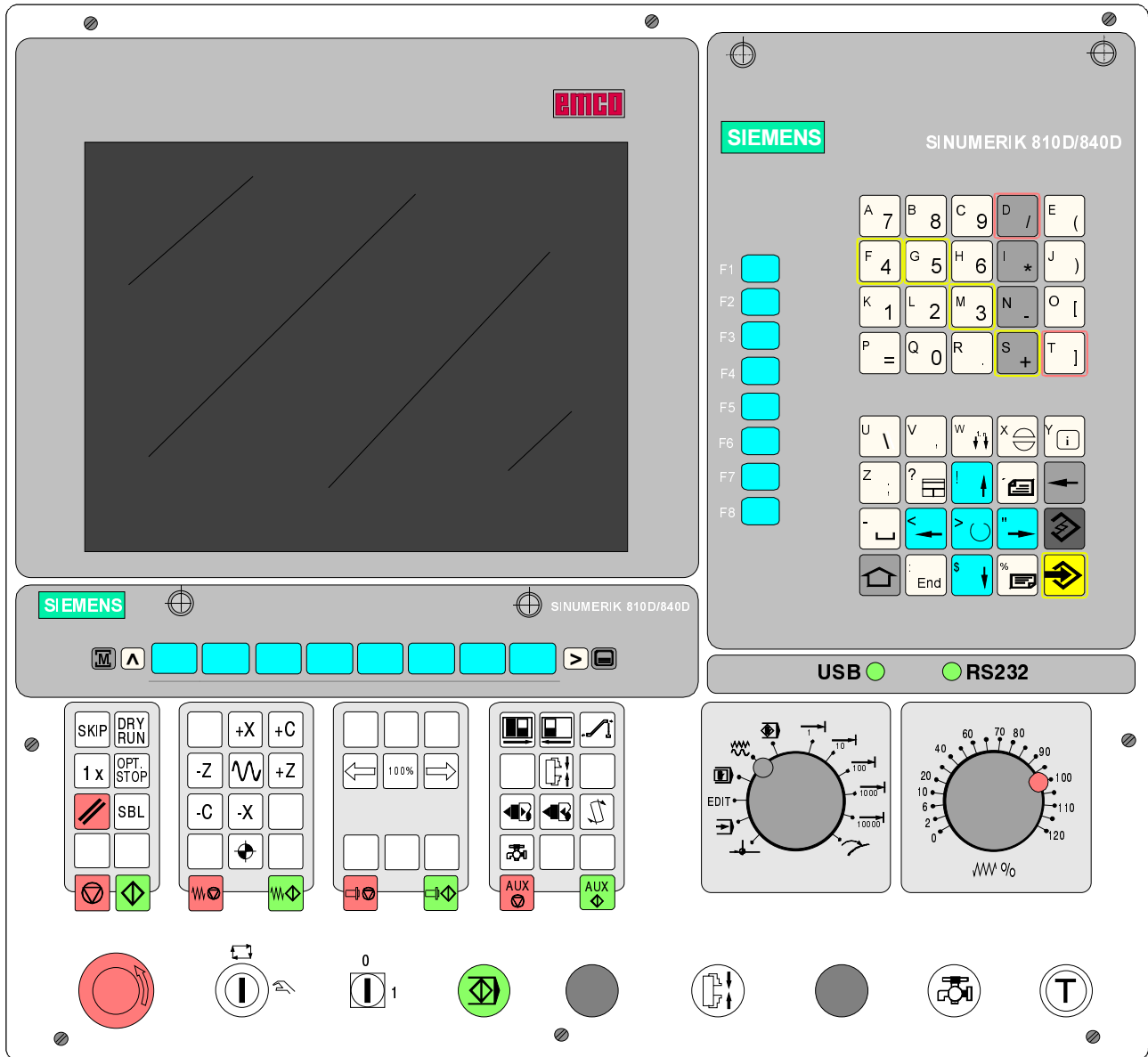


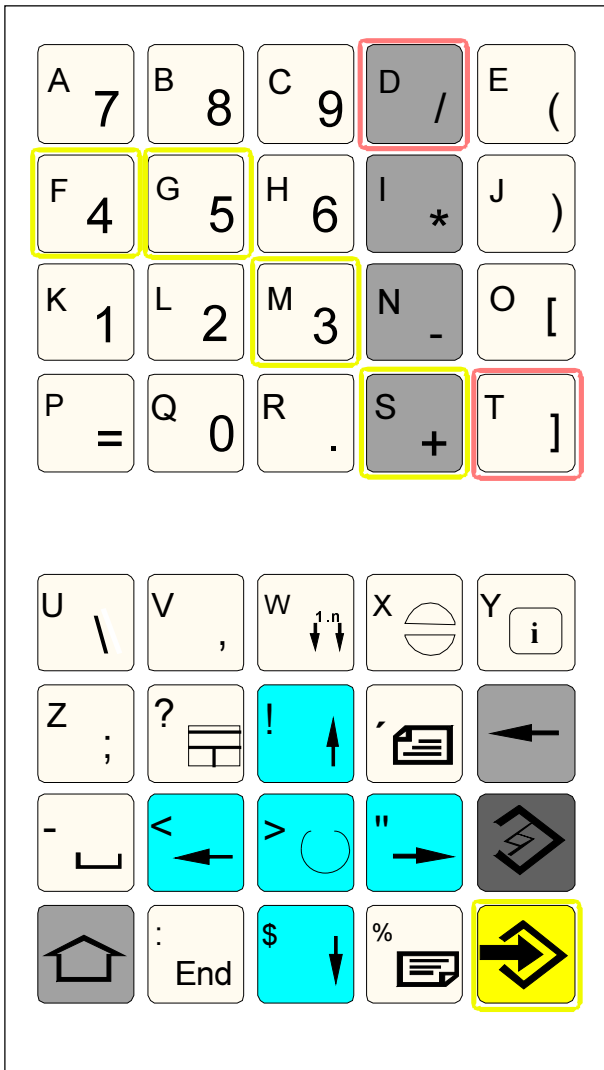
*Length correction*



# B: Key Description

## Control Keyboard, Digitizer Overlay





Address and numeric keyboard

### Address and Numeric Keyboard

The shift key bottom left shifts to the second key function (indicated in the left top edge of the keys).

Example:



### Double-Shift Function

1 x Shift:

For the following key press the second key function will be done, for all following inputs the first key function.

2 x Shift:

For all following key presses the second key function will be done (shift lock).

3 x Shift:

For the following key press the first key function will be done, for all following inputs the second key function.

4 x Shift:

Deselect the 2x or 3x shift function.

## Key Functions



Direct jump to the Operating Area Machine



Jump back to the superior menu (recall)



Expanding the softkey line in the same menu



Show basic menu (selection Operating Areas)

If pressed again jump back to the previous menu



Confirm alarm

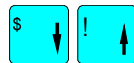


Show information for the actual operating status - works only when the dialogue line shows an "I".

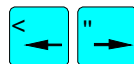


Select window (when several windows are on the screen)

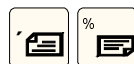
Keyboard inputs are valid for the selected window only.



Cursor down / up



Cursor left / right



Leaf backward / forward



Blank



Clear (Backspace)



Selection key / Toggle key

- Selection of predefined input values in input fields and lists, which are marked with this symbol
- Activate / deactivate switch box / radio button
  - = active
  - = not active



Edit key / Undo

- Switch to edit mode in tables and input fields
- Undo function for table elements and input fields (leaving a field with this key does not store the entered value but reestablishes the old value)



Jump to line end (list end)



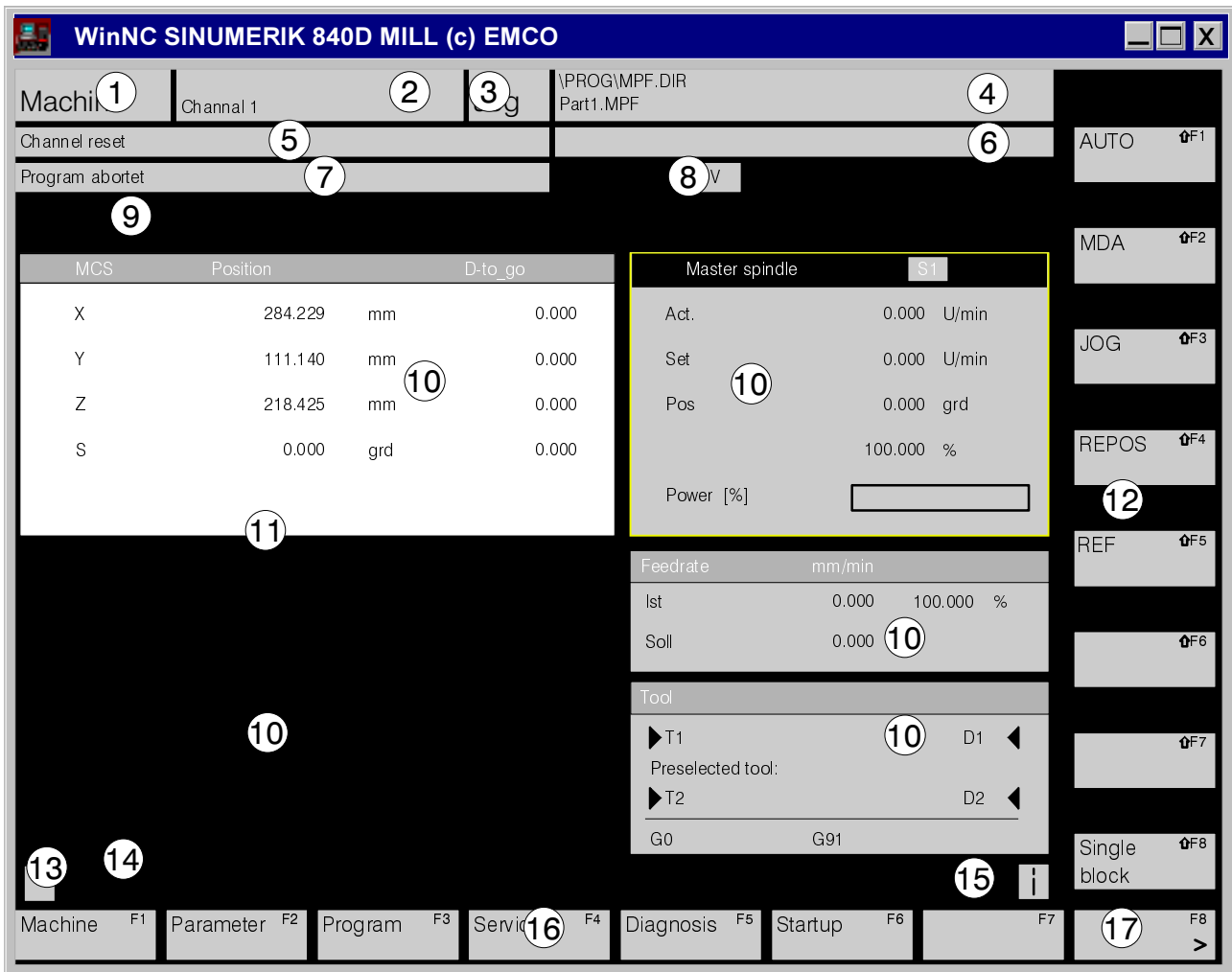
Input key




- Take over an edited value
- Open / close directory
- Open file



Shift key

### Screen Division

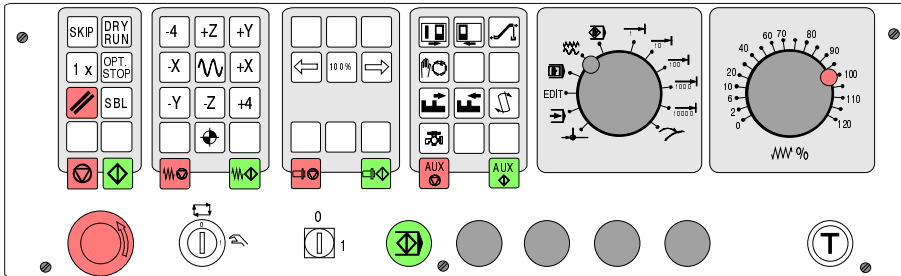


- 1 Display of the active Operating Area
- 2 Display of the active channel
- 3 Operating mode, when a submode is active, it also will be displayed (e.g. REF, INC)
- 4 Program path and name of the selected program
- 5 Channel status
- 6 Channel operating messages
- 7 Program status
- 8 Channel status display (SKIP, DRY, SBL, ...)
- 9 Alarm and message line
- 10 Working window, NC display  
The working windows (program editor) and NC displays (feed, tool) available in the active Operating Area are displayed here.
- 11 The selected window is marked with a border and the headline is displayed inverted. The keyboard inputs are effective here.
- 12 Vertical softkeys  
These 8 fields show the functions of the keys right beside. (at the PC: Shift F1..F8)
- 13 When this symbol is displayed, the key  is active (jump back to superior menu is possible).
- 14 Dialogue line with operator notes
- 15 When this symbol is displayed, the key  is active (information available).
- 16 Horizontal softkeys  
These 8 fields show the functions of the keys below. (at the PC: F1..F8)
- 17 When this symbol is displayed, the key  is active (more softkey functions available in this line).

## Machine Control Keys

The machine keys are in the lower part of the control keyboard or digitizer overlay.

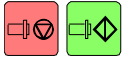
Depending on the used machine and accessory not all of these functions are active.



Machine control keyboard of the EMCO Concept-Mill Series

## Key Description

	SKIP (skip blocks will not be executed)
	DRY RUN (test run of programs)
	Single piece mode
	OPT STOP (program stop at M01)
	RESET
	Single block machining
	Program stop / program start
	Manual axis movement
	Rapid
	Approaching the reference point in all axes
	Feed stop / feed start
	Spindle override lower / 100% / higher



Spindel stop / spindle start; spindle start in JOG and electronic handwheel

Clockwise: press key short, Counterclockwise: press min. 1 sec.



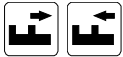
Consent key



Open / close door



Swivel dividing head



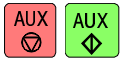
Open / close clamping device



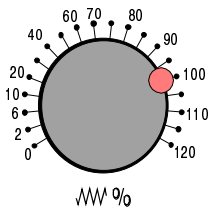
Swivel tool turret



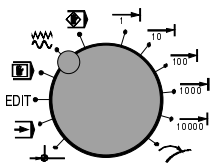
Coolant on/off



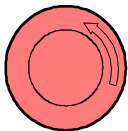
AUX OFF / AUX ON (auxiliary drives off / on)



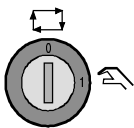
Feed / rapid feed override switch



Mode selector



EMERGENCY OFF (Unlock: pull out button)



Key switch for special operations (siehe Maschinenbeschreibung)



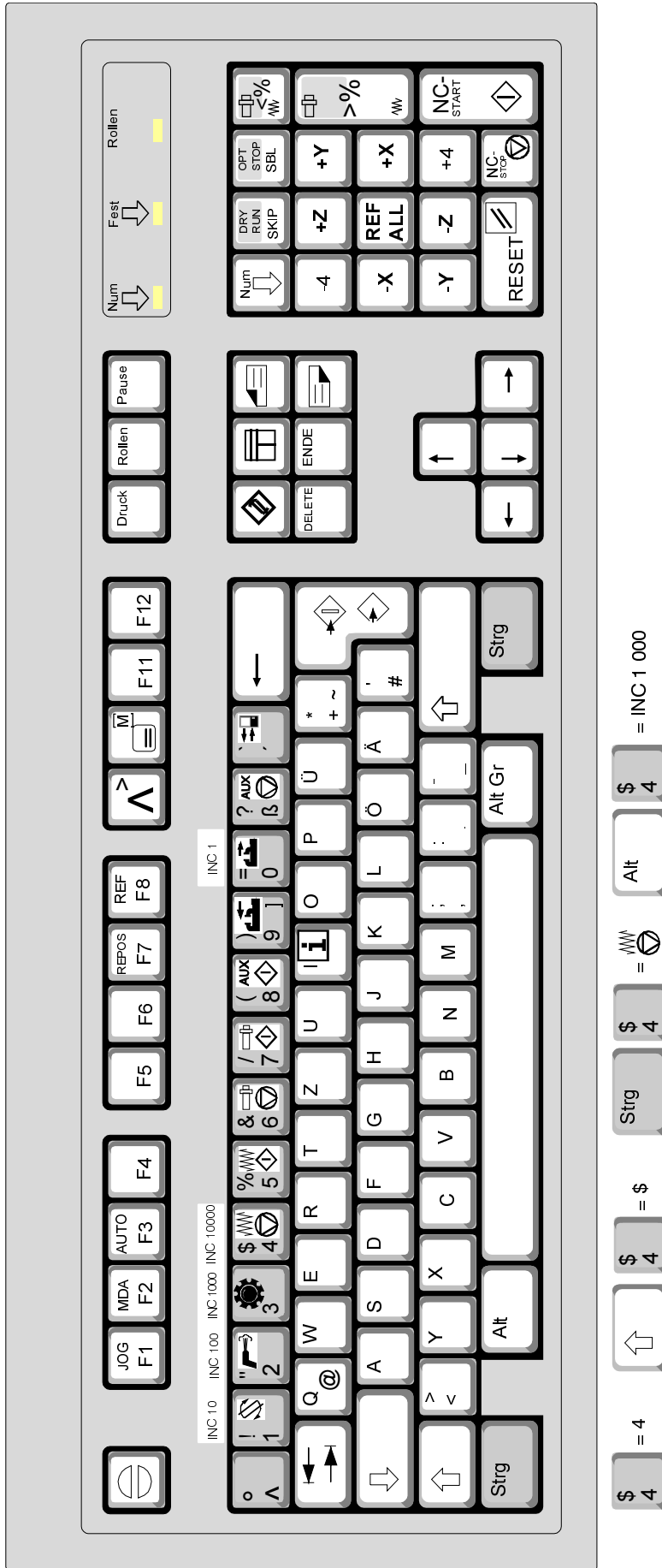
Additional NC start key



No function



PC Keyboard



Bold lined keys are special functions for control and machine, to activate hatched key functions press the ALT or CTRL key at the same time.

Pressing F10 shows the Operating Areas (Machine, Parameter, ...) in the horizontal softkey line.  
 Pressing Shift F10 shows the operating modes (AUTOMATIC, JOG, ...) in the vertical softkey line.

Pressing ESC confirms some alarms.

The meaning of the key combination CTRL 2 depends on the machine:

- MILL 50/55: Puff blowing ON/OFF
- MILL 100/125/155: Coolant ON/OFF

The assignment of the accessory functions is described in the chapter "Accessory Functions"

The machine functions in the numeric key block are active only with inactive NUM lock





## C: Operation

### Operation principle


The operation of the SINUMERIK 810D/840D is organized in 6 menus, so-called Operating Areas:

- Machine
- Parameter
- Program
- Services
- Diagnosis
- Start-Up

These six operating areas are displayed in the basic menu in the horizontal softkey line.

### Call basic menu



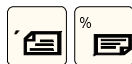
Press the key  to display the basic menu with the six operating areas in the horizontal softkey line. From any menu the basic menu can be called with this key.

By pressing again this key you will go back to the previous menu.

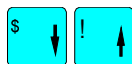


### Navigation in the menu window

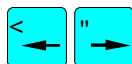
- Change menu window  
With this key you can change the active window (the active window is marked with a coloured border). Inputs can be done in the active window only.

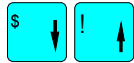


- Scrolling in the menu window  
Scrolls page forward or back.



- Place cursor in the menu window.



**Navigation in the directories**

- Select file / directory




- Open / close directory  
Open / close selected directory.



- Open file  
Open file when it should be processed in the editor..

**Edit inputs / values**

- Use the key  to activate / deactivate radiobuttons or switchboxes.

= active

= not active



- Input fields

Change into input mode.


Enter a value or term (e.g. file name) with the alphanumeric keyboard.

You automatically enter the input mode, when the cursor was placed on the input field previously.



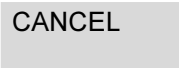
Acknowledge your input with the key "Input". The value will be taken over.



Use the key  to toggle between predefined values (e.g. forward-backward).

**Confirm / cancel input**

- Confirm input  
Save inputs and leave actual menu window (return to calling menu).



- Cancel input  
Reject inputs and leave actual menu window (return to calling menu).



- Reject inputs and leave actual menu window (return automatically to the next higher menu level).



- Reject inputs and stay in actual menu window.

**Mouse operation**

- 1x Click means:  
Activate menu window  
Put cursor on desired input field  
Select directory  
Press softkey  
Activate / deactivate radio button / switchbox  
Activate input field  
Open selection list
- 2x Click (double click) means:  
List selection  
Take over value / input  
Open directory

- Right mouse key means



Show operating areas

## Survey Operating Areas


The functionalities of the control are organized in operating areas.

Operating Area	executable functions
Machine	Work off part program Manual operation of the machine
Parameter	Editing data for programs and tool administration
Program	Creating and adapting part programs
Services	Read in / out programs and data
Diagnosis	Alarm display Service display
Start-Up	Adjusting the NC data to the machineSystem settings


## Operating Area Machine


The Operating Area Maschine covers all functions and influences, that lead to actions at the machine tool or detect its state.


Three operation modes:


- JOG   
Jog is used for manual operation and set-up of the machine.


Set-up functions::

Approach reference point (Ref) 

Repositioning (Repos) 

Increment traverse 





- MDA   
half-automatic operation  
Part programs can be created and worked off block-by-block.

- AUTOMATIC   
Fully automatic operation  
Working off part programs.  
Part programs will be selected, started, corrected, intentionally influenced (e.g. single block) and worked off.

These operating modes can be selected by softkey (PC keyboard) or with the operating mode selector switch.

### Approach reference point

By approaching the reference point the control will be synchronized to the machine.

- Select REF mode (  or Alt+F8 at the PC).
- Press the direction key  or  to approach the reference point in this axis, same for all other axes.
- With the key  all axes will be approached automatically (PC keyboard).









### Danger of collisions

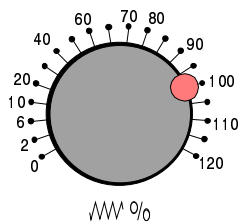
Mind for obstacles in the working area (clamping devices, clamped workpieces etc.).

After reaching the reference point its position will be displayed as actual position. Now the control is synchronized to the machine.

### Traverse slides manually

The machine axes can be traversed manually with the direction keys.

- Change to JOG mode (  or Alt+F1 at the PC).
- The keys , , , , , , etc. move the axes in the desired direction as long as the keys are pressed.
- Feed will be set with the override switch.
- When the key  is pressed simultaneously, the axes move with rapid speed.




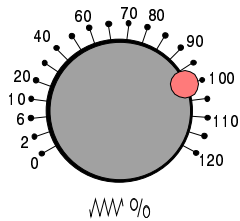


### Traverse slides in increments

With the direction keys you can traverse the slides in increments.


INC 1	1/1000 mm	per key press
INC 10	1/100 mm	per key press
INC 100	1/10 mm	per key press
INC 1000	1 mm	per key press
INC VAR	variable step measure	

- Change into INC mode (→|...|← or Alt+0 ... Alt+4 at the PC).  
(Alt 0..10<sup>0</sup>=1, Alt 1..10<sup>1</sup>=10, Alt 2..10<sup>2</sup>=100, ...)
- The keys **-X**, **+X**, **-Y**, **+Y**, **-Z**, **+Z**, etc. move the axes in the desired direction for the desired increment.
- Feed will be set with the override switch.
- When the key  is pressed simultaneously, the axes move with rapid speed.



### MDA mode

In the MDA mode (Manual Data Automatic) part programs can be created and ran block-by-block. Therefore enter the desired movements via the keyboard as single part program blocks into the control.

The control works off the blocks after pressing the key .

For a MDA program run the same preconditions are valid as for an automatic program run.

### Automatic mode

In the Automatic mode part programs can run fully automatic.

Preconditions for working off part programs:

- The reference point was approached
- The part program is loaded into the control.
- The necessary correction values have been checked or entered (e.g. zero offsets, tool corrections)
- The safety locks are active (e.g. chip guard door closed).

Possibilities in automatic mode:

- Program correction
- Block search
- Overstore
- Program influence

see chapter F - Program Run.

## Operating Area Parameter

In the Operating Area Parameter you can enter and edit data for program and tool correction.

### Tool data



see chapter E - Tool Measuring / Tool Administration.

### R Parameter (arithmetic parameter)

R Parameter are variables which can be used as calculation parameter within programs. These parameter can be edited manually in this operating range.

Press the softkey R PARAMETER.

You can page up and down in the R parameter list

using the keys  and .

Change parameter:

Position the cursor on the appropriate input field and enter the new value.

Delete parameter:


With the softkey DELETE AREA you can delete all parameters in the range R.. to R.. .

The softkey DELETE ALL deletes the entire R parameter range.

With the softkeys ABORT and OK you can abort or confirm deleting.

Find parameter:

Press the softkey SEARCH and enter the number of the parameter to be found.

When you press  the cursor will be placed on the specified parameter if it exists.

R para-  
meter

Delete  
area

Delete all

Abort

OK

Search

100 computation variables (= R parameter) of the REAL type are available as a standard under the address R with the control Sinumerik 840 D. The range R0 to R89 is at the free disposal of the user, R90 to R99 is reserved for EMCO.

### Workpiece counter (R90, R91)

Actual workpiece number

The actual workpiece number is indicated under parameter R90.

Nominal workpiece number

The nominal workpiece number is indicated under parameter R91.

Function

- If the workpiece number is entered in R91, parameter R90 counts from the preset nominal workpiece number downward to 0.

When the preset workpiece number is worked off, the message "Nominal workpiece number reached" is displayed.

- If the workpiece number R91 and the actual workpiece number R90 is set to 0, parameter R90 counts from 0 upward.

Programming

The call-up of the workpiece counter in the program is carried out just before the M30 command with **L700 P1**.

### Example

250 workpieces are manufactured.


- Nominal workpiece number R91 = 250 set  
The workpiece counter counts from 250 to 0 downward and then emits the message "Nominal workpiece number reached".
- Nominal workpiece number R91 = 0 set  
Actual workpiece number R90 = 0 set  
The workpiece counter counts from 0 to 250 and does not emit a message.

**Setting data****Work. area limitation**

- Working area limitation

With the working area limitation you can set limits in which the tool can be moved.

Place the cursor on the desired input field and enter the new value.

Activate your input with the key .

**Note:**

In MDA and Automatic modes, within an NC program the working area limitation is active only after the WALIMON command.

**JOG data**

- JOG data

**JOG feedrate**

Feed for the axes for manual traversin in JOG operation.

Enter the new value

**Spindle data**

- Spindle data

**Max. / min:**

Limitation of the spindle speed. It is permitted only within the maximum and minimum values that are defined in the machine data


**Spindle speed limitation at G96:**

Programmable upper spindle speed limit (G96) for constant cutting speed.

**Gear:**

Enter the selected gear step for machines with mechanical gear.

By that the speed of the main spindle can be monitored correctly.

Place the cursor on the input field and enter the new value or select the value with the key .

**Feedrate  
DRY**

- Dry run feedrate

This feedrate is used in DRY RUN instead of the programmed feedrate.

**Starting  
angle**

- Starting angle for thread cutting

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

### Zero offset

- Change settable zero offset (G54 - G57):

#### Coarse offset:

On most machine tools, this value is secured against unauthorized changes with a key switch.

#### Fine offset:

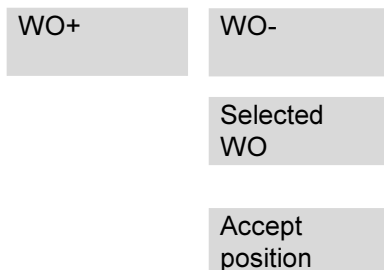
Is used for fine corrections of the coarse offset (e.g. wear correction) and is not secured against changes. The input value for fine offset is limited to  $\pm 1$  mm.

The effective settable zero offset is the total sum of fine and coarse offset.

#### Rotation, scale, mirror:

Determination like programming of "Frames". Rotation can only be set around geometry axes.


See chapter D - Programming - Coordinates, Zero offsets.



Select the desired zero offset G54 - G57 with the softkeys WO- and WO+.

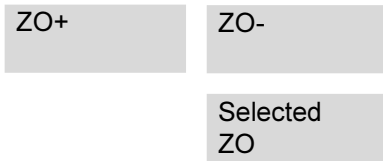
Select the active zero offset (in MDA or Automatic) with the softkey SELECTED WO.

The softkey ACCEPT POSITION is displayed only if an axis position is entered in the input field. This position is transferred to the control when you press ACCEPT POSITION.

Enter the values into the input field or select a new value with the key  (mirror).



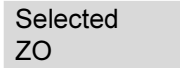
SAVE will save the new values, REJECT resets the altered values to the original values.



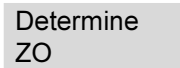
- Measure settable zero offset (G54 - G57) (touch):

Select the desired zero offset G54 - G57 with the softkeys ZO- and ZO+.

Select the active zero offset (in MDA or Automatic) with the softkey SELECTED ZO.



You must be in JOG mode.




Press the softkey DETERMINE ZO.

Enter the corresponding tool data for your touching tool:

T-No. tool

D-No. correction (edge)

Use the key  to select:

- the relevant length parameter (1, 2, 3) and direction (+, -, without)
- inclusion and direction of radius (+, -, without)
- inclusion and direction of a freely definable offset (+, -, without)



Confirm the touching tool with OK.



Place the cursor on the corresponding input field of the offset (e.g. Z coarse).

Touch on the corresponding position of the zero offset (e.g. clamping device position in Z) and press the softkey OK.

The position will be taken over.

Touch on all desired positions of the zero offset.



SAVE will save the new values, REJECT resets the altered values to the original values.



- Change base zero offset:

The base zero offset is a zero offset which is always active (without special call-up).

The base zero offset is used for e.g. spacer tables on milling machines or intermediate flanges on lathes.

Coarse offset:

On most machine tools, this value is secured against unauthorized changes with a key switch.

Fine offset:

Is used for fine corrections of the coarse offset (e.g. wear correction) and is not secured against changes. The input value for fine offset is limited to  $\pm 1$  mm.

The effective base zero offset is the total sum of fine and coarse offset.

Rotation, scale, mirror:

Determination like programming of "Frames". Rotation can only be set around geometry axes.

See chapter D - Programming - Coordinates, Zero offsets.

Overview

Base  
ZO

Select base zero offset:

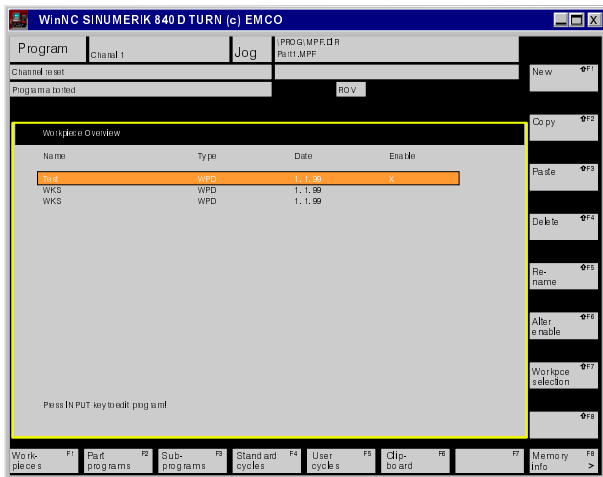
Press the softkeys OVERVIEW and BASE ZO.

Input of the base zero offset is the same like for settable zero offsets.

### Total effective zero offset

The total zero offset, that is active in a part program, is the sum of base zero offset + settable zero offset G54-G599 + Frames.

Total ZO = Base + G54-G599 + Frames



## Operating Area Program

In the Operating Area Program, part programs can be created and adapted, and part programs can be administrated.

### Program types

- Part program

A part program is a sequence of commands for machining a workpiece.

- Subprogram

A subprogram is a sequence of part program commands, that can be called up multiple with different supply parameters.

Cycles are a kind of subprograms

- Workpiece

In this context a workpiece is a directory, that includes programs or other data.

- Cycles

Cycles are subprograms to execute repeated used machining steps at the workpiece.

The preprogrammed standard cycles can not be altered.

User cycles can be created and altered at will.

**Program administration**

- File and directory types

name.MPF	main program
name.SPF	subprogram
name.TOA	tool data
name.UFR	zero offset / Frame
name.INI	initializing file
name.COM	comment
name.DEF	definition for global user data and makros
name.DIR	common directory, contains programs, workpiece directories and other directories with the extension .DIR. The names of these directories (MPF.DIR, DPF.DIR, CLIP.DIR, ...) are predefined and can not be altered.
name.WPD	workpiece directory, contains program and data modules, which belong to the workpiece (it must not contain other directories with the extension .DIR or .WPD)
name.CLP	clipboard directory, may contain all types of files and directories.


- Copy / Paste

Place the cursor on the file to be copied and press the softkey COPY.

The file will be marked as copy source.

Enter the directory, in which the marked file should be copied and press the softkey PASTE.

With paste into a workpiece directory the type can be

altered with the key .

When the name of the source file and the copied file should be the same, press the softkey OK.

When the copied file should get a new name, enter the new name with the keyboard and press the softkey OK.

Copy

Paste

OK


Re-  
name

- Rename

Place the cursor on the file to be renamed and press the softkey RENAME.

The rename dialogue window will be opened.

Enter the new name.

The file type can be changed with the key .


Delete

OK

- Delete

Place the cursor on the file to be deleted.

To mark several files, place the cursor on the first file,

press the key  and place the cursor on the last file.

Press the softkey DELETE.

Confirm the safety query with the softkey OK and all marked files / directories will be deleted.

- Programs can be deleted only when they are not in process.
- To delete a workpiece directory, no program in this workpiece directory must be selected.
- When a workpiece directory is deleted, all files within this directory will be deleted.

Alter  
enable

- Enable

A program can be worked off only when it is enabled. Enabled programs are marked with an "X" in the program list.

To enable or disable a program, mark the program and press the softkey ALTER ENABLE.

Work-  
piece

new

### Create workpiece directory

Press the softkey WORKPIECE. Die Übersicht aller Werkstückverzeichnisse wird eingeblendet.

Press the softkey NEW. The input window will be opened.

Enter the name for the new workpiece directory with the keyboard.

The name of the first part program will be asked for and the editor for this part program will be opened.

new

### Create / edit program

- Create new program

Open the workpiece directory in which the new program should be created.


Press the softkey NEW and enter the new program name.

The corresponding file name can be selected with the

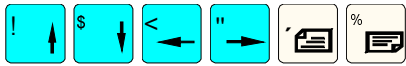
key .

- Select existing program

Place the cursor on the program that should be edited.

Press the key .

The text editor for the selected file will be opened.



• Edit program

Place the cursor in the text with the cursor keys and the keys page up / page down.

The delete key deletes the character left from the cursor.



The input key closes a block. A LF sign (line feed) will be written into the program. Only after that the entered block will be accepted for machining.



Overwrite

Paste

With the softkeys OVERWRITE and PASTE you can toggle between overwrite and insert mode.

Mark block

Mark, copy, paste, delete block:

Place the cursor on the beginning of the block and press the softkey MARK BLOCK. Move the cursor on the end of the block and the block will be marked automatically.

Copy block

COPY BLOCK copies the block in the clipboard. Also with program change the block stays in the clipboard.

Insert block

INSERT BLOCK inserts the block from the clipboard into the text before the cursor position.

Delete block

DELETE BLOCK deletes the marked block.

By pressing again MARK BLOCK you will leave the marking mode.

Renumber

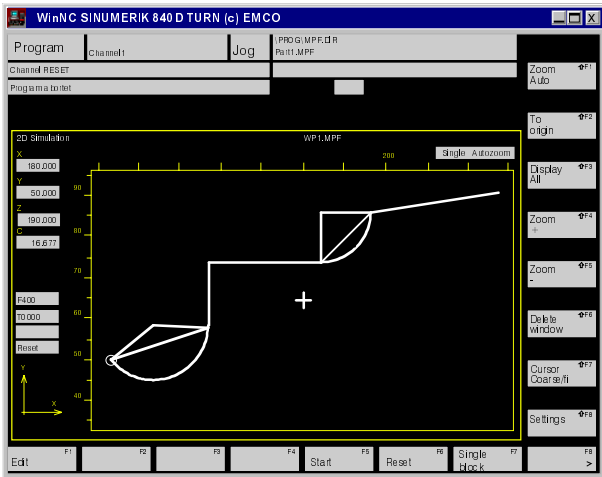
Renumber:

With the softkey RENUMBER the program blocks in the editor will be renumbered.

Close

CLOSE

A query will be displayed whether the changes should be stored. The the text editor will be closed and the screen shows the program overview.



**Program simulation**

While editing a program the programmed tool movements can be simulated graphically at the screen.

By that the program can be tested for geometrical and formal correctness.

Not recognized will be technological mistakes (e.g. wrong sense of rotation, wrong feedrate, etc.).

Display colors:

- light green traversing path with feed
- dark green traversing path with rapid feed
- yellow reticule, tool symbol, symmetry axes etc.

blue circle auxiliary lines

The simulation window shows the actual axis positions, feed, tool, status Run/Reset and the settings Autozoom and Single.

Simulation

Press the softkey SIMULATION.

Start

The softkey START starts the simulation.

Reset

RESET sets back the simulation.

Single

With SINGLE the simulation runs block-by-block (continue with START).

Zoom Auto

ZOOM AUTO displays all traversing pathes in the graphic traversing area adapted to the window size.

To Origin

TO ORIGIN reestablishes the basic picture (cancel zoom functions)

Display all

DISPLAY ALL shows the complete traversing area of the machine.

Zoom +

Zoom -

ZOOM + and ZOOM - set the zoom factor of the display.

Previously put the reticule with the cursor keys to the desired centre of display.

Delete window

DELETE WINDOW clears the simulation display.

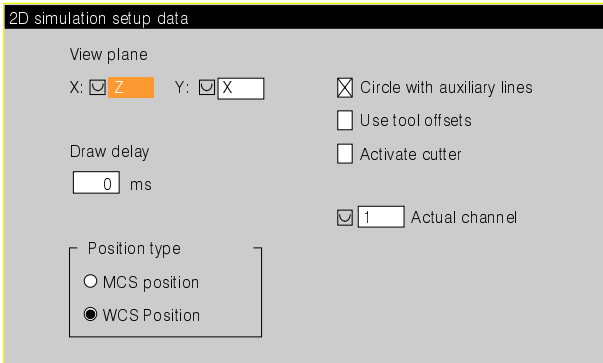
Cursor coarse/fine

CURSOR COARSE/FINE changes the step with of the cursor steps.

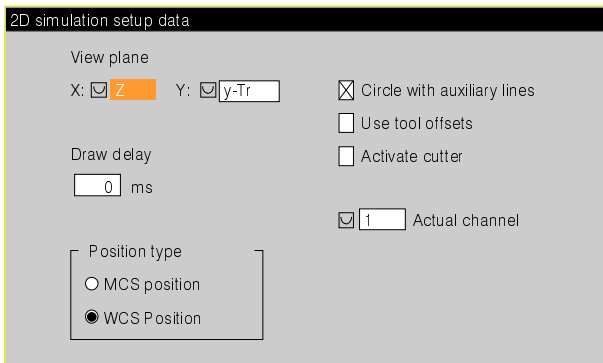
Edit

EDIT returns to the program editor.

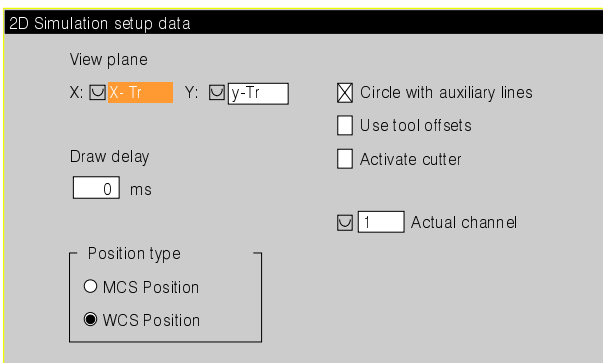
Settings



Normal presentation level



Presentation levels - selection for Tracyl




Presentation levels - selection for Transmit

Simulation settings:

The softkey SETTINGS opens the settings window for the simulation.

With view plane you can decide, which plane is displayed in simulation.

Below draw delay you can enter a delay time between the simulation blocks. By that the simulation will be more representative.

When you press the key  while a running simulation, the rest of the simulation will run without delay.

Below position type you can select whether the simulation data will be displayed in the machine coordinate system (MCS) or in the workpiece coordinate system (WCS).

Circle with auxiliary lines

Select / deselect the display of radii and the string between the circle end points.

Use tool offsets


Simulation with / without tool offsets.

Activate cutter

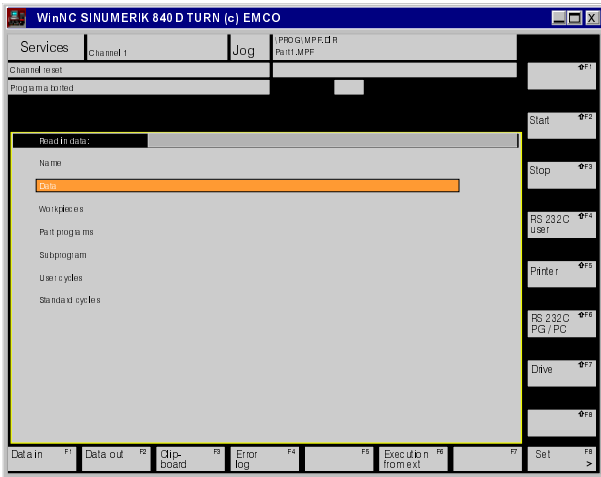
Simulation with cutter radius compensation or without cutter radius compensation (display of tool center path).

Actual channel

Selection of the actual channel (only on machines with more than one channel).

 The main program, the subroutines and the cycles must be released to enable simulation.

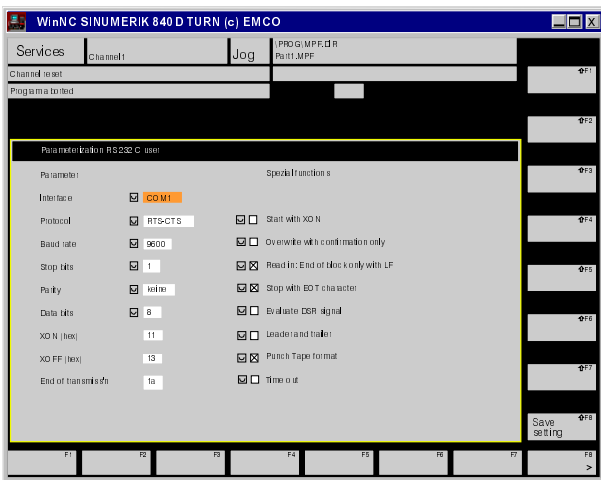




## Operating Area Services

The Operating Area Services is used to read in or send data via interface COM1 - COM4.

Print data with the softkey PRINTER, and transmit data to / from disk drive with the softkey DRIVE.



### Interface settings

For data transmission the settings of sender and receiver must be the same, otherwise transmission would not work.


With EMCO WinNC you can send only via the interface RS 232 C User.

The function RS 232C PG/PC is not active.

Press the softkeys RS232 C USER and SET.

Place the cursor on the input fields and enter the corresponding values.

The values of the parameter Interface, Protocol, Baud rate, Stop bits, Parity, Data bits can be selected

with the toggle key ().

The parameters Protocol, Baud rate, Stopbits, Parity, Data bits, XON-Sign, XOFF-Sign and DSR Signal are getting separately saved for every interface. While changing the interface the new parameters are shown and can be modified.

Also the special functions can be selected with the toggle key.

The settings can be saved with the softkey SAVE SETTING.

### Drive settings

Press the softkeys DRIVE and SET.

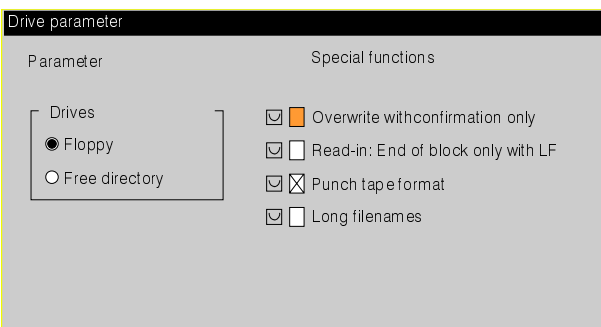
Select Floppy (Drive A:) or Free Directory.

With the option Free Directory you can select any directory, e.g. on hard disk C:.

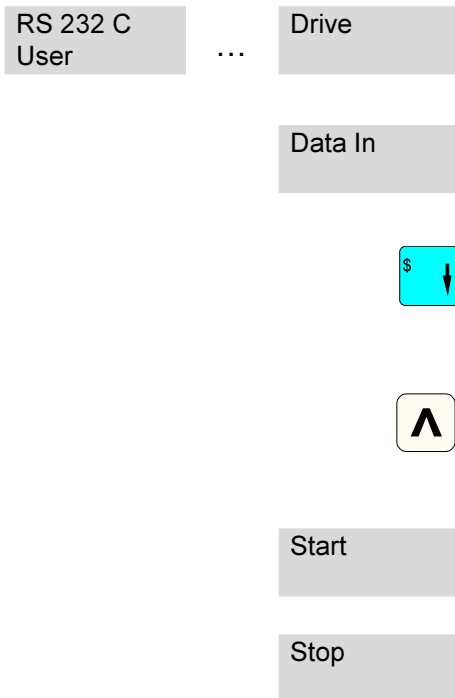
The special functions can be selected with the toggle key.

The settings can be saved with the softkey SAVE SETTING.

Save setting



Save setting




### Read-in data

Select data source with softkey RS 232 C USER to DRIVE.

Press the softkey DATA IN.

Place the cursor on the desired directory in the directory list.

The data read-in will be written into this selected directory.

With the softkey  you will get back into the superior directory.

The softkey START starts reading in. All data will be written into the previous defined directory.

The softkey STOP stops reading in.

#### Notes:

- While reading in it is possible to call up certain selected data by the control only from DRIVE.
- When the option "Overwrite with confirmation only" is selected, existing data will be overwritten only after confirmation. On reject, read-in will be continued with the next file.
- Only data with a valid extension (e.g. .MPF) can be read in (except read-in into clipboard).

A prerequisite for the data exchange is that the parameterization of transmitter and receiver is in accordance (see operating range services).

"No" tool correction values from the WinNC may be imported into the machine tool administration, since this procedure may cause system errors.

### Data exchange WinNC - machine

- Set receiver to ready-to-receive state as described under "Read in data".
- Set receiver to ready-to-receive state as described under "read out data" and confirm by means of the softkey START.
- Actuate read-in on receiver by means of the softkey STOP.

RS 232 C  
User

...

Drive

Data Out

### Send data

Select sending target with softkey RS 232 C USER to DRIVE.

Press the softkey DATA OUT.

You can send following data types:

- Data (tool and magazine data, R parameter, zero offsets)
- Workpieces
- Part programs
- Subprograms
- User cycles
- Standard cycles

Selection of data to be sent:

Place the cursor on the desired data type.



E.g. the cursor is on "Workpiece".

When you start transmission now, all workpieces will be sent.




Press INPUT and a list of workpieces will be displayed and you can mark a workpiece.

When you start transmission now, all data included in the workpiece (programs, subprograms etc.) will be sent.

Press INPUT again and a list of the data included in the workpiece will be displayed, etc..

The you can send single files.



With the softkey  you will get back into the superior directory.

Start

The softkey START starts sending data.

Stop

The softkey STOP stops sending data.

**Copying and pasting data from the clipboard**

In the clipboard all types of data may be stored (e.g. after read-in from serial interface etc.). You can sort these data (programs, subprograms etc.) in the corresponding directories (MPF.DIR, SPF.DIR, etc.).

Clip-board

Press the softkey CLIPBOARD.

The cursor is placed on a file in the clipboard window (lower window). Place the cursor on the file that should be put into the directory structure.



Place the cursor into the upper window (target window).



Place the cursor on the desired directory in the structure. Into this directory the file from the clipboard will be written.

Copy and paste

Press the softkey COPY AND PASTE.

The control asks for a file name. Keep the old name or enter a new name. Confirm with OK.

OK

The file is copied into the target directory.



Go back to the directory overview with the softkey



Delete

Notes:

- The files will be copied from the clipboard but not automatically deleted in the clipboard.
- Use the softkey DELETE to delete data in the clipboard.

## Operating Area Diagnosis

The Operating Area Diagnosis shows alarms and messages in the full length form.

With an active alarm or message change to the Operating Area Diagnosis, to get following information:

**Number:**

Alarm number. With several active alarms the will be displayed in time sequence.

**Date:**

Exact date, time when the alarm occurred.

**Delete criteria:**

Display of the key that must be pressed to delete the alarm.

**Text:**

Full length text of the alarm.

### Display of software versions

Press the softkey SW VERSIONS to show the versions of the individual software components. These versions must be indicated when requesting service.

## Operating Area Start-up

The Operating Area Start-up is not active in WinNC.

For including accessories, automatisations etc. use the auxiliary software "WinConfig"

## D: Programming

**Note**


This programming chapter describes all functions that can be done with WinNC.

Depending on the machine that is operated with WinNC not all of these functions may work.

The parameters or cycles marked with (\*) are only available for cycle version 6.02.

Example:

The milling machine Concept MILL 55 has no position controlled main spindle, therefore no spindle position can be programmed.



When programming in editor, set a blank between every single word (z.B.: G0 X20 Z-35).

# Surveys

## G- commands

COMMAND	MEANING
G0	Rapid traverse
G1	Machining traverse
G2	Circular interpolation clockwise
G3	Circular interpolation counterclockwise
CIP	Circular interpolation via intermediate point
G4	Dwell time
G9	Exact stop non-modal
G17	Working plane selection XY
G18	Working plane selection XZ
G19	Working plane selection YZ
G25	Minimum programmable working area limitation / programmable spindle speed limitation
G26	Maximum programmable working area limitation / programmable spindle speed limitation
G33	Thread pitch constant
G331	Tread tapping
G332	Tread tapping / retraction
G40	Cutter radius compensation OFF
G41	Cutter radius compensation ON Left
G42	Cutter radius compensation On Right
G53	Deselection settable zero offset
G54-G57	Settable zero offset
G500	Deselection zero offset
G505-G599	Settable zero offset
G60	Exact stop modal
G601	Step enable if positioning window fine reached
G602	Step enable if positioning window coarse reached
G603	Step enable if setpoint reached
G63	Tread tapping without synchronization
G64	Contouring mode
G641	Contouring mode with programmable transition rounding
G70	Input system inch
G71	Input system metric
G90	Absolute dimensioning
G91	Incremental dimensioning
G94	Feed in mm/min, inch/min
G95	Rotational feed rate in mm/rev, inch/rev
G96	Constant cutting speed ON
G97	Constant cutting speed OFF
G110	Pole parameter, with reference the last approached position
G111	Pole parameter, absolute in the workpiece coordinate system
G112	Pole parameter, with reference to the last valid pole
G140	Soft approach and leaving
G141	Approch from the left and/or leaving from the left



COMMAND	MEANING
G142	Approach from the right and/or leaving from the right
G143	Direction of approach and/or leaving according to the relative position from start and/or end point to the tangent direction
G147	Approach with a straight line
G148	Leaving with a straight line
G247	Approach with a quarter circle
G248	Leaving with a quarter circle
G340	Approach and leaving in space (starting position value)
G341	Approach and leaving in the plane
G347	Approach with a semicircle
G348	Leaving with a semicircle
G450	Approach and leave contour
G451	Approach and leave contour

## M- Commands

COMMAND	MEANING
M0	Programmed stop
M1	Optional stop
M2	End of program
M3	Spindle on clockwise
M4	Spindle on counterclockwise
M5	Spindle stop
M6	Tool change
M8	Coolant on
M9	Coolant off
M10	Lock dividing head
M11	Unlock dividing head
M17	Subprogram end
M25	Clamping device/ vice open
M26	Clamping device/ vice close
M27	Swivel dividing head
M30	Program end
M70	Spindle positioning
M71	Puff blowing on
M72	Puff blowing off

## Cycles

BEFEHL	BEDEUTUNG
CYCLE 71	Face milling
CYCLE 72	Contour milling
CYCLE 81	Drilling, Centering
CYCLE 82	Drilling, Counterboring
CYCLE 83	Deep hole drilling
CYCLE 84	Rigid tapping
CYCLE 840	Tapping with compensation chuck
CYCLE 85	Boring 1
CYCLE 86	Boring 2
CYCLE 87	Boring 3
CYCLE 88	Boring 4
CYCLE 89	Boring 5
CYCLE 90	Thread cutting
HOLES 1	Row of hole with MCALL
HOLES 2	Circle of hole with MCALL
LONGHOLE	Longholes on a circle
POCKET 1	Rectangular pocket
POCKET 2	Circular pocket
POCKET 3	Rectangular pocket
POCKET 4	Circular pocket
SLOT 1	Slots on a circle
SLOT 2	Circular slot

## Command shortcuts

COMMAND	MEANING
AC	Enter single axis position absolute
ACN	Approach circular axis position in negative direction
ACP	Approach circular axis position in positive direction
AND	Logical AND connection
AP	Polar angle
AR	Circle segment angle
AXIS	Variable type
AX	Axis operator
AXNAME	String operation
AMIRROR	Mirror coordinate system, additive
AROT	Rotate coordinate system, additive
ASCALE	Alter scale, additive
ATRANS	Programmable zero offset
B_AND B_NOT B_OR B_XOR	Bit operators
BOOL	Variable type
CASE	Vocabulary word for jump instruction
CIP	Circular interpolation with intermediate point (circle through points)
CHAR	Variable type
CHF	Insert chamfer (chamfer)
CR	Circle radius
CFC	Constant feed at contour
CFIN	Constant feed at the contour for internal radius, constant feed at tool centre for external radius (constant feed at internal radius)
CFTCP	Constant feed in tool centre point
CONTPRON	Activate contour preparation in tabular form
CHR	Insert chamfer
D	Tool offset number
DC	Approach circular axis position on shortest way (direct to contour)
DIAMOF	Radius as dimension
DIAMON	Diameter as dimension
DEF	Define variable
DISPLOF	Suppress current block display
DISPLON	Display in the program window on
DIV	Integer division
DEFAULT	Check structures
DEFINE AS	Macro programming
DISC	Compensation at outside corners Flexible programming of the approach and retraction instruction
DISCL	Distance of the end point of the machining level during soft approach and leaving
DISR	Distance of the milling cutter edge from the starting point during soft approach and leaving
ELSE	Check structures
ENDFOR	Check structures
ENDLOOP	Check structures
ENDWHILE	Check structures
ENDIF	Check structures
EXECTAB	Block by block execution of contour elements of a table
EXECUTE	End definition
F	Feed

COMMAND	MEANING
FOR	Check structures
FRAME	Variable type
FAD	Speed of the slow feed movement with soft approach and leaving
GOTOB	Jump instruction with jump destination backwards
GOTOF	Jump instruction with jump destination forwards
I1	Address for circle interdot
IC	Enter single axis position incremental (incremental coordinates)
IF	Vocabulary word for condition
INT	Integers with leading sign
INTERSEC	Calculate contour preparation in tabular form
ISAXIS	Is a certain axis available (Enquiry via axis number)
J1	Address for circle interdot
KONT	Cutter radius compensation, surround contour in start and end point
K1	Address for circle interdot
LIMS	speed limitation
LOOP	Check structures
MCALL	Modal cycle or subprogram call (the following call stays active until it is deselected with MCALL without parameter) (modal call)
MSG	Message output in an alarm
MIRROR	Mirroring
N	Address of block number
NOT	Negation
NORM	Cutter radius compensation, direct contour approach (normal approach)
OFFN	Offset countur-normal
OR	Logical OR connection
P	Program repeat
PM	Indication of the feed FAD for approaching and departing as linear feed FAD=PM(...)
PR	Indication of the feed FAD for approaching and departing as feed per revolution
PROC	Subprogram start
R	R-parameter R[0]-R[99]
ROT	Rotate coordiante system (rotation)
REAL	Real number
RET	Subprogramm retreat
RND	Insert radius at contour corner (round)
RNDM	Insert radius at contour corner modal (round modal)
RP	Polar radius (radius polar)
RPL	Rotation of plane
REP	Field initialization
S	Spindle address
SAVE	Saving the register at subprogramm calling
SETAL	Alarms are programmed by inserting this keyword
SET	Initialization of value lists
SETMS	Define master spindle
SF	Starting point offset, only needed for multiple threads
SPCOF	Position controlled spindle operation off
SPCON	Position controlled spindle operation on
STRING	Variable type
SCALE	Alter scale factor (scale)
STRLEN	String operation
SPOS	Spindle position
SPOSA	Spindle position
SUBSTR	Define the part of a string

COMMAND	MEANING
SUPA	Blockwise suppression of all zero offsets (suppress all)
SBLOF	Single set suppression on
SBLON	Single set suppression off
TRANS	Programmed zero offset (translation)
T	Tool address
TURN	Number of turns for helical line (turns)
TRAFOOF	Deactivates an active transformation (transformation off)
TRACYL	XZ- plane transformation
TRANSMIT	XY-plane transformation
UNTIL	Check structures
VAR	Define variable
WAITP	Wait for the axis to finish traversing
WAITS	Wait for spindle position to be reached
WHILE	Check structures
WALIMOF	Working field limiting off
WALIMON	Working field limiting on
XOR	Exklusiv OR

### Arithmetic functions

COMMAND	MEANING
+, -, *, /, %, ^	Arithmetic function
SIN()	Sine function
COS()	Cosine function
TAN()	Tangent function
ASIN()	Arc sine function
ACOS()	Arc cosine function
ATAN()	Arc tangent function
ATAN2(,)	Arc tangent function (section X, section Y)
SQRT()	Root function
POT()	Power function
SQR()	Squaring function
EXP()	Exponential function (base e)
LOG()	Logarithm function (base e)
LN()	Natural logarithm function
PI	Circle partition number (3.141592...)
TRUE	logic True (1)
FALSE	logic False (0)
ABS()	Absolute function
TRUNC()	Integer number part function
ROUND()	Rounding function
MOD()	Modular function


## Calculator

Mathematical expressions can be directly evaluated in an input field.

You can use any number of parentheses in the mathematical expressions.

To calculate the terms, please press the "Enter" key or leave the input field.

In case errors occur during the formula evaluation, the term having been entered last will be displayed and WinNC will release an error message.

The currently displayed value can be used in a formula with .



The formula being entered last can be edited again by means of the key "?".

COMMAND	DESCRIPTION	EXAMPLE	RESULT
+	addition	1+1	2
-	subtraction	3-2	1
*	multiplication	5*3	15
/	division	15/3	5
%	modulus (remainder)	10%4	2
^	power	5^2	25
PI	circular graduation number	PI	3,141593
SIN()	sine	SIN(90)	1
ASIN()	arc sine	ASIN(-1)	-90
COS()	cosine	COS(90)	0
ACOS()	arc cosine	ACOS(-1)	180
TAN()	tangent	TAN(45)	1
ATAN()	arc tangent (value)	ATAN(1)	45
ATAN2( , )	arc tangent (X segment, Y segment)	ATAN2(0,1)	0
EXP()	exponential function (base e)	EXP(1)	2,718282
LOG()	logarithm (base e)	LOG(5)	1,609
SQRT()	square root	SQRT(2)	1,414
MOD( , )	modulus function	MOD(10,4)	2
TRUE	true	TRUE	1
FALSE	false	FALSE	0
AND	conjunction	1AND1	1
OR	disjunction	1OR1	1
NOT	negation	NOT(1OR1)	0

*Functions of the calculator*

## System variables

System variable	Description
\$P_AXN1	Current address of the geometry axis- abscissa
\$P_AXN2	Current address of the geometry axis- ordinate
\$P_AXN3	Current address of the geometry axis- applicate
\$P_IFRAME	Current settable frame
\$P_PFRAME	Current programmable frame
\$P_BFRAME	Actual base frame variable
\$P_ACTFRAME	Current total frame
\$P_UIFR[]	Adjustable frames (G54)
\$P_F	Path feed F last programmed
\$P_DRYRUN	0 (FALSE): Dry run on 1 (TRUE): Dry run off
\$P_SEARCH	1 (TRUE): Block search (with or without calculation) is active
\$P_TOOLR	Active tool radius (total)
\$P_TOOLNO	Active tool number T0 - T32000
\$AC_MSNUM	Master spindle number
\$MN_SCALING_SYSTEM_IS_METRIC	Ground system Metric (1: Metric, 2: Inch)
\$MN_SCALING_VALUE_INCH	conversion faktor from Metric to Inch (25,4)
\$MN_INT_INCR_PER_MM	Computing precision of the linear positions
\$MN_MIRROR_REF_AX	Reference axis for frame elements
\$P_SIM	1 (TRUE): Simulation in progress
\$P_SDIR[]	Last spindle rotational direction to be programmed
\$P_GG	Current G function of a G group lind. As for PLC interface
\$P_EP	Setpoint last programmed
\$MA_SPIND_ASSIGN_TO_MACHAX	Allocation of the spindle to the machine axis
\$MA_NUM_ENCS	Number of transducers
\$AA_S	Actual spindle speed: Sign corresponds to direction of rotation
\$MC_DIAMETER_AX_DEF	Plane axis number
\$P_AD[]	Active tool offset
\$P_TOOL	Active tool edge D0 - D9
\$P_MC	0 (FALSE): No modal subprogram call 1 (TRUE): Modal subprogramm call
\$P_TOOLL	Avtive overall tool length
\$A_IN[]	Digital input NC (1-16)
\$A_OUT[]	Digital output NC
\$A_INA[]	Analog input NC (1-4)



## Working Movements

### G0, G1 Linear interpolation (cartesian)

G0: Traverse with rapid feed e.g. for fast positioning  
 G1: Traverse with programmed feedrate F, e.g. machining the workpiece

#### Format

G0 X.. Y.. Z.,  
 G1 X.. Y.. Z.. F..

### G0, G1 Linear interpolation (polar)

#### Format

G0 AP.. RP..  
 G1 AP.. RP..

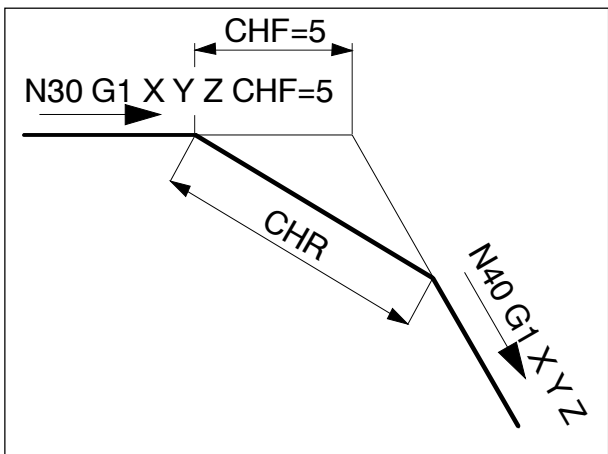
### Insert chamfer, radius

Chamfers or radii can be inserted between straights and arch in any combination.

#### Format

G..	X..	Y..	Z..	CHR=..	Chamfer
G..	X..	Y..	Z..	CHF=..	Chamfer
G..	X..	Y..	Z..	RND=..	Radius

Before programming, define the origin of the workpiece coordinate system with G111.



### Chamfer

The chamfer will be inserted after the block in which it was programmed. The chamfer is always in the working plane (G17).

The chamfer will be inserted symmetrically into the contour corner.

The CHF value is the length of the chamfer.

Example:

N30	G1	X..	Y..	CHF=5
N35	G1	X..	Y..	

### Radius

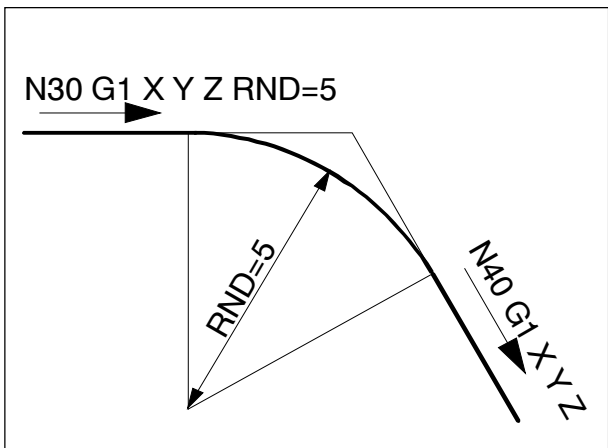
The radius will be inserted after the block in which it was programmed. The radius is always in the working plane (G17).

The curve is a circular arc and will be inserted into the contour corner with tangential connections.

The RND value is the radius of the arc.

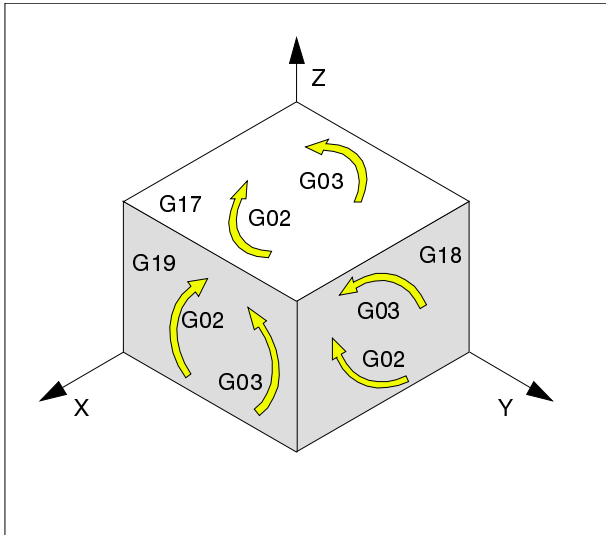
Example:

N30	G1	X..	Y..	RND=5
N35	G1	X..	Y..	



### Modal radius RNDM

At every following contour corner a radius will be inserted until the modal radius is deselected with RNDM=0.



**G2, G3, CIP Circular interpolation**

- G2 clockwise
- G3 counterclockwise
- CIP via intermediate point (**C**ircle through **P**oints)

Display of the circular movement in the different planes.

For a circular movement the start and end point are in one plane (level).

If a movement of the 3rd axis is programmed (e.g. for G17 the Z axis) a helix line will result.

Helix line:

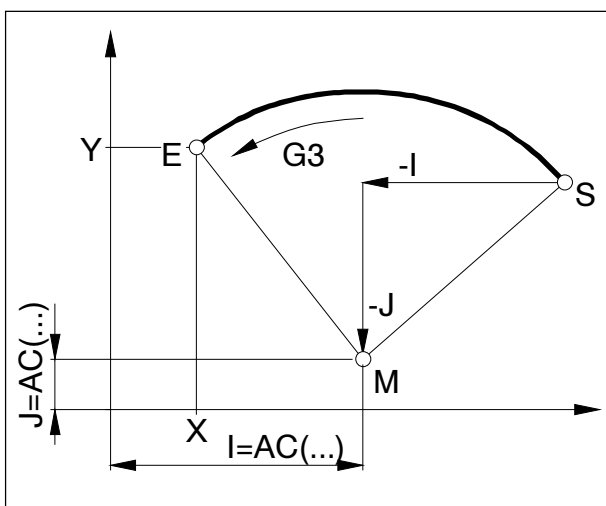
Programming an arc, start and end point have different Z values (G17).

If more than one circular run is requested the number of full circles must be indicated with TURN= .

**Programming with start point, end point, centre point**

G2/G3 X.. Y.. Z.. I.. J.. K..

- X, Y, Z end point E in cartesian coordinates
- I, J, K circle centre point M in cartesian coordinates, related to start point S



Start point

The start point is the position of the tool at the time of the call-up of G2/G3.

End point

The end point is programmed with X, Y, Z.

Circle centre point

The circle centre point is programmed incremental with I, J, K based on the start point or with I=AC(..), J=AC(..), K=AC(..) absolute from the workpiece zero point.

**Programming with start point, end point, circle radius**

G2/G3 X.. Y.. Z.. CR=±..

X, Y, Z end point E in cartesian coordinates  
 CR=± circle radius

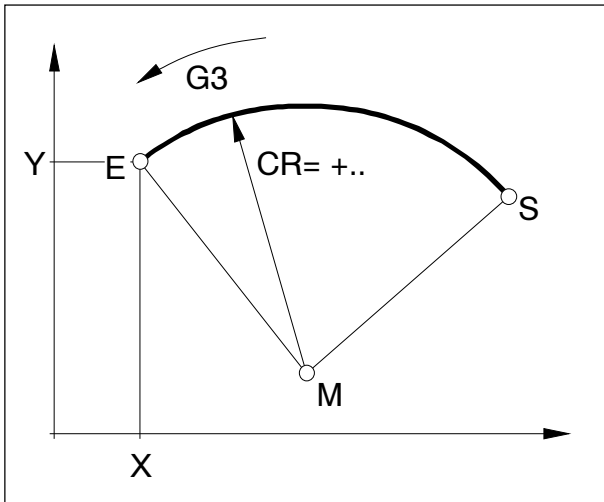
**Start point**  
 The start point is the position of the tool at the time of the call-up of G2/G3.

**End point**  
 The end point is programmed with X, Y, Z.

**Circle radius**  
 The circle radius is indicated with CR. The sign indicates whether the circle arc is smaller or larger than 180°.

CR=+ angle smaller or equal 180°  
 CR=- angle larger 180°.

Full circles can not be programmed with CR.



**Programming with start point, circle centre point or end point, spread angle**

G2/G3 X.. Y.. Z.. AR=.. or  
 G2/G3 I.. J.. K.. AR=..

X, Y, Z end point E in cartesian coordinates  
 or

I, J, K circle centre point M in cartesian coordinates, related to start point S

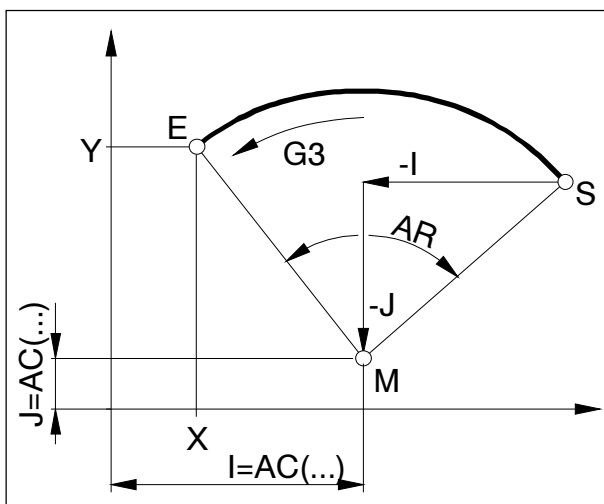
AR= spread angle

**Start point**  
 The start point is the position of the tool at the time of the call-up of G2/G3.

**End point**  
 The end point is programmed with X, Y, Z.

**Circle centre point**  
 The circle centre point is programmed incremental with I, J, K based on the start point or with I=AC(..), J=AC(..), K=AC(..) absolute from the workpiece zero point.

**Spread angle**  
 The spread angle must be smaller than 360°.  
 Full circles can not be programmed with AR.



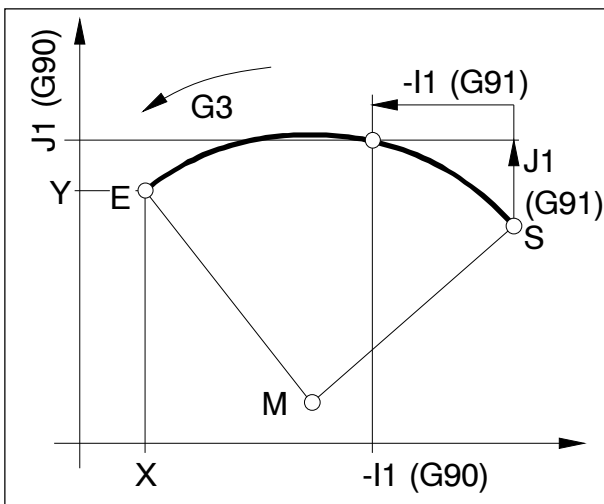
**Programming with polar coordinates**

G2/G3 AP=.. RP=..

AP= end point E polar angle,  
the pole is the circle centre point

RP= polar radius, at the same time circle radius

The pole of the polar coordinate system must be at the circle centre point (previous put on circle centre point with G111)



**Programming with start point, intermediate point, end point**

CIP X.. Y.. Z.. I1=.. J1=.. K1=..

X, Y, Z end point E in cartesian coordinates

I1, J1, K1 intermediate point Z in cartesian coordinates

**Start point**

The start point is the position of the tool at the time of the call-up of G2/G3.

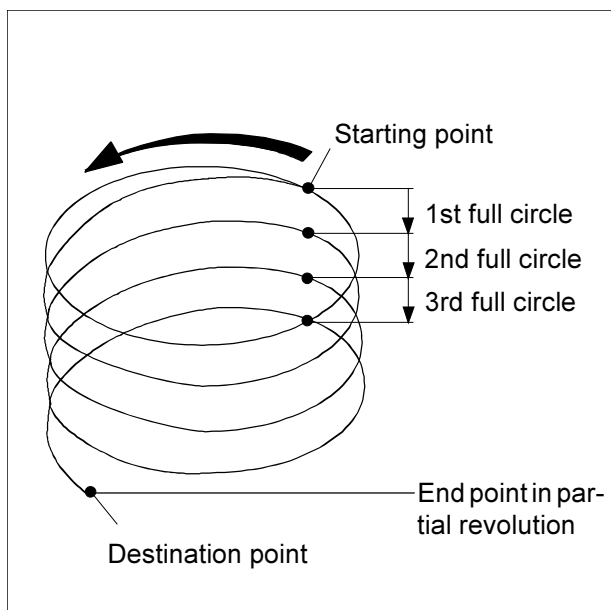
**End point**

The end point is programmed with X, Y, Z.

**Intermediate point**

The intermediate point is programmed with I1, J1, K1.

With G91 (incremental programming) the intermediate point is related to the start point.



### Helical interpolation

G2/G3 X... Y... Z... I... J... K... TURN=  
 G2/G3 X... Y... Z... CR=... TURN=  
 G2/G3 AR=... I... J... K... TURN=  
 G2/G3 AR=... X... Y... Z... TURN=  
 G2/G3 AP... RP=... TURN=

X, Y, Z	End point in Cartesian coordinates
I, J, K	Circle center point in Cartesian coordinates
CR=	Circle radius
AR=	Arc angle
AP=	Polar angle
RP=	Polar radius
TURN=	Number of additional circle passes within the range 0 to 999

For detailed informations of the interpolations parameter see "Circular interpolation".

## G4 Dwell time

### Format

N... G04 F... [sec]  
N... G04 S... [U]

F dwell time in seconds

S dwell time in number of main spindle revolutions

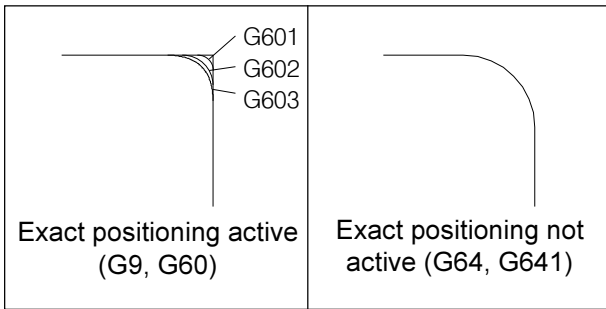
The tool will be stopped in the last reached position - sharp edges - transitions, clean drill / groove ground, exact positioning.

### Note

- The dwell time starts after the feed rate of the previous block has become 0.
- S and F are used as time values in the G4 block only. A previous programmed feedrate F or spindle speed S will be kept.

### Example

N75 G04 F2.5 (dwell time = 2,5 sec)



### G9, G60, G601, G602, G603 Exact positioning

- G9 Exact positioning, effective blockwise
- G60 Exact positioning, modal
- G601 Step enable if positioning window fine reached
- G602 Step enable if positioning window coarse reached
- G603 Step enable if end of interpolation reached

G601/G602/G603 are effective only with active G60 or G9.

The commands G64, G641 - Contouring mode deselect G60.

#### G9/G60:

Activation of G601, G602 or G603.

G9 is effective in the block only in which it was programmed, G60 is effective as long as it will be cancelled with G64 or G641.

#### G601, G602:

The following block will be started only after in the G9 / G60 the slides were stopped to standstill (short standstill delay at block end).

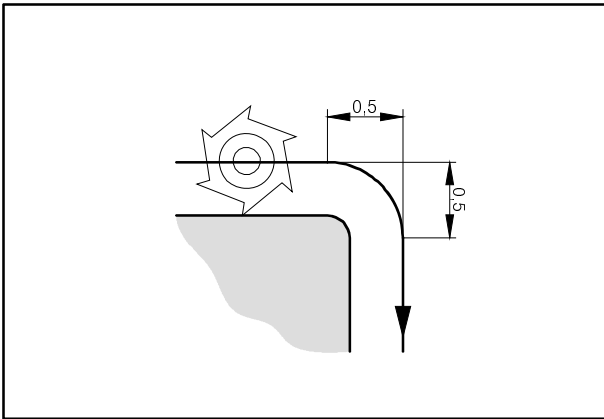
By that the corners will not be rounded and the transitions will become sharp.

The target position can be in a fine (G601) or coarse (G602) tolerance window.

#### G603:

The block change will be triggered in that moment when the control calculates the nominal feedrate 0 for the included axes (no standstill). At this point in time the actual value is back for a run after amount. By that the edge will be rounded.

With G603 the edge will be rounded most.



**G64, G641 Contouring mode**

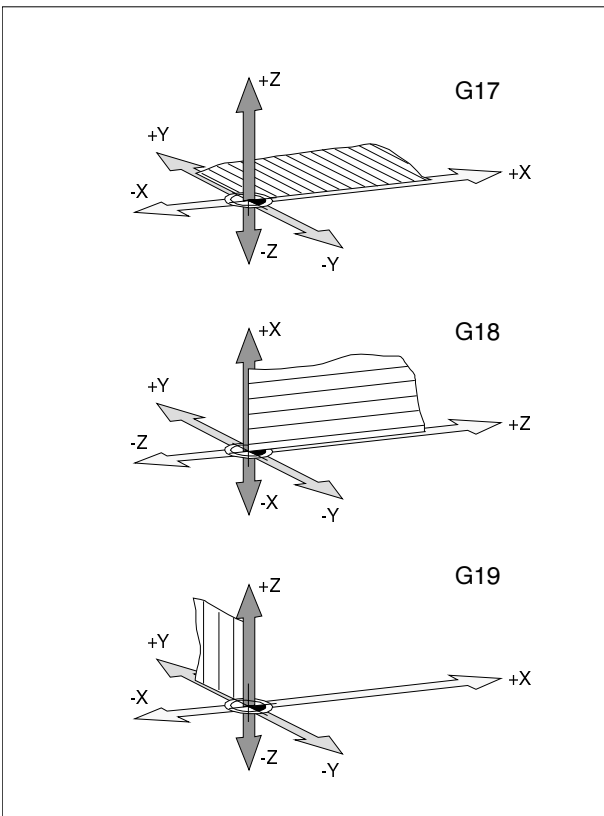
G64 Contouring mode  
 G641 Contouring mode with programmable rounding

The contour will be produced with path feed rate as constant as possible. This results in shorter machining time and rounded transitions.

At tangential transitions the tool traverses with a path feed rate as constant as possible, in corners the feed rate will be reduced correspondingly.

The larger the feed rate F the larger is rounding of the edges (contour error).

The command G641 allows to indicate the rounding distance.



**G17, G18, G19 Working plane selection**

**Format**  
 N... G17/G18/G19

G17 XY plane  
 G18 ZX plane  
 G19 YZ plane

G17-G19 selects the working plane.

- The tool axis is vertical to the working plane.
- Circle interpolation G2/G3/CIP occurs in the working plane.
- Polar coordinate interpolation occurs in the working plane.
- Cutter radius compensation G41/G42 occurs in the working plane.
- The infeed movements e.g. for drilling cycles are vertical to the working plane.



## G25, G26 Programmable working area limitation

### Format

N... G25/G26 X... Y... Z...

G25/G26 limitates the working area in which the tool can traverse.

By that in the working area, a safety area can be established which are locked for tool movements.

G25 and G26 must be programmed in a separate program block.

The programmable working area limitation will be defined in the program by G25 and G26, and switched on and off with WALIMON and WALIMOF.

G25	Lower working area limitation
G26	Upper working area limitation
WALIMON	Working area limitation on
WALIMOF	Working area limitation off

## G25, G26 Programmable spindle speed limitation

### Format

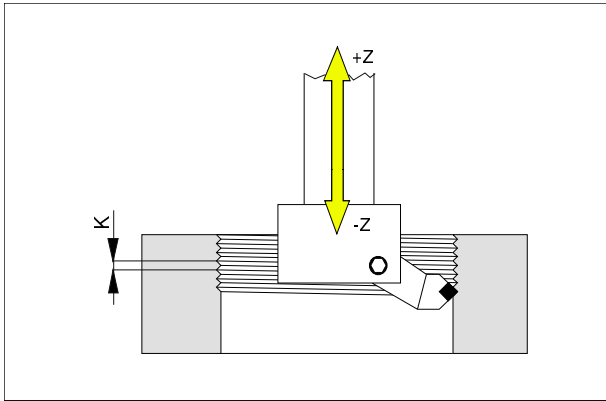
N... G25/G26 S...

A minimum and a maximum spindle spee can be defined with G25/G26 .

G25 and G26 must be programmed in a separate program block.

The programmable spindle speed limitation with G25/G26 overwrites the values in the setting data and therefore is kept also after program end.

G25	Lower spindle speed limitation
G26	Upper spindle speed limitation
S	Min / max spindle speed



### G33 Thread cutting

**Format**

N... G33 Z... K...

K ..... Thread pitch [mm]

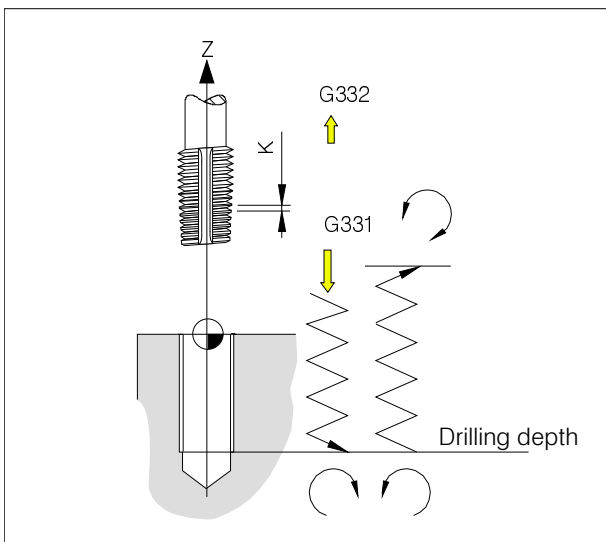
Z ..... Thread depth

Threads can be cut with an adequate tool (hollow-out tool or facing head) .

The pitch (K) corresponding to the main direction of the thread must be indicated.

**Note**

- Feed and spindle override are not active with G33 (100%).
- A fair sized undercut has to be machined first.



### G331/G332 Tapping without compensation chuck

(only for PC Mill 155)

**Format**

N... G331 X... Z... K...

N... G332 X... Z... K...

X, Z ..... Drilling depth (End point)

K ..... Thread lead

Drilling depth, Thread lead

Drilling in Z- direction, Thread lead K

**G331 Tapping:**

Tapping is described by the drilling depth (end point of the thread) and the lead.

**G332 Retraction movement:**

This movement is described with the same lead as the G331 movement. The reversal in the direction of the spindle is performed automatically.

SPOS have to be used to prepare the spindle for tapping.

## G63 Thread tapping with compensating chuck

### Format

G63 X.. Y.. Z.. F.. S..

Thread tapping without synchronisation.

Programmed spindle speed S, programmed feed rate F and pitch P of the tap must match:

$F \text{ [mm/min]} = S \text{ [U/min]} \times P \text{ [mm/U]}$  resp.

$F \text{ [mm/U]} = P \text{ [mm/U]}$

The dive-in movement of the tap is programmed with G63.

G63 is effective blockwise. While G63 the feed and spindle override are locked to 100%.

The retraction movement (with opposite spindle direction) also must be programmed with G63.

Example:

Tap M5 (pitch P = 0,8 mm)

spindle speed S = 200, therefore F = 160

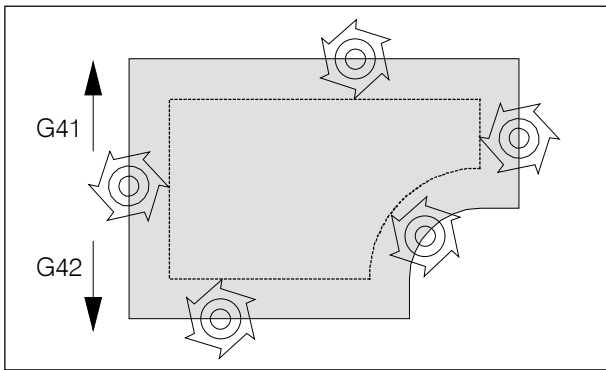
```
N10 G1 X0 Y0 S200 F1000 M3  
      (approach start point)
```

```
N20 G63 Z-50 F160  
      (thread tapping, thread depth 50)
```

```
N30 G63 Z3 M4  
      (retraction, change of spindle direction)
```

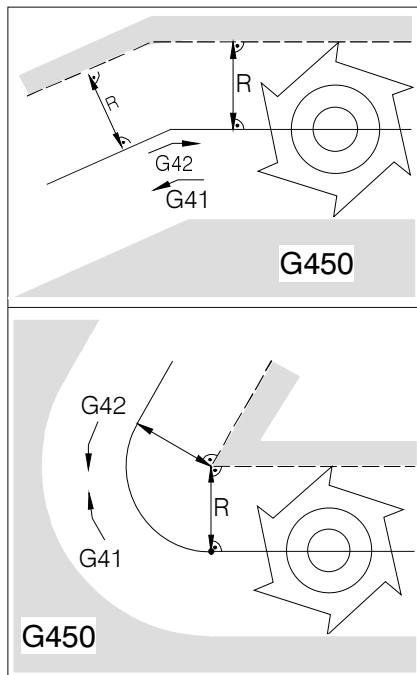
### Tool Radius Compensation G40-G42, G450

- G40** Tool radius compensation OFF
- G41** Tool radius compensation LEFT (climb milling)
- G42** Tool radius compensation RIGHT (upcut milling)



With G41/42 the tool traverses an equidistant path to the programmed contour. The path distance is the dimension of the cutter radius.

Look in feed direction to determine the correction side G41/42 (left / right from the contour).



#### Corner traverse characteristics G450

G450: The tool traverses around a workpiece corner on a arc with tool radius.

- — — Programmed tool path
- real tool path with correction

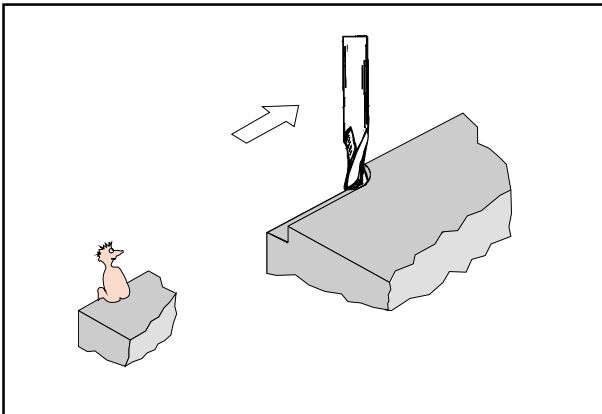
**G40 Cutter radius compensation OFF**

Cutter radius compensation will be deselected by G40.

Deselection is allowed only in combination with a straight movement (G00, G01) (Departing movement).

G40 can be programmed in the same block as G00 or G01 or in the previous block.

G40 is normally programmed with retraction to the tool change point.



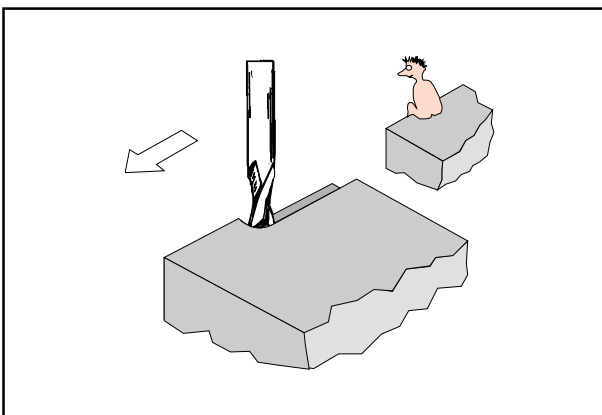
Definition G41 Cutter radius compensation left

**G41 Cutter radius compensation left**

When the tool is **left** of the contour to be machined (viewed in direction of feed), G41 must be programmed.

**Notes**

- Direct change between G41 - G42 is not allowed - previous deselection with G40.
- Selection must occur in combination with G0 or G1 (Approaching movement).
- No change of tool correction while active cutter radius compensation.



Definition G42 Cutter radius compensation right

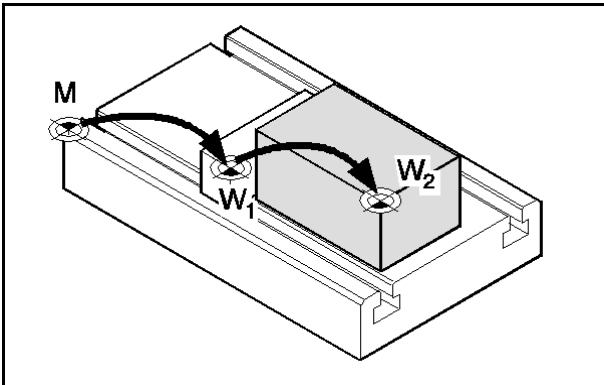
**G42 Cutter radius compensation right**

When the tool is **right** of the contour to be machined (viewed in direction of feed), G42 must be programmed.

Notes see G41!

### Zero offsets G53-G57, G500-G599, SUPA

- G53 The zero offsets will be suppressed for one block.
- G500 G54 - G599 are switched off.
- G54-57 Settable zero offsets.
- G505-599 Settable zero offsets.
- SUPA Blockwise suppressing, including programmed zero offsets and DRF offsets (**SUPpress All**)



Zero points are used to indicate the position of the workpiece on the machine.

Normally the coordinate system is shifted with G54-G599 to a stop point ( $W_1$ ) at the clamping device (fix stored), and the following shift to the workpiece zero point ( $W_2$ ) occurs with TRANS (variable).

### Inch dimensions G70, Metric dimensions G71

Depending on G70 / G71 the following values can be entered in inch or mm:

- Path informationen X, Y, Z
- Circle parameter I1, J1, K1, I, J, K, CR
- Thread pitch
- Programmable zero offsets TRANS, ATRANS
- Polar radius RP

All other values e.g. feed rates, tool corrections or settable zero offsets will be calculated in that unit that was present in the machine data.

The entire zero point offset effective in a workpiece program is the sum of basis zero point offset + adjustable zero point offsets + frames.

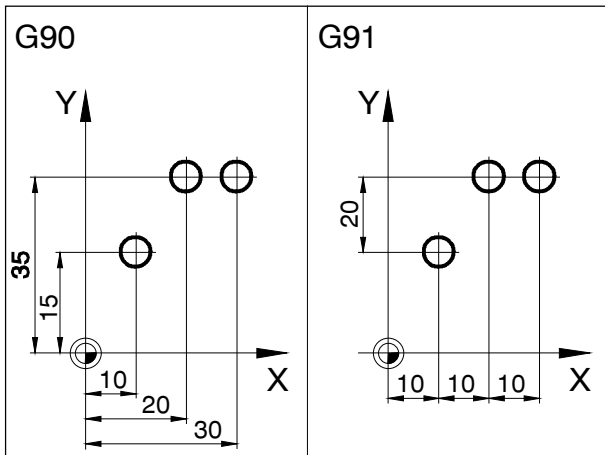


## Coordinaten, Zero Offset

### Working plane G17-G19

In the working plane the tool radius is effective, vertical to the working plane the tool length.

Main working plane for turning: G18 (ZX)



### G90 Absolute dimensions

The dimensions refer to the actual zero point. The tool traverses **ON** a programmed position.

### G91 Incremental dimensions

The dimensions refer to the last programmed tool position. The tool traverses **FOR** a distance to the next position.

Single axes can be programmed absolute or incremental without respect to G90 / G91.

Examples:

G90

G0 X40 Z=IC(20)

Here is the Z value incremental although G90 Absolute dimensions is active.

G91

G0 X20 Z=AC(10)

Here is the Z value absolute although G91 Incremental dimensions is active.

## Feed Programming G94, G95

### General

- Feed values are not influenced by G70/71 (inch-mm), only the machine setting data is valid.
- After every change G93-95 F must be programmed again.
- The feed rate F is valid for path axes only, but not for synchronous axes.

### Feed rate in mm/min G94

Slide movement X, Z:

The address F is the feed rate in mm/min.

Main application for milling.

### Vorschub in mm/U G95

Slide movement X, Z:

The address F is the feed rate in mm/rev of the milling spindle.

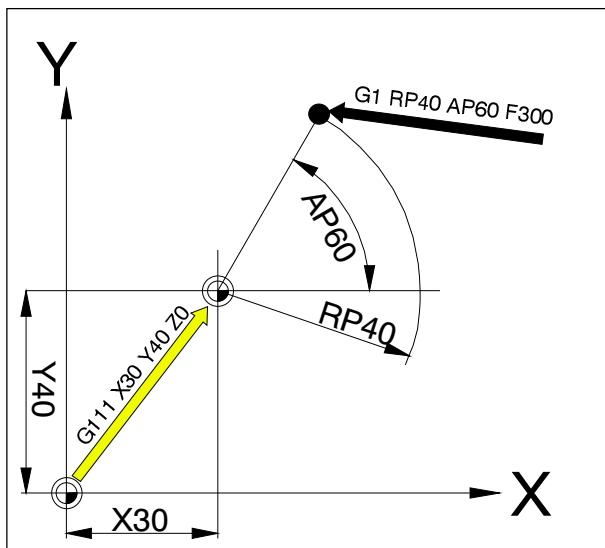
**Main application for turning.**



## Polar coordinates G110-G112

With polar coordinate programming the positions will be indicated as angle and radius, related to the pole (zero point of the polar coordinate system).

In NC blocks with polar end point data, for the selected working plane no cartesian coordinates must be programmed as e.g. interpolation parameter, axis addresses etc.



### Determination of the pole

G110 Pole position related to the last programmed tool position.

G111 Pole position, related to the actual zero point of the workpiece coordinate system.

G112 Pole position related to the last valid pole.

The pole can be indicated in cartesian or polar coordinates.

X,Z coordinates of the pole (cartesian)

RP polar radius (= distance pole - target)

AP polar angle between distance pole - target and angle reference axis (first programmed pole axis)

### Example

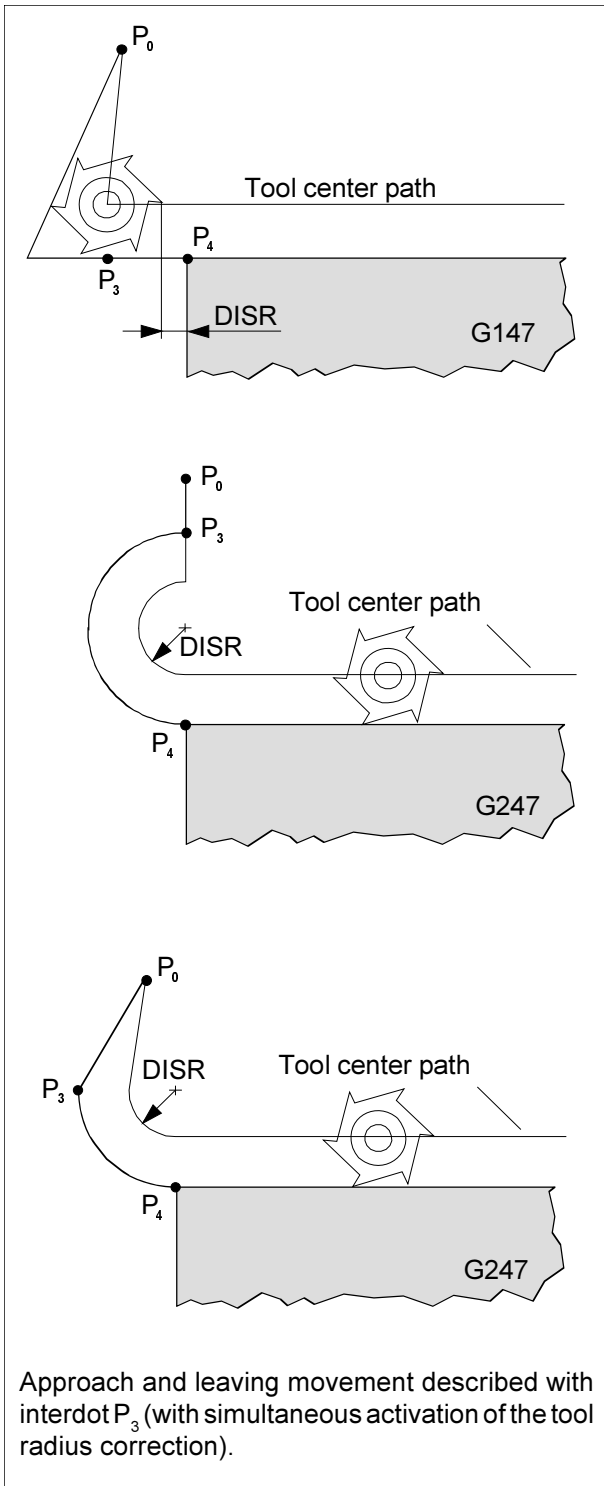
```
G111 Z30 X40
```

```
G1 RP=40 AP=60 F300
```

G111 puts the pole on the absolute position 30/40.

G1 moves the tool from its previous position to the polar coordinate RP40/AP60.

The angle relates to the Z axis, because Z was programmed first in the G111 block.



Traversing movements with G0/G1 have to be programmed prior to the soft departure and approach. The programming of G0/G1 is not possible in the same block.

### Soft approach and leaving G140 - G341, DISR, DISCL, FAD

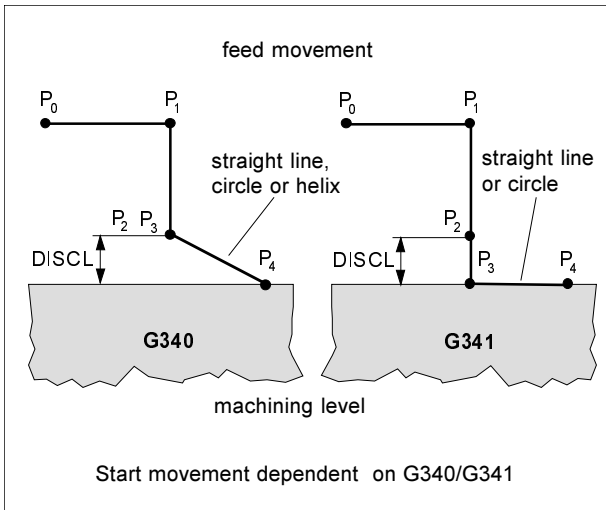
- G140 Soft approach and leaving
- G141 Approach from the left and/or leaving from the left
- G142 Approach from the right and/or leaving from the right
- G143 Direction of approach and/or leaving dependent on the relative position of start and/or end point to the tangent direction
- G147 Approach with a straight line
- G148 Leaving with a straight line
- G247 Approach with a quarter circle
- G248 Leaving with a quarter circle
- G340 Approach and leaving in space (starting position value)
- G341 Approach and leaving in the plane
- G347 Approach with a semicircle
- G348 Leaving with a semicircle
- G450 Approach and leave contour
- G451 Approach and leave contour
- DISR • Approach and leaving with straight line, distance of the milling cutter edge from starting point to contour
- Approach and leaving with circles. Radius of the tool centre path
- DISCL Distance of the end point of the rapid motion of the machining level
- DISCL=AC Indication of the absolute position of the end point of the rapid movement
- DISCL=0 G340:  $P_1, P_2, P_3$  coincide  
G341:  $P_2, P_3$  coincide
- FAD Indication of the feed speed  
G341: from  $P_2$  to  $P_3$   
G340: from  $P_2$  and/or  $P_3$  to  $P_4$
- FAD=PM Linear feed (like G94)
- FAD=PR Rotation feed (like G95)

The function soft approach and leaving serves to approach the starting point of a contour tangentially irrespective of the position of the initial point. The function is mainly used in connection with the tool radius correction, however, it is not obligatory.

The approach and leaving movement consists of a max. of 4 partial movements:

- starting point of the movement ( $P_0$ )
- interdots ( $P_1, P_2, P_3$ )
- end point ( $P_4$ )

The points  $P_0, P_3,$  and  $P_4$  are always defined. The interdots  $P_1$  and  $P_2$  can be omitted according to the machining conditions.



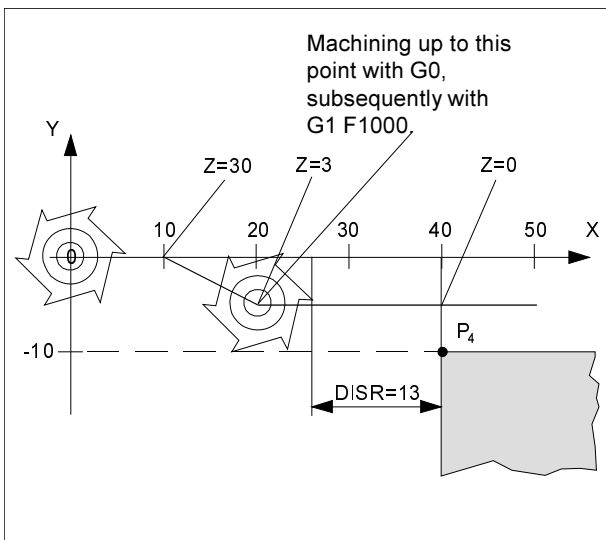
**Selection of the approach and/or leaving direction**  
 Determination of the approach and leaving direction by means of the tool radius correction

with positive tool radius:  
 G41 active - approach from the left  
 G42 active - approach from the right

**Division of the movement from the starting to the end point (G340 and G341)**

The characteristic approach of P<sub>0</sub> to P<sub>4</sub> is represented in the adjoining figure.

In those cases, in which the position of the active levels is programmed with G17 to G19, a possibly active rotating FRAME is taken into consideration.



```
$TC_DP1 [1,1]=120
$TC_DP6 [1,1]=7
```

```
N10 G90 G0 X0 Y0 Z30 D1 T1
N20 X10
N30 G41 G147 DISCL=3 DISR=13 Z=0 F1000
N40 G1 X40 Y-10
N50 G1 X50
```

...

N30/40 can be substituted by:

```
1.
N30 G41 G147 DISCL=3 DISR=13 X40 Y-10
ZO F1000
```

or

```
2.
N30 G41 G147 DISCL=3 DISR=13 F1000
N40 G1 X40 Y-10 ZO
```

## Collision Detection NORM, KONT

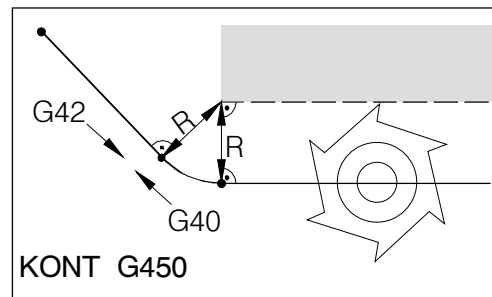
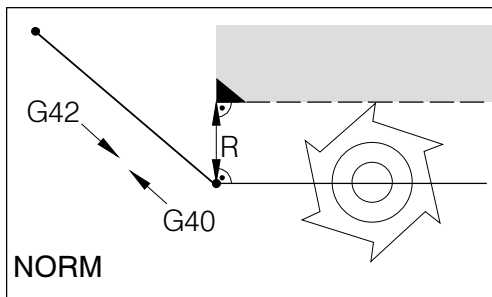
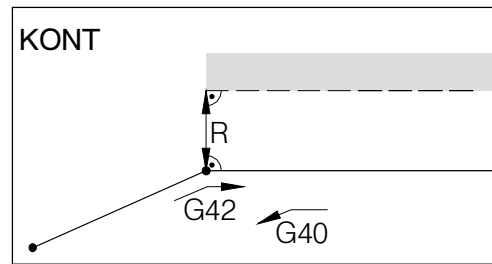
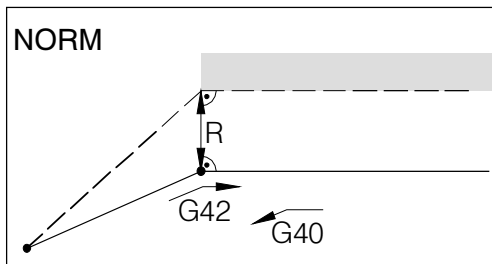
### Approach and leave contour NORM/KONT

**NORM:** The tool approaches straight and stands vertical to the start point.

If the start / end point is not on the same side of the contour as the first / last point of the contour a contour violation will happen.

**KONT:** The tool traverses around the contour point as programmed with G450/451.

— — — programmed tool path  
 ————— real tool path with correction



With approaching or leaving with NORM a contour violation (black) will happen, if the start or end point is behind the contour.

With approaching or leaving with KONT the tool traverses around the corner with an arc (G450) or a straight (G451).

## Cycle call

Cycle call occurs as following:

Cycle (Parameter 1, Parameter 2, ...)

In the survey pictures and in the cycle description you will find all needed parameter for the cycles.

In the call-up of the cycles the only parameter values will be entered (no parameter term).

Therefore the sequence of parameter must be kept, for that the values can not be misinterpreted.

When a parameter is not needed, on its place a additional comma must be given.

Example:

A drill hole should be made with CYCLE 81. A safety distance is not necessary (e.g. there is already a groove in the workpiece).

The hole should be 15 mm deep, related to the zero point.

CYCLE81 (5,0,,,-15)

CYCLE 81 . Drilling, centering

5 ..... Retraction plane 5 mm over zero point  
(= workpiece surface)

0 ..... Reference plane at zero level

,, ..... Here the safety distance would be programmed. For that the control knows that the next value is the drilling depth, an additional comma is set.

-15 ..... End depth absolute

) ..... The parameter DPR is not indicated. Since no further parameter follows, no additional comma is necessary.

### Notes for cycle description

The cycle description starts with a survey table containing the cycles and their parameter.

Following is the exact description of the parameter.

In the survey table the cycles are based on the previous cycle, that means only those parameter are described that are different to the previous cycle or new.

Example:

CYCLE 82 has the same parameter as CYCLE 81, only parameter 6 DTB is added.

CYCLE 83 has the first 5 parameter like CYCLE 81, the parameter 6 - 12 are added.

CYCLE 84 has the parameter 1-5 like CYCLE 81, the parameter 6 like CYCLE 82 and the parameter 7 - 12 are added, etc..



Cycles can also be called up by means of MCALL.  
(See "Modal subroutine MCALL")

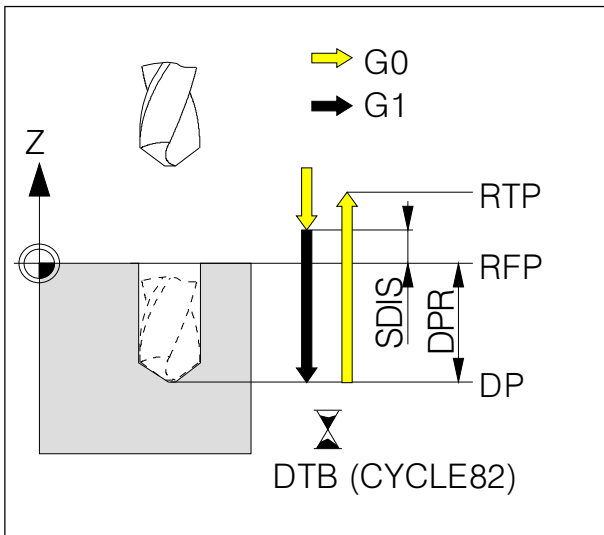
## Drilling Cycles

DRILLING CYCLES	
Cycle 81	Drilling, Centering
Cycle 82	Drilling, Counterboring
Cycle 83	Deep hole drilling
Cycle 84	Rigid tapping
Cycle 840	Tapping with compensation chuck
Cycle 85	Boring 1
Cycle 86	Boring 2
Cycle 87	Boring 3
Cycle 88	Boring 4
Cycle 89	Boring 5
Holes 1	Row of hole with MCALL
Holes 2	Circle of hole with MCALL
Cycle 801	Linear pattern

**CYCLE81 Drilling, Centering**  
**CYCLE82 Drilling, Counterboring**

CYCLE81 (RTP,RFP,SDIS,DP,DPR)  
 CYCLE82 (RTP,RFP,SDIS,DP,DPR,DTB)

- |  |   |                                    |
|--|---|------------------------------------|
| RTP  | retraction plane absolute   | <b>Re</b> Traction Plane           |
|  | After the cycle the tool is placed on this height. RTP must be higher than the reference plane.     |                                    |
| RFP  | reference plane absolute  | <b>Re</b> FeRence Plane            |
|  | Height of the workpiece surface, most time the workpiece reference point is on the surface (RFP=0)  |                                    |
| SDIS   | safety distance without sign  | <b>Sa</b> feTy <b>DIS</b> tance    |
|  | The tool traverses with rapid feed until SDIS over reference plane and then changes to working feed |                                    |
| DP   | end depth absolute  | <b>De</b> Pth                      |
|  | Depth of the hole, related to the workpiece reference point.  |                                    |
| DPR  | end depth relative to reference plane   | <b>De</b> Pth <b>Re</b> lative     |
|  | Depth of the hole, related to the reference plane, without sign.                                    |                                    |
| Either DP or DPR can be programmed. If nevertheless both parameters are programmed, DPR will be valid. |   |                                    |
| DTB  | dwll time at hole ground in [s]   | <b>Dwell</b> Time at <b>Bottom</b> |
|  | The tool retracts only after a dwell time to clean the hole ground (only CYCLE82).                  |                                    |

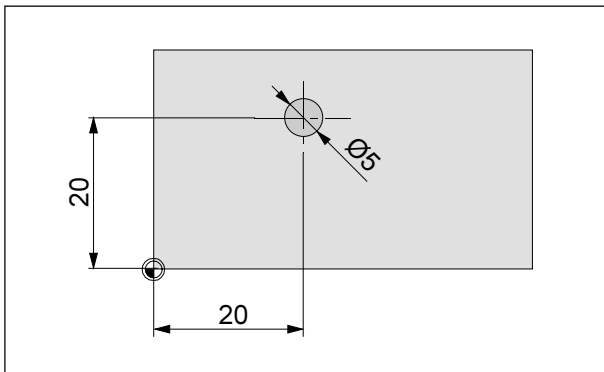


Previous to the cycle the tool must be placed over the hole position.

The tool drills with programmed feed to hole depth DP/DPR and retracts with rapid feed.

CYCLE82 allows a dwell time DTB at hole ground.

**Example Cycle81**



Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Final drilling depth absolute .....	-10
Depth incr. ....	0
Dwell time (only Cycle82) .....	0

```

G54
TRANS Z20
T1 D1 M6
S1500 M3 F120
G0 X20 Y20 Z2
Cycle81 (2, 0, 1, -10, 0)
G0 Z40
M30
    
```



## CYCLE83 Deep hole drilling

CYCLE83 (RTP,RFP,SDIS,DP,DPR,FDEP,FDPR,DAM,DTB,DTS,FRF,VARI,AXN,MDEP,VRT,DTD,DIS1)

Parameter additional to Cycle82:

FDEP	1st drilling depth absolute	<b>First DEPth</b>
FDPR	1st drilling depth relative	<b>First DePth Relative</b>
DAM	degression amount	<b>Degression AMount</b>
DTS	dwel time before infeed in [s]	<b>Dwell Time at infeed Start</b>
FRF	feed reduction factor for 1st infeed	<b>Feed Reduction Factor</b>
VARI	machining variant	<b>VARIant</b>
AXN(*)	Tool axis	<b>AXis</b>
MDEP(*)	Minimum drilling depth	<b>Min. drilling DEPth</b>
VRT(*)	Variable return path	<b>Variable ReTurn path</b>
DTD(*)	Dwell time at final drilling depth	<b>Dwell Time</b>
DIS1(*)	Lead distance	<b>DIStance</b>

Function:

The tool drills up to the requested final drilling depth at the programmed spindle speed and feed rate. The deep hole is machined by repeated incremental plunging depths - whose maximum value can be determined - until the total hole depth is reached. After every plunging depth the drill can be either retracted to the reference plane+safety distance for chip removal or it can be retracted by 1mm each time for chip breaking.

**FDEP**

Depth of the first drilling infeed related to the workpiece zero point.

**FDPR**

Depth of the first drilling infeed related to the reference plane, without sign.

**DAM**

Starting from the first drilling depth each of the following infeeds will be reduced each by the value DAM.

**DTB**

The dwell time at the final drilling depth (chip breaking) is programmed in seconds or revolutions of the main spindle.

DTB < 0 Entry in revolutions

DTB = 0 Entry in seconds

**DTS**

The tool is retracted after each infeed and will go forward again after the dwell tim DTS.

**FRF**

With this factor FRF the programmed feed can be reduced for the first infeed, possible input: 0,001 - 1.

**VARI**

VARI=0 - chip breaking

After each infeed the tool retracts for 1 mm to break the chips.

VARI=1 - chip removal

After each infeed the tool retracts out of the boring to the reference plane to remove the chips out of the boring.

**AXN**

Selection of the tool axis:

Z=3

**MDEP**

For drilling stroke calculations above a decrease factor, a minimum drilling depth can be determined. In case the calculated drilling stroke is below the minimum drilling depth, the remaining drilling depth is machined in strokes from the size of the minimum drilling depth.

**VRT**

Return path during chip breaking. With VRT=0 (parameter not programmed) the drill is retracted by 1 mm each.

**DTD**

The dwell time at the final drilling depth can be entered in seconds or in revolutions.

DTD > 0 Entry in seconds

DTD < 0 Entry in revolutions

DTD = 0 Dwell time as programmed under DTB

**DIS1**

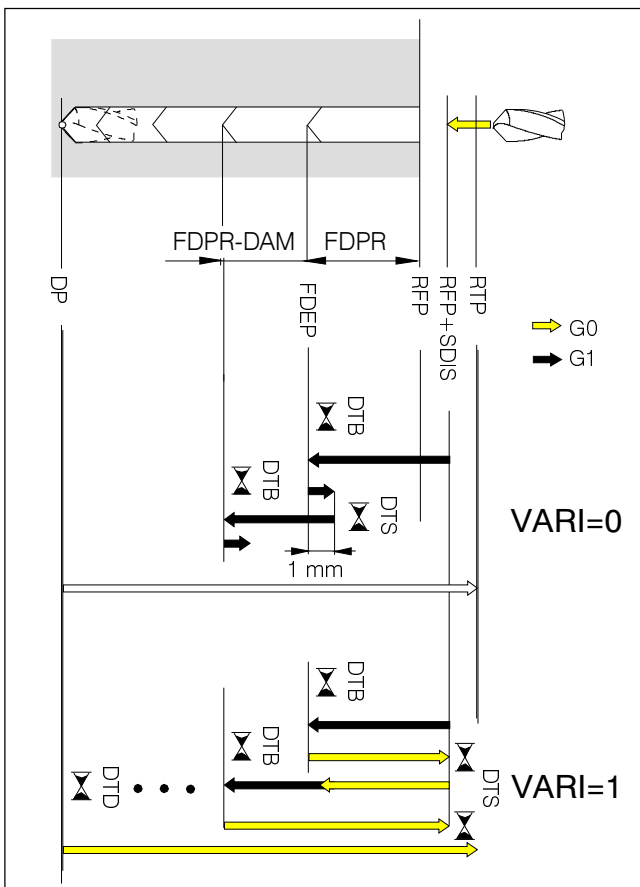
The lead distance after penetrating again into the bore can be programmed (for VARI=1).

DIS1 > 0 Positioning at the programmed value

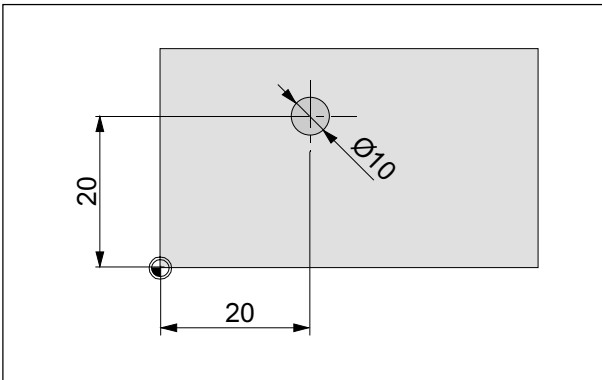
DIS1 = 0 Automatic calculation

Previous to the cycle the tool must be placed over the hole position (X=0).

The tool drills with programmed feed to the first drilling depth FDEP/FDPR, retracts in rapid, next infeed etc.. The infeed depth will be reduced each for DAM.



**Example Cycle83**



Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Final drilling depth absolute .....	-10
Depth incr. ....	0
First drilling depth absolute .....	-5
First drilling depth relative .....	0
Degression .....	5
Dwell time .....	0
Dwell time .....	0
Feedrate factor .....	1
Machining type .....	0
Tool axis .....	3
Minimum drilling depth .....	1
Variable return path .....	0
Dwell time at final drilling depth .....	0
Lead distance .....	0

```

G54
TRANS Z20
T2 D1 M6
S2000 M3 F150
G0 X20 Y20 Z2
Cycle83 (2, 0, 1, -10, 0, -5, 0, 5, 0, 0, 1, 0,3,1,0,0,0)
G0 Z40
M30
    
```

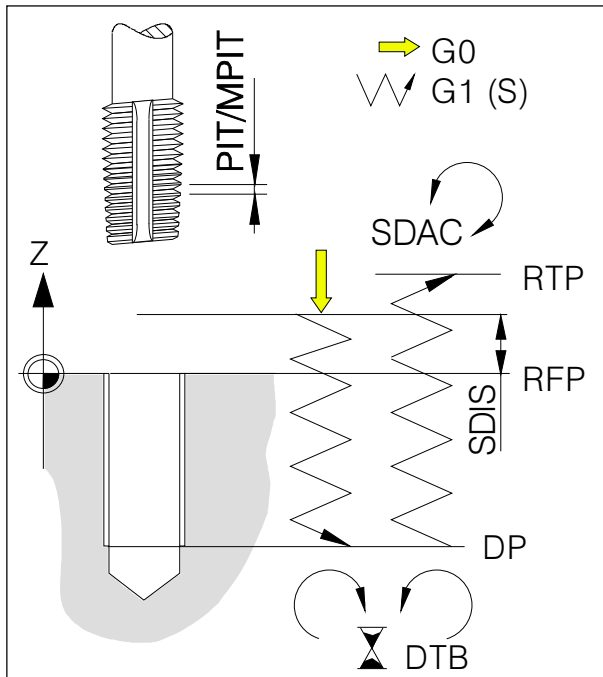
**CYCLE84 Rigid tapping**

CYCLE84 (RTP,RFP,SDIS,DP,DPR,DTB,SDAC,MPIT,PIT,POSS,  
SST,SST1,AXN,PTAB,TECH,VARI,DAM,VRT)

Parameter additional to Cycle81:

SDAC	spindle direction after cycle end 3: right, 4: left, 5: spindle stop	<b>Spindle Direction After Cycle</b>
MPIT	thread pitch as nominal value Thread pitch for regular metric thread, value range 3 (M3) - 48 (M48).	<b>Metrical PITch</b>
PIT	thread pitch in mm Value range 0,001 - 2000 mm. (with sign for the rotation direction) Program either MPIT or PIT. Contradictious values trigger an alarm.	<b>PITch</b>
POSS	spindle position for exact stop Before the cycle the spindle will be positioned with POSS.	<b>POSITION of Spindle</b>
SST	spindle speed for tapping	<b>Spindle Speed for Tapping</b>
SST1	spindle speed for retraction	<b>Spindle Speed for Tapping</b>
AXN(*)	Tool axis Selection of the tool axis: Z=3	
PTAB(*)	Evaluation of the threading pitch PIT 0: correspondingly programmed measuring system inch/metric 1: pitch in mm 2: pitch in threads per inch 3: pitch in inch/revolution	
TECH(*)	Technological settings UNITS DIGIT: exact stop 0: programmed as before a cycle call 1: (G601) 2: (G602) 3: (G603) TENS DIGIT: pilot control 0: programmed as before a cycle call 1: with pilot control (FFWON) 2: without pilot control (FFWOFF) HUNDREDS DIGIT: acceleration 0: programmed as before a cycle call 1: jerk-limited acceleration of the axes 2: rapid acceleration of the axes 3: reduced acceleration of the axes THOUSANDS PLACE: 0: activate spindle operation again (for MCALL) 1: remain in position-controlled operation (for MCALL)	

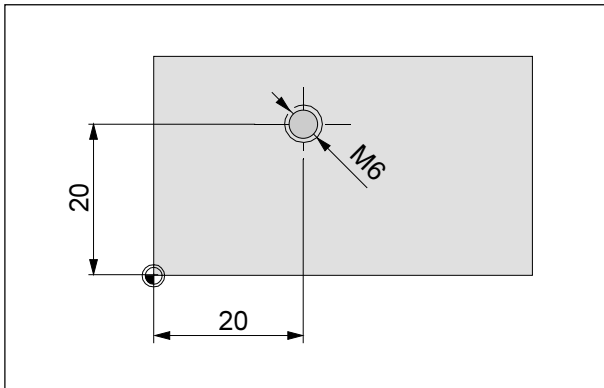
- VARI(\*) Machining mode  
 0: continuous tapping  
 1: deep-hole tapping with chip breaking  
 2: deep-hole tapping with chip removal
- DAM(\*) Incremental drilling depth  
 without arithmetic sign
- VRT(\*) Variable retraction path for chip breaking  
 without arithmetic sign



#### Machining sequence:

- Previous to the cycle the tool must be placed over the hole position.
- Rapid traverse to safety distance.
- Oriented spindle stop POSS.
- Tapping to end depth DP with spindle speed SST, spindle rotation and feed are synchronized.
- Dwell time at end depth.
- Spindle direction change.
- Retraction to safety distance with spindle speed SST1.
- Rapid retraction to retraction plane RTP.
- Establishing the spindle direction SDAC.

**Example Cycle84**



Return plane absolute .....	5
Reference plane absolute .....	0
Safety distance .....	2
Final drilling depth absolute .....	-15
Depth inc. relative .....	0
Dwell time .....	0
Direction of rotation after end of cycle .....	3
Thread lead as thread size .....	0
Thread lead value .....	1
Spindle position .....	0
Speed .....	500
Speed for retract .....	800
Tool axis .....	3
Evaluation of thread pitch .....	0
Technological settings .....	0
Machining mode .....	0
Incremental drilling depth .....	0
Variable return path .....	0

This cycle only works for machines that have a spindle with position control.

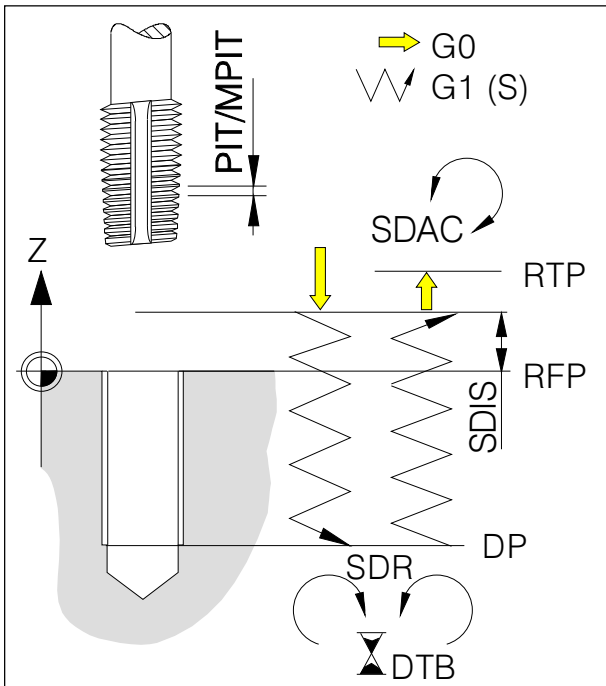
```
G54
TRANS Z20
T4 D1 M6
G0 X20 Y20 Z5
Cycle84 (5, 0, 2, -15, 0, 0, 3, 0, 1, 0, 500, 800,
3,0,0,0,0,0)
G0 Z40
M30
```

**CYCLE840 Tapping with compensation chuck**

CYCLE840 (RTP,RFP,SDIS,DP,DPR,DTB,SDR,SDAC,ENC,  
MPIT,PIT,AXN,PTAB,TECH)

Parameter additional to Cycle81:

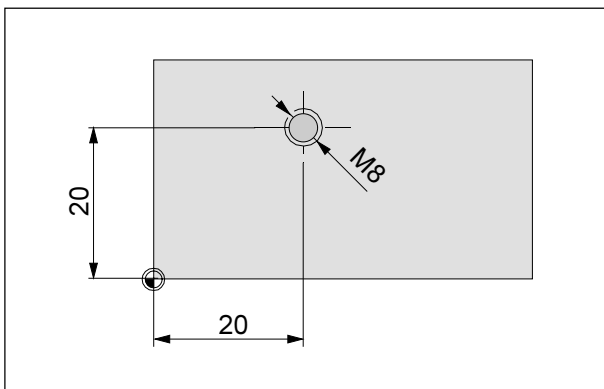
SDR	spindle direction for retraction 0: automatic change, 3: right, 4: left	<b>Spindle Direction for Retraction</b>
SDAC	spindle direction after cycle end 3: right, 4: left, 5: spindle stop	<b>Spindle Direction After Cycle</b>
ENC	use encoder 0: use encoder, 1: do not use encoder, on machines without encoder this parameter will be ignored	<b>ENCoder</b>
MPIT	thread pitch as nominal value Thread pitch for regular metric thread, value range 3 (M3) - 48 (M48).	<b>Metrical PITch</b>
PIT	thread pitch in mm Value range 0,001 - 2000 mm. (with sign for the rotation direction) Program either MPIT or PIT. Contradictious values trigger an alarm.	<b>PITch</b>
AXN(*)	Tool axis Selection of the tool axis: Z=3	
PTAB(*)	Evaluation of the threading pitch PIT 0: correspondingly programmed measuring system inch/metric 1: pitch in mm 2: pitch in threads per inch 3: pitch in inch/revolution	
TECH(*)	Technological settings UNITS DIGIT: exact stop 0: programmed as before a cycle call 1: (G601) 2: (G602) 3: (G603) TENS DIGIT: pilot control 0: programmed as before a cycle call 1: with pilot control (FFWON) 2: without pilot control (FFWOFF) HUNDREDS DIGIT: brake actuation point 0: without calculation 1: with calculation	



**Machining sequence:**

- Previous to the cycle the tool must be placed over the hole position.
- Rapid traverse to safety distance.
- Tapping to end depth DP with programmed spindle speed.
- Dwell time at end depth.
- Change of spindle direction according SDR.
- Retraction to safety distance
- Rapid retraction to retraction plane RTP.
- Establishing the spindle direction SDAC.

**Example Cycle840**



Return plane absolute .....	5
Reference plane absolute .....	0
Safety distance .....	2
Final drilling depth absolute .....	-15
Depth incr. relative .....	0
Dwell time .....	0
Direction of rotation for retraction .....	4
Direction of rotation after end of cycle .....	3
Machining type .....	0
Thread lead as thread size .....	0
Thread lead as value .....	1,25
Tool axis .....	3
Evaluation of thread pitch .....	0
Technological settings .....	0

```

G54
TRANS Z20
T4 D1
S600 M3
G0 X20 Y20 Z2
Cycle840 (5, 0, 2, -15, 0, 0, 4, 3, 0, 0, 1.25,3,0,0)
G0 Z40
M30
    
```



**CYCLE85 Boring 1, CYCLE89 Boring 5**

CYCLE85 (RTP,RFP,SDIS,DP,DPR,DTB,FFR,RFF)

CYCLE89 (RTP,RFP,SDIS,DP,DPR,DTB)

Boring 1 and 5 work similar CYCLE82.

Differences to CYCLE82:

- The infeed feed rate is not the last programmed F value but will be programmed with parameter FFR in the cycle call.
- The retraction feed is not rapid feed but will be programmed with parameter RFF in the cycle call.

FFR infeed feed rate

RFF retraction feed rate

**Forward Feed Rate****Retraction Feed**

### CYCLE86 Boring 2

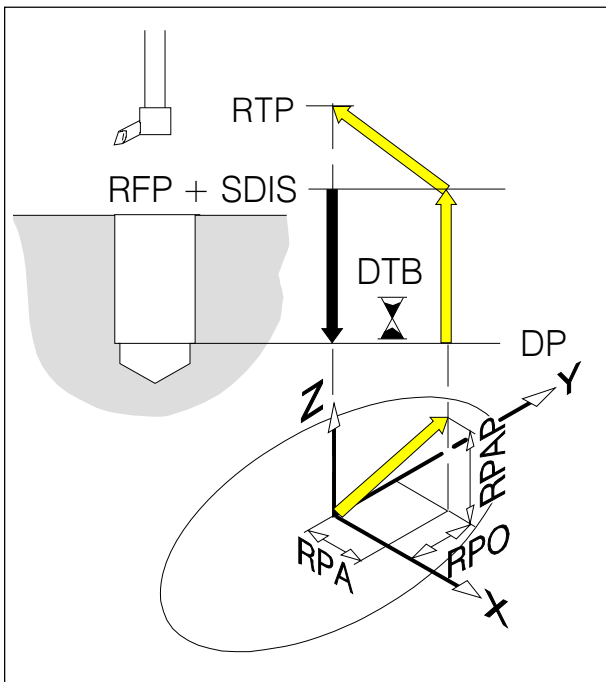
CYCLE86 (RTP,RFP,SDIS,DP,DPR,DTB,SDIR,RPA,RPO,RPAP,POSS)

Boring 2 works similar CYCLE82.  
Only a boring head tool is allowed.

Differences to CYCLE82:

- The rotational direction is programmed in the cycle with SDIR.
- At the boring ground is an oriented spindle stop (POSS) and the boring head can be lifted from the surface with RPA, RPO, RPAP in X/Y/Z to avoid scratching the surface while retraction.

SDIR	spindle direction 3: right, 4: left	<b>Spindle DIR</b> ection
RPA	lift off movement in X incremental with sign	<b>Retraction Position Abscissa</b>
RPO	lift off movement in Y incremental with sign	<b>Retraction Position Ordinate</b>
RPAP	lift off movement in Z incremental with sign	<b>Retraction Position AP</b> plicate
POSS	spindle position for exact stop	<b>POS</b> ition of <b>Spindle</b>



The lift off movement must occur in opposite direction to the boring head cutter edge.

The lift off path must be smaller than the outstanding length of the cutter edge out of the boring head.

**CYCLE87 Boring 3**

CYCLE87 (RTP,RFP,SDIS,DP,DPR,SDIR)

CAUTION: Boring with program stop M0 at hole ground,  
Retraction is done after pressing NC Start without spindle rotation.

**CYCLE88 Boring 4**

CYCLE87 (RTP,RFP,SDIS,DP,DPR,DTB,SDIR)

CAUTION: Boring with dwell time and program stop M0 at hole ground,  
Retraction is done after pressing NC Start without spindle rotation.

### Row of hole HOLES1, Circle of hole HOLES2

With these functions the single drilling position will be approached in rapid one-by-one, and at each drilling position the drilling cycle will be executed that was previous called as modal cycle.

The sequence of drill positions will be approached path-optimized.

#### Programming

- |   |                          |
|---|--------------------------|
| 1. Modal call of the requested drilling cycle | N60 MCALL CYCLE81 (....) |
| 2. Program drilling pattern                   | N65 HOLES1(....)         |
| 3. Deselect modal cycle call                  | N70 MCALL                |

#### Row of holes HOLES1

HOLES1 (SPCA,SPCO,STA1,FDIS,DBH,NUM)

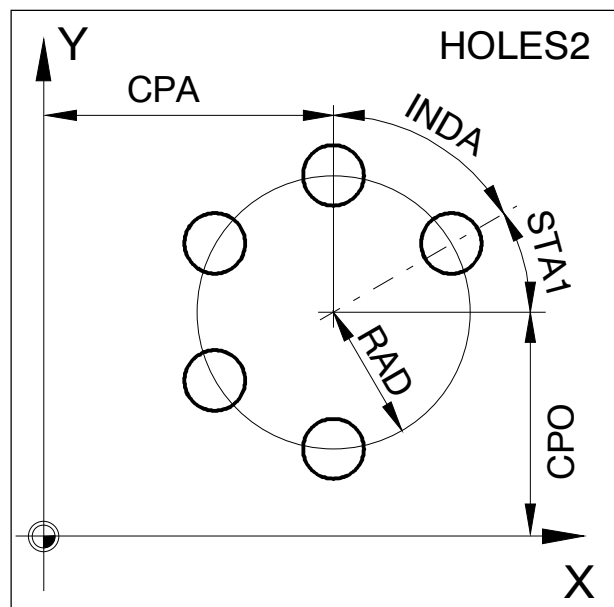
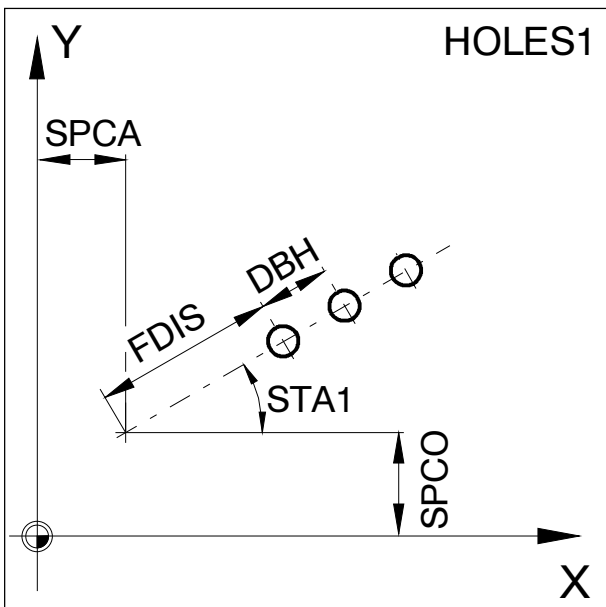
SPCA	start point in X	<b>Start Point of Cycle Abscissa</b>
SPCO	start point in Y	<b>Start Point of Cycle Ordinate</b>
STA1	angle of the row of holes related to X	
FDIS	distance start point - first hole incremental without sign	<b>First DISTance</b>
DBH	distance between holes incremental without sign	<b>Distance Between Holes</b>
NUM	number of holes	<b>NUMBER</b>

#### Circle of holes HOLES2

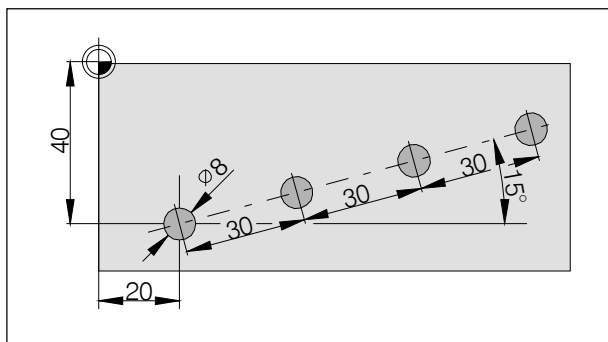
HOLES2 (CPA,CPO,RAD,STA1,INDA,NUM)

CPA	centre point in X	<b>Centre Point Abscissa</b>
CPO	centre point in Y	<b>Centre Point Ordinate</b>
RAD	radius of the hole circle	<b>RADius</b>
STA1	starting angle related to X	
INDA	indexing angle	<b>INDexing Angle</b>
NUM	number of holes	<b>NUMBER</b>

When INDA=0, the holes will be divided even on the circle of holes



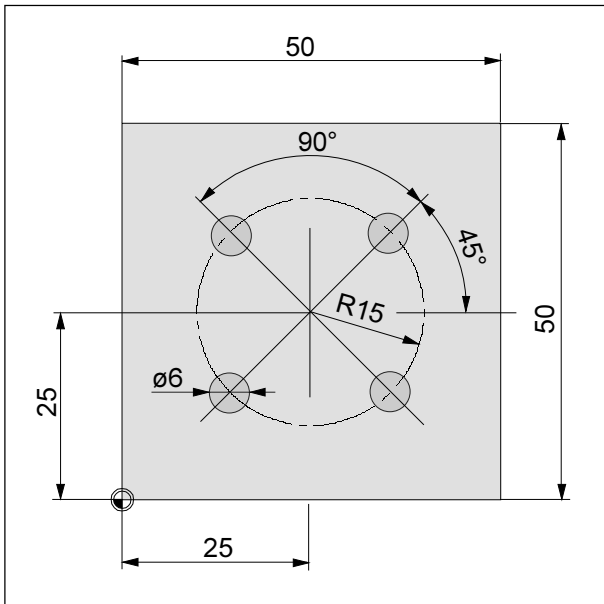
**Example Holes1**



Reference point, abscissa .....	20
Reference point ordinate .....	-40
Angle to abscissa .....	15°
Distance of first hole from reference point .....	0
Distance between holes .....	30
Number of holes .....	4

```

G54
TRANS Z40
T1 D1 M6
S1000 M3 F200
G0 X0 Y0 Z50
M8
MCALL Cycle81 (10, 0, 2, -12, 0)
Holes1 (20, -40, 15, 0, 30, 4)
MCALL
M9 G0 Z50
M30
    
```



**Example Holes2**

Center point abszissa .....	25
Center point ordinate .....	25
Radius of hole circle .....	15
Initial angle .....	45°
Indexing angle .....	90°
Number of holes .....	4

```

G54
TRANS Z20
T1 D1 M6
S1200 M3 F140
G0 X25 Y25 Z10
MCALL Cycle83(2, 0, 1, -10, 0, -5, 0, 5, 0, 0, 1, 0, 3,
2, 0, 0, 0)
Holes2(25, 25, 15, 45, 90, 4)
MCALL
G0 Z50
M30
    
```

### Linear pattern CYCLE801(\*)

This cycle enables to machine hole patterns. The bore type is determined by the previous modally selected drilling cycle.

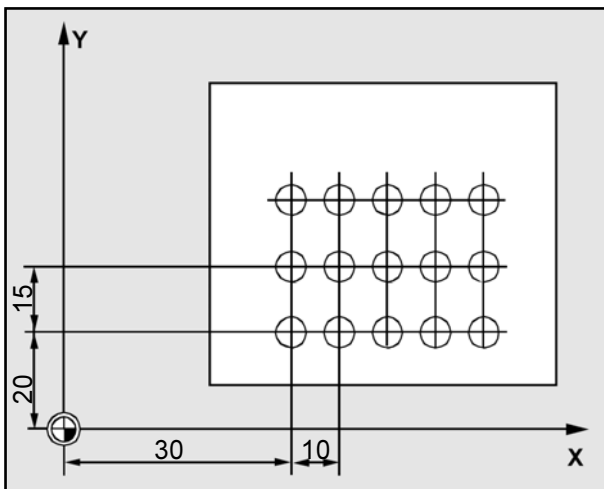
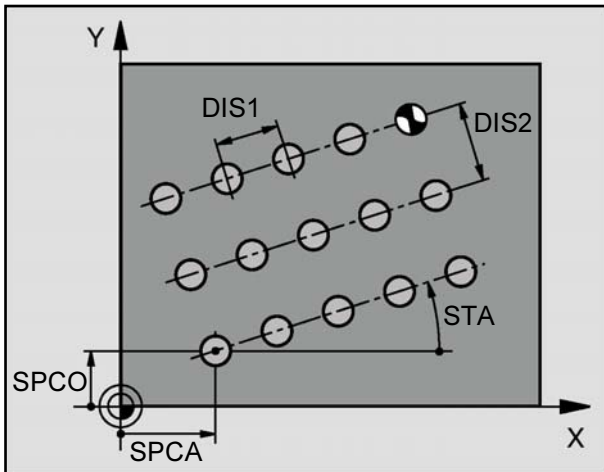
#### Programming

1. Call the requested drilling cycle modally     N60 MCALL CYCLE81 (....)
2. Program the hole pattern                     N65 CYCLE801(....)
3. Deselect the modal call                     N70 MCALL

CYCLE801 (SPCA,SPCO,STA1,DIS1,DIS2,NUM1,NUM2)

Additional parameters to HOLES1:

- STA1   Angle towards X-axis
- DIS1   Distance between the columns (without arithmetic sign)
- DIS2   Distance between the lines (without arithmetic sign)
- NUM1   Number of columns
- NUM2   Number of lines



#### Example Cycle801

Reference point absciss (in X) .....	30
Reference point ordinate (in Y) .....	20
Angle of hole pattern .....	0°
Distance between columns .....	10
Distance between lines .....	15
Number of columns .....	5
Number of lines .....	3

```

G54
TRANS Z20
T1 D1 M6
S1200 M3 F140
G0 X0 Y0 Z50
M8
MCALL Cycle81(10, 0, 2, -12, 0)
Cycle801(30, 20, 0, 10, 15, 5, 3)
MCALL
M9 G0 Z50
M30
    
```





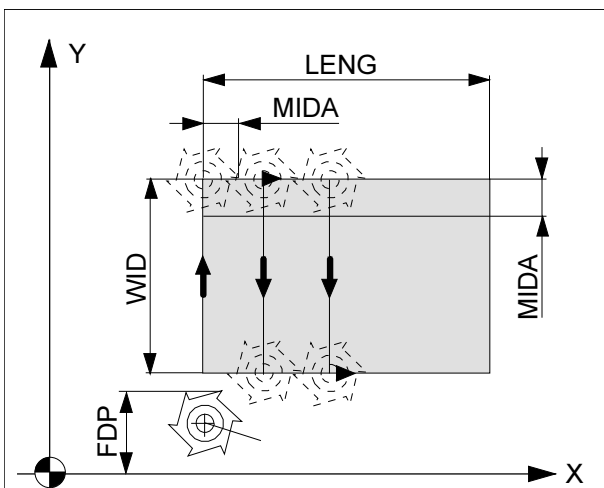
## Milling Cycles

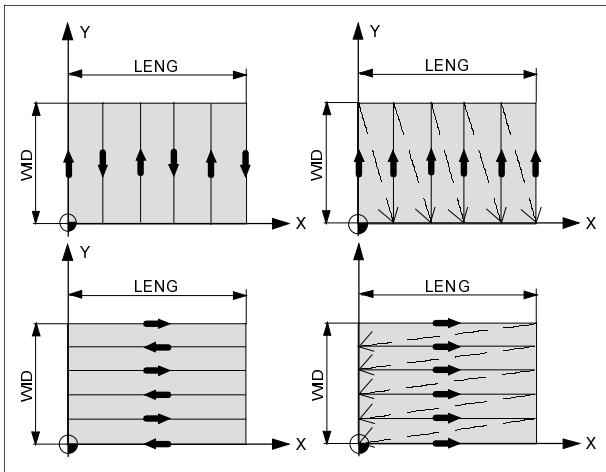
MILLING CYCLES	
Cycle 71	Face milling
Cycle 72	Contour milling
Cycle 90	Thread cutting
Longhole	Longholes on a circle
Slot 1	Slots on a circle
Slot 2	Circular slot
Pocket 1	Rectangular pocket
Pocket 2	Circular pocket
Pocket 3	Rectangular pocket
Pocket 4	Circular pocket
Cycle 76	Rectangular stud milling
Cycle 77	Circular stud milling

### CYCLE71 Face milling

CYCLE71 (RTP,RFP,SDIS,DP,PA,PO,LENG,WID,STA,MID,MIDA,  
FDP,FALD,FFP1,VARI)

- |      |  |  |
|------|--|--|
| RTP  | Retraction plane (absolute)  | <b>Re</b> Traction Plane   |
|      | After the cycle the tool stands at this height. RTP must be higher than the reference level.                     |  |
| RFP  | Reference plane (absolute)   | <b>Re</b> FeRence Plane  |
|      | Height of the workpiece surface, in most cases the workpiece zero point lies on the surface (RFP=0)              |  |
| SDIS | Safety clearance without sign  | <b>Sa</b> feTy <b>DIS</b> tance  |
|      | The tool moves in rapid motion up to the height SDIS across the reference level and then changes to working feed |  |
| DP   | Depth (absolute)   | <b>De</b> Pth  |
| PA   | Starting point, abscissa (absolute)  | <b>Point Ab</b> cissa  |
| PO   | Starting point, ordinate (absolute)  | <b>Point Or</b> dinate   |
| LENG | Rectangle lenght along the 1st axis  | <b>LENG</b> th   |
|      | The angle from which dimensioning is carried out results from the sign.  |  |
| WID  | Rectangle lenght along the 2nd axis  | <b>WID</b> th  |
|      | The angle from which dimensioning is carried out results from the sign.  |  |
| STA  | Angle between the longitudinal axis of the rectangle and the 1st axis of the plane (enter abscissa without sign) |  |
|      | Value range: $0^\circ \leq \text{STA} < 180^\circ$   |  |
| MID  | Maximum infeed depth (enter without sign)  | <b>Max</b> imal Infeed <b>De</b> pth   |
| MIDA | Maximum infeed width value for clearance in the plane (enter without sign)                                       | <b>Max</b> imal Infeed <b>De</b> pth <b>A</b>  |
| FDP  | Retraction travel in the plane (incremental, enter without sign)   |  |
| FALD | Final machining allowance in depth (incremental, enter without sign)   |  |
|      | In machining mode smoothing FALD stands for the residual material on the area.                                   |  |
|      |  | <b>Fin</b> ishing <b>A</b> llowance <b>De</b> pth  |
| FFP1 | Feedrate for surface machining   | <b>Fe</b> ed <b>F</b> or <b>P</b> lane   |
| VARI | Type of machining  | <b>VARI</b> ant  |
|      | UNITS DIGIT  |  |
|      | Value:   | 1... Clear up to final machining allowance<br>2... Finishing   |
|      | TENS DIGIT   |  |
|      | Value:   | 1... parallel to the abscissa, in one direction<br>2... parallel to the ordinate, in one direction<br>3... parallel to the abscissa, With changing direction<br>4... parallel to the ordinate, with changing direction |





**Function:**

With cycle CYCLE71, you can face mill any rectangular surface. The cycle differentiates between roughing (machining the surface in several steps until the finishing allowance) and finishing (end milling the surface in one step). Maximum infeed can be defined in width and depth.

The cycle does not include any cutter compensation. Depth infeed is performed in the open.

**Example Cycle 71**

Return plane absolute .....	10
Reference plane absolute .....	0
Safety distance .....	2
Depth absolut .....	-6
Initial point abszissa .....	0
Initial point ordinate .....	0
Lenght of rectangle, abszissa .....	60
Lenght of rectangle, ordinate .....	40
Angle between longitudinal axis and abszissa ...	10
Max. infeed depth for one infeed .....	2
Maximal infeed width, incr. ....	10
Retraction path in the plane .....	5
Final machining .....	0
Feedrate for surface machining .....	400
Machining type .....	31
Overrun path in direction of plane infeed .....	2

```

G54
TRANS Z20
T6 D1 M6
S2500 M3 F400
G0 X0 Y0 Z20
CYCLE           71           (10,0,2,-
6,0,0,60,40,10,2,10,5,0,400,31,2)
G0 Z40
M30
    
```

### CYCLE72 Contour milling

CYCLE72 (KNAME, RTP, RFP, SDIS, DP, MID, FAL, FALD, FFP1, FFD, VARI, RL, AS1, LP1, FF3, AS2, LP2)

Parameter additional to Cycle81:

- KNAME** Name of contour subroutine  
The contour to be milled is completely programmed in a subroutine. With the parameter KNAME you can define the name of the contour subroutine.
- FAL** Final machining allowance at the edge contour (enter without sign)
- FFD** Feedrate for depth infeed (enter without sign)
- RL** Following the contour clockwise or counterclockwise with G40, G41 or G42  
Value: 40 .... G40 (approach and leaving only linear)  
41 .... G41  
42 .... G42
- LP1/LP2** Length, radius  
LP1: Length of approach travel (along a straight line) or radius of the mill centre path of the arc approach (along a circle)  
LP2: Length of return travel (along a straight line) or radius of the mill centre path of the arc approach (along a circle)  
This values must be programmed >0.

**FF3** Return feedrate and feedrate for intermediate positioning in the plane  
If no feedrate is programmed, the intermediate motions are carried out with surface feedforward gor G01.

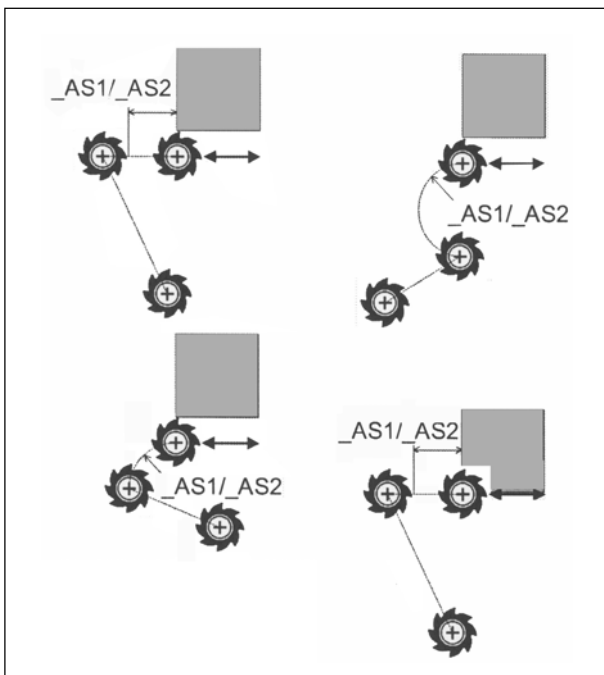
**AS1/AS2** Specification of approach/retraction travel  
AS1: approaching travel; AS2: retraction travel  
If AS2 is not programmed, the behaviour of retraction is the same os that of the approach.

**UNITS DIGIT:**

- Value: 1 .... Straight tangential line
- 2 .... Semi- circle
- 3 .... Quadrant

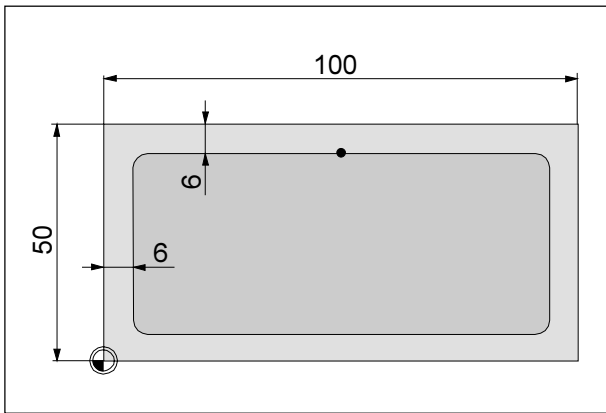
**TENS UNIT:**

- Value: 0 Approach to the contour in the plane
- 1 Approach to the contour of a physical path



AS1/AS2 approach/retractin travel

**Example Cycle 72**



Name .....	Kontur1
Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Depth absolute .....	-4
Infeed depth .....	4
Final machining .....	0
Final machining .....	0
Feedrate surface .....	250
Feedrate depth .....	100
Machining type .....	11
Machining type .....	41
Approach path .....	2
Length, radius .....	5
Retraction feed .....	0
Retract path .....	2
Length, radius .....	5

```

G54
TRANS Z20
T1 D1 M6 (Fräser Ø16)
S2500 F400 M3
Cycle72 ("Kontur1",2,0,1,-
4,4,0,0,250,100,11,41,2,5,0,2,5)
G0 Z40
M30
    
```

Subprogramm "Kontur1"

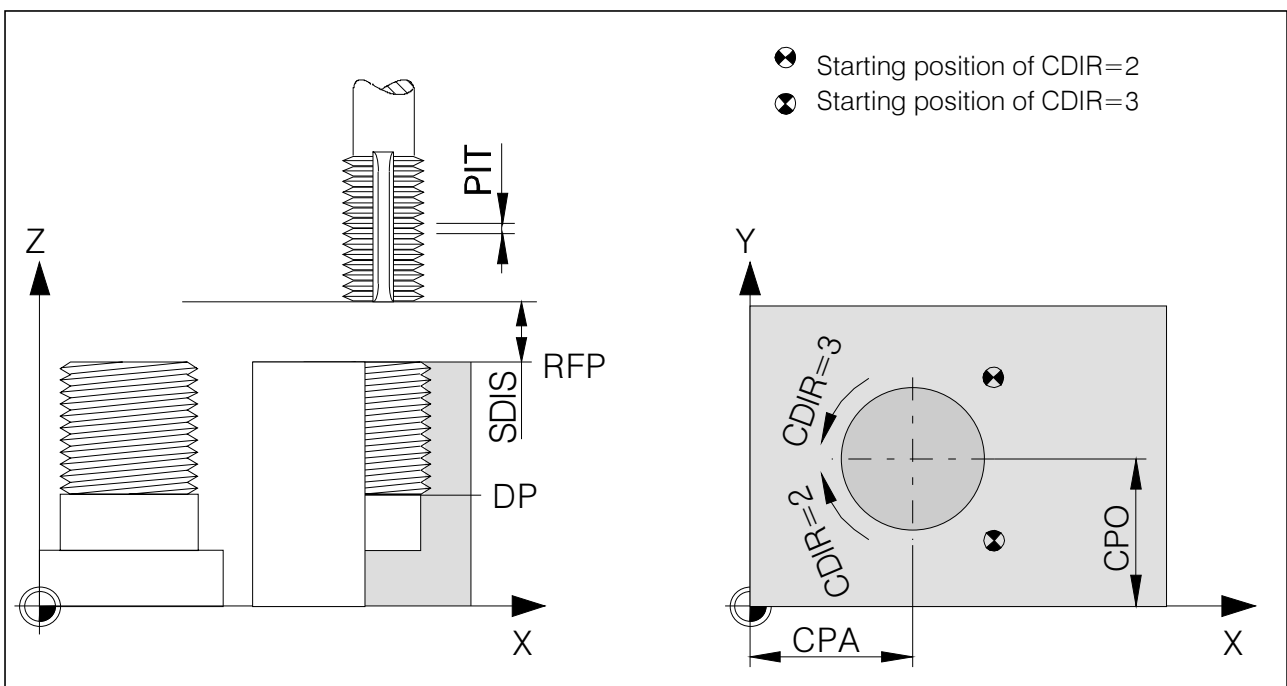
```

G1 X50 Y44
X94 RNDM=6
Y6
X6
Y44
X50 RNDM=0
M17
    
```

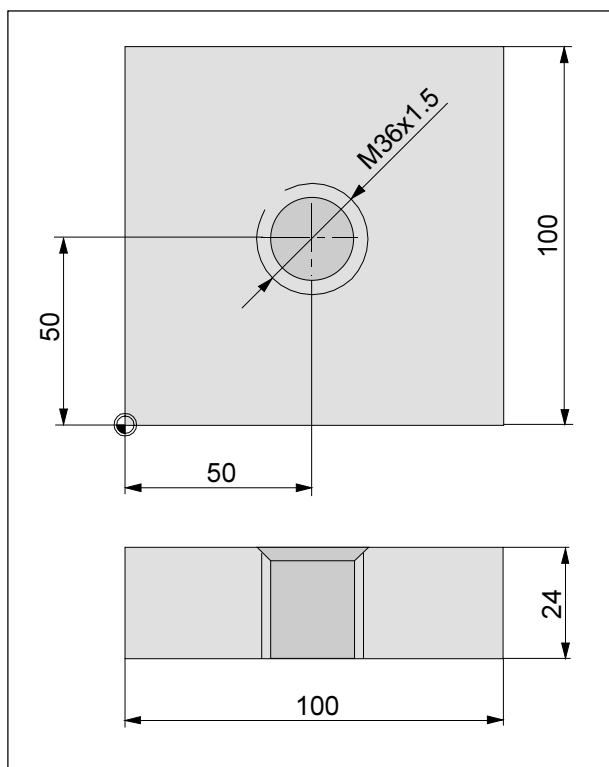
### CYCLE 90 Thread cutting

CYCLE90 (RTP,RFP,SDIS,DP,DPR,DIATH,KDIAM,  
PIT,FFR,CDIR,TYTH,CPA,CPO)

RTP	retraction plane absolute	<b>Re</b> Traction Plane
RFP	reference plane absolute	<b>Re</b> FeRence Plane
SDIS	safety distance without sign see CYCLE81.	<b>Sa</b> FeTy <b>DI</b> Stance
DP	thread length absolute Length of thread, related to workpiece zero point.	<b>De</b> Pth
DPR	thread length relative to reference plane Length of thread, related to reference plane, without sign. Either DP or DPR can be programmed. If nevertheless both values are programmed, the parameter DPR will be valid.	<b>De</b> Pth <b>Re</b> lative
DIATH	nominal thread diameter Outside diameter for internal and external threads	<b>DI</b> Ameter of <b>Th</b> read
KDIAM	core / inside diameter Internal thread: core boring diameter External thread: thread core diameter	<b>Ke</b> rnEl <b>DI</b> AMeter
PIT	thread pitch	<b>PIT</b> ch
FFR	feed rate on helical path	<b>For</b> ward <b>Fe</b> ed <b>Re</b> ate
CDIR	cutting direction 2: G2 (clockwise)    3: G3 (counterclockwise)	<b>Cu</b> ttinG <b>DI</b> Rection
TYPTH	type of thread 0: internal thread    1: external thread Internal thread: start position is thread centre Außengewinde: start position at CDIR=2: X>CPA, Y>CPO start position at CDIR=3: X>CPA, Y<CPO	<b>TY</b> Pe of <b>Th</b> read
CPA	Centre point in X	<b>Cent</b> re <b>PO</b> int <b>Ab</b> scissa
CPO	Centre point in Y	<b>Cent</b> re <b>PO</b> int <b>Or</b> dinatE



## Example Cycle 90



Return plane absolute .....	1
Reference plane absolute .....	0
Safety distance .....	1
Thread depth absolut .....	-25
Depth incr. ....	0
Nominal diameter .....	36
Core hole .....	34,38
Thread lead .....	1.5
Feedrate .....	400
Milling direction .....	2
Thread type 0= inside 1=outside .....	0
Centre point, abszissa .....	50
Centre point, ordinate .....	50

```

G54
TRANS Z20
T1 D1 M6
S2500 M3
G0 X50 Y50 Z2
Cycle90 (1, 0, 1, -25, 0, 0, 36, 34, 38, 1.5, 400, 2,0,5 0,5
0)
G0 X80 Y100 Z50
M30

```

### General: Longhole, Slot milling

- The tool radius must have been entered in the tool data register.
- The tool must cut cross centre (must be suitable to dive in).

The Siemens cycles assume longholes or slots on a circle.

#### Programming parallel longholes and slots

Programming the longhole or slot cycle with the values:

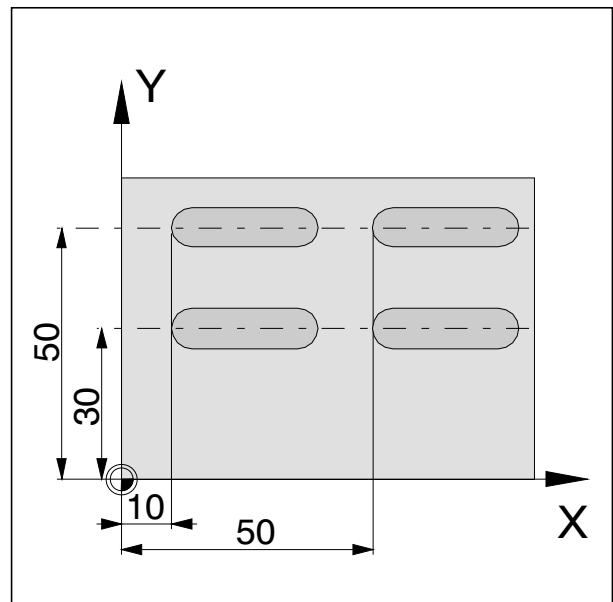
NUM=1, RAD=0, INDA=0

For each longhole or slot a separate cycle must be called.

Example:

```

LONGHOLE (RTP, RFP, SDIS, DP, NUM=1, LENG, CPA=10, CPO=30,
          RAD=0, STA1=0, INDA=0, FFD, FFP1, MID)
LONGHOLE (RTP, RFP, SDIS, DP, NUM=1, LENG, CPA=10, CPO=50,
          RAD=0, STA1=0, INDA=0, FFD, FFP1, MID)
LONGHOLE (RTP, RFP, SDIS, DP, NUM=1, LENG, CPA=50, CPO=50,
          RAD=0, STA1=0, INDA=0, FFD, FFP1, MID)
LONGHOLE (RTP, RFP, SDIS, DP, NUM=1, LENG, CPA=50, CPO=30,
          RAD=0, STA1=0, INDA=0, FFD, FFP1, MID)
    
```



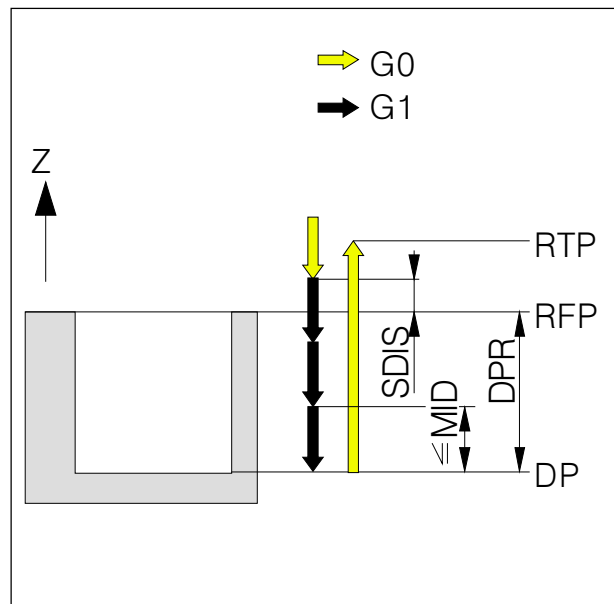
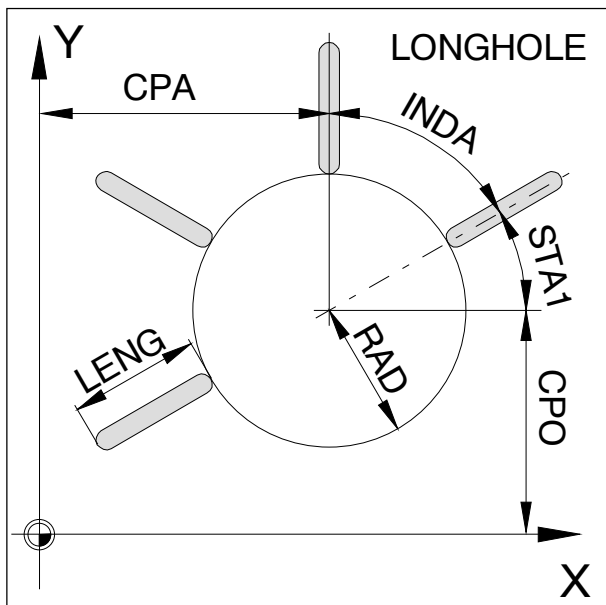


### Longholes on a circle LONGHOLE

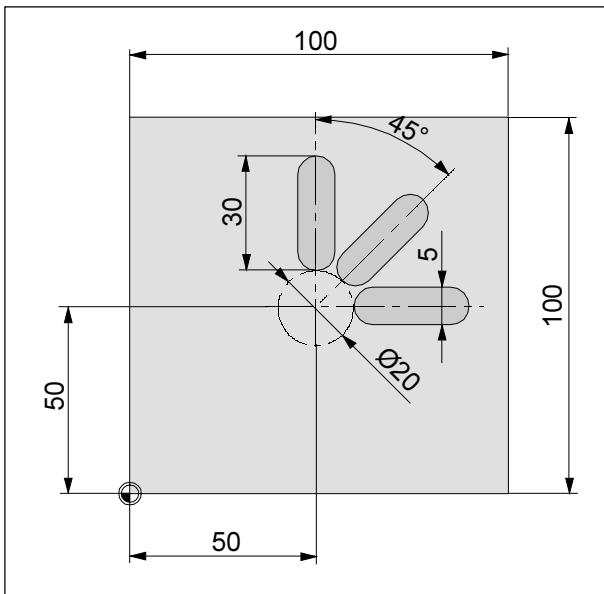
LONGHOLE (RTP,RFP,SDIS,DP,DPR,NUM,LENG,CPA,CPO,  
RAD,STA1,INDA,FFD,FFP1,MID)

With this function each longhole position is approached with rapid feed and the programmed longhole will be machined. Longhole width = tool diameter.  
The sequence of longhole positions will be approached path-optimized.

RTP	retraction plane absolute After the cycle the tool will be at this height.	<b>Re</b> TracTion Plane
RFP	reference plane absolute Height of workpiece surface, normally RFP=0.	<b>Re</b> FeRence Plane
SDIS	safety distance without sign Rapid traverse until SDIS over reference plane, below working feed rate	<b>Sa</b> feTy <b>DIS</b> tance
DP	longhole depth absolute Depth of the longhole, related to the workpiece zero point.	<b>De</b> Pth
DPR	longhole depth relative to reference plane Depth of the longhole, related to the reference plane, without sign. Either DP or DPR can be programmed.	<b>De</b> Pth <b>Re</b> lative
NUM	number of longholes	<b>NUM</b> ber
LENG	length of longhole	<b>LEN</b> Gth
CPA	centre point in X	<b>Cent</b> re <b>Point</b> <b>Ab</b> scissa
CPO	centre point in Y	<b>Cent</b> re <b>Point</b> <b>Ord</b> inate
RAD	radius of the inner circle	<b>RAD</b> ius
STA1	starting angle related to X	
INDA	indexing angle When INDA=0 the longholes will be divided even on the circle.	<b>IND</b> exing <b>AN</b> gle
FFD	feed rate for depth infeed	<b>Fe</b> ed <b>For</b> <b>De</b> pth
FFP1	feed rate for surface machining	<b>Fe</b> ed <b>For</b> <b>PL</b> ane
MID	max. infeed depth for one infeed without sign	<b>MA</b> ximum <b>IN</b> feed <b>De</b> pth



**Example Longhole**



Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Elongated hole .....	-6
Depth incr. ....	0
Number of elongated holes .....	3
Hole length .....	30
Center point, abszissa .....	50
Center point, ordinate .....	50
Radius of circle .....	10
Initial angle .....	0
Incremental angle .....	45
Feedrate depth .....	80
Feedrate surface .....	350
Infeed depth for one infeed .....	2

```

G54
TRANS Z20
T1 D1 M6
S2500 M3
G0 X50 Y50 Z2
Longhole (2, 0, 1, -6, 0, 3, 30, 50, 50, 10, 0, 45, 80, 350,
2)
G0 Z50
M30
    
```

### Slots on a circle SLOT, Circular slot SLOT2

SLOT1 (RTP,RFP,SDIS,DP,DPR,NUM,LENG,WID,CPA,CPO,RAD,STA1, INDA,FFD,FFP1,MID,CDIR,FAL,VARI,MIDF,FFP2,SSF,FALD,STA2)  
 SLOT2 (RTP,RFP,SDIS,DP,DPR,NUM,AFSL,WID,CPA,CPO,RAD,STA1, INDA,FFD,FFP1,MID,CDIR,FAL,VARI,MIDF,FFP2,SSF)

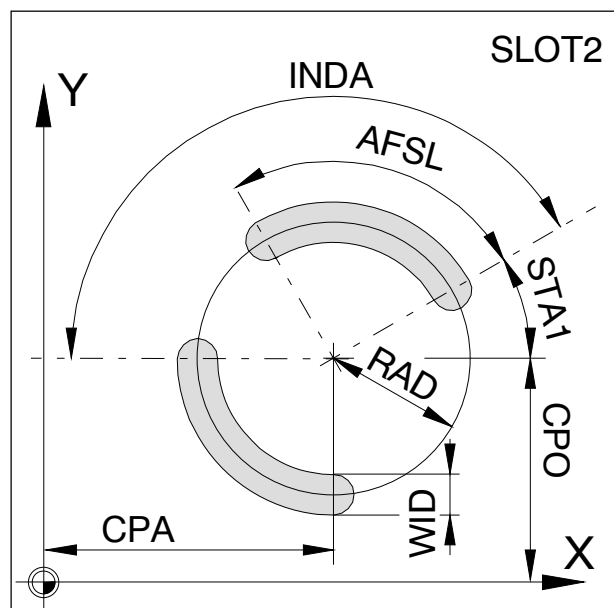
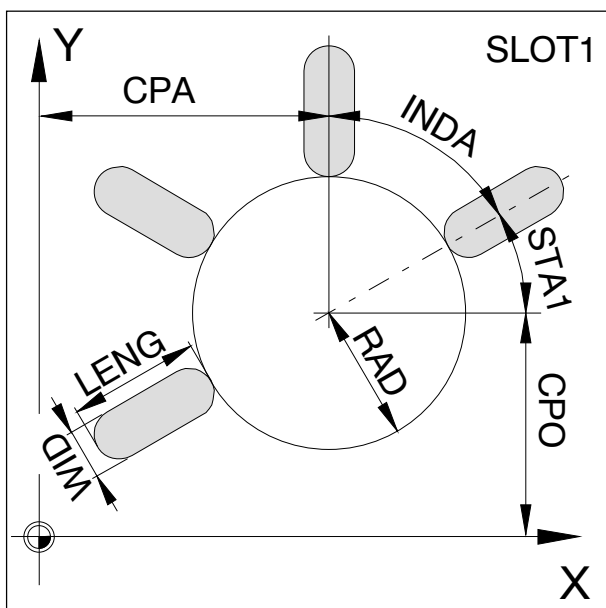
With this function each slot position is approached with rapid feed and the programmed slot will be machined.  
 The slot width will be programmed and must be larger than the tool diameter, but maximum twice the tool diameter.  
 The sequence of slot positions will be approached path-optimized.  
 SLOT1 produces straight slots radial to the slot circle, SLOT2 produces arc-shaped slots on the slot circle.

Parameter additional to LONGHOLE:

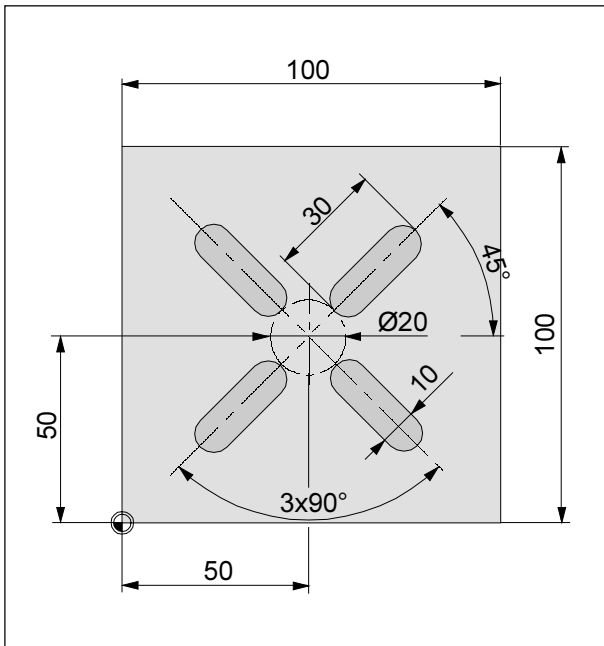
WID	slot width without sign	<b>WIDTH</b>
CDIR	cutting direction 2: G2 (clockwise) 3: G3 (counterclockwise)	<b>Cutting DIRection</b>
FAL	finishing offset without sign	<b>Finishing Allowance</b>
VARI	machining variant	<b>VARIante</b>
UNITS DIGIT:		
0: roughing and finishing to final dimension		
1: roughing until finishing offset		
2: finishing from finishing offset to final dimension		
TENS DIGIT:		
0: Perpendicular with G0		
1: Perpendicular with G1		
3: Oscillation with G1		

When the slot width WID or slot length AFSL is so large or the indexing angle INDA so small, that the slots would touch each other, the cycle will be aborted with an alarm.

MIDF	max. infeed depth for finishing without sign	<b>Maximum Infeed Depth Finishing</b>
FFP2	feed for finishing	<b>Feed For Plane</b>
SSF	spindle speed for finishing	<b>Spindle Speed Finishing</b>
AFSL	angle for slot length without sign only for SLOT2	<b>Angle For Slot Length</b>
FALD(*)	Finishing allowance at the slot bottom	
STA2(*)	Maximum plunging angle for reciprocating movement	



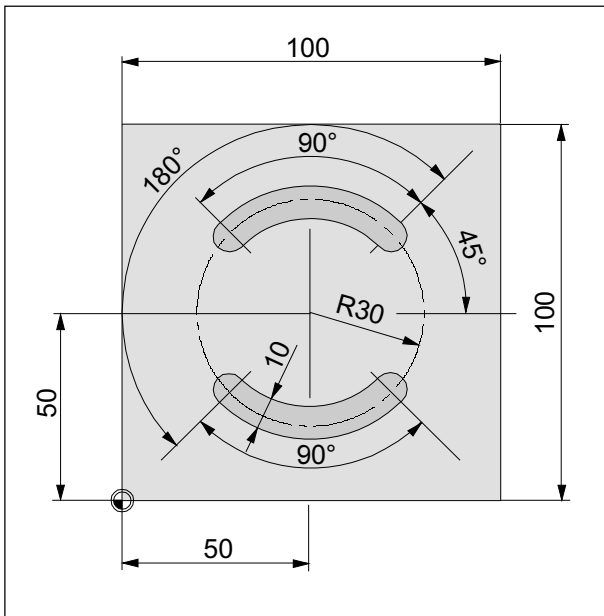
**Example SLOT1**



Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Slot depth absolute .....	-6
Depth incr. ....	0
Number of slots .....	4
Slot length .....	30
Slot width .....	10
Center point, abszissa .....	0
Center point, ordinate .....	0
Radius of circle .....	10
Initial angle .....	45
Incremental angle .....	90
Feedrate depth .....	80
Feedrate surface .....	350
Infeed depth .....	2
Milling direction .....	3
Final machining .....	0.2
Machining type .....	0
0= complete	
1= roughing	
2= finishing	
Maximum infeed depth for finish cutting .....	6
Feedrate for finish cutting .....	400
Speed for finish cutting .....	3500
Finishing allowance at the slot bottom .....	0.5
Maximum plunging depth .....	5

```

G54
TRANS Z20
T1 D1 M6
S2500 M3
G0 X50 Y50 Z2
Slot1 (2, 0, 1, -6, 0, 4, 30, 10, 50, 50, 10, 45, 90,
      80, 350, 2, 3, 0.2, 0, 6, 400, 3500,0.5,5)
G0 Z50
M30
    
```



**Example Slot 2**

Return plane absolute .....	2
Reference plane absolute .....	0
Safety plane .....	1
Circumferential slot depth absolute .....	-6
Circumferential slot depth incr. ....	0
Number of circumferential slots .....	2
Angle for slot length .....	90
Circumferential slot width .....	10
Center point, abszissa .....	50
Center point, ordinate .....	50
Radius of circle .....	30
Initial angle .....	45
Incremental angle .....	180
Feedrate for depth infeed .....	80
Feedrate for surface machining .....	300
Max. infeed depth for one infeed .....	2
Milling direction .....	3
Final machining .....	0.2
Machining type .....	0
0= complete	
1= roughing	
2= finishing	
Maximum infeed depth for finish cutting .....	6
Feedrate for finish cutting .....	400
Speed for finish cutting .....	4000

```

G54
TRANS Z20
T1 D1 M6
S2500 M3
G0 X50 Y50 Z2
Slot2 (2, 0, 1, -6, 0, 2, 90, 10, 50, 50, 30, 45, 180,
      80, 300, 2, 3, 0.2, 0, 6, 400, 4000)
G0 Z50
M30
    
```

**Rectangular pocket POCKET1,  
Circular pocket POCKET2**

POCKET1 (RTP,RFP,SDIS,DP,DPR,LENG,WID,CRAD,CPA,CPO,STA1,FFD,FFP1,MID,CDIR,FAL,VARI,MIDF,FFP2,SSF)

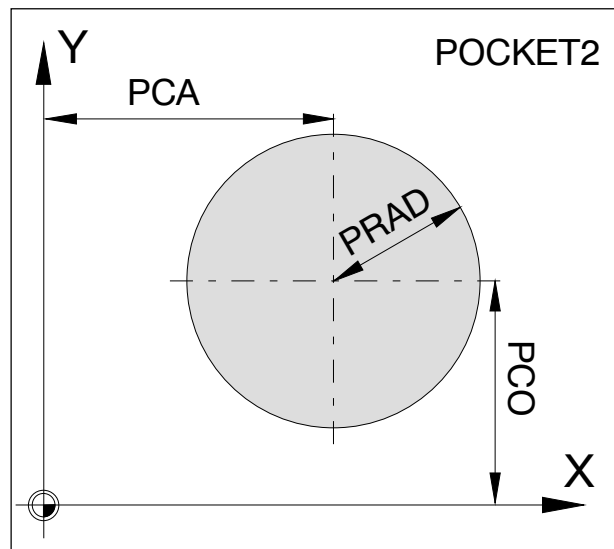
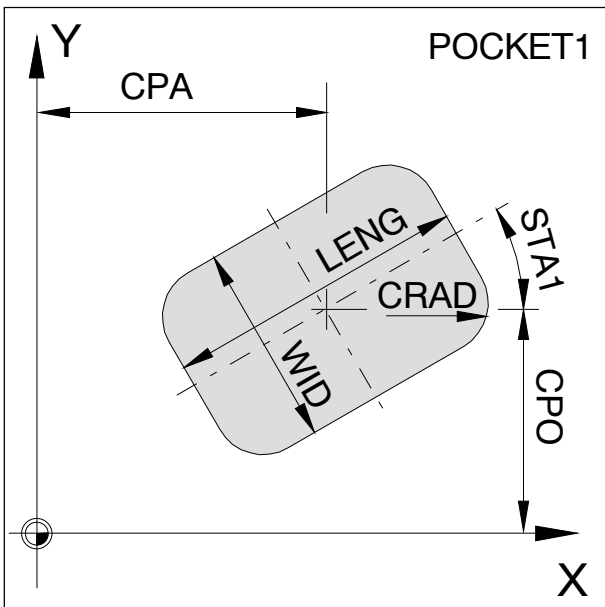
POCKET2 (RTP,RFP,SDIS,DP,DPR,PRAD,CPA,CPO,FFD,FFP1,MID,CDIR,FAL,VARI,MIDF,FFP2,SSF)

The tool traverses with rapid feed to the pocket center at level safety distance over reference plane and works off the pocket from centre outwards.

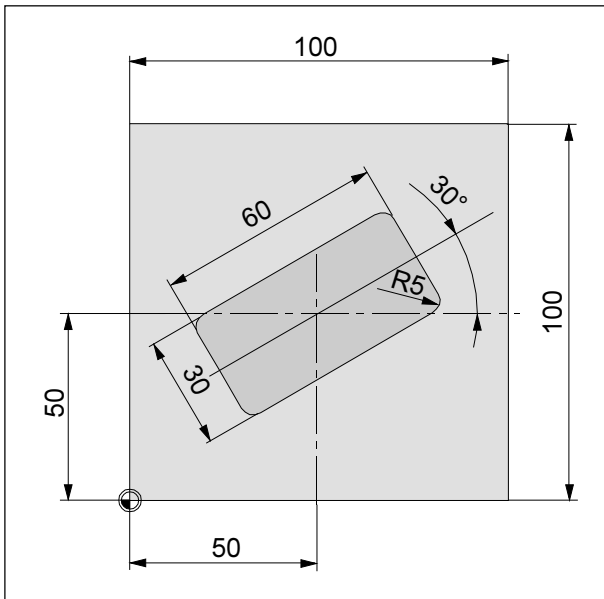
Pocket length and width must be larger than the tool diameter, otherwise cycle abort with alarm.

RTP retraction plane, RFP reference plane, SDIS safety distance, DP, DPR end depth like previous cycles.

PRAD	○ pocket radius without sign	<b>Pocket RADIus</b>
LENG	▭ pocket length without sign	<b>LENGth</b>
WID	▭ pocket width without sign	<b>WIDth</b>
CRAD	▭ corner radius without sign	<b>Corner RADIus</b>
CPA	centre point in X	<b>Centre Point Abscissa</b>
CPO	centre point in Y	<b>Centre Point Ordinate</b>
STA1	pocket angle related to X	
FFD	feed for depth infeed	<b>Feed For Depth</b>
FFP1	feed for surface machining	<b>Feed For Plane</b>
MID	max. infeed depth for cutting without sign	<b>Maximum Infeed Depth</b>
CDIR	machining direction 2: G2 (clockwise)    3: G3 (counterclockwise)	<b>Cutting DIRection</b>
FAL	finishing offset without sign	<b>Finishing Allowance</b>
VARI	machining variant 0: roughing and finishing to final dimension 1: roughing until finishing offset 2: finishing from finishing offset to final dimension	<b>VARIante</b>
MIDF	max. infeed depth for finishing without sign	<b>Maximum Infeed Depth Finishing</b>
FFP2	feed for finishing	<b>Feed For Plane</b>
SSF	spindle speed for finishing	<b>Spindle Speed Finishing</b>



**Example Pocket 1**

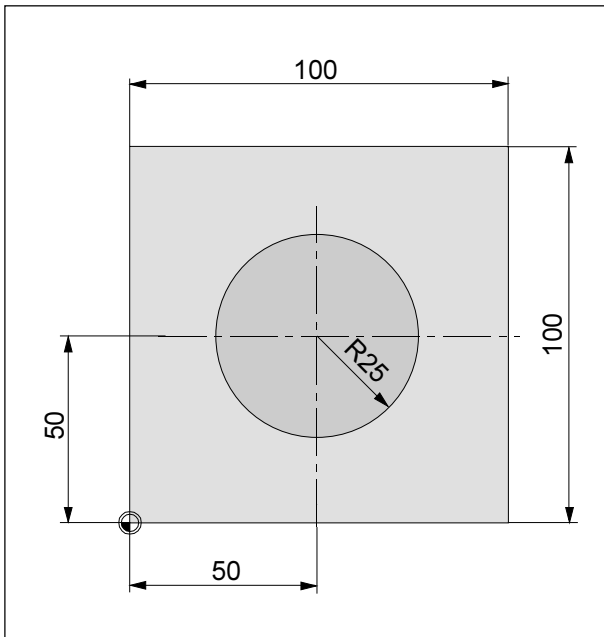


Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Pocket depth absolute .....	-6
Pocket depth incr. ....	0
Pocket length .....	60
Pocket width .....	30
Corner radius .....	5
Center point, abszissa .....	50
Center point, ordinate .....	50
Angle between abszissa and ordinate .....	30
Feedrate for depth infeed .....	80
Feedrate for surface machining .....	300
Max. infeed depth for one infeed .....	2
Milling direction .....	3
Final machining allowance .....	0.2
Machining type .....	0
0= complete	
1= roughing	
2= milling	
Maximum infeed depth for finish cutting .....	6
Feedrate for finish cutting .....	400
Speed for finish cutting .....	4000

```

G54
TRANS Z20
T1 D1 M6
G0 X50 Y50 Z2
Pocket 1 (2, 0, 1, -6, 0, 60, 30, 5, 50, 50, 30, 80, 400,
          2, 3, 0.2, 0, 6, 400, 4000)
G0 Z50
M30
    
```

**Example Pocket 2**



Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Pocket depth absolute .....	-6
Pocket depth incr. ....	0
Pocket radius .....	25
Center point, abszissa .....	50
Center point, ordinate .....	50
Feedrate for depth infeed .....	80
Feedrate for surface machining .....	300
Max. infeed depth for one infeed .....	2
Milling direction .....	3
Final machining allowance .....	0.2
Machining type .....	0
0= complete	
1= roughing	
2= milling	
Maximum infeed depth for finish cutting .....	6
Feedrate for finish cutting .....	400
Speed for finish cutting .....	4000

```

G54
TRANS Z20
T1 D1 M6
G0 X50 Y50 Z2
Pocket 2 (2, 0, 1,-6, 0, 25, 50, 50, 80, 300, 2,
          3, 0.2, 0, 6, 400, 4000)
G0 Z50
M30
    
```



### **Rectangular pocket milling POCKET3, Circular pocket milling POCKET4**

POCKET3 (RTP,RFP,SDIS,DP,LENG,WID,CRAD,PA,PO,STA,MID,  
FAL,FALD,FFP1,FFD,CDIR,VARI,MIDA,AP1,AP2,AD,RAD1,DP1)  
POCKET4 (RTP,RFP,SDIS,DP,PRAD,PA,PO,MID,FAL,FALD,FFP1,FFD,  
CDIR,VARI,MIDA,AP1,AD,RAD1,DP1)

These cycles can be applied to roughing and finishing.

For finishing is, a face cutter is needed.

The depth infeed will always start at the pocket centre point and performed vertically from there; thus a pilot drill can be suitable performed in this position.

A tool offset must therefore be programmed before the cycle is called. Otherwise the cycle is aborted and a alarm is output.

#### **New functions from POCKET3 concerning POCKET 1:**

- Milling direction can be defined with a G instruction (G2/G3) or as compared to the spindle direction.
- For solid machining, the maximum infeed width in the plane is programmable.
- Finishing allowance considering the pocket.
- Three different:
  - vertically at the pocket center point.
  - along the helicoidal path around the pocket centre.
  - oscillating around the pocket central axis.
- Shorter approach path around the pocket centre.
- Consideration of a workpiece contour in the plane and a basic size at the base (optimum processing of pre-formed pockets possible).

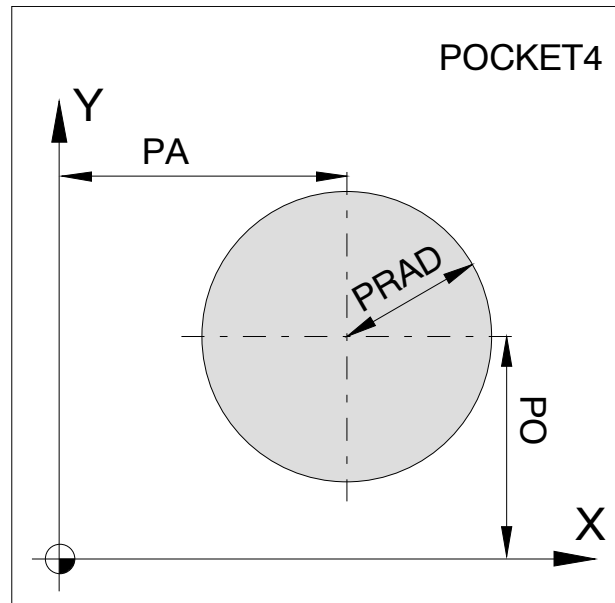
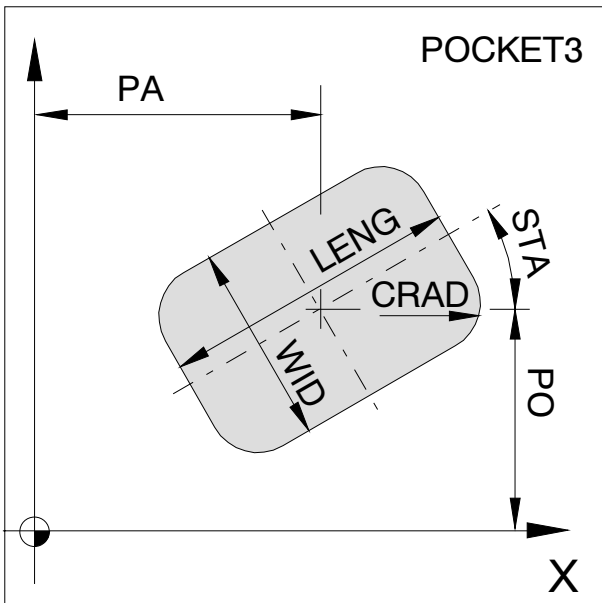
#### **New function from POCKET4 concerning POCKET 2:**

- Milling direction can be defined with a G instruction (G2/G3) or as compared to the spindle direction.
- For solid machining, the maximum infeed width in the plane is programmable.
- Finishing allowance for the pocket base.
- Two different penetration strategies:
  - vertically from the pocket centre point
  - along a helicoidal path around the pocket centre.
- Shorter approach path in the plane for finishing.
- Consideration of a workpice contour in the plane and a basic size at the base (optimum processing of pre-formed pockets possible)
- MIDA is newly calculated during edge machining.

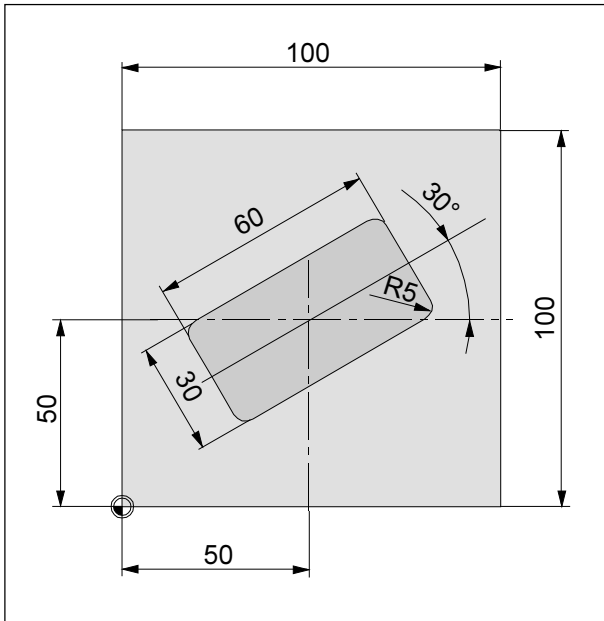
Working outlet see POCKET3, POCKET4

RTP,RFP,DP,SDIS,PRAD,LENG,WID,CRAD,FFD,FAL,FFP1,MID,CDIR,VARI,  
MIDA, STA the same as in the previous cycles.

- CDIR Milling direction: (enter without sign)  
 0: same direction (as spindle rotation)  
 1: opposite direction  
 2: with G2 (independent of spindle direction)  
 3: with G3
- VARI Type of machining: (enter without sign)  
 UNIT DIGIT  
 1: Solid machining to final machining allowance  
 2: Finishing  
 TENS DIGIT:  
 0: Vertically to the pocket centre with G0  
 1: Vertically to the pocket centre with G1  
 2: Along a helix
- PA Pocket centre point in X **Point Abscissa**  
 PO Pocket centre point in Y **Point Ordinate**  
 FALD Final allowance at base **Finishing Allowance Depth**  
 AP1 Basic size pocket length  
 AP2 Basic size pocket width  
 AD Basic size pocket depth  
 RAD1 Radius of the helicoidal path on plunge  
 DP1 Infeed depth per revolution during penetration along a helicoidal path



**Example Pocket 3**

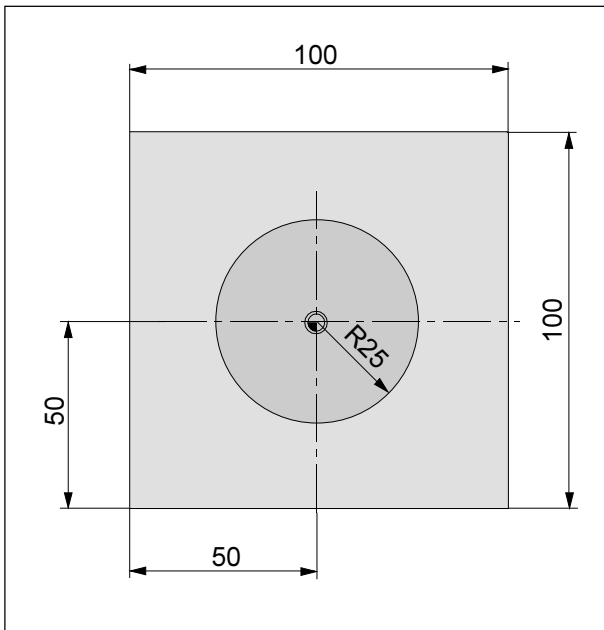


Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Pocket depth .....	-6
Pocket length .....	60
Pocket width .....	30
Corner radius .....	5
Reference point (Pocket center), abszissa .....	50
Reference point (Pocket center), ordinate .....	50
Angle between longitudinal axis and abszissa ...	30
Max. infeed depth for one infeed .....	2
Final machining edge .....	0.2
Final machining bottom .....	0.1
Feedrate surface .....	300
Feedrate depth .....	80
Milling direction .....	3
0= synchronized milling	
1= backward milling	
2= G2	
3= G3	
Machining type .....	21
UNITS DIGIT	
1: Roughing	
2: Finishing	
TENS DIGIT	
0: G0	
1: G1	
2: Helix	
3: Oscillation	
Maximum infeed width .....	7
Unfinished length .....	0
Unfinished width .....	0
Unfinished depth .....	0
Radius, angle .....	10
Depth for infeeding .....	2

```

G54
TRANS Z20
T2 D1 M6
S3000 M3
G0 X50 Y50 Z2
Pocket 3 (2, 0, 1, -6, 60, 30, 5, 50, 50, 30, 2, 0.2,
          0.1, 300, 80, 3, 21, 7, 10, 2)
G0 Z50
M30
    
```

**Example Pocket 4**



Return plane absolute .....	2
Reference plane absolute .....	0
Safety distance .....	1
Pocket depth absolute .....	-6
Pocket radius .....	25
Pocket centre point, abszissa .....	0
Pocket centre point, ordinate .....	0
Max. infeed depth for one infeed .....	3
Final machining edge .....	0.2
Final machining bottom .....	0.1
Feedrate surface .....	400
Feedrate depth .....	80
Milling direction .....	3
0= synchronized milling	
1= backward milling	
2= G2	
3= G3	
Machining type .....	21
UNITS DIGIT	
1: Roughing	
2: Finishing	
TENS DIGIT	
0: G0	
1: G1	
2: Helix	
3: Oscillation	
Maximum infeed width .....	7
Unfinished plane .....	0
Unfinished depth .....	0
Radius only for plunge on helix path .....	10
Depth for infeeding .....	3

```

G54
TRANS X50 Y50 Z20
T2 D1 M6
S3000 M3
G0 X0 Y0 Z2
Pocket 4 (2, 0, 1, -6, 25, 0, 0, 3, 0.2, 0.1, 400,
          80, 3, 21, 7, 0, 0, 10, 3)
G0 Z50
M30
    
```

### Rectangular stud milling **CYCLE76**(\*), Circular stud milling **CYCLE77**(\*)

CYCLE76 (RTP,RFP,SDIS,DP,DPR,LENG,WID,CRAD,PA,PO,STA,MID,  
FAL,FALD,FFP1,FFD,CDIR,VARI,AP1,AP2)

CYCLE77 (RTP,RFP,SDIS,DP,DPR,PRAD,PA,PO,MID,FAL,FALD,FFP1,FFD,  
CDIR,VARI,AP1)

These cycles can be used for roughing and finishing.

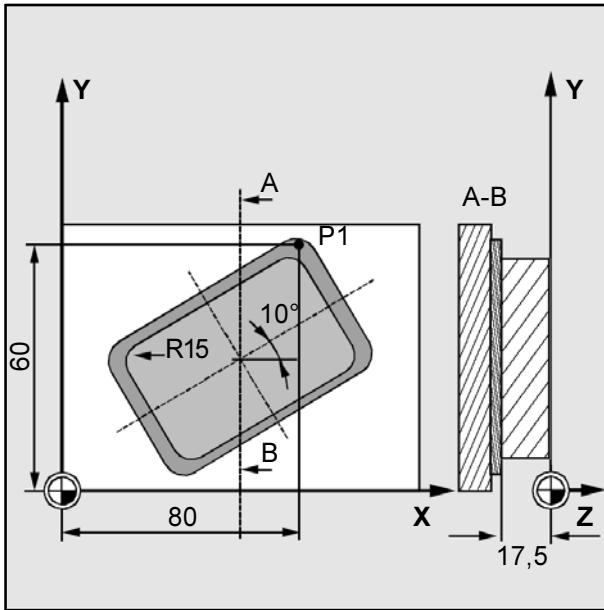
For finishing a face cutter is necessary.

The plunging depth is always carried out in the position prior to approaching the contour in a semicircular way.

Before calling the cycle, you have to activate a tool correction. Otherwise the cycle is aborted and followed by an alarm.

RTP	Retraction plane absolute After the cycle the tool is positioned in this plane.	<b>Re</b> Tracti <b>o</b> n Plane
RFP	Reference plane absolute Height of the workpiece surface, mostly RFP=0.	<b>Re</b> Fe <b>r</b> e <b>n</b> ce Plane
SDIS	Safety distance without arithmetic sign Rapid traverse up to SDIS above reference plane, below in working feed rate.	<b>S</b> afety <b>DI</b> St <b>A</b> n <b>C</b> e
DP	Absolute depth Depth of the stud, referred to the workpiece datum.	<b>De</b> P <b>th</b>
DPR	Depth in relation to reference plane Depth of the stud, referred to the reference plane, without arithmetic sign. Either DP or DPR is programmed.	<b>De</b> P <b>th</b> <b>R</b> elative
PRAD	○-stud radius without arithmetic sign	
LENG	□-stud length with arithmetic sign	<b>LE</b> NG <b>th</b>
WID	□-stud width with arithmetic sign	<b>WI</b> D <b>th</b>
CRAD	□-corner radius without arithmetic sign	<b>C</b> orner <b>RA</b> D <b>i</b> u <b>s</b>
PA	Reference point stud absciss (in X)	
PO	Reference point stud ordinate (in Y)	
STA	Angle between longitudinal axis of the stud and X-axis	
MID	Maximum infeed depth for a single cutting depth without arithmetic sign	<b>M</b> aximum <b>I</b> nfeed <b>D</b> e <b>p</b> th
FAL	Finishing allowance without arithmetic sign	
FALD	Finishing allowance at the bottom without arithmetic sign	
FFP1	Feed for plane machining	<b>F</b> eed <b>F</b> or <b>P</b> lane
FFD	Feed for plunging depth	<b>F</b> eed <b>F</b> or <b>D</b> e <b>p</b> th
CDIR	Cutting direction 0: down-cut milling                      1: up-cut milling 2: G2 (clockwise)                      3: G3 (counter-clockwise)	<b>C</b> utting <b>DI</b> R <b>e</b> cti <b>o</b> n
VARI	Machining mode 1: Roughing until finishing clearance 2: Finishing of finishing clearance until absolute measurement	<b>V</b> AR <b>I</b> ante
AP1	Length/ diameter blank stud without arithmetic sign	
AP2	Width blank stud without arithmetic sign	

**Example for rectangular stud milling Cycle 76**



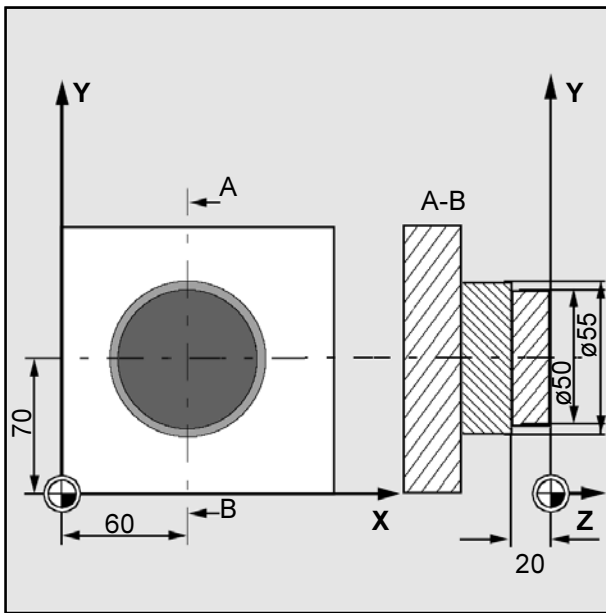
Retraction plane absolute .....	10
Reference plane absolute .....	0
Safety distance .....	2
Depth absolute .....	-17.5
Depth in relation to reference plane .....	0
Stud length .....	-60
Stud width .....	-40
Corner radius .....	15
Reference point 1st axis of plane .....	80
Reference point 2nd axis of plane .....	60
Angle between longitudinal axis and abscissa .....	10
Plunging depth for one infeed at maximum .....	11
Finishing allowance without arithmetic sign .....	0
Finishing allowance bottom without arithmetic sign .....	0
Feed for plane machining .....	900
Feed for plunging depth .....	800
Milling direction .....	0
0: down-cut milling	
1: up-cut milling	
2: direction G2	
3: direction G3	
Machining: .....	1
1: roughing until finishing clearance	
2: finishing of finishing clearance until absolute measurement	
Length of blank stud .....	80
Width of blank stud .....	50

Cycle setting data:

\_ZSD [2]=0     Dimension of rectangular pocket or rectangular spigot from the center point  
 \_ZSD [2]=1     Dimension of rectangular pocket or rectangular spigot from a corner

```
G54
TRANS Z20
T2 D1 M6     (end mill cutter 10mm)
S3000 M3
G0 X50 Y50 Z2
_ZSD [2]=1     (Dimensioning of spigot referred to corners)
Cycle 76 (10,0,2,-17.5,0,-60,-40,15,80,60,10,11,0,0,900,800,0,1,80,50)
G0 Z50
M30
```

**Example for rectangular stud milling Cycle 77**



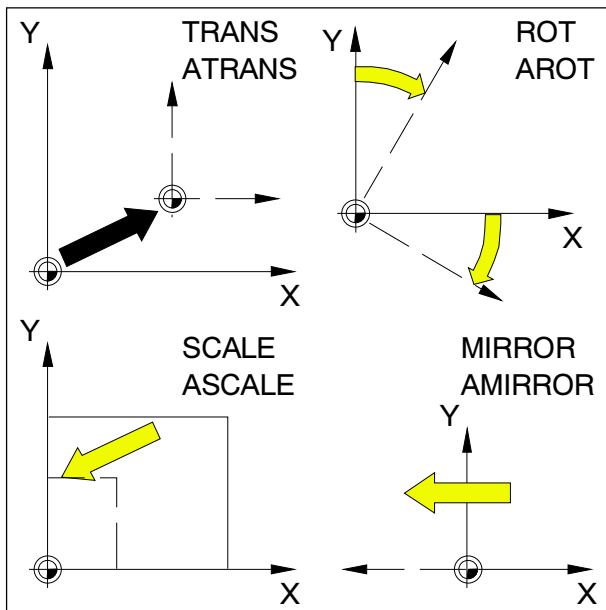
Retraction plane absolute .....	10
Reference plane absolute .....	0
Safety distance .....	3
Depth absolute .....	-20
Depth in relation to reference plane .....	
Stud radius .....	50
Reference point 1st axis of plane .....	60
Reference point 2nd axis of plane .....	70
Plunging depth for one infeed at maximum .....	10
Finishing allowance without arithmetic sign .....	0.5
Finishing allowance bottom without arithmetic sign	0
Feed for plane machining .....	900
Feed for plunging depth .....	800
Milling direction .....	1
0: down-cut milling	
1: up-cut milling	
2: direction G2	
3: direction G3	
Machining .....	1
1: Roughing until finishing clearance	
2: Finishing of finishing clearance until absolute measurement	
Diameter blank stud .....	55

```

G54
TRANS Z20
T2 D1 M6 (end mill cutter 10mm)
S1800 M3
G0 X50 Y50 Z2
Cycle 77 (10,0,3,-20,0,50,60,70,10,0.5,0,900,800,
          1,1,55)
G0 Z50
M30
    
```





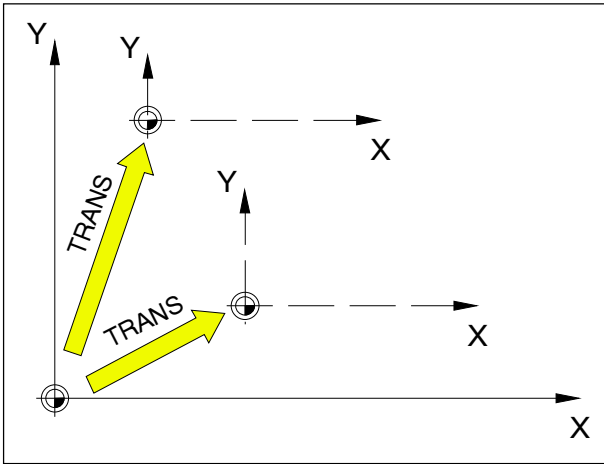


### Frames

Frames alter the actual coordinate system.

- Shift coordinate system: TRANS, ATRANS
- Rotate coordinate system: ROT, AROT
- Programmable scale factor: SCALE, ASCALE
- Mirror coordinate system: MIRROR, AMIRROR

The frame commands will be programmed in a separate NC block and executed in the programmed sequence.



ATRANS relates to the last valid zero point G54 - G599, TRANS.

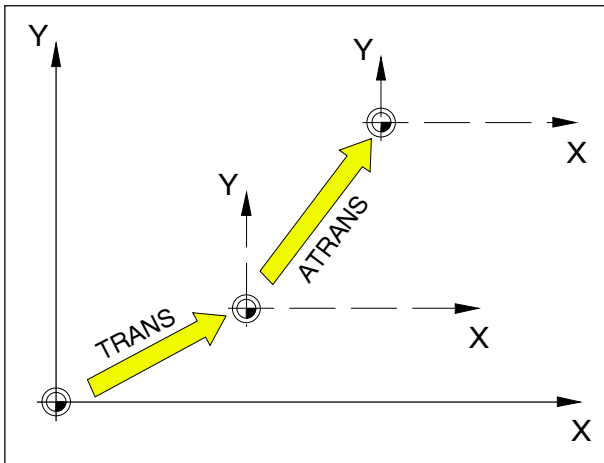
**Programmable zero offset TRANS, ATRANS**

Format:

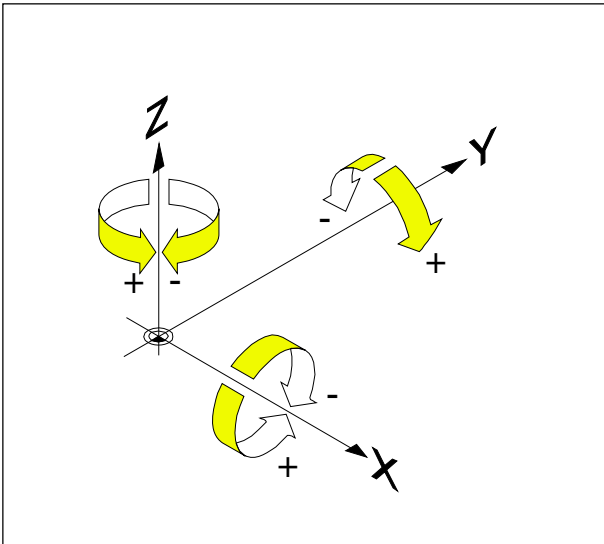
TRANS/ATRANS X... Z...

**TRANS** Zero offset absolute, related to the actual zero point G54-G599. (TRANS deletes all previous programmed frames (TRANS, ATRANS, ROT, AROT, ...)).

**ATRANS** Zero offset additive, related to the actual settable (G54-G599) or programmed (TRANS/ATRANS) zero point. A zero shift that builds-up on existing frames (TRANS, ATRANS, ROT, AROT, ...) is programmed with ATRANS.



TRANS relates always to the actual zero point G54 - G599.



**Programmable rotation ROT, AROT**

ROT/AROT is used to rotate the workpiece coordinate system around each of the geometry axes X, Y, Z or through an angle RPL in the selected working plane G17, G18, G19.

This allows easier programming of contours with main axes that are inclined to the geometry axes.

**Format:**

ROT/AROT X.. Y.. Z..

ROT/AROT RPL=..

**ROT** Rotation absolute, related to the actual zero offset G54-G599.

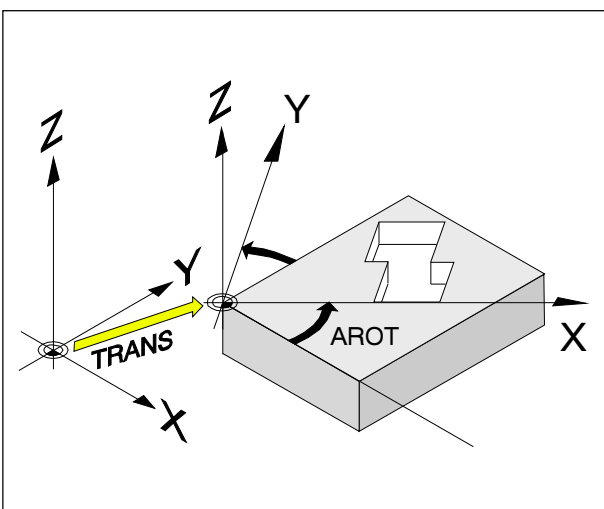
(ROT deletes all previous programmed frames (TRANS, ATRANS, ROT, AROT, ...)).

**AROT** Rotation additive, related to the actual settable (G54-G599) or programmed (TRANS/ATRANS) zero offset.

A rotation that builds-up on existing frames (TRANS, ATRANS, ROT, AROT, ...) is programmed with AROT.

X, Y, Z Rotation in space (in degrees); geometry axis around which the rotation takes place.

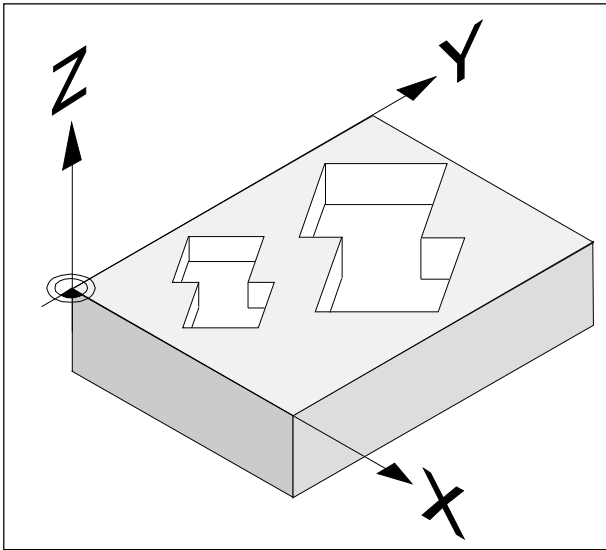
RPL= Rotation in the plane (e.g. G17) (in degrees).



**Example:**

For easier programming of the contour beside the coordinate system should be rotated for 30°, the existing programmable zero offset TRANS should be kept.

```
N.. G17
N.. TRANS ...
N..
N60 AROT Z30
or
N60 AROT RPL=30
```



**Programmable scale factor SCALE, ASCALE**

SCALE/ASCALE allows to set a separate scale factor for every axis X, Y, Z. When different scale factors are used for X, Y, Z the contour becomes distorted.

**Format:**

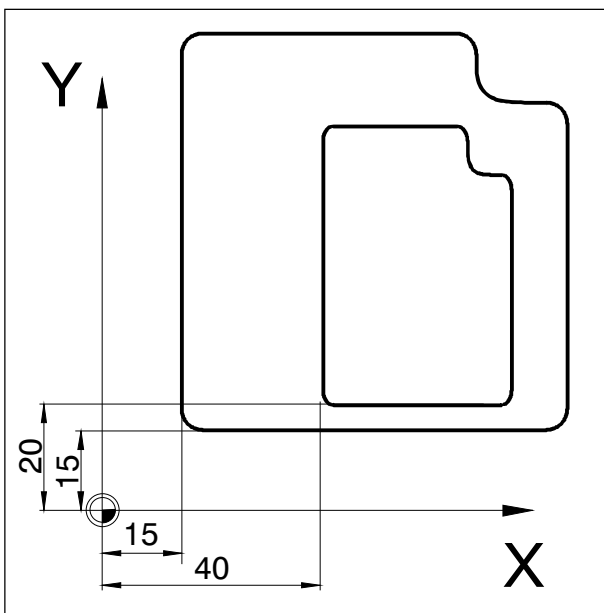
SCALE/ASCALE X.. Y.. Z..

When after SCALE/ASCALE a zero offset is programmed with ATRANS it also will be scaled.

**SCALE** Scale absolute, related to the actual settable zero offset G54-G599. SCALE deletes all previous programmed frames (TRANS, ATRANS, ROT, AROT, ...). SCALE without axis address deselects the scale factor (and all other frames).

**ASCALE** Scale additive, related to the actual settable (G54-G599) or programmed (TRANS/ATRANS) zero point. A scale that builds-up on existing frames (TRANS, ATRANS, ROT, AROT, ...) is programmed with ASCALE.

X, Y, Z Scale factor for each axis.

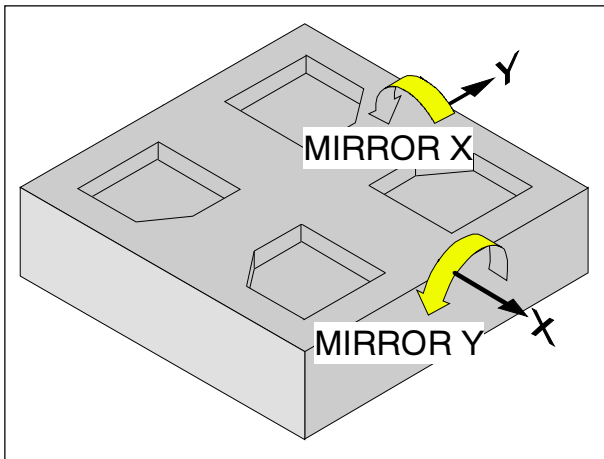


**Example**

The contour beside is programmed in subprogram L10. It is used twice in a different size and distorted.

```

N.. G54                zero offset
N35 TRANS X15 Y15     absolute prog. zero offset
N40 L10               large contour
N45 TRANS X40 Y20     absolute prog. zero offset
N55 ASCALE X0.5 Y0.7 scale factor
N60 L10               small contour
N75 SCALE              deselect frames
    
```



**Programmable mirroring MIRROR, AMIRROR**

MIRROR/AMIRROR mirrors workpiece shapes on coordinate axes X, Y, Z.

**Format:**

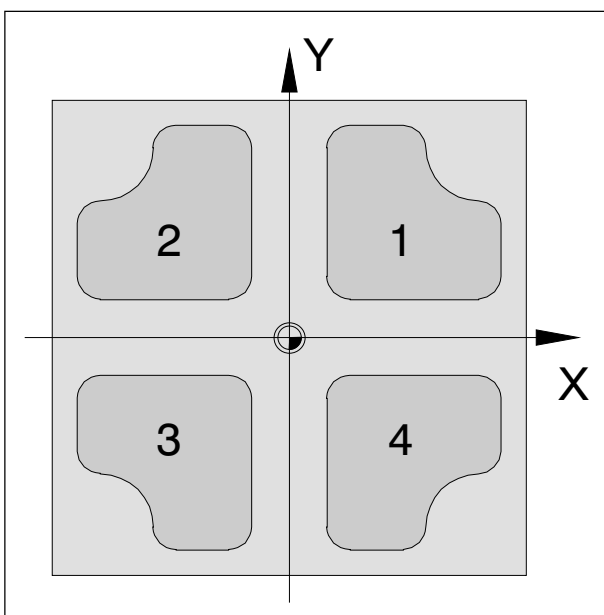
MIRROR/AMIRROR X.. Y.. Z..

When a contour is mirrored, the circle direction G2/G3 and the cutter radius compensation G41/G42 are changed automatically.

**MIRROR** Mirroring absolute, related to the actual settable zero offset G54-G599. (MIRROR deletes all previous programmed frames (TRANS, ATRANS, ROT, AROT, ...)). MIRROR without axis address deselects mirroring (and all other frames).

**AMIRROR** Mirroring additive, related to the actual settable (G54-G599) or programmed (TRANS/ATRANS) zero point. Mirroring, that builds-up on existing frames (TRANS, ATRANS, ROT, AROT, ...) is programmed with AMIRROR.

**X, Y, Z** Geometry axis to be mirrored on. The value indicates the distance from the mirror axis to the geometry axis, e.g. X0.



**Example**

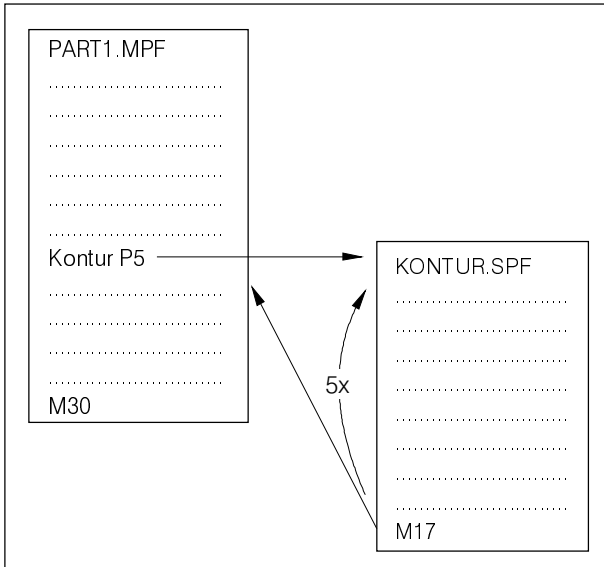
Contour 1 is programmed in the subprogram L10. The further three contours are programmed with mirroring. The zero point G54 is in the middle of the workpiece.

```

N.. G54          zero offset
N40 L10         contour top right
N45 MIRROR X0  mirror on Y
N50 L10         contour top left
N55 AMIRROR Y0 additive mirror on X
N60 L10         contour bottom left
N65 MIRROR Y0  mirror on X
N70 L10         contour bottom right
N75 MIRROR     mirror off
    
```



## Subprograms



Program run with subprogram

Functions which are repeated multiple can be programmed as subprograms.

The subprograms are called with here names

R parameter can be transfered in subprograms  
**Subprogram Call in Part Program**

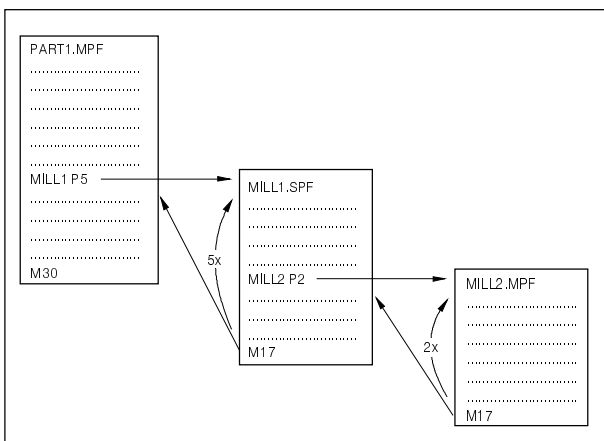
z.B.: MILL1 P1 LF

MILL1 Subprogram number

P1 Number of Subprogram runs (max. 99)

### Subprogram End with M17

z.B.: N150 M17 LF



Nesting of subprograms

### Subprogram Nesting

A eleven-fold nesting of subprograms is possible. Block search is possible into the eleventh subroutine level.

Cycles also count as subprograms, that means e.g. a drilling cycle can be called max. in the 10th subprogram level.

**Subprogram with SAVE- mechanism**

With this function, the operating data which are currently valid in the main program, such as G functions or overall Frame, are stored when the subprogram is called. On return to the calling program the old state is automatically restored. For this, specify the additional command SAVE with the definition statement with PROC.

**Subprograms with passing parameters**

Beginning of program, PROC

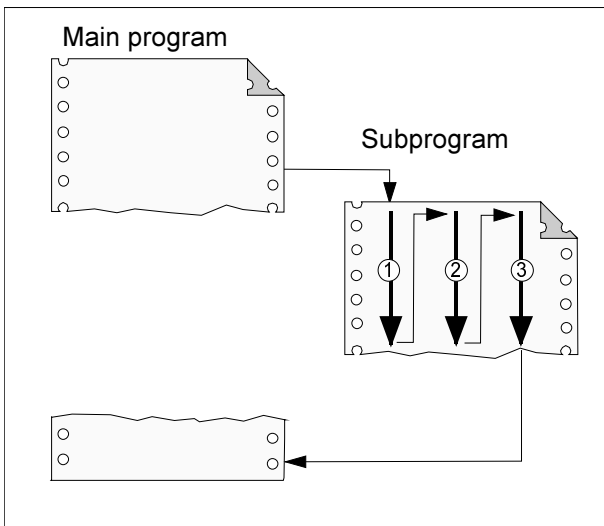
A subprogram that is to take over parameters from the calling program when the program runs is designated with the vocabulary word PROC.

End of program M17, RET

The command M17 designates the end of subprogram and is also an instruction to return to the calling main program.

The vocabulary word RET stands for end of subprogram without interruption of continuous path mode and without function output to the PLC.

Subprogram calls must be programmed in a separate NC block.



**Subprogram with program repeating, P**

If you want to execute a subprogram several times in succession, you can program the required number of program repetitions in the block in the subprogram call under address P.

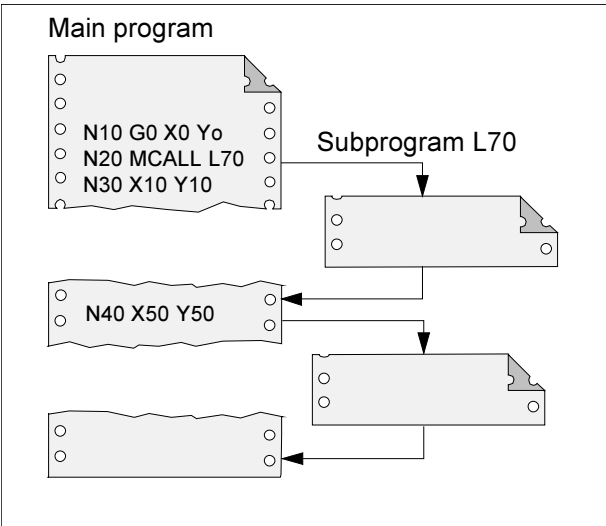
Parameters are only passed on during the program call or the first pass. The parameters remain unchanged for the repetitions.



In a program run, only one MCALL call can apply at any one time. Parameters are only passed once with MCALL.

**Modal subprogram MCALL**

With this function the subprogram is automatically called an executed after every block with motion. In this way you can automate the calling os subprograms that are to be executed at different positions on the workpiece. For example, for drilling patters.



**Example**

```
N10 G0 X0 Y0
N20 MCALL L70
N30 X10 Y10
N40 X50 Y50
```

**Deactivating the modal subprogram call**

With MCALL without a subprogram call or by programming a new modal subprogram call for a new subprogram.



## Program jumps

### Uncontitional program jumps

Format

Label:

GOTOB LABEL

or

GOTOF LABEL

Label:

**GOTOB** Jump instruction with jump destination backwards (towards the start of program)


**GOTOF** Jump instruction with jump destination forwards (towards the end of program)

**LABEL** Destination (label within the program)

**LABEL:** Jump destination

Programs working in standard manner (main programs, subroutines, cycles,...) can be changed in order by means of program jumps. Destination addresses can be approached within a program by means of GOTOF and/or GOTOB.

The program continues processing with the instruction following immediately the destination address.



Program jumps must be programmed in a separate NC block.

### Conditional program jumps

Format:

Label:

IF expression GOTOB LABEL

oder

IF expression GOTOF LABEL

LABEL:

**IF** Conditions

**GOTOB** Jump instruction with jump destination backwards (towards the start of program)

**GOTOF** Jump instruction with jump destination forwards (towards the end of program)

**LABEL** Destination (label within the program)

**LABEL:** Jump destination


Jump conditions can be formulated with IF statements. The jump to the programmed destination only occurs if the jump condition is fulfilled.

## Programming messages, MSG

Messages can be programmed to provide the user with information about the current machining situation during program execution.

A message is generated in an NC program by inserting the keyword "MSG" in parentheses "()" followed by the message text in double quotation marks.

A message can be cleared by programming "MSG()".



A message text can be up to 124 characters long and is displayed in two lines (2x62 characters). Contents of variables can also be displayed in message text.

Example:

```
N10 MSG ("Roughing of contour")
N20 X... Y...
N ...
N90 MSG ()
```

You can also set alarms in addition to messages in an NC program. Alarms are displayed in a separate field on the screen display. An alarm is associated with a reaction on the control which depends on the alarm category.

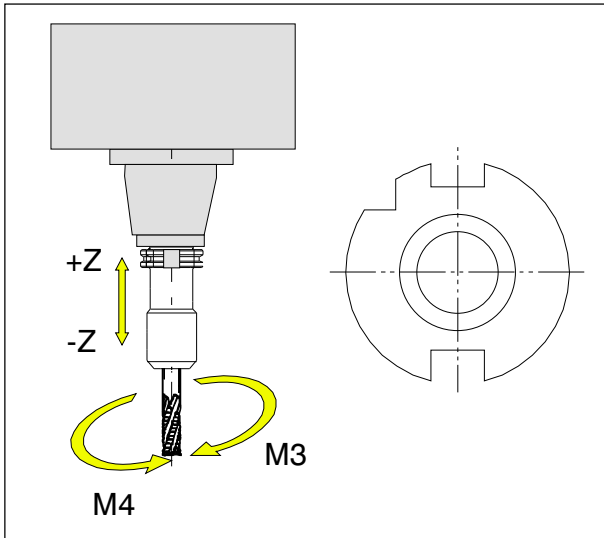
Alarms are programmed by inserting the keyword "SETAL" followed by the alarm number in parentheses.

Alarms are always programmed in a separate block.

Example:

```
N100 SETAL (65000) ;Set alarm 65000
```

**Spindle ON M3 / M4, Speed S, Spindle STOP M5, Spindle positioning SPOS**



**View of the rotational direction**

View the spindle from +Z in direction -Z, to determine the rotational sense

**M3** . clockwise ..... right-hand turning

**M4** . counterclockwise ..... left-hand turning

**M5** . spindle stop

**Spindle speed programming**

The spindle speed is programmed with the address S.

Example:

N20 M3 S2000

Spindle ON clockwise (right) with 2000 rpm

**Spindle positioning**

SPOS stops the milling spindle at the programmed position.

Format:

SPOS=... [°]



## A axis (Dividing device)

For milling during the cylinder surface transformation, the A axis and tool slide must be moved in a definite relation against each other.

Therefore, the A axis (dividing device) is always a round axis which can be operated and programmed like an ordinary linear axis.

The indication of the A axis is always carried out in degrees.

e.g. G0 A90

## TRACYL

Is used for contour milling at the surface area.

The cylinder surface curve transformation provides the following capabilities:

- Longitudinal grooves on cylindrical bodies,
- Transverse grooves on cylindrical bodies,
- Any other groove shapes on cylindrical bodies.

The shape of the grooves is programmed with reference to the processed level cylinder surface area.

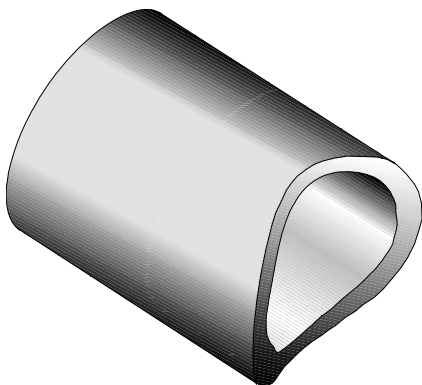
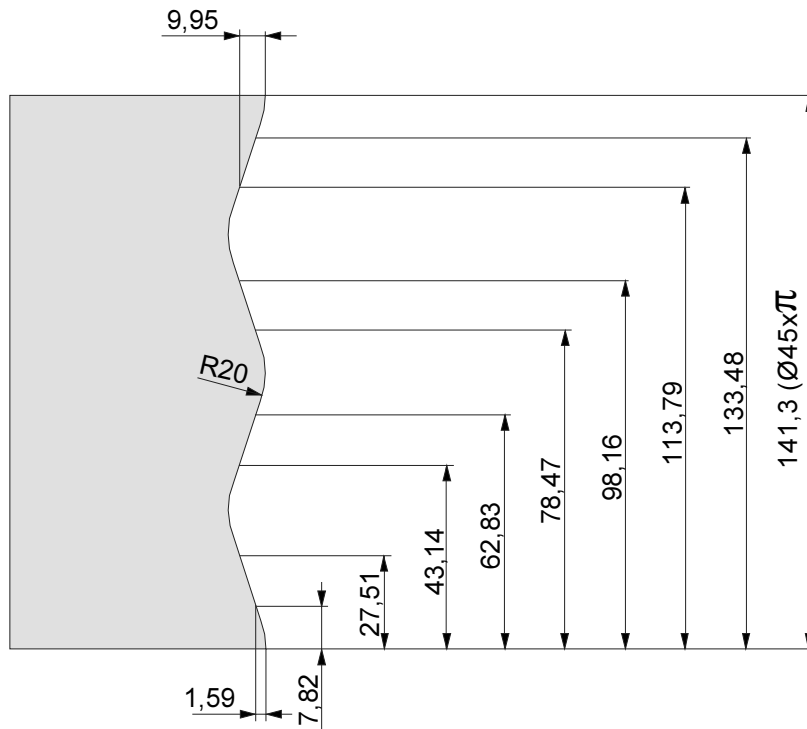
Selection:  
 general ..... TRACYL( )  
 Deselection:  
 general ..... TRAFOOF

**Note:**

Due to an actual transformation or deselection of transformation, the zero point offset and the previous transformations (e.g. Transmit) are deselected and must be programmed again.







```

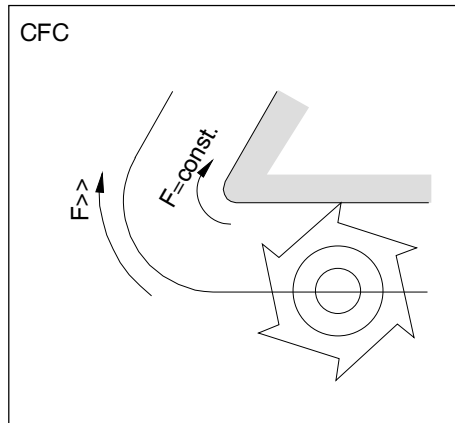
G54
TRANS X64.5
T1 D1 M6
S2000 F200 M3 M8
G0 X10 Y0 Z20
TRACYL(45)
G55
TRANS X61.5
M11
G0 Y0
G1 X0 G42
G3 X-1.591 Y7.871 CR=20
G1 X-9.952 Y27.508
G2 X-9.952 Y43.142 CR=20
G1 X-1.591 Y62.833
G3 X-1.591 Y78.467 CR=20
G1 X-9.952 Y98.158
G2 X-9.952 Y113.483 CR=20
G1 X-1.591 Y133.483
G3 X0 Y141.3 CR=20
G1 X10 G40
TRAFOOF
M10
G54
TRANS X64.5
G0 X60 Y60 Z60
M30
    
```

## Feed optimizing CFTCP, CFC, CFIN

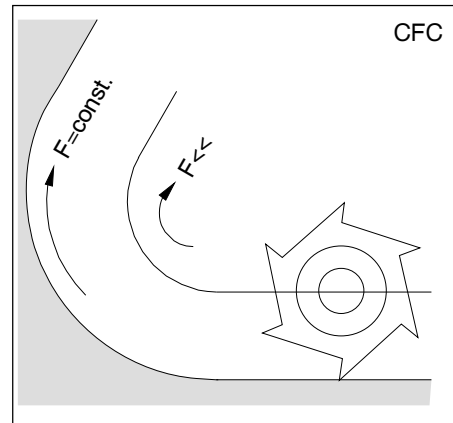
Basic status (CFC):

With active cutter radius compensation G41/42 the feed is valid at the programmed contour.

The basic setting CFC can result in unwanted high or low feedrates in curves at the workpiece side opposite to the contour.



Tool outside the curve



Tool inside the curve

The feed characteristic can be determined with the following commands

### CFTCP

(Constant Feed in Tool Centre Point)

The shape of the contour does not influence the feed rate at the tool centre.

Application:

The tool cuts at the whole diameter.

(e.g. roughing)

### CFIN

(Constant Feed at Internal radius)

The feed rate of the tool centre path will be reduced when the tool is inside a curve.

Outside curves do not increase the tool path feed rate (important for machining with fully invadated tool, end face finishing etc.).

### CFC

(Constant Feed at Contour)

Basic setting.

Constant feed at the curve.

The feed rate of the tool centre path will be increased when the tool is outside a curve and reduced when the tool is inside a curve.

Application:

The tool cut only at the circumference.

(e.g. finishing)


## Command Description M Commands

### M00 Programmed Stop

This command effects a machining stop within a part program.


The milling spindle, feeds and coolant will be switched off.

The machine door can be opened without releasing an alarm.

With "NC START"  the program run can be continued. After that the main drive will be switched on with all values which were valid before.

### M01 Programmed Stop, Conditional

M01 works like M00, but only when the function "PROGRAMMED STOP YES" was switched on via softkey in the menu "PROGRAM control".

With "NC START"  the program run can be continued. After that the main drive will be switched on with all values which were valid before.

### M02 Main Program End

M02 works like M30.

### M03 Milling Spindle ON Clockwise

The spindle will be switched on provided that a cutting speed has been programmed, the machine doors are closed and a workpiece is correctly clamped. M03 must be used for all right hand cutting tools.

### M04 Milling Spindle On Counter clockwise

The same conditions as described under M03 apply here.

M04 must be used for all left hand cutting tools.

### M05 Milling Spindle OFF

The main drive is braked electrically.

At the program end the milling spindle is automatically switched off.

### M06 Tool Change

The by T previously selected tool will be swivelled in.

After that the main drive will be switched on with all values which were valid before.

M6 and T..D.. must not be in the same block.

### M08 Coolant ON

The coolant will be switched on.

### M09 Coolant OFF

The coolant will be switched off.

### M10 Dividing device clamping on

The dividing device is clamped.

### M11 Release clamping dividing device

Clamping of the dividing device is released.

### M17 Subprogramm End

M17 will be written in the last block of a subprogram. It can stand alone in this block or with other functions. The call-up of a subroutine and M17 must not stand in the same block (nesting).

**Caution:**

When M commands are programmed that can not be executed by the machine, the respective M command will be ignored and the program continues.

This can cause collisions (e.g. with missing workpiece manipulation).

**M25 Clamping device/ OPEN machine vice**

The clamping device and/or the machine vice is opened.

**M26 CLOSE clamping device/machine vice**

The clamping device and/or the machine vice is closed.

**M27 Swivel Dividing Head**

Only for accessory dividing head.

The dividing head will be swivelled for one step (step angle mechanically adjusted).

**M30 Main Program End**

With M30 all drives will be switched off and the control will be resetted to program start.

Furthermore the workpiece counter will be increased by 1.

**M70 Spindle positioning****M71 Puff Blowing ON**

only for accessory blow off device.

The blow off device will be switched on.

**M72 Puff Blowing OFF**

only for accessory blow off device.

The blow off device will be switched off.

## Free contour programming

The free contour programming is a support tool for the editor.

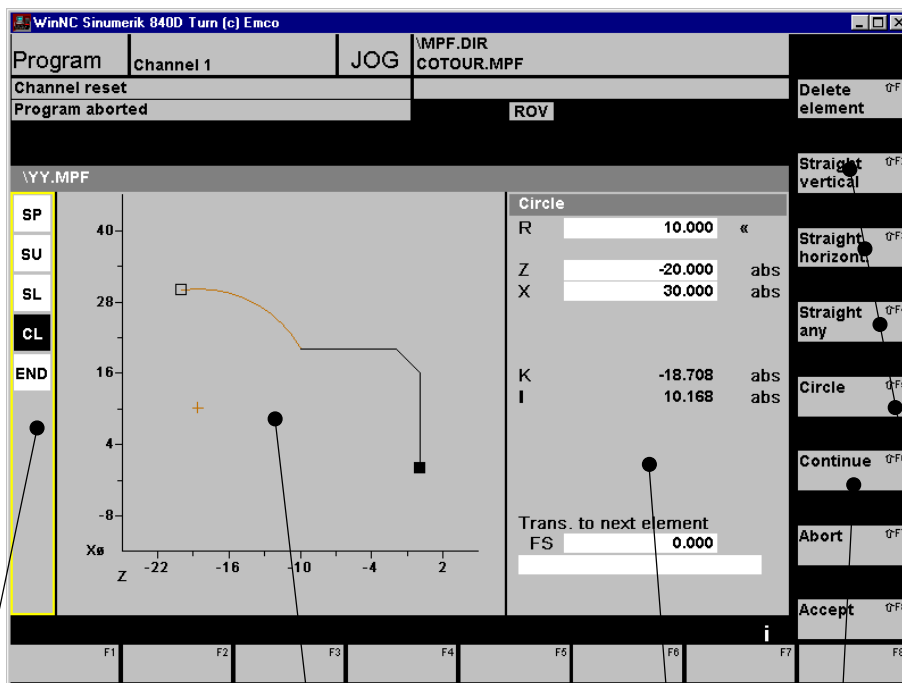
An integrated contour calculator calculates possibly missing parameters as soon as they result from other parameters.

Additionally, contour transition elements such as chamfer or radius are available.

The selection is carried out in the operation area Program .

You select an already existing program and/or open a new part program via the softkeys "Workpiece" and "Part programs"

You open the contour editor with the softkeys "Support" and "New contour" and/or "Recompile" .



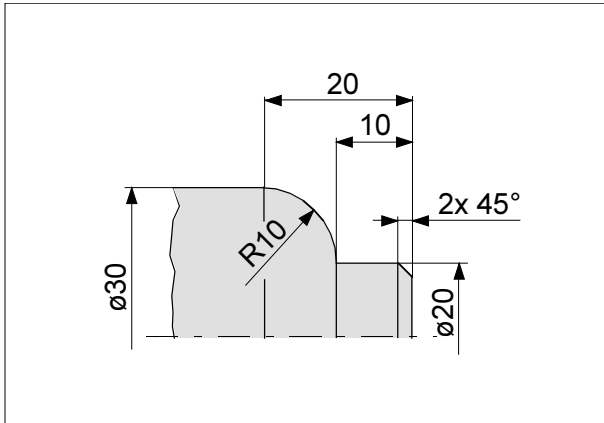
Programmed contour elements (can be opened again by double click)

Graphic representation of the programmed contour elements

Programming window

Contour elements

Further contour functions (close pole, contour)



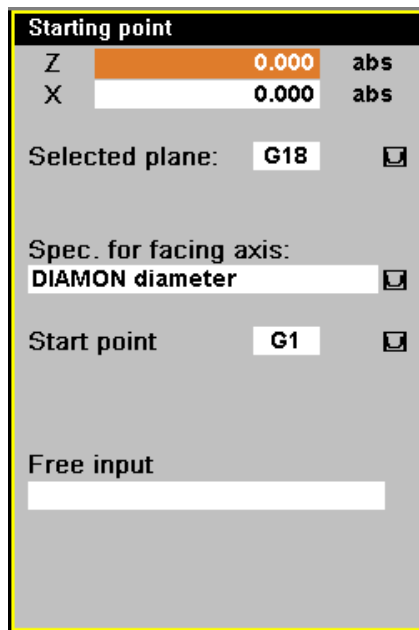
Programming example

**Contour programming:**

**Determine starting point:**

With the input of contours the starting point is determined at first.

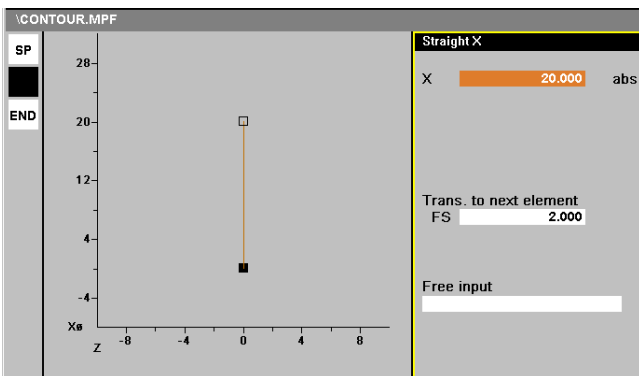
- The coordinates for X and Z must be programmed absolutely.
- Plane selection: G17 / G18 / G19
- Indication dimension facing axis
  - DIAMON (diameter)
  - DIAMOF (radius)
  - DIAM90 (diameter/radius)
- Start point (approach): G0 / G1



The programmed values are accepted in the editor with the softkey "accept element".

Further parameters for the contour definition are opened with the softkey "All parameters".

You can select between the selection options with the space bar or the softkey "Alternative".



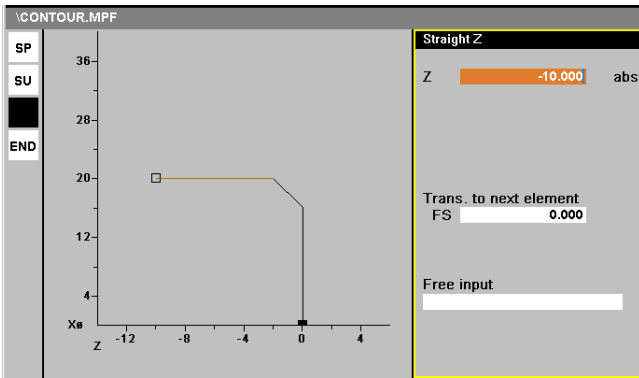
**Selection softkey "Straight vertical"**

End point X ..... 20.000  
 Chamfer/radius/undercut FS ..... 2.000

During the transition to the next element you have the possibility to select between a chamfer or a radius.

Contour description "Straight vertical"

The contour description is accepted in the editor with the softkey "accept element".

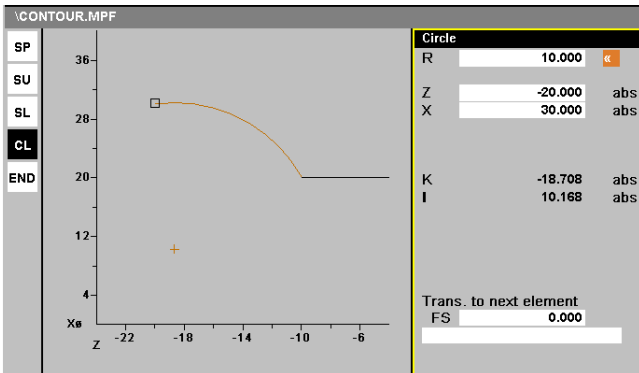


Contour description "Straight horizontal"

**Selection softkey "Straight horizontal"**

End point X..... -10.000  
 Transition to next element ..... 0.000

The contour description is accepted in the editor with the softkey "accept element".



Contour description "Circle"

**Selection softkey "Circle"**

radius ..... 10  
 End point Z ..... -20.000  
 End point X ..... 30  
 Transition to next element ..... 0.000

With complete indication of radius and end point the values for I and K are calculated automatically.

The right selection of the propositions can be made with the softkey "Dialog select". The selection is accepted with "Dialog accept". The contour description is accepted in the editor with the softkey "Accept element".

You change from free contour programming back to the editor with the softkey "Accept".

During retranslation only the contour elements created with free contour programming are generated again. Texts carried out subsequently in the program text get lost during this procedure.

A contour already existing can be modified by means of the softkey "Recompile". During this procedure the cursor of the editor must be positioned within the contour.

### Softkey "Straight any"

Any straight lines can be programmed with this function.

Any straight line is an oblique line in X or Z- direction the end point of which is programmed via a coordinate point or an angle.

Unless parameter input fields are programmed, the control presumes that those values are unknown and tries to calculate them from other parameters.

### Softkey "Continue"

You find the softkey "Pole" and "Close contour" under the softkey "Continue".

The softkey "Pole" is not active.

The softkey "Close contour" is used for closing a contour with one straight line.

## Differences contour processor EMCO – Siemens KP Version 1.0.5

### Graphic

- alternative solution is not indicated
- different scaling logic
- chamfers are only drawn between straight lines
- Undercuts are not drawn (are generally not implemented)

### Inputs

- 'tangential' is represented as  $a_2 = 0.000$
- Softkey 'Tangent prev. elem' is represented activated for elements with tangential transition
- 'Close contour' creates only one (any) and not two straight lines (vertical and horizontal)
- it is not possible to work with polar coordinates

### Contour elements

- Undercut is not implemented

### Contour processor:

- with coinciding solutions however, occasionally a selection dialogue appears
- occasionally an alternative solution (for starting point or end point) of a contour element cannot be selected. In this case a change to another contour element concerned (preceding, next element) is useful. Then the selection of the alternative solution can be carried out there.

### Created code

- constant coordinate values are not generally omitted in the code (only with horizontal and/or vertical straight lines)
- selected multiple solutions cannot be retranslated from a Siemens code (the first solution is displayed)
- code with selected multiple solutions cannot be retranslated on an original control
- the text from the free input is deposited in an EMCO specific field but does not create an NC-code
- chamfers between straight line and circle and/or circle and circle create a different code

### Error messages

- „Chamfer/radius too large“  
The value for the transition with chamfer or radius is too large.

Remedy: select smaller transition value



- „Inconsistent geometry values!“  
Due to the last input a contradiction for the contour element was noted.  
Examples:
  - Circle end points outside of circles
  - Intersection point at infinity
  - Tangents of points within a circleRemedy: correct the last input
- „Illegal function!“  
The last input or selection is not admissible at the moment.  
Examples:
  - Transitions to following element with radii with elements with tangential connection to the previous element
  - Cut-ins as transition (are generally not implemented)
  - Transitions at indefinite contour elementsRemedy: select a valid transition to the following element
- „Make dialog selection first!“  
You want to make an input while waiting for the selection of a solution.  
Remedy: First make a selection, then further inputs can be carried out.

**Help**

- Help images are in a modal window, therefore the contour processor cannot be operated while a help image is displayed
- Not all help images of the original control are implemented



## E: Tool Correction / Tool Measuring

### Tool Correction

#### Tool programming

T...: Tool number in magazine  
D...: Tool correction number

To every tool number T up to 9 correction numbers D can be assigned.

The control SINUMERIK 810D/840D describes the correction data D as edge, but that does not mean the single edges (teeth) of the tool, but the correction data that are assigned to this tool.

A tool can have also several correction numbers (e.g. a plane miller can be used as chamfer miller, on the same tool two different points are measured). Depending on usage in the program this tool is called e.g. with T1 D1 or T1 D2.

The command T..D.. calls up the tool correction data D, but the tool will not be changed in yet. The data for tool correction will be read from the tool data register (tool length, tool radius, ...).

Possible tool numbers:  
T 1..32000, D 1..9

#### Tool change

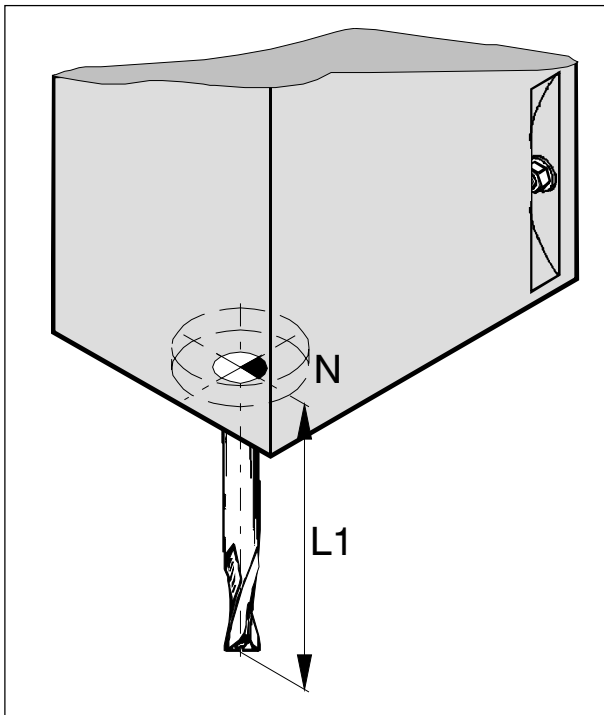
M6: Change tool

The command M6 does automatically all necessary movements for tool change.

To avoid collisions the tool must be lifted previously (traverse free).

#### Example

```
N50 G0 X200 Y120 Z80
      Approach tool change position
N55 T4 D1
      Call tool and correction
N60 M6
      Change in tool
N65 ...
```



#### Tool length correction

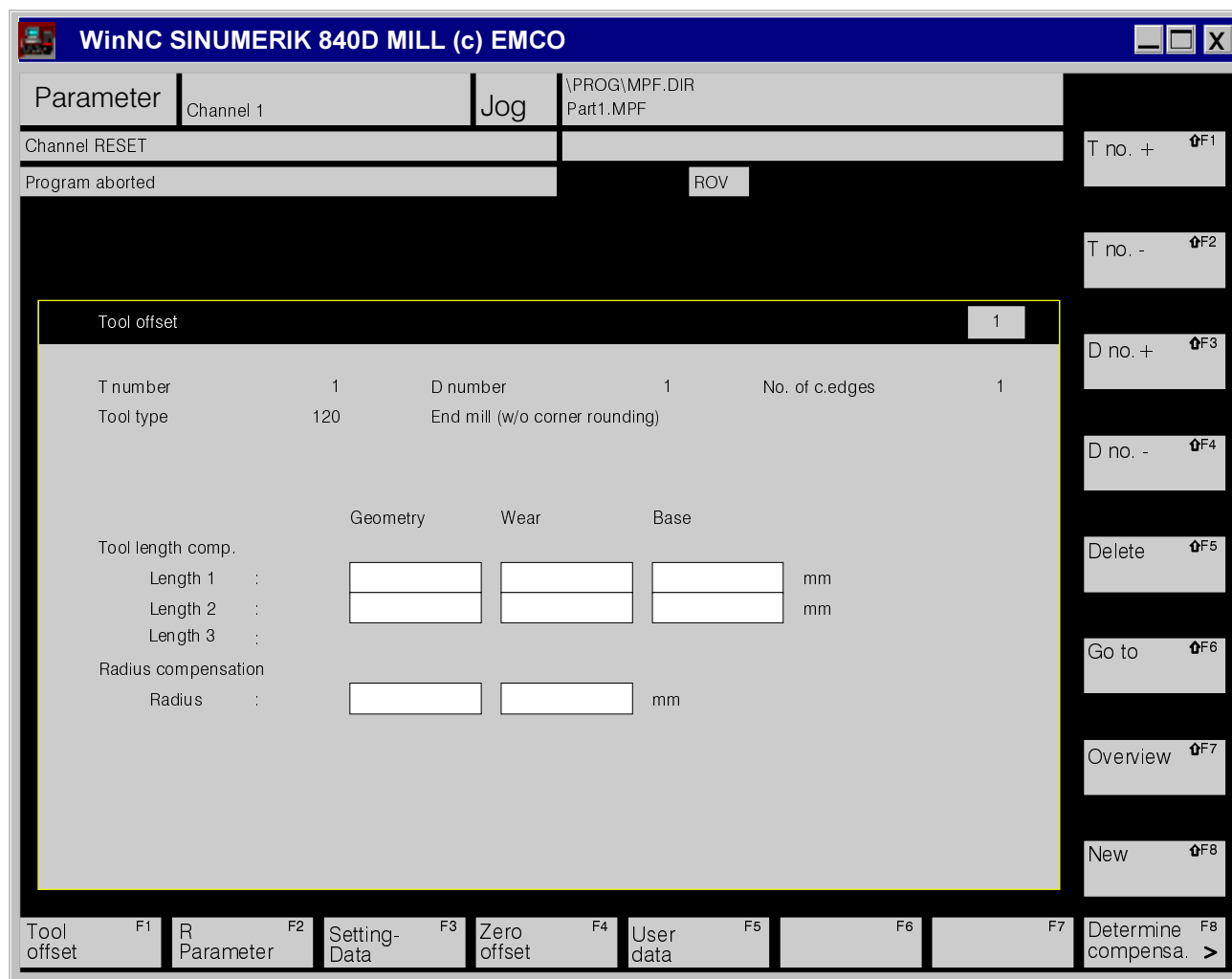
The tool length correction is effective vertical to the working plane (G17-G19).

Main application for vertical milling: G17 - tool length correction in Z

The tool length correction shifts the tool zero from the tool mount reference point N to the tool tip.

By that all positions are in relation to the tool tip.

On most milling machines the tool mount reference point is in the center of the face of the spindle nose.



**T number**

With this number the tool will be called up (position number in tool turret)

**D number**

Number of the tool correction. A tool can have also several correction numbers (e.g. a plane miller can be used as chamfer miller, on the same tool two different points are measured).

**No. of c. edges**

Number of D numbers for the tool (but not tooth number etc.).

**Tool type**

This number determines the kind of tool.

**Geometry**

Dimensions of the tool

**Wear**

Deviation from the geometry value.

**Base**

Dimension of a tool holder (e.g. gear head), in which the tools will be clamped.

The sum of geometry, wear and base is the total effective tool correction.

Softkeys:

**T no +, T no -**

Switches to the next higher or lower tool number.

**D no +, D no -**

Switches in the tool to the next higher or lower tool correction number.

**Delete**

Delete a tool from the list or delete a correction of the actual tool.

Press the softkey DELETE. The vertical softkey line shows the softkeys DELETE CUT. EDGE, DELETE TOOL and ABORT.

**Delete tool**

The actual tool and all its edges (corrections D) will be deleted.

**Delete cutting edge**

Always the cutting edge with the highest D number will be deleted.

The D numbers must be continuous without gap, e.g. a tool with four edges must have D1, D2, D3, D4 and only D4 can be deleted.

D1 can not be deleted, in this case the whole tool must be deleted (a tool must have at least one edge).

**Abort**

Exit without deleting.

**Go to**

Direct selection of tool.

Press the softkey GO TO. The vertical softkey list shows the selection softkeys and beside a input window.

**Preselected tool**

The number selected in a CNC program (while or after program run).

**Active tool**

The tool that is swivelled in in the tool turret.

**Input field**

Here you can enter the requested T and D number

and take over with .

**OK**

Changes to the requested tool.

**Abort**

Exit without tool selection.

**Overviews**

Display of tool list.

Place the cursor on the requested tool and take over with softkey "OK".


**New**

Establish a new tool or a new correction (edge).

**New tool edge**

A correction data set will be added to an existing tool.

Enter the T number to which the new edge should be added (the actual tool is suggested) and the tool type of the new edge.


Confirm your input with .

The softkey OK establishes the new edge, ABORT leaves without new edge.

**New tool**

A new tool will be added to the list.

Enter the T number and tool type of the new tool.

Confirm your input with .

The softkey OK establishes the new tool, ABORT leaves without new tool.

**Determine compensation**

Automatic tool data measuring, see chapter "Tool measuring".

**Direction of length correction**

The direction of compensation depends on the active plane G17 - G19.

G17 (main application for vertical milling):

Length 1 in Z

Radius in XY plane

This manual describes the usual case G17 if not indicated different.

G18 (main application for horizontal milling or working with angular gear head):

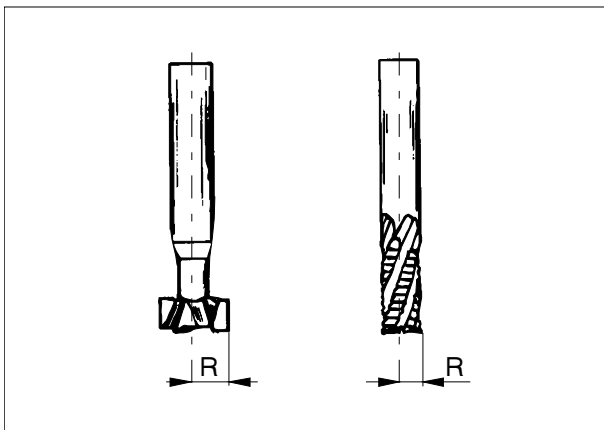
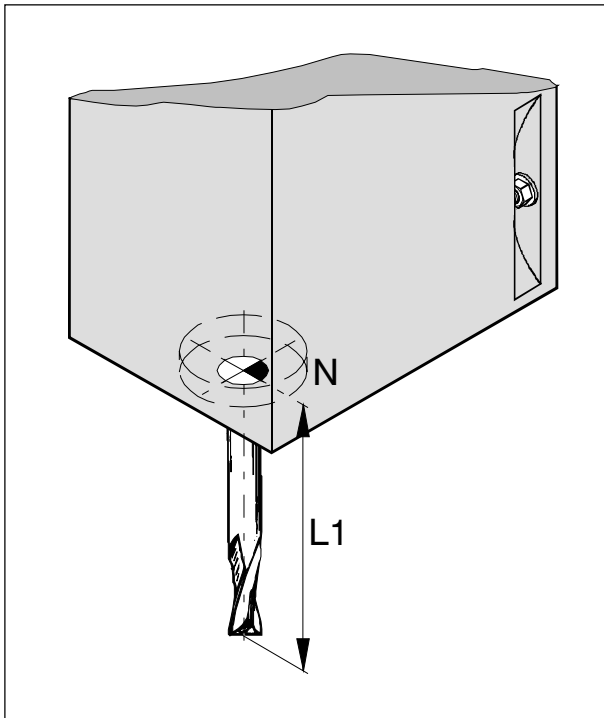
Length 1 in X

Radius in ZX plane

G19 (working with angular gear head):

Length 1 in Y

Radius in YZ plane

**Required correction values for milling and drilling tools**

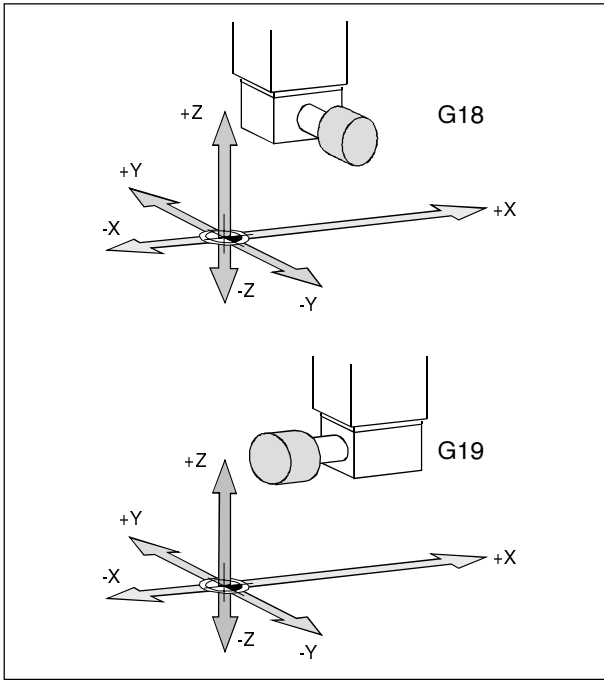
For milling tools you must indicate:

- Length 1
- Radius

For drilling tools enter radius = 0, because drills do not cut at the circumference.

In the fields length and length 3 the value 0 must be entered.

All other inputs will be ignored by the control



**Required correction values for angular gear head tools**

For angular gear head tools the plane G18 or G19 is used.

**G18**

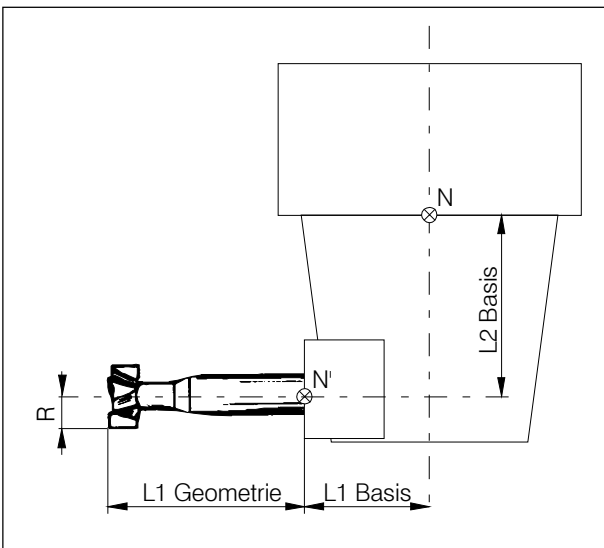
The tool axis is parallel to the Y machine axis (vertical to machine front).

- Length 1 in Y
- Length 2 in Z
- (Length 3 in X)
- Radius in X/Z

**G19**

The tool axis is parallel to the X machine axis (parallel to machine front).

- Length 1 in X
- Length 2 in Z
- (Length 3 in Y)
- Radius in Y/Z



For angular gear head tools it is recommended to set the dimensions of the angular gear head (always the same) as base value and the dimensions of the milling tool as geometry value.

By that the tool length L1 (geomtry) can be measured on a tool measuring device and N' is used as tool mount reference point.

The base values indicate the distance from N to N'.

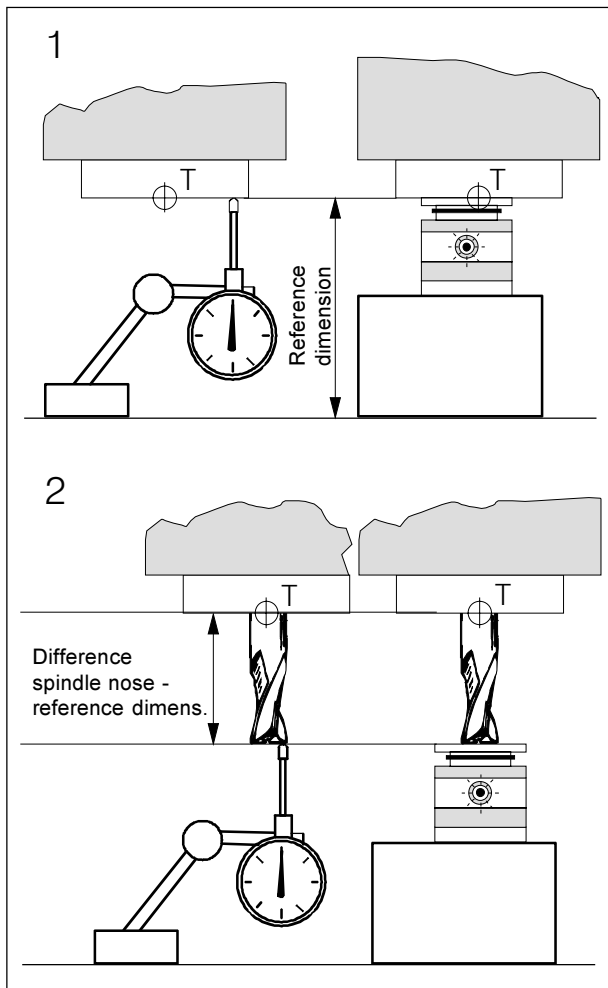
**Angular gear head with side offset**

When tool axis and main spindle axis have no intersection point, the tool has a side offset.

This side offset must be entered additionally as base value Length 3 in X (for G18) or in Y (for G19).

For angular gear heads without side offset the value 0 must be entered for Length 3.

## Tool Measuring



- 1 Mount measuring cell or metering clockwork
- 2 Select JOG mode
- 3 Traverse spindle nose onto measuring cell / metering clockwork (1), set metering clockwork to 0.
- 4 Call up tool data register  
Operating Area Parameter - Tool offset
- 5 Press softkey DETERMINE COMPENSA.
- 6 In the field "Reference" set the axis on Z.
- 7 Take over spindle nose position with OK  
The reference value (position of spindle nose) is displayed under Length 1. Enter actual position under Ref. value.
- 8 Enter Length 1 in the field "Reference dimensions" as "reference value".
- 9 Change in the tool to be measured.
- 10 Traverse tool tip onto measuring cell / metering clockwork. Positioning cursor on length 1 (control T- number)
- 11 Press "determin.compensa". Chose axis Z and press OK. Tool length are saved under L1.
- 12 Select next T number and repeat from step 9 until all tools are measured.

### Enter tool radius

- A Direct measuring (caliper gauge) and manual writing into tool data register
- B Use the sequence above analogous for the tool radius:  
Touch direction in X or Y  
at 6: Set axis on X or Y (depending on touch direction)



## F: Program Run

### Preconditions

#### Zero offsets G54-G57

The used zero offsets must be measured and entered.

#### Tools

The used tools must be measured and entered.  
The tools must be at the corresponding positions (T) in the tool change system.

#### Reference point

The reference point must be approached in all axes.

#### Machine

The machine must be ready for operation.  
The workpiece must be clamped safe.  
Loose parts (clamping keys etc.) must be removed from the working area to avoid collisions.  
The machine door must be closed for program run.

#### Alarms

No alarms must be active.

## Program Selection

### Program overview

Operating Area Machine, Automatic mode.  
Press the softkey PROGRAM OVERVIEW.

With the horizontal softkeys the programs of the specified type (workpieces, part programs, subprograms, standard cycles, user cycles, clipboard) can be displayed.

### Program enable

To work off a program it must be enabled previous.



- Operating Area Machine, Automatic mode
- Press the softkey PROGRAM OVERVIEW
- Mark the desired program or workpiece with the

keys  .

- You can enable / disable the selected program / workpiece with the softkey ALTER ENABLE.
- Enable is displayed with an (X) in the list:  
(X) enabled  
( ) disabled
- When a program is part of a workpiece (directory), the program and the workpiece must be enabled.

### Select program for working off

Select program

- Operating Area Machine, Automatic mode
- Press the softkey PROGRAM OVERVIEW
- Mark the program with the keys  .
- Press the softkey PROGRAM SELECTION.
- The program name is displayed right in the headline.

Select workpiece

- In the workpiece overview select a workpiece with

the keys  .

- Press the softkey WORKPIECE SELECTION.
- If there exists a part program with the same name in this directory, it will be selected for machining automatically (e.g. with selection of the workpiece PART1.WPD the part program PART1.MPF will be selected automatically).
- The program name and the workpiece information is displayed right in the headline.
- If there exists a initialisation file with the same name in this directory, it will be executed immediately with selection of the part program (e.g. PART1.INI).


## Program Start, Program Stop

Select a program for machining.

Change into Operating Area Machine, Automatic mode.

Press the key  for program start.

Stop program with , continue with .


Abort program with .

### Messages while program run


3 Stop: Emergency stop active  
The EMERGENCY OFF button was pressed.

4 Stop: Alarm active with stop  
An alarm stopped the program.

5 Stop: M0/M1 active  
Programmed stop of program run.


Continue with .

6 Stop: Block ended in SBL mode  
A block was finished in single block mode.

Continue with .

7 Stop: NC Stop active

The program was stopped with the key .

Continue with .

8 Wait: Read-in enable missing  
Read-in enable is a signal from the machine to the control.  
The actual block has not been worked off yet (e.g. tool change, dividing device, bar loader, etc.).  
The next program block will be worked off only after the previous was finished.

9 Wait: Feedrate enable missing  
Feed enable is a signal from the machine to the control.  
The actual block has not been worked off yet (e.g. spindle did not yet reach the programmed speed, etc.).  
The next program block will be worked off only after the previous was finished.

10 Wait: Dwell time active  
Program run was stopped for the programmed dwell time.


17 Wait: Feedrate override to 0%  
The feed override switch is on 0% position.

18 Stop: NC block incorrect  
Programming error


21 Wait: Block search active  
While block search all blocks before the search target will be simulated internally first and at the search target machining will start.

## Program Control

Press the softkey PROGRAM CONTROL.

Select the desired function with the cursor keys .

and .

Activate / deactivate the function with the key .

### SKIP Skip block

When this function is active, all blocks that are signed with a slash before the block number (/N...) will not be executed while program run.


### DRY Dry run feedrate

For test run without workpiece (no machining). All blocks with a programmed feedrate (G1, G2, G3, G33, ...) will traverse with the predefined dry run feed instead of the programmed feedrate. The spindle does not run.

### ROV Rapid traverse override

### SBL1 Single block with stop after machine function blocks

Program run will be stopped after every movement.

Continue with .


### SBL2 Single block with stop after every block

Program run will be stopped after every block, also when no movement is programmed in the block (calculating block).

### M01 Programmed stop

With M01 in the program the program normally will not stop at this command.

When this function is active, the program stops at M01.

Continue .

### DRF DRF selection

Additional incremental zero offset with the electronic handwheel.

### PRT Program test

Program test without axis movement.

## Block Search

Block search allows to run a program forward until a required block and then start machining. Two types of block search are available.

### 1. With calculation at the contour

While block search the same calculations are preceded as with a normal program run (the program is simulated internally).




At the block start of the search block that machine status will be established, that would be active also with normal program run. Afterwards the search block will be worked off like a normal program run block.

### 2. With calculation at the block end point

While block search the same calculations are preceded as with a normal program run (the program is simulated internally).

At the block end of the search block that machine status will be established, that would be active also with normal program run. The block end of the search block will be approached direct, the search block itself will not be worked off.

Sequence:

- Operating Area Machine, AUTO mode is selected.
- The program for block search is selected.
- The control is in RESET status.
- Press the softkey BLOCK SEARCH.
- Place the cursor on the search block.
- Block search will be started with the softkey CALCULATE CONTOUR or CALCULATE BLK ENDPT.
- The control calculates all blocks until search target but executes no movements.
-  aborts block search.
-  starts the program run. The screen shows a security query. Confirm it with .
- The position for the search target will be approached with a compensation movement and from that time on the program runs automatically.



## G: Flexible NC- Programming

### Variable and arithmetic parameters

Variables can be used instead of fixed values to increase the flexibility of a program. You can respond to signals such as measured values or, by storing setpoints in the variables, you can use the same program for different geometries.

A skilled programmer can use variable calculation and program jumps to create a highly flexible program archive which will considerably reduce the programming work required.

### Variable types

- User defined variables
- Arithmetic parameters
- System variables

### Variable types

#### INT

Integers with leading sign

Value range:  $\pm(2^{31} - 1)$

#### REAL

Fractions with decimal point

Value range:  $\pm(10^{-300} \dots 10^{+300})$

#### BOOL

Boolean values: TRUE (1) and FALSE (0)

Value range: 1, 0

#### CHAR

1 ASCII character specified by the code

Value range: 0 ... 255

#### STRING

Character string, number of character in [...], maximum 200 Characters

Value range: Folge von Werten mit 0 ... 255

#### AXIS

Axis names (Axis addresses) only

Value range: All axis identifiers and spindles of a channel.

#### FRAME

Geometrical parameters for translation, rotation, scaling, mirroring.

### System variable

Variables provided by the control which can be processed in the program. System variables provide access to zero offsets, tool offset, actual values, measured values on the axes, control states, etc. System variables return values of the defined type. Some of the system variables cannot be assigned values.

The name of a system variable always identified by the "\$" character followed by the specific names.

### Overview of the system variable types

1st letter	Meaning
\$M	Machine data
\$S	Setting data
\$T	Tool management data
\$P	Programmed values
\$A	Current values
\$V	Service data
2nd letter	Meaning
N	NCK- global
C	Channel- specific
A	Axis- specific

Example:

\$AA\_IM ..... Current axis specific value in the machine coordinate system.

## Variable definition

### User defined variables

In addition to the predefined variables, the programmer can also define his own variables and assign values to them.

Local variables are only valid in the program in which they are defined.

Global variables apply in all programs.

#### Variable name

A variable name consists of up to 32 characters. The first two characters must be a letter or an underscore. The "\$" character cannot be used for user defined variables, as it is reserved for system variables.

Format:

DEF INT name  
or DEF INT name=Value

DEF REAL name  
or DEF REAL name1, name2=3, name4  
or DEF REAL name [array index1, array index2]

DEF BOOL name

DEF CHAR name  
or DEF CHAR name [array index]=("A","B",...)

DEF STRING [string length] name

DEF AXIS name  
or DEF AXIS name [array index]

DEF FRAME name

### Example

Variable type INT	
DEF INT NUMBER	A variable type integer is created with the name NUMBER. The system initializes the variable with zero.
DEF INT NUMBER=7	A variable type integer is created with the name NUMBER. The system initializes the variable with the value 7.
Variable type REAL	
DEF REAL DEPTH	A variable type Real is created with the name DEPTH. The system initializes the variable with zero.
DEF REAL DEPTH=6,25	A variable type Real is created with the name DEPTH. The initial value is 6,25.
Variable type BOOL	
DEF BOOL IF_TOOMUCH	A variable of type Bool is created with the name IF_TOOMUCH. The system initializes the variable with zero (FALSE)
DEF BOOL IF_TOOMUCH=1 DRF BOOL IF_TOOMUCH=TRUE DEF BOOL IF_TOOMUCH=FALSE	A variable of type Bool is created with the name IF_TOOMUCH.
Variable type CHAR	
DEF CHAR GUSTAV_1=65	You can assign a code for the ASCII character to the variable of type Char or assign the ASCII character directly (65 is the code for the letter A)
DEF CHAR GUSTAV_1="A"	
Variable type STRING	
DEF STRING [6]SAMPLE_1="START"	Variables of type String can store a string of characters. The maximum number of characters is enclosed in square brackets after the variable type.
Variable type AXIS	
DEF AXIS AXISNAME=(X1)	The variables of type Axis have the name Axisname and contain the axis identifier of a channel, here X1

If a value is not assigned to a variable when it is defined, the system initializes it with zero. Variables must be defined at the beginning of the program before use. The definition must be made in a separate block. Only one variable type can be defined per block.

A variable of type AXIS stores names and spindle identifiers of a channel. Axis names with extended addresses must be enclosed in parentheses.



### Array definition

Programming

```

DEF CHAR NAME[n,m]
DEF INT NAME[n,m]
DEF REAL NAME[n,m]
DEF AXIS NAME[n,m]
DEF FRAME NAME[n,m]
DEF STRING[string length] NAME[m]
DEF BOOL [n,m]
    
```

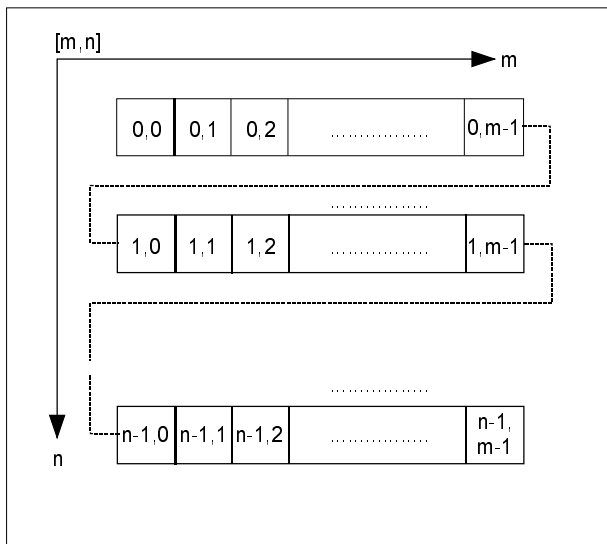
INT NAME[n,m]      Variable type (CHAR, INT, REAL, AXIS, FRAME, BOOL)

DEF STRING[string length] NAME[m]  
 The data type STRING can only be defined with one dimensional arrays

NAME                      Variable name

Arrays with a maximum of 2 dimensions can be defined.

Arrays with STRING variables may only be one dimensional. The string length is specified after the data type String.



### Array index

The elements of an array can be accessed via the array index.

The array elements can either be read or assigned values using this array index.

The first array elements begins with the index [0,0]. With an array size of [3,4], for example, the maximum array index is [2,3].

In the marginal example, the initialization values match the index of the array element in order to illustrate the order of the individual array elements.

### Initialization of arrays

Initialization values can be assigned to arrays elements during program execution or when arrays are defined.

The right hand array index is incremented first on two dimensional arrays.

**Initialization of value lists, SET****Options during array definition**

```
DEF Typ VARIABLE=SET(Value)
DEF Typ ARRAY[n,m]=SET(Value,Value,...)
oder
DEF Typ VARIABLE=Value
DEF Typ ARRAY[n,m]=(Value,Value,...)
```

- The number of array elements assigned corresponds to the number of initialization values programmed.
- Array elements without values are automatically assigned the value "0".
- There may be no gaps in the value list for variables of the AXIS type.
- If more values are programmed than remaining array elements exist, the system trigger an alarm.

**Options during program execution**

```
ARRAY[n,m]=SET(Value, Value,...)
ARRAY[n,m]=SET(Expression, Expression,...)
```

- Field elements are initialized as described above for array definition.
- Expressions may also be used here as initialization values.
- Initialization starts at the programmed array indices. Values can also be assigned selectiely to subarrays.

**Example**

```
Assignment of expressions
DEF INT ARRAY[5,5]
ARRAY[0,0]=SET(1,2,3,4,5)
ARRAY[2,3]=SET(Variable,4*5.6)
```

The axis index is not processed for axis variables.

**Example**

```
Initialization on one line
$MA_AX_VELO_LIMIT[1,AX1]=SET(1.1,2.2,3.3)
```

Corresponds to:

```
$MA_AX_VELO_LIMIT[1,AX1]=1.1
$MA_AX_VELO_LIMIT[2,AX1]=2.2
$MA_AX_VELO_LIMIT[3,AX1]=3.3
```

**Initialization with identical values, REP****Options during array definition**

```
DEF Typ ARRAY[n,m]=REP(Value)
```

All array elements are assigned the same value (constant).



Variables of type FRAME cannot be initialized.


**Example**

```
DEF REAL ARRAY5[10,3]=REP(9.9)
```

**Options during program execution**

```
ARRAY[n,m]=REP(value)
ARRAY[n,m]=REP(expression)
```

- Expressions may also be used here as initialization values.
- All array elements are initialized with the same value.
- Initilization starts at the programmed array indices. Values can also be assigned selectively to subarrays.



Variables of the FRAME type are permitted and can be initialized very simple using this method.

**Example**

Initialization of all elements with one value.

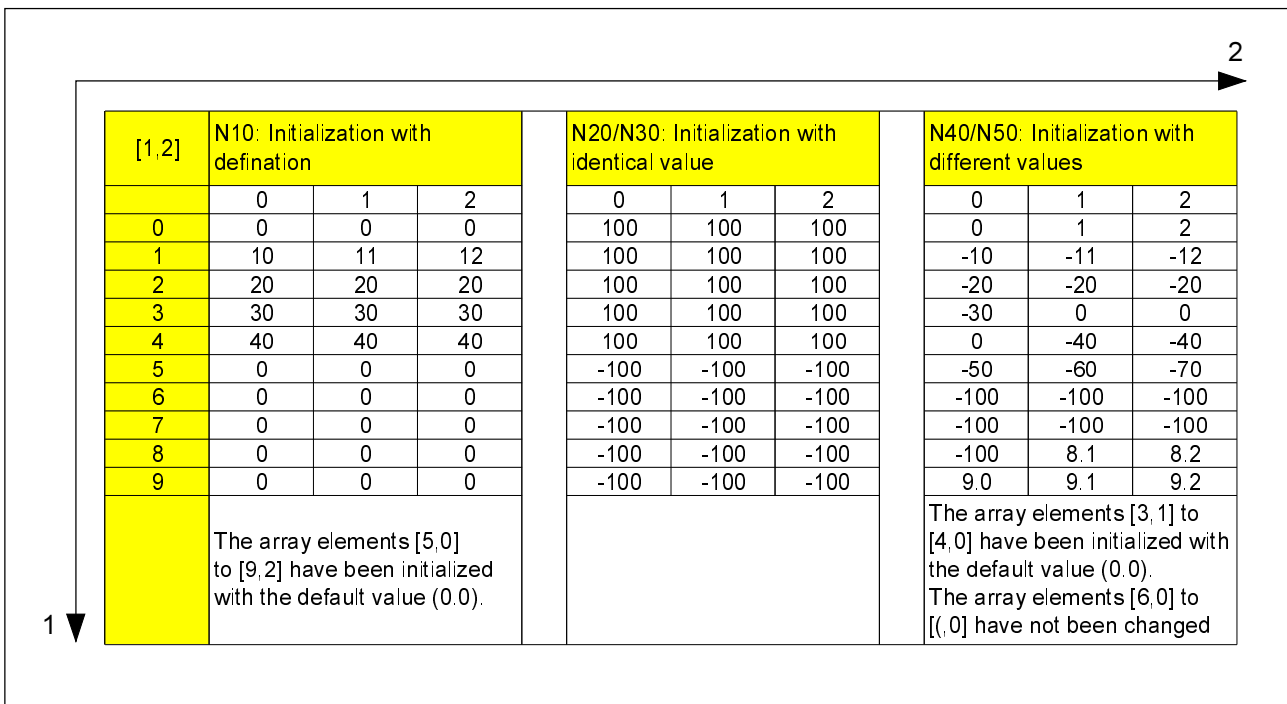
```
DEF FRAME FRM[10]
FRM[5]=REP(CTTRANS(X,5))
```

Example

Initialization of complete variable arrays.  
 The drawing shows the current assignment.

```

N10 DEF REAL ARRAY1 [10, 3] = SET(0, 0, 0, 10, 11, 12, 20, 20, 20, 30, 30, 30, 40, 40, 40, )
N20 ARRAY1 [0,0] = REP (100)
N30 ARRAY1 [5,0] = REP (-100)
N40 ARRAY1 [0,0] = SET (0, 1, 2, -10, -11, -12, -20, -20, -30, , , , -40, -40, -50, -60, -70)
N50 ARRAY1 [8,1] 0 SET (8.1, 8.2, 9.0, 9.1, 9.2)
    
```



## Indirect programming

Indirect programming enables programs to be used universally. The extended address (index) is substituted by a variable of suitable type.

All addresses can be configured, except for::

- N- block number
- G- G command
- L- subprogram

Indirect programming is not possible for any settable addresses. (X[1] is not permitted instead of X1).

Example

S1=300                      Direct programming

```
DEF INT SPINU=1
S[SPINU]=300
```

Indirect programming:  
Speed 300rpm for the spindle  
whose number is stored in the  
variable SPINU.

## Assignments

Values of matching types can be assigned to variables/arithmetic parameters in the program.

The assignment is always made in a separate block. Up to two assignments are possible per block.

Assignments to axis addresses always require a separate block to variable assignments.

Example

```
R1=10.518 R2=4 Vari1=45
X=47.11 Y=R2
```

```
R1=R3 VARI1=R4
```

Assignment of  
numeric value

```
R4=-R5 R7=-VARI8
```

Assignment of a variable  
of matching type  
Assignment of  
opposite leading sign  
(only allowed with  
types INT/REAL).

## Assignment to string variables

A distinction is made between upper and lower case characters within a CHAR or STRING.

Example

```
MSG("Finishing contour")
```

Displays the text "Finishing contour" on the screen.



## Comparison and logic operations

### Comparison operators

The comparison operators can be used for variables of type CHAR, INT, REAL and BOOL.

The code value is compared with the CHAR type.

The following are possible with types STRING, AXIS and FRAME: == und <>.

The result of a comparison operation is always type BOOL.

Comparison operations can be used, for example, to formulate a jump condition.

==	Equal to
<>	Not equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to
<<	Chaining of strings

#### Example

```
IF R10>=100 GOTOF DEST
```

oder

```
R11=R10>=100
```

```
IF R11 GOTOF DEST
```

The result of the comparison R10>=100 is first buffered in R11.

### Logic operators

Logic operators are used to logically combine truth values. AND, OR, NOT and XOR can generally only be used on variables of type BOOL, however, they can also be used on the data types CHAR, INT and REAL by means of implicit type conversion.

Spaces must be inserted between Boolean operands and operators.

In logic (Boolean) operations the following applies to the data types BOOL, CHAR, INT and REAL:

0 is equivalent to FALSE

not equal to 0 is equivalent to TRUE

AND	AND
OR	OR
NOT	NOT
XOR	Exklusiv OR

Parentheses can be used in arithmetic expressions to define the order of execution for all operators and thus to override the normal priority rules.

```
IF (R10<50) AND ($AA_IM[X])>=17.5) GOTOF DEST
IF NOT R10 GOTOB START
```

### Bit operators

Bit for bit logic operations can also be performed on variables of the type CHAR and INT. Type conversion takes place automatically.

B_AND	Bit AND
B_OR	Bit OR
B_NOT	Bit NOT
B_XOR	Bit exclusive OR

The operator B\_NOT refers only to an operand; this follows the operator.

#### Example

```
IF $MC_RESET_MODE_MASK B_AND 'B10000'
GOTOF ACT_PLANE
```

**Priority of operators**

Each operator is assigned a priority. When an expression is evaluated, the operators with the highest priority are always applied first. Where operators have the same priority, the evaluation is from left to right.

Paratheses can be used in arithmetic expressions to define the order of execution for all operators and thus to override the normal priority rules.

**Priority of operators (highest to lowest)**

1. NOT, B_NOT	Negation, bit negation
2. *, /, DIV, MOD	Multiplication, division
3. +, -	Addition, subtraction
4. B_AND	Bit AND
5. B_XOR	Bit exclusive OR
6. B_OR	Bit OR
7. AND	AND
8. XOR	exclusive OR
9. OR	OR
10. <<	Chaining of strings, result type STRING
11. ==, <>, >, <, >=, <=	Comparison operators

The chain operator „:“ for frames may not appear with other operators in an expression. A priority level is this not required for this operator.

**Type conversion**

The constant numeric value, variable or expression assigned to a variable must be compatible with the type of this variable. If this is this case, the type is automatically converted when the value is assigned.

**Possible type conversion**

from	to REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
REAL	yes	yes*	yes <sup>1)</sup>	yes*	-	-	-
INT	yes	yes	yes <sup>1)</sup>	yes <sup>2)</sup>	-	-	-
BOOL	yes	yes	yes	yes	yes	-	-
CHAR	yes	yes	yes <sup>1)</sup>	yes	yes	-	-
STRING	-	-	yes <sup>4)</sup>	yes <sup>3)</sup>	yes	-	-
AXIS	-	-	-	-	-	yes	-
FRAME	-	-	-	-	-	-	yes

\* On type conversion from REAL to INT, a fraction >=0.5 is rounded up, otherwise this fraction is rounded down (same effect as ROUND function)

- 1) Values <> 0 are TRUE, Values == 0 are FALSE
- 2) If the value is in the permitted value range
- 3) If only 1 character
- 4) String runs 0 = >FALSE, otherwise TRUE

If a value is greater than the target range on conversion, an error message is generated.

If mixed types occur in an expression, a type conversion is performed automatically.

**Length of strings, STRLEN**

This functionality allows the length of a string to be specified.

Syntax:

INT_ERG = STRLEN (STRING)	Result type: INT
---------------------------	------------------

Semantics:

A number of characters is returned that - counting from the beginning of the string- are not 0 characters.

Example:

This function can be used to determine the end of the string, for example, in connection with the single character access described below:

```
IF(STRLEN(BAUSTEIN_NAME)>10)GOTOF FEHLER
```



## CASE statement

Format:

```
CASE (expression) OF constant1 GOTOF LABEL1 DEFAULT GOTOF LABELn
CASE (expression) OF constant1 GOTOB LABEL1 DEFAULT GOTOB LABELn
```

CASE	Vocabulary word for jump instruction
GOTOF	Jump instruction with jump destination forwards
GOTOB	Jump instruction with jump destination backwards
LABEL	Destination (label within the program)
LABEL:	The name of the jump destination is followed by a colon
Expression	arithmetic expression
Constant	Constant of type INT
DEFAULT	Program path if none of the previously named constants applies

The CASE statement enables various branches to be executed according to a value of type INT.

The program jumps to the point specified by the jump destination, depending on the value of the constant evaluated in the CASE statement.

In cases where the constant matches none of the predefined values, the DEFAULT instruction can be used to determine the jump destination.

If the DEFAULT instruction is not programmed, the jump destination is the block following the CASE statement.

```
CASE(expression) OF 1 GOTOF LABEL1 2 GOTOF LABEL2 ... DEFAULT
GOTOF LABELn
„1“ and „2“ are possible constants.
If the value of the expression = 1 (INT-constant), jump to block with LABEL1
If the value of the expression = 2 (INT-constant), jump to block with LABEL2
...
otherwise jump to the block with LABELn
```

Example

```
DEF INT VAR1 VAR2 VAR3
CASE(VAR1+VAR2-VAR3) OF 7 GOTOF LABEL1 9 GOTOF LABEL2 DEFAULT
GOTOF LABEL3
LABEL1: G0 X1 Y1
LABEL2: G0 X2 Y2
LABEL3: G0 X3 Y3
```

## Check structures

IF-ELSE-ENDIF .....	Selection between 2 alternatives
LOOP-ENDLOOP .....	Endless loop
FOR-ENDFOR .....	Count loop
WHILE-ENDWHILE .....	Loop with condition at beginning of loop
REPEAT-UNTIL .....	Loop with condition at end of loop

The control processes the NC blocks as standard in the programmed sequence.

In addition to the program branches described in this Section, these commands can be used to define additional alternatives and program loops.

These commands enable the user to produce wellstructured and easily lrgible programs.

### IF-ELSE-ENDIF

An IF-ELSE-Endif- block is used to select one of two alternatives:

```
IF (expression)
N50...
N60...
ELSE
N120...
ENDIF
```

If the value of the expression is TRUE, i.e. the condition is fulfilled, then the next program block is executed. If the condition is not fulfilled, then the ELSE program branch is executed. The ELSE branch can be omitted.

### Endless- Program loop, LOOP

Endless loops are used in endless programs. At the end of the loop, there is always a branch back to the beginning.

```
LOOP
N50...
N60...
ENDLOOP
```

### Count loop, FOR

The FOR loop is used if it is necessary to repeat an operation by a fixed number of runs. The variable must be of the INT type.

```
FOR Variable = start value TO endvalue
N50...
N60...
ENDFOR
```

**Program loop with condition at beginning of loop, WHILE**

The WHILE program loop is executed for as long as the condition is fulfilled.

```

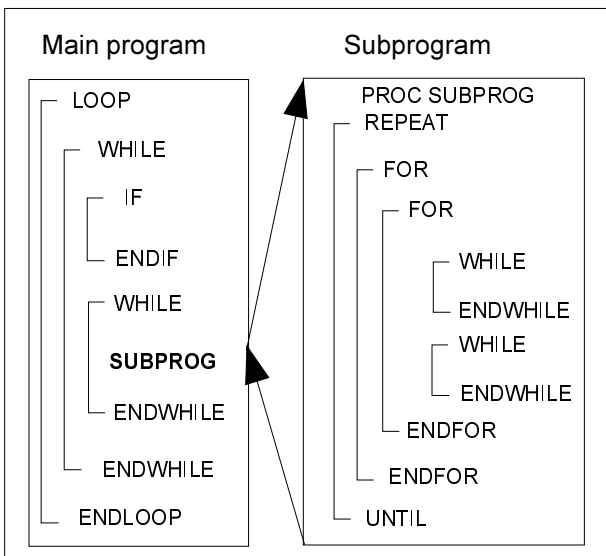
WHILE expression
N50...
N60...
ENDWHILE
    
```

**Program loop with condition at the end of loop, REPEAT**

The REPEAT loop is executed once and repeated continuously until the condition is fulfilled.

```

REPEAT
N50...
N60...
UNTIL(expression)
    
```



**Nesting depth**

Check structures apply locally within programs. A nesting depth of up to 8 check structures can be set up on each subprogram level

**Runtime response**

In interpreter mode (active as standard), it is possible to shorten program processing times more effectively by using program branches than can be obtained with check structures.

There is no difference between program branches and check structures in precompiled cycles.

### Supplementary conditions

Blocks with check structures elements cannot be suppressed. Labels may not be used in blocks of this type.

Check structures are processed interpretively. When a loop end is detected, a search is made for the loop beginning, allowing for the check structures found in the process.

For this reason, the block structures of a program is not checked completely in interpreter mode.

It is not generally advisable to use a mixture of check structures and program branches.

A check can be made to ensure that check structures are nested correctly when cycles are preprocessed.

Check structures may only be inserted in the statement section of a program. Definitions in the program header may not be executed conditionally or repeatedly.

It is not permissible to superimpose macros on vocabulary words for check structures or on branch destinations. No such check is made when the macro is defined.

#### Example

(Endless program)

```
%_N_LOOP_MPF
LOOP
  IF NOT $P_SEARCH ;no block search
    G01 G90 X0 Z10 F1000
    WHILE $AA_IM[X] <= 100
      G1 G91 X10 F500 ;Drilling pattern
      Z-5 F100
      Z5
    ENDWHILE
    Z10
  ELSE ; Block search
    MSG(„No drilling during block search“)
  ENDIF
  $A_OUT[1] = 1 ;next drilling plate
  G4 F2
ENDLOOP
M30
```

Example (Production of a fixed quantity of parts)

```
%_N_WKPCCOUNT_MPF
DEF INT WKPCCOUNT
FOR WKPCCOUNT =0 TO 100
G01 ...
ENDFOR
M30
```

**Suppress current block display, DISPLOF, DISPLON**

Format

PROC ... DISPLOF

With DISPLOF the current block display for a subprogram is suppressed. DISPLOF is placed at the end of the PROC statement.

Instead of the current block, the call of the cycle or the subprogram is displayed.

By default the block display is activated. Deactivation of block display with DISPLOF applies until the return from the subprogram or end of program. If further subprograms are called from the subprogram with the DISPLOF attribute, the current block display is suppressed in these as well. If a subprogram with suppressed block display is interrupted by an asynchronous subprogram, the blocks of the current subprogram are displayed.

**Single set suppression SBLOF, SBLON**

Format

PROC ... SBLOF  
PROC ... SBLON

SBLOF ..... Single set suppression OF  
SBLON ..... Single set suppression ON

**Single set suppression program specific**

With SBLOF qualified programs are worked out as a complete block.

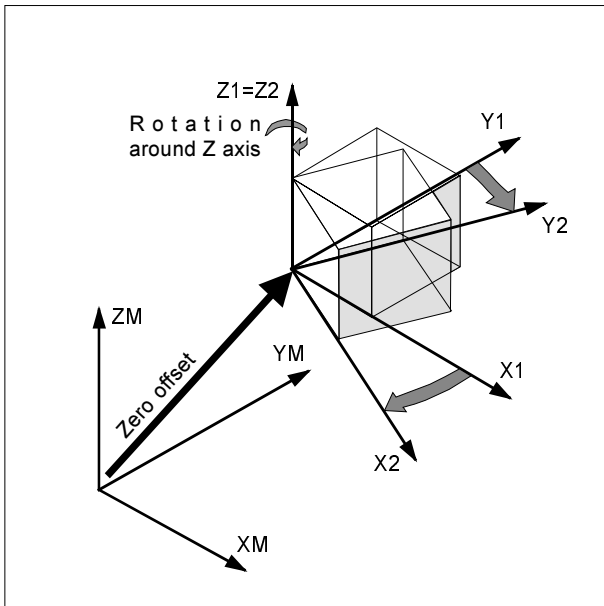
Example:

```
PROC Example SBLOF  
G1 X10  
RET
```

**Single set suppression at the program**

SBLOF can stand alone in the block. From this block single block is switched off until the next SBLON or until the end of the active subroutine level.

## Frames



Frame is the conventional term for a geometrical expression that describes an arithmetic rule, such as translation or rotation.

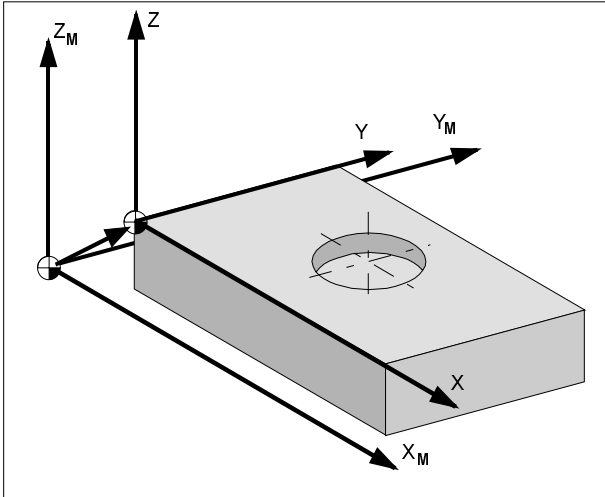
Frames are used to describe the position of a destination coordinate system by specifying coordinates or angles starting from the current workpiece coordinate system.

### Possible frames:

- **Basis frames (basis offset)**  
Describe the transformation of coordinates from the basis system of coordinates (BKS) into the zero point system (BOS) and have the same effect as adjustable frames.
- **adjustable frames (G54...G599)**  
Adjustable frames are zero point offsets to be called up from any NC program with the commands G54 to G599. The offset values are preset by the operator and stored in the zero point memory of the control.
- **programmable frames.**  
Programmable frames (TRANS, ROT,...) are valid in the actual NC-program and refer to the adjustable frames. They are used to determine the workpiece coordinate system.
- **additive frames**  
The actually adjusted workpiece zero point or the one programmed last via frames serves as reference.

### Frame variable/ frame relationship

Using the frames described above, arithmetic rules are specified to describe how the positions of the coordinate systems are related



Frame variable:

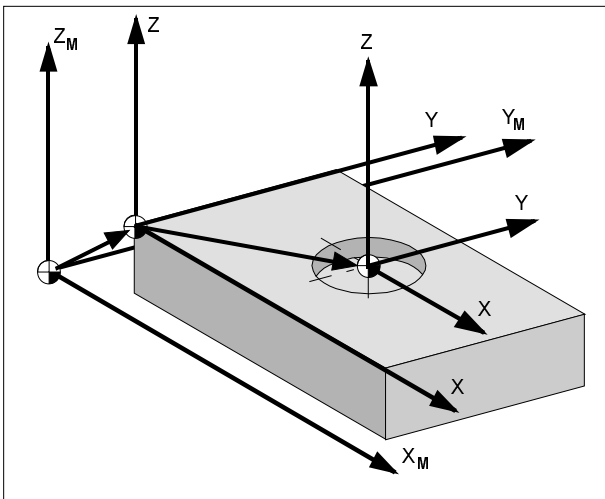
\$P\_PFRAME means: current programmable frame.

### Predefined frame variables

#### \$P\_IFRAME

Current settable frame variable that sets up the reference between the machine and workpiece coordinate systems.

\$P\_IFRAME contains the translation or rotation, etc., defined by G54.



#### \$P\_BFRAME

Actual basis frame variable which establishes the reference between the basis system of coordinates and the basis zero point system which can be defined by the operator.

#### \$P\_PFRAME

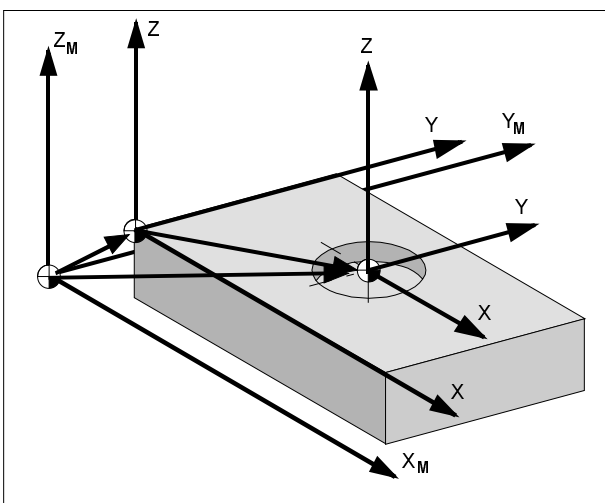
Actual programmable frame variable which establishes the reference between workpiece zero point system and the workpiece coordinate system.

\$P\_PFRAME contains the resulting frame, which results from programming TRANS/ATRANS, ROT/AROT, SCALE/ASCALE, MIRROR/AMIRROR to the programmable frame.

#### \$P\_ACTFRAME

Actual, resulting total frame consisting of the actual basis frame variable \$P\_BFRAME, the actual adjustable frame variable \$P\_IFRAME and the actual programmable frame variable \$P\_PFRAME.

\$P\_ACTFRAME describes the actually valid workpiece zero point.



**Predefined adjustable frames \$P\_UBFR[n]**

Writing on the predefined frame variable \$P\_UBFR[n] does not activate the basis frame at the same time but the activation is carried out only with the first execution of a G500, G54,... G599 instruction.

**Predefined adjustable frames \$P\_UIFR[n]**

The adjustable zero point offsets G54 until G599 can be read or written from the workpiece program by means of the predefined frame variable \$P\_UIFR[n].

5 adjustable frames (\$P\_UIFR[0] until \$P\_UIFR[4]) and/or 5 synonymous G commands - G500 (switch-off) and G54 until G57 - are preset as a standard .



A total of 100 systems of coordinates can be created by predefined adjustable frames, which can be programmed to overlap the program e. g. as zero point for various devices.

\$P\_UIFR[0] corresponds G500  
 \$P\_UIFR[1] corresponds G54  
 \$P\_UIFR[2] corresponds G55  
 \$P\_UIFR[3] corresponds G56  
 \$P\_UIFR[4] corresponds G57



**Axis function AXNAME, ISAXIS, AX**

AXNAME("TRANSVERSE AXIS")

AX[AXNAME("STRING")]

SPI(spindle number)

ISAXIS(geometry axis number)

AXNAME ... Converts an input string to an axis identifier.

AX ..... Variable axis identifier.

ISAXIS ..... Checks whether the specified geometry axis exists

**AXNAME**

Is used, for example, to create generally applicable cycles when the name of the axes are not known.

**ISAXIS**

Is used in universal cycles in order to ensure that a specific geometry axis exists and thus that any following \$P\_AXNX- call is not aborted with an error message.



# H: Alarms and Messages

## Machine Alarms 6000 - 7999

These alarms will be triggered by the machines. There are different alarms for the different machines.

The alarms 6000 - 6999 normally must be confirmed with RESET. The alarms 7000 - 7999 are messages which normally will disappear when the releasing situation is finished.

### PC MILL 50 / 55 / 100 / 105 / 125 / 155 Concept MILL 55 / 105 / 155

#### 6000: EMERGENCY OFF

The EMERGENCY OFF key was pressed. Remove the endangering situation and restart machine and software.

#### 6001: PLC-CYCLE TIME EXCEEDING

Contact EMCO Service.

#### 6002: PLC - NO PROGRAM CHARGED

Contact EMCO Service.

#### 6003: PLC - NO DATA UNIT

Contact EMCO Service.

#### 6004: PLC - RAM MEMORY FAILURE

Contact EMCO Service.

#### 6005: OVERHEAT BRAKEMODUL

Main drive was braked too often, large changes of speed within a short time. E4.2 active

#### 6006: OVERLOAD BRAKE RESISTOR

see 6005

#### 6007: SAFETY CIRCUIT FAULT

Axis and main drive contactor with machine switched off not disabled. Contactor got stuck or contact error. E4.7 was not active during switch-on.

#### 6009: SAFETY CIRCUIT FAULT

Defective step motor system.

A running CNC program will be interrupted, the auxiliary drives will be stopped, the reference position will be lost.

Contact EMCO Service.

#### 6010: DRIVE X-AXIS NOT READY

The step motor board is defective or too hot, a fuse or cabling is defective.

A running program will be stopped, the auxiliary drives will be switched off, the reference position will be lost.

Check fuses or contact EMCO service.

#### 6011: DRIVE Y-AXIS NOT READY

see alarm 6010.

#### 6012: DRIVE Z-AXIS NOT READY

see alarm 6010.

#### 6013: MAIN DRIVE NOT READY

Main drive power supply defective, main drive too hot, fuse defective.

A running program will be stopped, the auxiliary drives will be switched off.

Check fuses or contact EMCO Service.

#### 6014: NO MAIN SPINDLE SPEED

This will be released, when the spindle speed is lower than 20 rpm because of overload.

Alter cutting data (feed, infeed, spindle speed).

The CNC program will be aborted, the auxiliary drives will be stopped.

#### 6019: VICE TIME EXCEED

The electric vice has not reached a stop position within 30 seconds.

The control or the clamping device board are defective, the vice is stuck. Adjust the proximity switches of the stop position.

#### 6020: VICE FAILURE

When the electric vice is closed, the signal "clamping device clamped" of the clamping device board has failed.

The control, the clamping device board or the wiring are defective.

A 2007-05

**6022: CLAMPING DEVICE BOARD DEFECTIVE**

The signal "clamping device clamped" is constantly released, although no command has been given.  
Replace the board.

**6024: MACHINE DOOR OPEN**

The door was opened while a machine movement.  
The program will be aborted.

**6027: DOOR LIMIT SWITCH DEFECTIVE**

The limit switch of the automatic door is displaced, defective, wrong cabled.  
Contact EMCO service.

**6028: DOOR TIMEOUT**

The automatic door sticks, the pressured air supply is insufficient, the limit switch is displaced.  
Check door, pressured air supply, limit switch or contact EMCO service.

**6030: NO PART CLAMPED**

No workpiece inserted, vice cheek displaced, control cam displaced, hardware defective.  
Adjust or contact EMCO service.

**6040: TOOL TURRET INDEX FAILURE**

After WZW procedure drum pressed down by Z-axis. Spindle position wrong or mechanical defect. E4.3=0 in lower state

**6041: TOOL CHANGE TIMEOUT**

Tool drum sticks (collision?), main drive not ready, fuse defective, hardware defective.  
A running CNC program will be stopped.  
Check for collisions, check fuses or contact EMCO service.

**6043-6046: TOOL DISK POSITION FAULT**

Position error of main drive, error of position supervising (inductive proximity switch defective or disadjusted, drum allowance), fuse defective, hardware defective.

The Z axis could have been slipped out of the tothing while the machine was switched off.  
A running CNC program will be stopped.  
Contact EMCO service.

**6047: TOOL DISK UNLOCKED**

Tool drum turned out of locked position, inductive proximity switch defective or disadjusted, fuse defective, hardware defective.

A running CNC program will be interrupted.  
Contact EMCO service.

When the tool drum is turned out of locked position (no defect), act as following:

Turn the drum into locking position manually  
Change into MANUAL (JOG) mode.

Turn the key switch. Traverse the Z slide upwards, until the alarm disappears.

**6048: DIVIDING TIME EXCEEDED**

Dividing head sticks, insufficient pressured air supply, hardware defective.

Check for collision, check pressured air supply or contact EMCO service.

**6049: INTERLOCKING TIME EXCEEDED**

see alarm 6048

**6050: M25 AT RUNNING MAIN SPINDLE**

Cause: Programming mistake in NC program.

A running program will be aborted.

The auxilliary drives will be switched off.

Remedy: Correct NC program

**6064: DOOR AUTOMATIC NOT READY**

Cause: pressure failure automatic door  
automatic door sticks mechanically  
limit switch for open end position defective  
security print circuits defect  
cabling defective  
fuses defective

A running program will be aborted.

The auxilliary drives will be switched off.

Remedy: service automatic door

**6069: CLAMPING FOR TANI NOT OPEN**

When opening the clamping pressure switch does not fall within 400ms. Pressure switch defective or mechanical problem. E22.3

**6070: PRESSURE SWITCH FOR TANI MISSING**

When closing the clamping pressure switch does not respond. No compressed air or mechanical problem. E22.3

**6071: DIVIDING DEVICE NOT READY**

Servo Ready Signal from frequency converter missing. Excess temperature drive TANI or frequency converter not ready for operation.

**6072: VICE NOT READY**

Attempt to start the spindle with an open vice or without clamped workpiece.  
Vice sticks mechanically, insufficient compressed air supply, compressed air switch defective, fuse defective, hardware defective.  
Check the fuses or contact EMCO service.

**6073: DIVIDING DEVICE NOT READY**

Cause: locking switch defective  
cabling defective  
fuses defective

A running program will be aborted.

The auxiliary drives will be switched off.

Remedy: service automatic dividing device  
lock the dividing device

**6074: DIVIDING TIME EXCEEDED**

Cause: dividing device sticks mechanically  
locking switch defective  
cabling defective  
fuses defective  
insufficient compressed-air supply.

A running program will be aborted.

The auxiliary drives will be switched off.

Remedy: Check for collision, check the compressed-air supply or contact the EMCO service.

**6075: M27 AT RUNNING MAIN SPINDLE**

Cause: Programming mistake in NC program.  
A running program will be aborted.  
The auxiliary drives will be switched off.  
Remedy: Correct NC program

**7000: INVALID TOOL NUMBER PROGRAMMED**

The tool position was programmed larger than 10.  
The CNC program will be stopped.  
Interrupt program with RESET and correct the program.

**7001: NO M6 PROGRAMMED**

For an automatic tool change you also have to program a M6 after the T word.

**7007: FEED STOP!**

The axes have been stopped by the robotics interface (robotics entry FEEDHOLD).

**7016: SWITCH ON AUXILIARY DRIVES**

The auxiliary drives are off. Press the AUX ON key for at least 0.5 sec. (to avoid accidentally switching on) to switch on the auxiliary drives.

**7017: REFERENCE MACHINE**

Approach the reference point.  
When the reference point is not active, manual movements are possible only with key switch at position "setting operation".

**7018: TURN KEY SWITCH**

With NC-Start the key switch was in position "setting operation".  
NC-Start is locked.  
Turn the key switch in the position "automatic" to run a program.

**7020: SPECIAL OPERATION MODE ACTIVE**

Special operation mode: The machine door is opened, the auxiliary drives are switched on, the key switch is in position "setting operation" and the consent key is pressed.  
Manual traversing the axes is possible with open door. Swivelling the tool turret is not possible with open door. Running a CNC program is possible only with standing spindle (DRYRUN) and SINGLE block operation.  
For safety: If the consent key is pressed for more than 40 sec. the function of this key is interrupted, the consent key must be released and pressed again.

**7021: INITIALIZE TOOL TURRET**

The tool turret operating was interrupted.  
No traversing operation is possible.  
Press tool turret key in JOG operation. Message occurs after alarm 6040.

**7022: INITIALIZE TOOL TURRET !**

see 7021

**7023: WAITING TIME MAIN DRIVE!**

The LENZE frequency converter has to be separated from the mains supply for at least 20 seconds before you are allowed to switch it on again. This message will appear when the door is quickly opened/ closed (under 20 seconds).

**7038: LUBRICATION SYSTEM FAULT**

The pressure switch is defective or gagged.  
NC-Start is locked. This can be reset only by switching off and on the machine.  
Contact EMCO service.

**7039: LUBRICATION SYSTEM FAULT**

Not enough lubricant, the pressure switch is defective.  
NC-Start is locked.  
Check the lubricant and lubricate manually or contact EMCO service.

**7040: MACHINE DOOR OPEN**

The main drive can not be switched on and NC-Start can not be activated (except special operation mode)

Close the machine to run a program.

**7042: INITIALIZE MACHINE DOOR**

Every movement and NC-Start are locked.

Open and close the machine door to initialize the safety circuits.

**7043: PIECE COUNT REACHED**

A predetermined number of program runs was reached. NC-Start is locked. Reset the counter to continue.

**7050: NO PART CLAMPED**

After switching on or after an the vice is neither at the open position nor at the closed position.

NC-Start is locked.

Traverse the vice manually on a valid end position.

**7051: DIVIDING HEAD NOT LOCKED!**

Either the dividing head is in an undefined position after the machine has been switched on, or the locking signal after a dividing process is missing. Initiate the dividing process, check, respectively adjust the proximity switch for locking.

**7054: VICE OPEN**

Cause: the workpiece is not clamped

When switching on the main spindle with M3/M4 alarm 6072 (vice not ready) will be released.

Remedy: Clamp

**7055: OPEN TOOL CLAMPING SYSTEM**

A tool is clamped in the main spindle and the control does not recognize the corresponding T number.

Eject the tool from the main spindle when the door is open by means of the PC keys "Strg" and " 1 ".

**7056: SETTING DATA INCORRECT**

An invalid tool number is stored in the setting data.

Delete the setting data in the machine directory xxxxx.pls.

**7057: TOOLHOLDER OCCUPIED**

The clamped tool cannot be positioned in the tool turret since the position is occupied.

Eject the tool from the main spindle when the door is open by means of the PC keys "Strg" and " 1 ".

**7058: RETRACTING THE AXES**

The position of the tool turret arm cannot be clearly defined during the tool change.

Open the machine door, push the tool turret magazine backwards to the stop. Move the milling head in the JOG mode upwards to the Z reference switch and then traverse the reference point.

**7270: OFFSET COMPENSATION ACTIVE !**

Only with PC-MILL 105

Offset compensation activated by the following operation sequence.

- Reference point not active
- Machine in reference mode
- Key switch in manual operation
- Press STRG (or CTRL) and simultaneously 4

This must be carried out if prior to the tool change procedure spindle positioning is not completed (tolerance window too large)

**7271: COMPENSATION FINISHED, DATA SAVED !**

see 7270



**PC TURN 50 / 55 / 105 / 120 / 125 / 155**  
**Concept TURN 55 / 105 / 155****6000: EMERGENCY OFF**

The EMERGENCY OFF key was pressed.  
The reference position will be lost, the auxiliary drives will be switched off.  
Remove the endangering situation and restart machine and software.

**6001: PLC-CYCLE TIME EXCEEDING**

The auxiliary drives will be switched off.  
Contact EMCO Service.

**6002: PLC - NO PROGRAM CHARGED**

The auxiliary drives will be switched off.  
Contact EMCO Service.

**6003: PLC - NO DATA UNIT**

The auxiliary drives will be switched off.  
Contact EMCO Service.

**6004: PLC - RAM MEMORY FAILURE**

The auxiliary drives will be switched off.  
Contact EMCO Service.

**6008: MISSING CAN SUBSCRIBER**

The SPS-CAN board is not identified by the control.  
Check the interface cable and the power supply of the CAN board.

**6009: SAFETY CIRCUIT FAULT**

Defective step motor system.  
A running CNC program will be interrupted, the auxiliary drives will be stopped, the reference position will be lost.  
Contact EMCO Service.

**6010: DRIVE X-AXIS NOT READY**

The step motor board is defective or too hot, a fuse is defective, over- or undervoltage from mains.  
A running program will be stopped, the auxiliary drives will be switched off, the reference position will be lost.  
Check fuses or contact EMCO service.

**6012: DRIVE Z-AXIS NOT READY**

see 6010.

**6013: MAIN DRIVE NOT READY**

Main drive power supply defective or main drive too hot, fuse defective, over- or undervoltage from mains.  
A running program will be stopped, the auxiliary drives will be switched off.  
Check fuses or contact EMCO Service.

**6014: NO MAIN SPINDLE SPEED**

This alarm will be released, when the spindle speed is lower than 20 rpm because of overload.  
Alter cutting data (feed, infeed, spindle speed).  
The CNC program will be aborted, the auxiliary drives will be switched off.

**6015: NO DRIVEN TOOL SPINDLE SPEED**

see 6014.

**6016: AUTOMATIC TOOL TURRET SIGNAL COUPLED MISSING****6017: AUTOMATIC TOOL TURRET SIGNAL UNCOUPLED MISSING**

In the tool turret that can be coupled, the position of the coupling and uncoupling magnet is monitored by means of two proximity switches. It has to be made sure that the coupling is in the rear stop position so that the tool turret can get to the next tool position. Equally, during operation with driven tools the coupling has to be safe in the front stop position.  
Check and adjust the cables, the magnet and the stop position proximity switches.

**6021: COLLET TIME OUT**

During closing of the clamping device the pressure switch has not reacted within one second.

**6022: CLAMPING DEVICE BOARD DEFECTIVE**

The signal "clamping device clamped" is constantly released, even though no command has been given. Replace the board.

**6023: COLLET PRESSURE MONITORING**

The pressure switch turns off when the clamping device is closed (compressed air failure for more than 500ms).



**6024: MACHINE DOOR OPEN**

The door was opened while a machine movement.  
The program will be aborted.

**6025: GEARBOX COVER NOT CLOSED**

The gearbox cover was opened while a machine movement. A running CNC program will be aborted.

Close the cover to continue.

**6027: DOOR LIMIT SWITCH DEFECTIVE**

The limit switch of the automatic door is displaced, defective, wrong cabled.

Contact EMCO service.

**6028: DOOR TIMEOUT**

The automatic door sticks, the pressured air supply is insufficient, the limit switch is displaced. Check door, pressured air supply, limit switch or contact EMCO service.

**6029: TAILSTOCK QUILL TIME EXCEED**

The tailstock quill does not reach a final position within 10 seconds.

Adjust the control and the stop position proximity switches, or the tailstock quill is stuck.

**6030: NO PART CLAMPED**

No workpiece inserted, vice cheek displaced, control cam displaced, hardware defective.

Adjust or contact EMCO service.

**6031: QUILL FAILURE****6032: TOOL CHANGE TIMEOUT**

see alarm 6041.

**6033: TOOL TURRET SYNC ERROR**

Hardware defective.

Contact EMCO service.

**6037: CHUCK TIMEOUT**

The pressure switch does not react within one second when the clamping device is closed.

**6039: CHUCK PRESSURE FAILURE**

The pressure switch turns off when the clamping device is closed (compressed air failure for more than 500ms).

**6040: TOOL TURRET INDEX FAILURE**

The tool turret is in no locked position, tool turret sensor board defective, cabling defective, fuse defective.

A running CNC program will be stopped.

Swivel the tool turret with the tool turret key, check fuses or contact EMCO service.

**6041: TOOL CHANGE TIMEOUT**

Tool drum sticks (collision?), fuse defective, hardware defective.

A running CNC program will be stopped.

Check for collisions, check fuses or contact EMCO service.

**6042: TOOL TURRET OVERHEAT**

Tool turret motor too hot.

With the tool turret a max. of 14 swivel procedures a minute may be carried out.

**6043: TOOL CHANGE TIMEOUT**

Tool drum sticks (collision?), fuse defective, hardware defective.

A running CNC program will be stopped.

Check for collisions, check fuses or contact EMCO service.

**6045: TOOL TURRET SYNC MISSING**

Hardware defective.

Contact EMCO service.

**6046: TOOL TURRET ENCODER FAULT**

Fuse defective, hardware defective.

Check fuses or contact EMCO service.

**6048: CHUCK NOT READY**

Attempt to start the spindle with open chuck or without clamped workpiece.

Chuck sticks mechanically, insufficient pressured air supply, fuse defective, hardware defective.

Check fuses or contact EMCO service.

**6049: COLLET NOT READY**

see 6048

**6050: M25 DURING SPINDLE ROTATION**

With M25 the main spindle must stand still (consider run-out time, evtl. program a dwell)

**6055: NO PART CLAMPED**

This alarm occurs when with rotating spindle the clamping device or the tailstock reach the end position.

The workpiece has been pushed out of the chuck or has been pushed into the chuck by the tailstock.

Check clamping device settings, clamping forces, alter cutting data.

**6056: QUILL NOT READY**

Attempt to start the spindle or to move an axis or to swivel the tool turret with undefined tailstock position.

Tailstock is locked mechanically (collision), insufficient pressured air supply, fuse defective, magnetic switch defective.

Check for collisions, check fuses or contact EMCO service.

**6057: M20/M21 DURING SPINDLE ROTATION**

With M20/M21 the main spindle must stand still (consider run-out time, evtl. program a dwell)

**6058: M25/M26 DURING QUILL FORWARD**

To actuate the clamping device in an NC program with M25 or M26 the tailstock must be in back end position.

**6059: C-AXIS SWING IN TIMEOUT**

C-axis does not swivel in within 4 seconds.  
Reason: not sufficient air pressure, and/or mechanics stuck.

**6060: C-AXIS INDEX FAILURE**

When swivelling in the C-axis the limit switch does not respond.  
Check pneumatics, mechanics and limit switch.

**6064: AUTOMATIC DOOR NOT READY**

Door sticks mechanically (collision), insufficient pressured air supply, limit switch defective, fuse defective.  
Check for collisions, check fuses or contact EMCO service.

**6065: LOADER MAGAZINE FAILURE**

Loader not ready.  
Check if the loader is switched on, correctly connected and ready for operation and/or disable loader (WinConfig).

**6066: CLAMPING DEVICE FAILURE**

No compressed air at the clamping device  
Check pneumatics and position of the clamping device proximity detectors.

**6067: NO COMPRESSED AIR**

Turn the compressed air on, check the setting of the pressure switch.

**7000: INVALID TOOL NUMBER PROGRAMMED**

The tool position was programmed larger than 8.  
The CNC program will be stopped.  
Interrupt program with RESET and correct the program.

**7007: FEED HOLD**

In the robotic mode a HIGH signal is at input E3.7.  
Feed Stop is active until a low signal is at E3.7.

**7016: SWITCH ON AUXILIARY DRIVES**

The auxiliary drives are off. Press the AUX ON key for at least 0.5 sec. (to avoid accidentally switching on) to switch on the auxiliary drives (also a lubricating pulse will be released).

**7017: REFERENCE MACHINE**

Approach the reference point.  
When the reference point is not active, manual movements are possible only with key switch at position "setting operation".

**7018: TURN KEY SWITCH**

With NC-Start the key switch was in position "setting operation".  
NC-Start is locked.  
Turn the key switch in the position "automatic" to run a program.

**7019: PNEUMATIC LUBRICATION MONITORING!**

Refill pneumatic oil

**7020: SPECIAL OPERATION MODE ACTIVE**

Special operation mode: The machine door is opened, the auxiliary drives are switched on, the key switch is in position "setting operation" and the consent key is pressed.  
Manual traversing the axes is possible with open door. Swivelling the tool turret is possible with open door. Running a CNC program is possible only with standing spindle (DRYRUN) and SINGLE block operation.  
For safety: If the consent key is pressed for more than 40 sec. the function of this key is interrupted, the consent key must be released and pressed again.

**7021: TOOL TURRET NOT LOCKED**

The tool turret operating was interrupted.  
NC start and spindle start are locked. Press the tool turret key in the RESET status of the control.

**7022: COLLECTION DEVICE MONITORING**

Time exceed of the swivelling movement.  
Check the pneumatics, respectively whether the mechanical system is jammed (possibly a workpiece is jammed).

**7023: ADJUST PRESSURE SWITCH !**

During opening and closing of the clamping device the pressure switch has to turn off and on once.  
Adjust the pressure switch. This alarm does not exist any more for versions starting with PLC 3.10.

**7024: ADJUST CLAMPING DEVICE PROXIMITY SWITCH !**

When the clamping device is open and the position stop control is active, the respective proximity switch has to feed back that the clamping device is "Open".  
Check and adjust the clamping device proximity switch, check the cables.

**7025 WAITING TIME MAIN DRIVE !**

The LENZE frequency converter has to be separated from the mains supply for at least 20 seconds before you are allowed to switch it on again. This message will appear when the door is quickly opened/ closed (under 20 seconds).

**7038: LUBRICATION SYSTEM FAULT**

The pressure switch is defective or gagged. NC-Start is locked. This alarm can be reset only by switching off and on the machine. Contact EMCO service.

**7039: LUBRICATION SYSTEM FAULT**

Not enough lubricant, the pressure switch is defective. NC-Start is locked. Check the lubricant and lubricate manually or contact EMCO service.

**7040: MACHINE DOOR OPEN**

The main drive can not be switched on and NC-Start can not be activated (except special operation mode)  
Close the machine to run a program.

**7041: GEARBOX COVER OPEN**

The main spindle cannot be switched on and NC start cannot be activated.  
Close the gearbox cover in order to start a CNC program.

**7042: INITIALIZE MACHINE DOOR**

Every movement and NC-Start are locked.  
Open and close the machine door to initialize the safety circuits.

**7043: PIECE COUNT REACHED**

A predetermined number of program runs was reached. NC-Start is locked. Reset the counter to continue.

**7048: CHUCK OPEN**

This message shows that the chuck is open. It will disappear if a workpiece will be clamped.

**7049: CHUCK - NO PART CLAMPED**

No part is clamped, the spindle can not be switched on.

**7050: COLLET OPEN**

This message shows that the collet is open. It will disappear if a workpiece will be clamped.

**7051: COLLET - NO PART CLAMPED**

No part is clamped, the spindle can not be switched on.

**7052: QUILL IN UNDEFINED POSITION**

The tailstock is in no defined position. All axis movements, the spindle and the tool turret are locked.  
Drive the tailstock in back end position or clamp a workpiece with the tailstock.

**7053: QUILL - NO PART CLAMPED**

The tailstock reached the front end position. Traverse the tailstock back to the back end position to continue.

**7054: NO PART CLAMPED**

No part clamped, switch-on of the spindle is locked.

**7055: CLAMPING DEVICE OPEN**

This message indicates that the clamping device is not in clamping state. It disappears as soon as a part is clamped.



## AC95 / ACC ALARMS

### Axis Controller Alarms

#### 8000 Fatal Error AC

#### 8100 Fatal init error AC

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

#### 8101 Fatal init error AC

see 8101.

#### 8102 Fatal init error AC

see 8101.

#### 8103 Fatal init error AC

see 8101.

#### 8104 Fatal system error AC

see 8101.

#### 8105 Fatal init error AC

see 8101.

#### 8106 No PC-COM card found

Cause: PC-COM board can not be accessed (ev. not mounted).

Remedy: Mount board, adjust other address with jumper

#### 8107 PC-COM card not working

see 8106.

#### 8108 Fatal error on PC-COM card

see 8106.

#### 8109 Fatal error on PC-COM card

see 8106.

#### 8110 PC-COM init message missing

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

#### 8111 Wrong configuration of PC-COM

see 8110.

#### 8113 Invalid data (pccom.hex)

see 8110.

#### 8114 Programming error on PC-COM

see 8110.

#### 8115 PC-COM packet acknowledge missing

see 8110.

#### 8116 PC-COM startup error

see 8110.

#### 8117 Fatal init data error (pccom.hex)

see 8110.

#### 8118 Fatal init error AC

see 8110, ev. insufficient RAM memory

#### 8119 PC interrupt no. not valid

Cause: The PC interrupt number can not be used.

Remedy: Find out free interrupt number in the Windows95 system control (allowed: 5,7,10,11,12,3,4 und 5) and enter this number in WinConfig.

#### 8120 PC interrupt no. unmaskable

see 8119

#### 8121 Invalid command to PC-COM

Cause: Internal error or defective cable

Remedy: Check cables (screw it); Restart software or reinstall when necessary, report to EMCO, if repeatable.

#### 8122 Internal AC mailbox overrun

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

#### 8123 Open error on record file

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

#### 8124 Write error on record file

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

#### 8125 Invalid memory for record buffer

Cause: Insufficient RAM, record time exceeding.

Remedy: Restart software, ev. remove drivers etc. to gain more RAM, reduce record time.

#### 8126 AC Interpolation overrun

Cause: Ev. insufficient computer performance.

Remedy: Set a longer interrupt time in WinConfig. This may result in poorer path accuracy.

#### 8127 Insufficient memory

Cause: Insufficient RAM

Remedy: Close other programs, restart software, ev. remove drivers etc. to gain more RAM.

#### 8128 Invalid message to AC

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

#### 8129 Invalid MSD data - axisconfig.

see 8128.

#### 8130 Internal init error AC

see 8128.

#### 8130 Internal init error AC

see 8128.

#### 8132 Axis accessed by multiple channels

see 8128.

**8133 Insufficient NC block memory AC**

see 8128.

**8134 Too much center points programmed**

see 8128.

**8135 No centerpoint programmed**

see 8128.

**8136 Circle radius too small**

see 8128.

**8137 Invalid for Helix specified**

Cause: Wrong axis for helix. The combination of linear and circular axes does not match.

Remedy: Program correction.

**8140 Maschine (ACIF) not responding**

Cause: Machine off or not connected.

Remedy: Switch on machine or connect.

**8141 Internal PC-COM error**

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

**8142 ACIF Program error**

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

**8143 ACIF packet acknowledge missing**

see 8142.

**8144 ACIF startup error**

see 8142.

**8145 Fatal init data error (acif.hex)**

see 8142.

**8146 Multiple request for axis**

see 8142.

**8147 Invalid PC-COM state (DPRAM)**

see 8142.

**8148 Invalid PC-COM command (CNo)**

see 8142.

**8149 Invalid PC-COM command (Len)**

see 8142.

**8150 Fatal ACIF error**

see 8142.

**8151 AC Init Error (missing RPG file)**

see 8142.

**8152 AC Init Error (RPG file format)**

see 8142.

**8153 FPGA program timeout on ACIF**

see 8142.

**8154 Invalid Command to PC-COM**

see 8142.

**8155 Invalid FPGA packet acknowledge**

see 8142 or hardware error on ACIF board (contact EMCO Service).

**8156 Sync within 1.5 revol. not found**

see 8142 or Bero hardware error (contact EMCO Service).

**8157 Data record done**

see 8142.

**8158 Bero width too large (referencing)**

see 8142 or Bero hardware error (contact EMCO Service).

**8159 Function not implemented**

Bedeutung: In normal operation this function can not be executed

**8160 Axis synchronization lost axis 3..7**

Cause: Axis spins or slide is locked, axis synchronisation was lost

Remedy: Approach reference point

**8161 X-Axis synchronization lost**

Step loss of the step motor. Causes:

- Axis mechanically blocked
- Axis belt defective
- Distance of proximity detector too large (>0,3mm) or proximity detector defective
- Step motor defective

**8162 Y-Axis synchronization lost**

see 8161

**8163 Z-Axis synchronization lost**

see 8161

**8164 Software limit switch max axis 3..7**

Cause: Axis is at traverse area end

Remedy: Retract axis

**8168 Software limit overtravel axis 3..7**

Cause: Axis is at traverse area end

Remedy: Retract axis

**8172 Communication error to machine**

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable. Check connection PC - machine, eventually eliminate distortion sources.

**8173 INC while NC program is running**

Remedy: Stop the program with NC stop or with Reset. Traverse the axis.

**8174 INC not allowed**

Cause: At the moment the axis is in motion.

Remedy: Wait until the axis stops and then traverse the axis.

**8175 MSD file could not be opened**

Cause: Internal error

Remedy: Restart software oder bei Bedarf neu installieren, report to EMCO, if repeatable.

**8176 PLS file could not be opened**

see 8175.

**8177 PLS file could not be accessed**

see 8175.

**8178 PLS file could not be written**

see 8175.

**8179 ACS file could not be opened**

see 8175.

**8180 ACS file could not be accessed**

see 8175.

**8181 ACS file could not be written**

see 8175.

**8183 Gear too high**

Cause: The selected gear step is not allowed at the machine.

**8184 Invalid interpolaton command****8185 Forbidden MSD data change**

see 8175.

**8186 MSD file could not be opened**

see 8175.

**8187 PLC program error**

see 8175.

**8188 Gear command invalid**

see 8175.

**8189 Invalid channel assignement**

see 8175.

**8190 Invalid channel within message**

see 8175.

**8191 Invalid jog feed unit**

Cause: The machine does not support the rotation feed in the JOG operating mode.

Remedy: Order a software update from EMCO.

**8192 Invalid axis in command**

see 8175.

**8193 Fatal PLC error**

see 8175.

**8194 Thread without length**

Cause: The programmed target coordinates are identical to the starting coordinates.

Remedy: Correct the target coordinates.

**8195 No thread slope in leading axis**

Remedy: Program thread pitch

**8196 Too many axis for thread**

Remedy: Program max. 2 axes for thread.

**8197 Thread not long enough**

Cause: Thread length too short.

With transition from one thread to the other the length of the second thread must be sufficient to produce a correct thread.

Remedy: Longer second thread or replace it by a linear interpolation (G1).

**8198 Internal error (to many threads)**

see 8175.

**8199 Internal error (thread state)**

Cause: Internal error

Remedy: Restart software or reinstall when necessary, report to EMCO, if repeatable.

**8200 Thread without spindle on**

Remedy: Switch on spindle

**8201 Internal thread error (IPO)**

see 8199.

**8201 Internal thread error (IPO)**

see 8199.

**8203 Fatal AC error (0-ptr IPO)**

see 8199.

**8204 Fatal init error: PLC/IPO running**

see 8199.

**8205 PLC Runtime exceeded**

Cause: Insufficient computer performance

**8206 Invalid PLC M-group initialisation**

see 8199.

**8207 Invalid PLC machine data**

see 8199.

**8208 Invalid application message**

see 8199.

**8212 Rotation axis not allowed**

see 8199.

**8213 Circle and rotation axis can't be interpolated****8214 Thread and rotation axis cant't be interpolated****8215 Invalid state**

see 8199.

**8216 No rotation axis for rotation axis switch**

see 8199.

**8217 Axis type not valid!**

Cause: Switching during the rotary axis operating mode when the spindle is running.

Remedy: Stop the spindle and switch over to the rotary axis operating mode.

**8218 Referencing round axis without selected round axis!**

see 8199.

**8219 Thread not allowed without spindle encoder!**

Cause: Thread cutting, respectively tapping is only possible with spindles with encoders.

**8220 Buffer length exceeded in PC send message!**

see 8199.

**8221 Spindle release although axis is no spindle!**

see 8199.

**8222 New master spindle is not valid**

Cause: The indicated master spindle is not valid when switching over to the master spindle.

Remedy: Correct the spindle number.

**8224 Invalid stop mode**

see 8199.

**8225 Invalid parameter for BC\_MOVE\_TO\_IO!**

Cause: The machine is not configured for touch probes. A traversing movement with rotary axis is not allowed during touch probe operating mode.

Remedy: Remove the rotary axis movement from the traversing movement.

**8226 Rotary axis switch not valid (MSD data)!**

Cause: The indicated spindle does not have a rotary axis.

**8228 Rotary axis switch not allowed while axis move!**

Cause: The rotary axis has moved during switching over to the spindle operating mode.

Remedy: Stop the rotary axis before switching.

**8229 Spindle on not allowed while rotary axis is active!****8230 Program start not allowed due to active spindle rotation axis!****8231 Axis configuration (MSD) for TRANSMIT not valid!**

Cause: Transmit is not possible at this machine.

**8232 Axis configuration (MSD) for TRACYL not valid!**

Cause: Tracyl is not possible at this machine.

**8233 Axis not available while TRANSMIT/ TRACYL is active!**

Cause: Programming of the rotary axis is not allowed during Transmit/ Tracyl.

**8234 Axis control grant removed by PLC while axis interpolates!**

Cause: Internal error

Remedy: Delete error with reset and inform EMCO.

**8235 Interpolation invalid while axis control grant is off by PLC!**

see 8234.

**8236 TRANSMIT/TRACYL activated while axis or spindle moves!**

see 8234.

**8237 Motion through pole in TRANSMIT!**

Cause: It is not allowed to move through the coordinates X0 Y0 in Transmit.

Remedy: Alter the traversing movement.

**8238 Speed limit in TRANSMIT exceeded!**

Cause: The traversing movement gets too close to the coordinates X0 Y0. In order to observe the programmed feed rate, the maximum speed of the rotary axis would have to be exceeded.

Remedy: Reduce the feed rate. Set the value of the C-axis feed limitation in WinConfig, machine data settings / general machine data/ to 0.2. Thus, the feed rate will be automatically reduced near the coordinates X0 Y0.

**8239 DAU exceeded 10V limit!**

Cause: Internal error

Remedy: Start the software again or install it anew. Report the error to EMCO.

**8240 Function not valid during active transformation (TRANSMIT/TRACYL)!**

Cause: The Jog and INC operating mode are not possible during Transmit in X/C and during Tracyl in the rotary axis.

**8241 TRANSMIT not enabled (MSD)!**

Cause: Transmit is not possible at this machine.

**8242 TRACYL not enabled (MSD)!**

Cause: Tracyl is not possible at this machine.

**8243 Round axis invalid during active transformation!**

Cause: It is not allowed to program the rotary axis during Transmit/Tracyl.

**8245 TRACYL radius = 0!**

Cause: When selecting Tracyl, a radius of 0 was used.

Remedy: Correct the radius.

**8246 Offset alignment not valid for this state!**

see 8239.

**8247 Offset alignment: MSD file write protected!****8248 Cyclic supervision failed!**

Cause: The communication with the machine keyboard is interrupted.

Remedy: Start the software again or install it anew. Report the error to EMCO.

**8249 Axis motion check alarm!**

see 8239

**8250 Spindle must be rotation axis !**

see 8239

**8251 Lead for G331/G332 missing !**

Cause: The threading pitch is missing or the starting coordinates are identical to the target coordinates.

Remedy: Program the threading pitch. Correct the target coordinates.

**8252 Multiple or no linear axis programmed for G331/G332 !**

Remedy: Program exactly one linear axis.

**8253 Speed value for G331/G332 and G96 missing !**

Cause: No cutting speed has been programmed.

Remedy: Program the cutting speed.

**8254 Value for thread starting point offset not valid!**

Cause: The thread starting point offset is not within the range of 0 to 360°.

Remedy: Correct the thread starting point offset.



**8255 Reference point not in valid software limits!**

Cause: The reference point has been defined outside the software limit switches.

Remedy: Correct the reference points in WinConfig.

**8256 Spindle speed too low while executing G331/G332!**

Cause: During tapping the spindle speed has decreased. Perhaps the incorrect threading pitch was used or the core drilling is not correct.

Remedy: Correct the threading pitch. Adapt the diameter to the core drilling.

**8257 Real Time Module not active or PCI card not found!**

Cause: ACC could not be started correctly or the PCI card in the ACC was not recognized.

Remedy: Report the error to EMCO.

**8258 Error allocating Linux data!**

see 8239.

**8259 Current thread in sequence not valid!**

Cause: One block of a thread in sequence has been programmed without thread G33.

Remedy: Correct the program.

**8261 Missing thread in sequence !**

Cause: A successive thread has not been programmed for a thread in sequence, the number has to be in accordance with the SETTHREADCOUNT () that has been defined before.

Remedy: Correct the number of threads in the thread in sequence and add a thread.

**8262 Reference marks are not close enough !**

Cause: The settings of the linear scale have been changed or the linear scale is defective.

Remedy: Correct the settings. Contact EMCO.

**8263 Reference marks are too close together!**

see 8262.

**22000 Gear change not allowed**

Cause: Gear step change when the spindle is active.

Remedy: Stop the spindle and carry out a gear step change.

**22270 Feed too high (thread)**

Cause: Thread pitch too large / missing, Feed for thread reaches 80% of rapid feed

Remedy: Program correction, lower pitch or lower spindle speed for thread



# I: Control Alarms

## Control Alarms 10000 - 59999

These alarms will be triggered by the control.

These are the same alarms as they would appear on the original SIEMENS control.

<b>10208</b>	<b>Channel %1 Continue program with NC Start</b>
Explanation:	%1 = Channel number After block search with calculation, the control is in the desired state. The program can now be started with NC Start or the state can be changed for the time being with overstore/jog.
Reaction:	Alarm display. NC Stop when alarm.
Remedy:	Clear alarm with NC Start and continue processing.
<b>10209</b>	<b>Channel %1 internal NC Stop after block search</b>
Explanation:	%1 = Channel number Internal alarm which serves for releasing NC Stop via the alarm reaction. The alarm is output if \$MN_SEARCH_RUN_MODE ==1 and the last action block is entered after block search in the main run. The alarm 10208 is activated depending on the VDI signal PLC -> NCK channel DBB1.6.
Reaction:	NC Stop when alarm.
Remedy:	Clear alarm with NC Start and continue processing.
<b>10620</b>	<b>Channel %1 block %3 axis %2 at software limit switch %4</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number, label %4 = String During the traversing motion, it is recognized that the software limit switch would be overtraveled in the displayed direction. It was not yet possible to detect in the block preparation that the traversing range would be exceeded: either there has been a motion overlay by the <b>handwheel</b> or a <b>coordinate transformation</b> is active.
Reaction:	Alarm display. Interface signals are set. NC Start disable.
Remedy:	Depending on the reason for this alarm being triggered, the following remedial measures should be undertaken: 1. <b>Handwheel override:</b> Cancel the motion overlay and avoid this or keep it smaller when the program is repeated. 2. <b>Transformation:</b> Check the preset/programmed zero offsets (current frame). If the values are correct, the tool holder (fixture) must be moved in order to avoid triggering the same alarm when the program is repeated, which would again cause the program to be aborted. Clear alarm with RESET key. Restart part program.
<b>10630</b>	<b>Channel %1 block %2 axis %3 at working area limit %4</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis, spindle number %4 = String (+ or -) The specified axis violates the working area limitation. This is recognized only in the main run because either the minimum axis values could not be measured before transformation or because there is a motion overlay.
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm at the block end. NC Start disable.
Remedy:	Program other motion or do not perform overlaid motion. Clear alarm with RESET key. Restart part program.
<b>10720</b>	<b>Channel %1 block %3 axis %2 software limit switch %4</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number, label %4 = String (+ or -) For the axis, the programmed path violates the currently valid software limit switch. (The 2nd software limit switch becomes active with the interface signal "2nd software limit switch plus/minus" in DB 31 - 48, DBX 12.2 and 12.3). The alarm is activated when preparing the part program block.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable.
Remedy:	Check the <b>position</b> of the axis as specified in the part program. Machine data: 36100 POS_LIMIT_MINUS/36120 POS_LIMIT_MINUS2 and 36110 POS_LIMIT_PLUS/36130 POS_LIMIT_PLUS2 must be checked for the software limit switches. Check the axis-specific <b>interface signals</b> : „Check 2nd software limit switch plus/minus" (DB 31 - 48, DBX 12.2 and 12.3) to see whether the 2nd software limit switch is selected. Check currently active <b>zero offsets</b> via the current frame. Clear alarm with NC Start and continue program.

<b>10730</b>	<b>Channel %1 block %3 axis %2 working area limitation %4</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number, label %4 = String (+ or -) This alarm is generated if it is determined during block preparation that the programmed path of the axis will result in exceeding the working area limitation.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable.
Remedy:	a) Check <b>NC program</b> for correct positional data b) Check <b>zero offsets</b> (current frame) c) Correct <b>working area limitation</b> via G25, or d) Correct <b>working area limitation</b> via setting data, or e) Deactivate <b>working area limitation</b> via setting data: 43410 WORKAREA_MINUS_ENABLE=FALSE Clear alarm with NC Start and continue program.
<b>10740</b>	<b>Channel %1 block %2 too many empty blocks in WAB programming</b>
Explanation:	%1 = Channel number %2 = Block number, label It is not allowed to program more blocks than specified by machine data MC_WAB_MAXNUM_DUMMY_BLOCKS between the WAB block and the block determining the approach and retraction tangent
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.
Remedy:	Modify part program Clear alarm with NC Start and continue program.
<b>10741</b>	<b>Channel %1 block %2 direction reversal with WAB infeed motion</b>
Explanation:	%1 = Channel number %2 = Block number, label A safety distance has been programmed which is located vertically to the machining plane and not located between the start and end point of the WAB contour.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.
Remedy:	Modify part program Clear alarm with NC Start and continue program.
<b>10742</b>	<b>Channel %1 block %2 WAB distance invalid or not programmed</b>
Explanation:	%1 = Channel number %2 = Block number, label
Possible causes:	<ul style="list-style-type: none"> <li>• In a WAB block, the parameter DISR has not been stated or its value is less than or equal to 0.</li> <li>• During approach or retraction with circle and active tool radius, the radius of the internally generated WAB contour is negative. The internally generated WAB contour is a circle with such a radius so that when it is offset with the current offset radius (sum of tool radius and offset value OFFN) the tool center point path with the programmed radius DISR result from this.</li> </ul>
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.
Remedy:	Modify part program Clear alarm with NC Start and continue program.
<b>10743</b>	<b>Channel %1 block %2 WAB programmed several times</b>
Explanation:	%1 = Channel number %2 = Block number, label
Reaction:	An attempt has been made to activate a WAB motion before a WAB motion activated previously was terminated. Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.
Remedy:	Modify part program Clear alarm with NC Start and continue program.
<b>10744</b>	<b>Channel %1 block %2 no valid WAB direction defined</b>
Explanation:	%1 = Channel number %2 = Block number, label The tangent direction for smooth approach or retraction is not defined.
Possible causes:	<ul style="list-style-type: none"> <li>• In the program, no block with travel information follows the approach block</li> <li>• Before a retraction block, no block with travel information has been programmed in a program.</li> <li>• The tangent to be used for WAB motion is vertical to the current machining plane.</li> </ul>
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.
Remedy:	Modify part program Clear alarm with NC Start and continue program.
<b>10745</b>	<b>Channel %1 block %2 WAB end positioning not clear</b>
Explanation:	%1 = Channel number %2 = Block number, label In the WAB block and in the following block, the position has been programmed vertically to the machining direction. In the WAB block, no position has been indicated in the machining plane.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.
Remedy:	Modify part program

Either remove the position data for the infeed axis from the WAB block or from the following block, or program also a position in the machining plane in the WAB block.  
Clear alarm with NC Start and continue program.

**10746 Channel %1 block %2 block search stop for WAB**

Explanation: %1 = Channel number

%2 = Block number, label

A block search stop has been inserted between a WAB approach block and the following block defining the tangent direction or between a WAB approach block and the following block defining the end position.

Reaction: Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.

Remedy: Modify part program.

Clear alarm with NC Start and continue program.

**10747 Channel %1 block %2 retraction direction not defined for WAB**

Explanation: %1 = Channel number

%2 = Block number, label

In a WAB retraction block with quarter circle or semi-circle (G248 or G348), the end point in the machining plane was not programmed, and either G143 or G140 without tool radius compensation is active.

Reaction: Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.

Remedy: Modify part program.

The following changes are possible:

- Indicate end point in the machining plane in the WAB block.
- Activate tool radius compensation (effective for G140 only, not for G143)
- State retraction side explicitly with G141 or G142.
- Perform retraction with a straight line instead of a circle.

Clear alarm with NC Start and continue program.

**10750 Channel %1 block %2 tool radius compensation activated without tool no.**

Explanation: %1 = Channel number

%2 = Block number, label

A tool T... must be selected so that the control can make allowance for the associated compensation values. A correction data block (D1) containing the correction values (parameter P1 -P25) is automatically assigned to each tool (T number). Up to 9 correction data blocks can be assigned to a tool by specifying the required data block with the D number (D1 - D9). The cutter radius compensation (CRC) is allowed for if function G41 or G42 is programmed. The correction values are contained in parameter P6 (geometry value) and P15 (wear value) of the active correction data block D x .

Reaction: Alarm display. Interface signals are set. Correction block is reorganized NC Start disable. NC Stop when alarm at block end.

Remedy: Before calling the CRC with G41/G42, program a tool number under the address T... .

Clear alarm with NC Start and continue program.

**10751 Channel %1 block %2 danger of collision due to tool radius compensation**

Explanation: %1 = Channel number

%2 = Block number, label

The „Bottleneck detection“ (calculation of intersection for the following compensated traversing blocks) has not been able to calculate a point of intersection for the reviewed number of traversing blocks. It is therefore possible that one of the equidistant paths violates the workpiece contour.

Reaction: Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Start when alarm at the block end.

Remedy: Check the part program and modify the programming if possible such that inside corners with smaller paths than the correction value are avoided. (Outside corners are not critical because the equidistants are lengthened or intermediate blocks are inserted so that there is always a point of intersection). Increase the number of reviewed traversing blocks via machine data 20240 **CUTCOM\_MAXNUM\_CHECK\_BLOCKS** (default: 3), resulting in an increase in the extent of calculation and therefore also the block cycle time.

Clear alarm with NC Start and continue program.

**10753 Channel %1 block %2: activate tool radius compensation in linear block**

Explanation: %1 = Channel number

%2 = Block number, label

Selection of cutter radius compensation with G41/G42 may only be performed in blocks where the G function G00 (rapid traverse) or G01 (feed) is active. In the block with G41/G42, at least one axis in the plane G17 to G19 must be written. It is always advisable to write both axes because as a rule, both axes are traversed when selecting the compensation.

Reaction: Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.

Remedy: Correct the NC program and put the compensation selection in a block with linear interpolation.

Clear alarm with NC Start and continue program.

**10754 Channel %1 block %2: deactivate tool radius compensation in linear block only**

Explanation: %1 = Channel number

%2 = Block number, label

Deselection of cutter radius compensation with G40 can only be performed in blocks where the G function G00 (rapid traverse) or G01 (feed) is active. In the block with G40, at least one axis in the plane G17 to G19 must be written.

	It is always advisable to write both axes because as a rule, both axes are traversed when deselecting the compensation.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Correct the NC program and put the compensation selection in a block with linear interpolation. Clear alarm with NC Start and continue program.
<b>10755</b>	<b>Channel %1 block %2: do not activate tool radius compensation via KONT at the current starting point</b>
Explanation:	%1 = Channel number %2 = Block number, label When activating the cutter radius compensation with <b>KONT</b> the starting point of the approach block is within the compensation circle and therefore already violates the contour. If the cutter radius compensation is selected with G41/G42, the approach behaviour (NORM or KONT) determines the compensation movement if the momentary actual position is <b>behind</b> the contour. With KONT, a circle is drawn with the cutter radius around the programmed initial point (= end point of the approach block). The tangent that passes through the current actual position and does not violate the contour is the approach movement. If the start point is within the compensation circle around the target point, no tangent passes through this point.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Place selection of the CLC such that the starting point of the approach movements comes to rest outside of the correction circle around the target point (programmed traversing movements > compensation radius). The following possibilities are available: <ul style="list-style-type: none"> <li>• Selection in the previous block</li> <li>• Insert intermediate block</li> <li>• Select approach behaviour NORM</li> </ul> Clear alarm with NC Start and continue program.
<b>10756</b>	<b>Channel %1 block %2: do not deactivate tool radius compensation via KONT at the programmed end point</b>
Explanation:	%1 = Channel number %2 = Block number, label On deselection of the cutter radius compensation, the programmed end point is within the compensation circle. If this point were in fact to be approached without compensation, there would be a contour violation. If the cutter radius compensation is deselected via G40, the approach behaviour (NORM or KONT) determines the compensation movement if the programmed end point is <b>behind</b> the contour. With KONT, a circle is drawn with the cutter radius about the last point at which the compensation is still active. The tangent passing through the programmed end position and not violating the contour is the retraction movement. If the start point is within the compensation circle around the target point, no tangent passes through this point.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Place deselection of the CLC such that the programmed end point comes to rest outside of the compensation circle about the last active compensation point. The following possibilities are available: <ul style="list-style-type: none"> <li>• Deselection in the next block</li> <li>• Insert intermediate block</li> <li>• Select retract behaviour NORM</li> </ul> Clear alarm with NC Start and continue program.
<b>10757</b>	<b>Channel %1 block %2: do not change the compensation plane while tool radius compensation is active</b>
Explanation:	%1 = Channel number %2 = Block number, label In order to change the compensation plane (G17, G18 or G19) it is first necessary to deselect the cutter radius compensation with G40.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Insert an intermediate block with compensation deselection in the part program. After plane change, the cutter radius compensation must be selected in an approach block with linear interpolation. Clear alarm with NC Start and continue program.
<b>10758</b>	<b>Channel %1 block %2 curvature radius with variable compensation value too small</b>
Explanation:	%1 = Channel number %2 = Block number, label The current cutter radius compensation (the cutter used) is too large for the programmed path radius. In a block with variable tool radius compensation, a compensation must be possible either anywhere or nowhere on the contour with the smallest <b>and</b> the largest compensation value from the programmed range. There must be no point on the contour in which the curvature radius is within the variable compensation range. If the compensation value varies its sign within a block, both sides of the contour are checked, otherwise only the compensation side.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Use smaller cutters or allow for a part of the cutter radius at the time of contour programming. Clear alarm with NC Start and continue program.
<b>10760</b>	<b>Channel %1 block %2 helical axis is not parallel to tool orientation</b>
Explanation:	%1 = Channel number %2 = Block number, label

	With active tool radius compensation a helix is only permissible if the helix axis is parallel to the tool, i.e. the circle plane and the compensation plane must be identical.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Orient helix axis perpendicular to the machining plane. Clear alarm with NC Start and continue program.
<b>10762</b>	<b>Channel %1 block %2 too many empty blocks between two traversing blocks with active tool radius compensation</b>
Explanation:	%1 = Channel number %2 = Block number, label The maximum permissible number of empty blocks are limited by a machine data.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	1. Modify part program: 2. Modify machine data 3. Check whether SBL2 is activated. With SBL2, a block is generated from each part program line which can lead to exceeding the maximum permissible number of empty blocks between two traversing blocks. Clear alarm with NC Start and continue program.
<b>10763</b>	<b>Channel %1 block %2 path component of the block in the compensation plane becomes zero.</b>
Explanation:	%1 = Channel number %2 = Block number, label Due to the collision monitoring with active tool radius compensation, the path component of the block in the compensation plane becomes zero. If the original block contains no motion information perpendicular to the compensation plane, it means that this block is excluded.
Reaction:	Alarm display.
Remedy:	<ul style="list-style-type: none"> <li>• The behaviour is correct at narrow locations that cannot be machined with the active tool.</li> <li>• Modify the part program if necessary</li> <li>• Use tool with smaller radius if necessary</li> <li>• Program CDOF.</li> </ul> Clear alarm with the Cancel key. No further operator action necessary.
<b>10764</b>	<b>Channel %1 block %2 discontinuous path with active tool radius compensation</b>
Explanation:	%1 = Channel number %2 = Block number, label This alarm occurs when, with active tool radius compensation, the starting point used for calculating the compensation is not identical to the end point of the preceding block. This situation can occur, for example, when a geometry axis is traversed between two positions as positioning axis or when, with an active kinematic transformation (e.g. 5-axis transformation) the tool length compensation is altered.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Modify part program. Clear alarm with NC Start and continue program.
<b>10770</b>	<b>Channel %1 block %2 change of corner type due to change of orientation with active tool radius compensation</b>
Explanation:	%1 = Channel number %2 = Block number, label The type of a corner (inside or outside corner) depends not only on the programmed path but also on the tool orientation. For this purpose, the programmed path is projected in the plane perpendicularly to the actual tool orientation and the corner type is determined there. If a change in orientation is programmed (in one or several blocks) between two traversing blocks, resulting in the type of corner at the end of the first traversing block being different from that at the start point of the second block, the above error message is issued.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Modify part program. Clear alarm with NC Start and continue program.
<b>10774</b>	<b>Channel %1 illegal tool dimensions with face cutting in block %2</b>
Explanation:	%1 = Channel number %2 = Block number, label This alarm occurs when illegal tool dimensions are programmed for face milling, e.g. negative tool radius, rounding radius zero or negative for tool types that require a rounding radius, taper angle zero or negative for tapered tools.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Modify part program Clear alarm with NC Start and continue program.
<b>10776</b>	<b>Channel %1 block %2 axis %3 must be geo axis if cutter compensation is active</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = axis name This alarm occurs when an axis that is required for tool radius compensation is not a geometry axis. With CUT2DF, the axis can be a positioning axis perpendicular to the machining plane; with all other types of compensation



Reaction:	(CUT2DF, CUT3DC, CUT3DF, CUT3DFF), all geometry axes must be operated as such. Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Modify part program. Clear alarm with NC Start and continue program.
<b>10777</b>	<b>Channel %1 block %2 tool radius compensation: too many blocks with suppression of compensation</b>
Explanation:	%1 = Channel number %2 = Block number, label The maximum permissible number of blocks with active compensation suppression with tool radius compensation is limited by the machine data CUTCOM_MAXNUM_SUPPRESS_BLOCKS.
Reaction:	Alarm display Interface signals are set Correction block mit Reorganisieren NC Start disable NC Stop when alarm at block end
Remedy:	<ul style="list-style-type: none"> <li>• Modify part program</li> <li>• Modify machine data</li> <li>• Check whether SBL2 is activated. With SBL2, a block is generated from each part program line which can lead to exceeding the maximum permissible number of empty blocks between two traversing blocks.</li> </ul> Clear alarm with NC Start and continue program.
<b>10 778</b>	<b>Channel %1 block %2 preparation stop with active tool radius compensation</b>
Explanation:	%1 = Channel number %2 = Block number, label If a preprocessing stop is detected with active tool radius compensation (either programmed by the user or generated internally) and the setting data \$SC_STOP_CC_STOPRE is set, then this warning is issued because in this situation machine movements which were not intended by the user can occur (termination of radius compensation and new approach). To continue machining, activate the CANCEL key and perform a restart.
Reaction:	Alarm display. NC Stop when alarm at block end.
Remedy:	<ul style="list-style-type: none"> <li>• Continue machining with CANCEL and Start</li> <li>• Modify part program</li> <li>• Set setting data \$SC_STOP_CC_STOPRE to FALSE.</li> </ul> Clear alarm with the Cancel key. No further operator action necessary.
<b>10 800</b>	<b>Channel %1 block %3 axis %2 is not a geometry axis</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number, label With an active <b>transformation</b> or a <b>frame with a rotation component</b> the geometry axes are needed for block preparation. If a geometry axis has previously been traversed as positioning axis, it retains its status of „positioning axis“ until it is again programmed as a geometry axis. Because of the POSA motion beyond block boundaries, it is not possible to identify in the preprocessing run whether the axis has already reached its target position when the block is executed. This is, however, an unconditional requirement for calculating the ROT component of the frame or of the transformation. If geometry axes are used as positioning axes, then: 1. <b>No rotation</b> may be specified in the current overall frame. 2. <b>No transformation</b> may be selected.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable.
Remedy:	After selecting transformation or frame, reprogram the geometry axis now operating as positioning axis (e.g. with WAITP) in order to revert the status to „geometry axis“.. Clear alarm with NC Start and continue program.
<b>10805</b>	<b>Channel %1 block %2 repositioning after switch of geoaxes or transformation</b>
Explanation:	%1 = Channel number %2 = Block number, label In the asynchronous subroutine the assignment of geometry axes to channel axes was changed or the active transformation modified.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program Clear alarm with RESET key. Restart part program.
<b>10810</b>	<b>Channel %1 block %2 master spindle not defined</b>
Explanation:	%1 = Channel number %2 = Block number, label The function „Revolutional feedrate“ (with G95 or G96), or „Rigid tapping“ (with G331/G332) has been programmed, although no master spindle is defined from which the speed could be derived. For the definition the MD 20090 SPIND_DEF_MASTER_SPIND is available for the default or the keyword SETMS in the part program, thus allowing each spindle of the channel to be redefined as master spindle.
Reaction:	Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.
Remedy:	Preset master spindle with MD 20090 SPIND_DEF_MASTER_SPIND[n]=m (n ... channel index, m ... spindle no.) or define it in the NC part program with an identifier, before a G function is programmed that requires a spindle. The machine axis to be operated as spindle must be provided with a spindle number in MD 35000 SPIND_ASSIGN_TO_MACHAX[n]=m (n ... machine axis index, m ... spindle no.). Furthermore, it must be assigned to a channel (channel axis index 1 or 2) with the MD 20070 AXCONF_MACHAX_USED[n]=m (n ... channel axis index, m ... machine axis index)..



	Clear alarm with NC Start and continue program.
<b>10820</b>	<b>Channel %1 rotary axis/spindle %2 not defined</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number Revolutional feed has been programmed for contouring and synchronous axes or for an axis/spindle. However, the rotary axis/spindle from which the feed is to be deduced is not available.
Reaction:	Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.
Remedy:	Correct part program or set the setting data 43300 ASSIGN_FEED_PER_REV_SOURCE correctly. Clear alarm with NC Start and continue program.
<b>10860</b>	<b>Channel %1 block %2 feedrate not programmed</b>
Explanation:	%1 = Channel number %2 = Block number, label In the displayed block, an interpolation type other than G00 (rapid traverse) is active. The F value has not been programmed.
Reaction:	Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.
Remedy:	Program feedrate in accordance with the interpolation type. <b>G93:</b> The feedrate is specified as a time-reciprocal value under address F in [1/min]. <b>G94</b> and <b>G97:</b> The feedrate is programmed under address F in [mm/min] or [m/min]. <b>G95:</b> The feedrate is programmed as revolutional feedrate under address F in [mm/revolution]. <b>G96:</b> The feedrate is programmed as cutting rate under address S in [m/min]. It is derived from the current <b>spindle speed</b> . Clear alarm with NC Start and continue program.
<b>10861</b>	<b>Channel %1 block %2 velocity of positioning axis %3 is zero</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = axis No axis velocity has been programmed and the positioning velocity set in the machine data is zero.
Reaction:	Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.
Remedy:	Enter a different velocity in machine data 32060 MA_POS_AX_VELO. Clear alarm with NC Start and continue program.
<b>10862</b>	<b>Channel %1 block %2 master spindle is axis of path</b>
Explanation:	%1 = Channel number %2 = Block number, label A contour has been programmed that also includes the master spindle as contouring axis. However, the velocity of the contour is derived from the rotational speed of the master spindle (e.g. G95).
Reaction:	Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.
Remedy:	Modify the program so that no reference is possible to the program itself. Clear alarm with NC Start and continue program.
<b>10870</b>	<b>Channel %1 block %2 facing axis not defined</b>
Explanation:	%1 = Channel number %2 = Block number, label When constant cutting speed is activated via the <b>G96</b> function, the spindle speed is controlled through the position of the facing axis such that the cutting speed programmed under S [mm/min] is applied at the tool tip. In the channel-specific MD 20100 DIAMETER_AX_DEF[n,m]=x (n ... channel index, m ... spindle index, x ... axis name), the name of the facing axis [String] can be specified for each of the 5 spindles used for speed calculation.
	$S[1/min] = \frac{S_{G96}[m/min] \cdot 1000}{D_{\text{face\_axis}}[\text{mm}] \cdot \pi}$
Reaction:	Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.
Remedy:	Enter the name of the facing axis in the channel-specific machine data 20100 DIAMETER_AX_DEF for the spindles used. Clear alarm with NC Start and continue program.
<b>10880</b>	<b>Channel %1 block %2 too many empty blocks between two traversing blocks when inserting chamfer or radius</b>
Explanation:	%1 = Channel number %2 = Block number, label Between 2 blocks containing contour elements and which are to be joined with a chamfer or a radius ( <b>CHF</b> , <b>RND</b> ), more blocks without contour information have been programmed than provided for in the machine data 20200 CHFRND_MAXNUM_DUMMY_BLOCKS.
Reaction:	Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.
Remedy:	Modify the <b>part program</b> in order that the permissible number of dummy blocks is not exceeded or adapt the channel-specific machine data 20200 CHFRND_MAXNUM_DUMMY_BLOCKS (dummy blocks with chamfers/radii) to the maximum number of dummy blocks. Clear alarm with NC Start and continue program.
<b>10882</b>	<b>Channel %1 block %2: do not activate chamfer or radius without traversing</b>
Explanation:	%1 = Channel number %2 = Block number, label No chamfer or radius has been inserted between 2 linear or circle contours (edge breaking) because:

- There is no straight line or circle contour in the plane
- There is a movement outside of the plane
- A plane change has taken place
- The permissible number of dummy blocks without traversing information has been exceeded

Reaction:

Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.

Remedy:

Correct the part program according to the above error description or change the number of dummy blocks in the channel-specific MD **CHFRND\_MAXNUM\_DUMMY\_BLOCKS** to comply with the maximum number allowed for in the program.

Clear alarm with NC Start and continue program.

#### 10900

#### **Channel %1 block %2 no S value programmed for constant cutting speed**

Explanation:

%1 = Channel number

%2 = Block number, label

If G96 is active, the constant cutting speed under address S is missing

Reaction:

Alarm display. Interface signals are set Correction block is reorganized. NC Start disable.

Remedy:

Program constant cutting speed under S in [m/min] or deselect the function G96. For example, with G97 the previous feed is retained but the spindle continues to rotate at the momentary speed.

Clear alarm with NC Start and continue program.

#### 10910

#### **Channel %1 block %2 excessive velocity of one path axis**

Explanation:

%1 = Channel number

%2 = Block number, label

With active transformation, an excessive increase in velocity occurs in one or several axes, e.g. because the path passes close by the pole.

Reaction:

Alarm display.

Remedy:

Divide the NC block into several blocks (e.g. 3) so that the path section with the excess is as small as possible and therefore of short duration. The other blocks are then traversed at the programmed velocity.

Clear alarm with Cancel key. No further operator action necessary.

#### 10911

#### **Channel %1 block %2 transformation prohibits to traverse the pole.**

Explanation:

%1 = Channel number

%2 = Block number, label

The given curve passes through the pole of the transformation.

Reaction:

Alarm display. Interface signals are set. NC Start disable.

Remedy:

Modify part program.

Clear alarm with RESET key. Restart part program.

#### 10914

#### **Movement not possible while transformation active - in channel %1 for block %2**

Explanation:

%1 = Channel number

%2 = Block number, label

The machine kinematics does not allow the specified motion.

Reaction:

If the working area limitation is violated (see machine position), the part program's working area must be changed such that the possible operating range be adhered to (e.g. modified part settings).

Remedy:

Clear alarm with the RESET key. Restart part program.

#### 10930

#### **Channel %1 block %2 interpolation type not allowed in stock removal contour**

Explanation:

%1 = Channel number

%2 = Block number, label

The contour of the stock removal cycle contains positioning commands other than G00, G01, G02 or G03. The contour program may contain only such contour elements as built up on these preparatory functions (i.e. no threading blocks, no spline blocks etc.).

Reaction:

Alarm display. Interface signals are set. NC Start disable.

Remedy:

In the contour subroutine, program only path elements that consist of straight lines and circular arcs.

Clear alarm with RESET key. Restart part program.

#### 10931

#### **Channel %1 block %2 error in programmed stock removal contour**

Explanation:

%1 = Channel number

%2 = Block number, label

In the subroutine for the contour there are the following errors during stock removal:

- Full circle
- Overlapping contour elements
- Wrong start position

Reaction:

Alarm display. Interface signals are set. NC Start disable.

Remedy:

The errors listed above must be corrected in the subroutine for the stock removal contour.

Clear alarm with RESET key. Restart part program.

#### 10932

#### **Channel %1 block %2 preparation of contour has been restarted**

Explanation:

%1 = Channel number

%2 = Block number, label

After contour segmentation has been started with the keyword **CONTPRON**, the contour to be prepared is described in the following block (as subroutine and/or main program). Following contour description, the contour segmentation must be ended with the keyword **EXECUTE** before a new call may occur.

Reaction:

Alarm display. Interface signals are set. NC Start disable.

Remedy:

Program the keyword **EXECUTE** for ending the previous conditioning in the part program before again calling up contour segmentation (keyword **CONTPRON**).

Clear alarm with RESET key. Restart part program.

<b>10933</b>	<b>Channel %1 block %2 contour program contains too few contour blocks</b>
Explanation:	%1 = Channel number %2 = Block number, label The subroutine in which the stock removal contour is programmed contains fewer than 3 blocks with movements in both axes in the machining plane. The stock removal cycle has been aborted.
Reaction:	Alarm display. Interface signals are set. NC reagiert innerhalb einer Bearbeitungsstation. NC Start disable.
Remedy:	Increase the size of the subroutine with the stock removal contour to include at least 3 NC blocks with movements in both axes of the current machining plane. Clear alarm with RESET key. Restart part program.
<b>10934</b>	<b>Channel %1 block %2 array for contour segmentation is set too small</b>
Explanation:	%1 = Channel number %2 = Block number, label During contour segmentation (activated with the keyword CONTPRON), the field for the contour table has been detected as too small. For every permissible contour element (circle or straight line) there must be a row in the contour table.
Reaction:	NC reacts within a machining station. Alarm display. Interface signals are set. NC Start disable.
Remedy:	Base the definition of the field variables of the contour table on the contour elements to be expected. The contour segmentation function divides up some NC blocks into as many as 3 machining cuts. Example: N100 DEF TABNAME_1 [30, 11] Field variables for the contour table provide for 30 machining cuts. The number of columns (11) is a fixed quantity. Clear alarm with RESET key. Restart part program.
<b>12000</b>	<b>Channel %1 block %2 address %3 programmed repeatedly</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string der Adresse Most addresses (address types) may only be programmed <b>once</b> in an NC block, so that the block information remains unambiguous (e.g. X... T... F... etc. - exception: G and M functions).
Reaction:	Alarm display. Interface signals are set Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. • Remove from the NC program addresses that occur more than once (except for those where multiple value assignments are allowed). • Check whether the address (e.g. the axis name) is specified via a user-defined variable (this may not be easy to see if allocation of the axis name to the variable is performed in the program through computational operations only). Clear alarm with NC Start and continue processing.
<b>12010</b>	<b>Channel %1 block %2 address %3 address type programmed too often</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string of the address For each address type, it is defined internally how often it may occur in a DIN block (for instance, all axes together form <b>one</b> address type for which a block limit also applies).
Reaction:	Alarm display. Interface signals are set Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. The program information must be split up over several blocks. But make sure that the functions are of the non-modal type Clear alarm with NC Start and continue processing.
<b>12020</b>	<b>Channel %1 block %2 combination of address modification not allowed</b>
Explanation:	%1 = Channel number %2 = Block number, label Valid address types are 'IC', 'AC', 'DC', 'CIC', 'CAC', 'ACN', 'ACP', 'CACN', 'CACP'. Not each of these address modifications can be used for each address type. The Programming Guide specifies which of these can be used for the various address types. If this address modification is applied to address types that are not allowed, then the alarm is generated, e.g.: N10 G02 X50 Y60 I=DC(20) J30 F100 interpolation parameters with DC.
Reaction:	Alarm display. Interface signals are set Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Apply non-modal address modifications only for permissible addresses, in accordance with the Programming Guide. Clear alarm with NC Start and continue processing.
<b>12030</b>	<b>Channel %1 block %2 invalid arguments or data types in %3</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string In polynomial interpolation, polynomials must not be greater than the 3rd degree. (Refer to Programming Guide.) $f(p) = a_0 + a_1 p + a_2 p^2 + a_3 p^3$ The coefficients $a_0$ (the starting points) are identical to the end points of the preceding block and need not be programmed. In the polynomial block, a <b>maximum of 3 coefficients per axis</b> is therefore allowed ( $a_1, a_2, a_3$

	)
Reaction:	Alarm display. Interface signals are set Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Clear alarm with NC Start and continue processing.
<b>12040</b>	<b>Channel %1 block %2 expression %3 is not of data type 'AXIS'</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string in the block Some keywords demand in their following parameter specification the data to be in variables of the type „AXIS“. For example, in the keyword PO the axis identifier must be specified in the parenthesized expression, and it must be defined as a variable of the AXIS type. With the following keywords only parameters of the AXIS type are possible: AX[.], FA[.], FD[.], FL[.], IP[.], OVRA[.], PO[.], POS[.], POSA[.] Example: N5 DEF INT INFEED=Z1 ; incorrect, this does not specify an axis; identifier but the number “26 161“ N5 DEF AXIS INFEED=Z1 ; correct : N10 POLY PO[X]=(0.1,0.2,0.3) PO[Y]=(22,33,44) &PO[INFEED]=(1,2,3)
Reaction:	Alarm display. Interface signals are set Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Correct the part program in accordance with the instructions given in the Programming Guide. Clear alarm with NC Start and continue processing.
<b>12060</b>	<b>Channel %1 block %2 same G group programmed repeatedly</b>
Explanation:	%1 = Channel number %2 = Block number, label The G functions that can be used in the part program are divided into groups that are <b>syntax defining</b> or <b>non-syntax defining</b> . Only <b>one G function</b> may be programmed from each G group. The functions within a group are mutually preclusive. The alarm refers only to the non-syntax defining G functions. If several G functions from these groups are called in one NC block, the <b>last</b> of these in a group is active <b>in each case</b> (the previous ones are ignored). G FUNCTIONS: Syntax defining G functions: 1st to 4th G group Non-syntax defining G functions: 5th to n G group
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Remedy is not necessary, but it should be checked whether the G function last programmed really is the one required. Clear alarm with NC Start and continue processing.
<b>12070</b>	<b>Channel %1 block %2 too many syntax-defining G functions</b>
Explanation:	%1 = Channel number %2 = Block number, label <b>Syntax defining G functions</b> determine the structure of the part program block and the addresses contained in it. <b>Only one</b> syntax defining G function may be programmed in each NC block. The G functions in the 1st to 4th G group are syntax defining.
Reaction:	Alarm display. Interface signals are set Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Analyze NC block and distribute the G functions over several NC blocks. Clear alarm with NC Start and continue processing.
<b>12080</b>	<b>Channel %1 block %2 syntax error in text %3</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source text area At the text position shown, the grammar in the block is incorrect. The precise reason for this error cannot be specified in more detail because there are too many possibilities. Example 1: N10 IF GOTOF ... ; the condition for the jump is missing! Example 2: N10 DEF INT VARI=5 N11 X VARI; the operation is missing for the X and VARI variables
Reaction:	Alarm display. Interface signals are set Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Analyze the block and correct it in accordance with the syntax rules given in the Programming Guide. Clear alarm with NC Start and continue processing.
<b>12090</b>	<b>Channel %1 block %2 unexpected argument %3</b>
Explanation:	%1 = Channel number %2 = Block number, label

%3 = Disallowed parameters in the text

The programmed function has been predefined; no parameters are allowed in its call. The first unexpected parameter is displayed.

Example:

On calling the predefined subroutine TRAF0F (switching off a transformation) parameters have been transferred (one or more).

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.  
Program function without parameter transfer.  
Clear alarm with NC Start and continue processing.

**12100 Channel %1 block %2 number of passes %3 not permissible**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Number of passes

The subroutines called with **MCALL** are modal, i.e. after each block with positional information a routine run is automatically performed **once**. For this reason, programming of the number of passes under address P is not allowed.

The modal call is effective until another **MCALL** is programmed, either with a new subroutine name or without (delete function).

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.  
Program the subroutine call **MCALL** without number of passes.  
Clear alarm with NC Start and continue processing.

**12110 Channel %1 block %2 syntax cannot be interpreted**

Explanation: %1 = Channel number

%2 = Block number, label

The addresses programmed in the block are not permissible together with the valid syntax defining G function. Example: G1 I10 X20 Y30 F1000. An interpolation parameter must not be programmed in the linear block.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.  
Check the block structure and correct in accordance with the programming requirements.  
Clear alarm with NC Start and continue processing.

**12120 Channel %1 block %2: Write special G function in separate block**

Explanation: %1 = Channel number

%2 = Block number, label

The G function programmed in this block must be alone in the block. No general addresses or synchronous actions may occur in the same block.

These G functions are:

**G25, G26** Working area and spindle speed limitation

**G110, G111, G112** Pole programming with polar coordinates

**G92** Spindle speed limitation with v constant **STARTFIFO, STOPFIFO** Control of preprocessing buffer.

E.g. G4 F1000 M100: no M function allowed in the G4 block.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Program G function by itself in the block.  
Clear alarm with NC Start and continue processing.

**12140 Channel %1 block %2 expression %3 not contained in this release**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Software construct in the source text

In the full configuration of the control functions are possible that are not yet implemented in the current version.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.  
The displayed function must be removed from the program.  
Clear alarm with NC Start and continue processing.

**12150 Channel %1 block %2 operation %3 not compatible with data type**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = String (violating operator)

The data types are not compatible with the required operation (within an arithmetic expression or in a value assignment).

Example 1:

Arithmetic operation

N10 DEF INT OTTO

N11 DEF STRING[17] ANNA

N12 DEF INT MAX

:



N50 MAX = OTTO + ANNA

Example 2:

Value assignment

N10 DEF AXIS BOHR

N11 DEF INT OTTO

:

N50 OTTO = BOHR

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.  
Alter the definition of the variables used such that the required operations can be executed.  
Clear alarm with NC Start and continue processing.

**12160 Channel %1 block %2 range of values exceeded**

Explanation: %1 = Channel number

%2 = Block number, label

The programmed constant or the variable exceeds the value range that has previously been established by the definition of data type.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Correct value of the constant or adapt data type. If the value for an integer constant is too great, it can be specified as real constant by adding a decimal point.

Example:

R1 = 9 876 543 210 Correct: R1 = 9 876 543 210.

Value range INTEGER: 2 31 - 1

Value range REAL:  $2^{-1022}$  bis  $2^{+1023}$

Clear alarm with NC Start and continue processing.

**12170 Channel %1 block %2 identifier %3 defined repeatedly**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Symbol in block

The symbol shown in the error message has already been defined in the active part program. Note that user-defined identifiers may occur more than once if the multiple definition occurs in other (sub)programs i.e. local variables may be redefined with the same name if the program has been exited (subprograms) or has already been concluded. This applies both to user-defined symbols (labels, variables) and to machine data (axes, DIN addresses and G functions).

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: The symbol already known to data management is displayed. This symbol must be looked for in the definition part of the current program using the program editor. The 1st or 2nd symbol must be given a different name.  
Clear alarm with NC Start and continue processing.

**12180 Channel %1 block %2 illegal chaining of operators %3**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Chained operators

Operator chaining means the writing in sequence of binary and unary operators without using any form of parentheses.

Example:

N10 ERG = VARA - ( - VARB ) ; correct notation

N10 ERG = VARA - - VARB ; error !

Reaction: Alarm display. Interface signals are set. Correction block.

Formulate the expression correctly and unambiguously making use of parentheses. This improves clarity and readability of the program.

Clear alarm with NC Start and continue processing.

**12190 Channel %1 block %2 variable of type ARRAY has too many dimensions**

Explanation: %1 = Channel number

%2 = Block number, label

Array with variables of type STRING may be no more than 1-dimensional, and with all other variables no more than 2-dimensional.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Correct the array definition, with multi-dimensional arrays define a second 2-dimensional array if necessary and operate it with the same field index.  
Clear alarm with NC Start and continue processing.

**12200 Channel %1 block %2 symbol %3 cannot be created**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Symbol in the source block

The symbol to be created with the DEF instruction cannot be created because:

- it has already been defined (e.g. as variable or function)
- the internal memory location is no longer sufficient (e.g. with large arrays).

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Make the following checks:

- Check with the text editor whether the name to be allocated in the active program cycle (main program and called subprograms) has already been used.
- Estimate the memory requirements for the symbols already defined and reduce these if necessary by using fewer global and more local variables.

Clear alarm with NC Start and continue processing.

#### 12210

##### Channel %1 block %2 string %3 too long

Explanation:

%1 = Channel number

%2 = Block number, label

%3 = String in the source block

- In the definition of a variable of type STRING, it has been attempted to initialize more than 100 characters.
- In an allocation, it has been found that the string does not fit in the given variable.

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.

- Select shorter string or divide up the character string into 2 strings
- Define larger string variable

Clear alarm with NC Start and continue processing.

#### 12220

##### Channel %1 block %2 binary constant %3 in string too long

Explanation:

%1 = Channel number

%2 = Block number, label

%3 = Binary constant

When initializing or allocating the value of a variable of type STRING more than 8 bits have been found as binary constant. DEF STRING[8] OTTO = "ABC'H55"B000011111'DEF"

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. In the window for the alarm message, the first characters of the binary constant are always displayed although the surplus bit might not be located until further on. Therefore, the **complete binary constant** must always be checked for an incorrect value.

Clear alarm with NC Start and continue processing.

#### 12230

##### Channel %1 block %2 hexadecimal constant %3 in string too long

Explanation:

%1 = Channel number

%2 = Block number, label

%3 = Hexadecimal constant

A string can also contain bytes that do not correspond to a character that can be entered or one that is available on a keyboard with a minimized number of keys. These characters can be input as binary or hexadecimal constants. They may occupy up to 1 byte each only - therefore be <256, e.g.

N10 DEF STRING[2] OTTO=" 'HCA' 'HFE' "

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. In the window for the alarm message, the first characters of the binary constant are always displayed although the surplus bit might not be located until further on. Therefore, the **complete hexadecimal constant** must always be checked for an incorrect value.

Clear alarm with NC Start and continue processing.

#### 12240

##### Channel %1 block %2 tool orientation %3 defined repeatedly

Explanation:

%1 = Channel number

%2 = Block number, label

%3 = Text

Only one tool orientation can be programmed per DIN block. This can either be defined via the 3 Euler angles, or the end points of the axes, or through direction vectors.

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Since the tool orientation can be set in 3 different ways, the most advantageous should be selected. For specifying in this way, the addresses and value assignments must be programmed and all other orientation parameters must be removed.

**Axis end points (additional axes):** A, B, C axis identifiers,

**Euler angles:** A2, B2, C2

**Direction vectors:** A3, B3, C3

Clear alarm with NC Start and continue processing.

#### 12250

##### Channel %1 block %2 do not nest macro %3

Explanation:

%1 = Channel number

%2 = Block number, label

%3 = Source string

The macro technique supplies a 1-line instruction or series of instructions with a new identifier by means of the keyword DEFINE. No further macro may be contained in the string of instructions (nesting).

Example:

N10 DEFINE MACRO1 AS G01 G91 X123 MACRO2 F100

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Nested macros must be replaced by the full program information. Clear alarm with NC Start and continue processing.

**12260 Channel %1 block %2 too many initialization values given for %3**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Source string

In the initialization of an array (array definition and value assignments to individual array elements) there are more initialization values than array elements.

Example:

N10 DEF INT OTTO[2,3]=(..., ..., {more than 6 values})

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.

Check the NC program to establish whether:

1. During array definition the number of array elements (n,m) was indicated correctly (DEF INT FIELDNAME[n,m] e.g. an array with 2 lines and 3 columns: n=2, m=3).

2. During initialization the value assignments have been made correctly (values of the individual field elements separated by **comma, decimal point** for variables of the type REAL)

Clear alarm with NC Start and continue processing.

**12270 Channel %1 block %2 macro identifier %3 already defined**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Source string macro name

The name of the macro to be selected by the instruction DEFINE is already defined in the control as:

**Macro name**

**Keyword**

**Variable**

**Configured identifier.**

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.

Select DEFINE instruction with another macro name.

Clear alarm with NC Start and continue processing.

**12290 Channel %1 block %2 arithmetic variable % 3 not defined**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Source string arithmetic variable

Only the R parameters are predefined as arithmetic variables. All other arithmetic variables must be defined with the DEF instruction before being used. The number of arithmetic parameters is defined via machine data. The names must be unambiguous and may not be repeated in the control (exception: local variables).

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Define the required variable in the definition part of the program (possibly in the calling program if it is to be a global variable).

Clear alarm with NC Start and continue processing.

**12300 Channel %1 block %2 call-by-reference argument missing on subroutine call %3**

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Source string

In the subroutine definition, a formal **REF parameter (call-by-reference parameter)** has been specified with no actual parameter assigned to it. The assignment takes place in the subroutine call on the basis of the position of the variable name and not on the basis of the name.

Example:

**Subroutine:** (2 call-by-value parameters X and Y, 1 call-by-reference parameter Z)

PROC XYZ (INT X, INT Y, VAR INT Z)

:

M17

ENDPROC

Main program:

N10 DEF INT X

N11 DEF INT Y

N11 DEF INT Z



:  
 N50 XYZ (X, Y) ; REF parameter Z missing!  
 or  
 N50 XYZ (X, Z) ; REF parameter Z missing!

Reaction: Alarm display. Interface signals are set. Correction block.  
 Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Assign a variable to all REF parameters (call-by-reference parameters) of the subroutine when calling. No variable must be assigned to „normal“ formal parameters (call-by-value parameters), as these are defaulted with 0.  
 Clear alarm with NC Start and continue processing.

**12310 Channel %1 block %2 axis argument missing on procedure call %3**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Source string  
 When calling the subroutine, an AXIS parameter is missing which, according to the EXTERN declaration, should be present. With the EXTERN instruction, user-defined subroutines (procedures) are made „known“ that have a parameter transfer. Procedures without parameter transfer require no EXTERN declaration.  
 Example:  
**Subroutine XYZ** (with the formal parameters):  
 PROC XYZ (INT X, VAR INT Y, AXIS A, AXIS B)  
**EXTERN instruction** (with variable types):  
 EXTERN XYZ (INT, VAR INT, AXIS, AXIS)  
**Subroutine call** (with actual parameters):  
 N10 XYZ ( , Y1, R\_TABLE)  
**Variable X** is defaulted with value 0  
**Variable Y** is supplied with the value of the variable Y1 and returns the results to the calling program after the subroutine run  
**Variable A** is supplied with the axis in R\_TABLE  
**Variable B missing!**

Reaction: Alarm display. Interface signals are set. Correction block.  
 Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Program the missing AXIS parameter in the call.  
 Clear alarm with NC Start and continue processing.

**12320 Channel %1 block %2 argument %3 must be call-by-reference**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Source string  
 A constant or the result of a mathematical expression has been assigned to a REF parameter instead of a variable at the time of the subroutine call, even though only variable identifiers are allowed.  
 Examples:  
 N10 XYZ (NAME\_1, 10, OTTO) or N10 XYZ (NAME\_1, 5 + ANNA, OTTO)

Reaction: Alarm display. Interface signals are set. Correction block.  
 Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.  
 Remove the constant or the mathematical expression from the NC block.  
 Clear alarm with NC Start and continue processing.

**12330 Channel %1 block %2 type of argument %3 incorrect**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Source string  
 When calling a procedure (a subroutine) it is found that the type of the actual parameter cannot be converted into the type of the formal parameter. There are two possible cases:

- **Call-by-reference parameter:** Actual parameter and formal parameter must be of precisely the same type, e.g. STRING, STRING.
- **Call-by-value parameter:** Actual parameter and formal parameter can in principle be different providing conversion is basically possible. In the present case, however, the types are generally not compatible, e.g. STRING-REAL.

from - to	REAL	INT	BOOL	CHAR	STRING	AXIS	FRAME
REAL	yes	yes*	yes**	yes*	-	-	-
INT	yes	yes	yes**	if value 0..255	-	-	-
BOOL	yes	yes	yes	yes	-	-	-
CHAR	yes	yes	yes**	yes	yes	-	-
STRING	-	-	yes***	only if 1 character	yes	-	-
AXIS	-	-	-	-	-	yes	-
FRAME	-	-	-	-	-	-	yes

\* At type conversation from REAL to INT fractional values that are >=0.5 are rounded up, others are rounded down

\*\* Value <>0 corresponds to TRUE, value ==0 corresponds to FALSE.

\*\*\* String length 0 => FALSE, otherwise TRUE

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Check transfer parameters of the subroutine call and define the application accordingly as call-by-value or call-by-reference parameter.  
Clear alarm with NC Start and continue processing.

**12340 Channel %1 block %2 number of arguments exceeded in %3**

Explanation: %1 = Channel number  
%2 = Block number, label  
%3 = Source string

When calling a function or a procedure (predefined or user-defined) more parameters were transferred than defined.

**Predefined functions and procedures:**

The number of parameters has been set permanently in the NCK.

**User-defined functions and procedures:**

The number of parameters is established by type and name in the definition

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Check whether the correct procedure/function has been called. Program the number of parameters in accordance with the procedure/function.  
Clear alarm with NC Start and continue processing.

**12350 Channel %1 block %2 argument %3 not accepted because AXIS argument is missing**

Explanation: %1 = Channel number  
%2 = Block number, label  
%3 = Source string

An attempt has been made to transfer actual parameters although axis parameters located before them have not been assigned. For procedure or function calls, assignment of parameters that are no longer required can be omitted, if **subsequently no further** parameters are to be transferred.

Example:

N10 FGROU(X, Y, Z, A, B) ; max. 8 axes possible

The following call-by-value parameters would then be defaulted with zero because the space-dependent assignment has been lost on account of the omitted axis parameters. Axes that can be omitted and following parameters do not occur in the predefined procedures and functions.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. In **predefined procedures and functions** either remove the following parameters or transfer any preceding axis parameters. In **user-defined procedures and functions**, parameter transfer must be programmed in accordance with the instructions given in the machine manufacturer's programming guide.  
Clear alarm with NC Start and continue processing.

**12360 Channel %1 block %2 dimension of argument %3 incorrect**

Explanation: %1 = Channel number  
%2 = Block number, label  
%3 = Source string

The following possibilities of error must be checked:

1. The current parameter is an array, but the formal parameter is a variable
2. The current parameter is a variable, but the formal parameter is an array
3. The current and formal parameters are arrays, but not with the dimensions to be defined.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.  
Correct the NC part program in accordance with the cause of error as listed above.  
Clear alarm with NC Start and continue processing.

**12370 Channel %1 block %2 range of values exceeded for %3**

Explanation: %1 = Channel number  
%2 = Block number, label  
%3 = Source string

Outside of an initialization block, a variable has been provided with a value range. The definition of program-global variables is allowed only in special initialization blocks. They can be provided with a value range.

Reaction: Alarm display. Interface signals are set. Correction block.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Remove specification of value range (begins with the keyword OF) or define the variable as global variable in the initialization block and provide it with a value range.  
Clear alarm with NC Start and continue processing.

**12390 Channel %1 block %2 type of initial value for %3 cannot be converted**

Explanation:

%1 = Channel number  
 %2 = Block number, label  
 %3 = Source string

During initialization, a value has been assigned to a variable that does not correspond to the type of the variable, nor can it be converted to the data type of the variable.

from - to	REAL	INT	BOOL	CHAR	STRING
REAL		yes*	yes	yes**	-
INT	yes		yes	yes**	-
BOOL	yes	yes		yes	-
CHAR	yes	yes	yes**		yes
STRING	-	-	yes	yes***	

\* Value <>0 corresponds to TRUE, value ==0 corresponds to FALSE.

\*\* String length 0 => FALSE, otherwise TRUE

\*\*\* If only one character

It is not possible to convert from type AXIS and FRAME nor into type AXIS and FRAME.

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.

- Define variable type such that the initialization value can be assigned, or
- Select initialization value in accordance with the variable definition.

Clear alarm with NC Start and continue processing.

**12400 Channel %1 block %2 element of array %3 does not exist**

Explanation:

%1 = Channel number  
 %2 = Block number, label  
 %3 = Source string

The following causes are possible:

- Impermissible index list; an axis index is missing
  - Array index does not match the definition of the variables
  - An attempt was made to access a variable at array initialization via SET or REP; this attempt did not correspond to the standard access. Single character access, partial frame access, omitted indices not possible.
- A nonexistent element was addressed on initializing this array.

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.

- **Array initialization:** Check the array index of the addressed element. The 1st array element is given the index [0,0], the 2nd array element [0,1] etc. The right array index (column index) is incremented first. In the 2nd row, the 4th element is also addressed with the index [1,3] (the indices start at zero).

- **Array definition:** Check the size of the array. The 1st number indicates the number of elements in the 1st dimension (number of rows), the 2nd number indicates the number of elements in the 2nd dimension (number of columns). An array with 2 rows and 3 columns must be defined by specifying [2,3].

Clear alarm with NC Start and continue processing.

**12410 Channel %1 block %2 incorrect index type for %3**

Explanation:

%1 = Channel number  
 %2 = Block number, label  
 %3 = Source string

In assigning a value to an element of an array variable, the array index was specified in a way that is not allowed. Only the following are allowed as array index (in square brackets):

- **Axis identifier**, provided the array variable was defined as data type FRAME.
- **Integer values** for all other data types.

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.

Correct indices of the array element with respect to variable definition or define the array variable differently.

Clear alarm with NC Start and continue processing.

**12420 Channel %1 block %2 identifier %3 too long**

Explanation:

%1 = Channel number  
 %2 = Block number, label

The symbol to be defined or the specified jump target has a name which is longer than the 32 characters allowed.

Reaction:

Alarm display. Interface signals are set. Correction block.

Remedy:

Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. The symbol to be created or the target of program jumps (label) must be selected within the system agreements, that means the name must begin with 2 letters (but the 1st sign must not be „\$“) and may be up to a maximum of 32 characters.

Clear alarm with NC Start and continue processing.

<b>12430</b>	<b>Channel %1 block %2 invalid index</b>
Explanation:	%1 = Channel number %2 = Block number, label In specifying an array index (in the array definition) an index was used that is outside the permissible range.
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Specify array index within the permissible range. Value range per array dimension: 1 - 32 767. Clear alarm with NC Start and continue processing.
<b>12440</b>	<b>Channel %1 block %2 maximum number of formal arguments exceeded</b>
Explanation:	%1 = Channel number %2 = Block number, label In the definition of a procedure (a subroutine) or in an EXTERN instruction, more than 127 formal parameters have been specified. Example: PROC ABC (FORMPARA1, FORMPARA2, ... .. FORMPARA127, <b>FORMPARA128</b> , ...) EXTERN ABC (FORMPARA1, FORMPARA2, ... .. FORMPARA127, <b>FORMPARA128</b> , ...)
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. A check must be made to determine whether all parameters really have to be transferred. If so, the formal parameters can be reduced by using global variables or R parameters, or by grouping together parameters of the same type to form an array and transfer them in this form. Clear alarm with NC Start and continue processing.
<b>12450</b>	<b>Channel %1 block %2 label defined repeatedly</b>
Explanation:	%1 = Channel number %2 = Block number, label The label of this block already exists. If the NC program is compiled off-line, the entire program is compiled block for block. During this procedure all multiple labels are recognized; this is not always the case with <b>on-line compilation</b> . (Only the actual program run is compiled here, i.e. program branches that are not passed through in this run are disregarded and could therefore contain programming errors.)
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer is positioned on the block where the displayed label occurs for the second time. Use the editor to search the part program where this label occurs for the first time, and change one of the names. Clear alarm with NC Start and continue processing.
<b>12460</b>	<b>Channel %1 block %2 maximum number of symbols exceeded with %3</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string The max. number of variable definitions (GUD, LUD), macro definitions, cycle programs, cycle parameters, that the controller's data management is able to handle, has been exceeded. If this alarm occurs in conjunction with alarm 15180 (initial.ini download failed), then this alarm shows the name of the block causing the error. (For a list of names and their meaning, please refer to alarm 6010)
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	Reduce the symbols in the block (possibly by using the array technique or by using R parameters), or adapt the machine data (if you have access rights). \$MC_MM_NUM_LUD_NAMES_TOTAL with error in LUD blocks (i.e. if more variable definitions were made in the active part programs than allowed by the MD). GUD data blocks can only cause errors as part of the 'initial.ini download' process. Macros and cycle program definitions are reloaded at each POWER ON/ NCK-RESET. This means that these blocks can only cause errors in conjunction with this process. See also the explanations for alarm 6010 Clear alarm with NC Start and continue processing.
<b>12470</b>	<b>Channel %1 block %2 unknown G function %3 used</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string In the displayed block, a non-defined G function has been programmed. Only "real" G functions are checked, which begin with the address G, e.g. G555. "Named" G functions such as CSPLINE, BRISK etc. are interpreted as subroutine names.
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. It must be decided on the basis of the machine manufacturer's programming guide as to whether or not the displayed G function is always omitted or not possible, or whether a standard G function has been reconfigured (or introduced by OEM). Remove G function from the part program or program function call in accordance with the machine manufacturer's programming guide.

Clear alarm with NC Start and continue processing.

<b>12480</b>	<b>Channel %1 block %2 subroutine %3 already defined</b>
Explanation:	<p>%1 = Channel number          %2 = Block number, label          %3 = Source string</p> <p>The name used in the PROC or EXTERN instruction has already been defined in another call description (e.g. for cycles).</p> <p>Example:          EXTERN CYCLE85 (VAR TYP1, VAR TYP2, ...)</p>
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	<p>Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. A program name must be selected that has not yet been used as identifier (theoretically, the parameter declaration of the EXTERN instruction could also be adapted to the existing subroutine in order to avoid the alarm output. However, it would have been identically defined twice).</p> <p>Clear alarm with NC Start and continue processing.</p>
<b>12520</b>	<b>Channel %1 too many machine data %3 in block %2</b>
Explanation:	<p>%1 = Channel number          %2 = Block number, label          %3 = Source symbol</p> <p>In the part program, in the machine data file (..._TOA) and in the initialization file (..._INI), no more than 2 machine data may be used per block.</p> <p>Example:          N ...          N 100 \$TC_DP1 [5,1] = 130, \$TC_DP3 [5,1] = 150.123, \$TC_DP4 [5,1] = 223.4,          \$TC_DP5 [5,1] = 200.12, \$TC_DP6 [5,1] = 55.02          N ...</p>
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	<p>Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.</p> <ul style="list-style-type: none"> <li>• Divide up the part program block into several blocks</li> <li>• If necessary, use the local variable for storing intermediate results</li> </ul> <p>Clear alarm with NC Start and continue processing.</p>
<b>12530</b>	<b>Channel %1 block %2 invalid index for %3</b>
Explanation:	<p>%1 = Channel number          %2 = Block number, label          %3 = Source string</p> <p>In macro definitions, an attempt was made to define a G function with more than 3 decades or an M function with more than 2 decades as identifier of the macro.</p> <p>Example:          _N_UMAC_DEF DEFINE G4444 AS G01 G91 G1234          DEFINE M333 AS M03 M50 M99          :          M17</p>
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	<p>Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Modify the macro definition in accordance with the Programming Guide.</p> <p>Clear alarm with NC Start and continue processing.</p>
<b>12540</b>	<b>Channel %1 block %2 is too long or too complex</b>
Explanation:	<p>%1 = Channel number          %2 = Block number, label</p> <p>The maximum internal block length after translator processing must not exceed 256 characters. After editing, for example, several macros in the block or a multiple nesting, this limit can be exceeded.</p>
Reaction:	Alarm display. Interface signals are set. Correction block.
Remedy:	<p>Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block.</p> <p>Divide up the program block into several subblocks.</p> <p>Clear alarm with NC Start and continue processing.</p>
<b>12550</b>	<b>Channel %1 block %2 identifier %3 not defined or option does not exist</b>
Explanation:	<p>%1 = Channel number          %2 = Block number, label          %3 = Source symbol</p> <p>The displayed identifier was not defined before being used.</p> <p><b>Macro:</b> Keyword, to be defined with the DEFINE ... AS ... instruction is missing in one of the files:          _N_SMAC_DEF, _N_MMAC_DEF, _N_UMAC_DEF, _N_SGUD_DEF, _N_MGUD_DEF, _N_UGUD_DEF</p> <p><b>Variable:</b> DEF instruction missing</p> <p><b>Program:</b> PROC declaration missing</p>

Reaction:	Alarm display. Interface signals are set. Correction block.																								
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. - Correct the names used (typing error) - Check the definition of variables, subroutines and macros - Check options. Clear alarm with NC Start and continue processing.																								
<b>12560</b>	<b>Channel %1 block %2 programmed value %3 exceeds allowed limits</b>																								
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string In a value assignment, the permissible value range of the data type has been exceeded.																								
Reaction:	Alarm display. Interface signals are set. Correction block.																								
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Assign value within the value range of the various data types, or if necessary use another type in order to increase the size of the value range, e.g. INT -> REAL.																								
	<table border="0"> <thead> <tr> <th>Variable type</th> <th>Property</th> <th>Value range</th> </tr> </thead> <tbody> <tr> <td><b>REAL</b></td> <td>Fractional number with dec. pt.</td> <td><math>\pm(2^{-1022} - 2^{+1023})</math></td> </tr> <tr> <td><b>INT</b></td> <td>Integers with signs</td> <td><math>\pm(2^{31} - 1)O</math></td> </tr> <tr> <td><b>BOOL</b></td> <td>Truth value TRUE, FALSE</td> <td>0,1</td> </tr> <tr> <td><b>CHAR</b></td> <td>1 ASCII character</td> <td>0 - 255</td> </tr> <tr> <td><b>STRING</b></td> <td>Character string (max. 100 values)</td> <td>0 - 255</td> </tr> <tr> <td><b>AXIS</b></td> <td>Axis addresses</td> <td>Axis names only</td> </tr> <tr> <td><b>FRAME</b></td> <td>Geometric information</td> <td>As for axis paths</td> </tr> </tbody> </table>	Variable type	Property	Value range	<b>REAL</b>	Fractional number with dec. pt.	$\pm(2^{-1022} - 2^{+1023})$	<b>INT</b>	Integers with signs	$\pm(2^{31} - 1)O$	<b>BOOL</b>	Truth value TRUE, FALSE	0,1	<b>CHAR</b>	1 ASCII character	0 - 255	<b>STRING</b>	Character string (max. 100 values)	0 - 255	<b>AXIS</b>	Axis addresses	Axis names only	<b>FRAME</b>	Geometric information	As for axis paths
Variable type	Property	Value range																							
<b>REAL</b>	Fractional number with dec. pt.	$\pm(2^{-1022} - 2^{+1023})$																							
<b>INT</b>	Integers with signs	$\pm(2^{31} - 1)O$																							
<b>BOOL</b>	Truth value TRUE, FALSE	0,1																							
<b>CHAR</b>	1 ASCII character	0 - 255																							
<b>STRING</b>	Character string (max. 100 values)	0 - 255																							
<b>AXIS</b>	Axis addresses	Axis names only																							
<b>FRAME</b>	Geometric information	As for axis paths																							
	Clear alarm with NC Start and continue processing.																								
<b>12600</b>	<b>Channel %1 block %2 invalid checksum of line</b>																								
Explanation:	%1 = Channel number %2 = Block number On processing an INI file or when executing a TEA file, an invalid line checksum has been detected.																								
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.																								
Remedy:	Correct INI file or correct MD and create new INI file (via „upload“). Steuerung AUS - EIN schalten.																								
<b>12610</b>	<b>Channel %1 block %2 accessing single char with call-by-reference argument not allowed %3</b>																								
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string An attempt has been made to use a single character access for a call-by-reference parameter.																								
Reaction:	Alarm display. Interface signals are set. Correction block.																								
Remedy:	Temporarily store single characters in user-defined CHAR variable and transfer this. Clear alarm with NC Start and continue processing.																								
<b>12620</b>	<b>Channel %1 block %2 accessing this variable as single char not allowed %3</b>																								
Explanation:	%1 = Channel number %2 = Block number, label %3 = Source string The variable is not a user-defined variable. The single character access is only allowed for user-defined variables (LUD/GUD).																								
Reaction:	Alarm display. Interface signals are set. Correction block.																								
Remedy:	Temporarily store variable in user-defined STRING, process this and put back into storage. Clear alarm with NC Start and continue processing.																								
<b>12630</b>	<b>Channel %1 block %2 skip / label not allowed</b>																								
Explanation:	%1 = Channel number %2 = Block number Blocks with control structures (FOR, ENDIF, etc.) cannot be concealed and must not contain any labels.																								
Reaction:	Alarm display. Interface signals are set. Correction block.																								
Remedy:	Teilprogramm korrigieren: Correct part program: Create conceal identifier by IF testing and write label on in its own in the block in front of the control structure block. Clear alarm with NC Start and continue processing.																								
<b>12640</b>	<b>Channel %1 block %2 invalid nesting of control structures</b>																								
Explanation:	%1 = Channel number %2 = Block number Error in program run: Opened control structures (IF-ELSE-ENDIF, LOOP-ENDLOOP etc.) are not terminated or there is no beginning of loop for the programmed end of loop. Example: LOOP ENDIF ENDLOOP																								
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.																								
Remedy:	Correct part program in such a way that all opened control structures are also terminated. Clear alarm with RESET key. Restart part program.																								



<b>12641</b>	<b>Channel %1 block %2 nesting level of control structures exceeds limit</b>
Explanation:	%1 = Channel number %2 = Block number Max. nesting depth control structures (IF-ELSE-ENDIF, LOOP-ENDLOOP etc.) exceeded. At the present time, the max. nesting depth is 8
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct part program. If necessary, move parts to a subroutine. Clear alarm with RESET key. Restart part program.
<b>12650</b>	<b>Channel %1 block %2 axis %3 name different in channel %4</b>
Explanation:	%1 = Channel number %2 = Block number %3 = Source symbol %4 = Channel number with different axis definition In cycles that are preprocessed at Power On, only those geometry and channel axis identifiers may be used that exist in all channels with the same meaning. In different channels, different axis indices are assigned to the axis identifier. The axis identifiers are defined via machine data 20060 AXCONF_GEOAX_NAME_TAB and 20080 AXCONF_CHANAX_NAME_TAB. Example: C is the 4th channel axis in channel 1 and the 5th channel axis in channel 2. If the axis identifier C is used in a cycle that is preprocessed at Power On, then this alarm is issued.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	1. Modify machine data: Select the same identifiers for geometry and channel axes in all channels. Example: The geometry axes are called X, Y, Z in all channels. They can then also be programmed directly in preprocessed channels. PROC DRILL G1 Z10 F1000 M17 or 2. Do not program the axis directly in the cycle but define it as parameter of the Axis type. Example: Cycle definition: PROC DRILL (AXIS DRILLAXIS) G1 AX[DRILLAXIS]=10 F1000 M17 Call from the main program: DRILL(Z) Clear alarm with RESET key. Restart part program.
<b>12661</b>	<b>Channel %1 block %2 technology cycle %3: no further program call possible</b>
Explanation:	%1 = Channel number %2 = Block number %3 = Name of the technology cycle call In a technology cycle it is not possible to call a subroutine or another technology cycle.
Reaction:	Alarm display. Interface signals are set. Correction block
Remedy:	Modify part program. Clear alarm with the RESET key.
<b>14000</b>	<b>Channel %1 block %2 Unzulaessiges Dateiende</b>
Explanation:	%1 = Channel number %2 = Block number, label Als Dateiende von Hauptprogrammen wird ein <b>M02</b> or ein <b>M30</b> erwartet, von Unterprogrammen <b>M17</b> . Von der Satzaufbereitung (Datenhaltung) wird kein Folgesatz geliefert, obwohl im vorhergehenden block kein Dateiende programmiert war.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Kontrollieren, ob das Programmende vergessen wurde einzugeben, or ob im letzten Programmsatz ein Sprung auf einen Programmabschnitt, in dem die Endekennung steht, erfolgt. Clear alarm with RESET key. Restart part program.
<b>14001</b>	<b>Channel %1 block %2 error at end of file, line feed missing</b>
Explanation:	%1 = Channel number %2 = Block number, label After system-internal data manipulation (e.g. when transferring blocks from an external source) a subfile can end without having LF as the last character.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Read out the part program, modify it with a text editor (e.g., insert blanks or comments before the displayed block), so that after reading it in again the part program has a different structure in the memory. Clear alarm with RESET key. Restart part program.
<b>14010</b>	<b>Channel %1 block %2 invalid default argument in subroutine call</b>
Explanation:	%1 = Channel number %2 = Block number, label In a subroutine call with parameter transfer, parameters have been omitted that cannot be replaced by default parameters (call-by-reference parameters or parameters of type AXIS. The other missing parameters are defaulted with the value 0 or with the unit frame in the case of frames).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	The missing parameters must be provided with values in the subroutine call. Clear alarm with RESET key. Restart part program.

<b>14011</b>	<b>Channel %1 block %2 program %3 not existing or not released for machining</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Program name An unknown identifier (string) was found in the part program. It is therefore assumed that this is a program name. The part program indicated in a subprogram call or SETINT statement does not exist or is not released for machining.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized.
Remedy:	The alarm may have different causes: - Typing error of the identifier stated in parameter 3 - Check subprogram call / SETINT statement or PROC statement. Reload part program and release for machining. - Parameter 3 can be a macro name. The macro definition file has an inappropriate content or it is not stored in the directory DEF_DIR or it has not been set active (via POWERON or via MMC operating step or by PI service 'F_COPY'). - Parameter 3 can be a GUD variable. There is no GUD definition file defining the variable or it is not stored in the directory DEF_DIR or it has not been set active (via the INITIAL_INI procedure or via MMC operating step or by PI service 'F_COPY'). Clear alarm with NC Start and continue program.
<b>14012</b>	<b>Channel %1 block %2 lowest subroutine level exceeded</b>
Explanation:	%1 = Channel number %2 = Block number, label The maximum nesting depth of 8 program levels has been exceeded. Subroutines can be called from the main program, and these in turn may have a nesting depth of 7.. <b>In interrupt routines the maximum number of levels is 4!</b>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify the machining program so that the nesting depth is reduced, e.g. using the editor copy a subroutine of the next nesting level into the calling program and remove the call for this subroutine. This reduces the nesting depth by one program level. Clear alarm with RESET key. Restart part program.
<b>14013</b>	<b>Channel %1 block %2 number of subroutine passes invalid</b>
Explanation:	%1 = Channel number %2 = Block number, label In a subroutine call the programmed number of passes P is zero or negative.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Program number of passes between 1 and 9 999. Clear alarm with RESET key. Restart part program.
<b>14014</b>	<b>Channel %1 block %2 selected program %3 or access permission not available</b>
Explanation:	%1 = Channel number The selected part program is not in the NCK memory or it is the access authorization for program selection at a higher level corresponding to the present status of the control. When this program was generated, it received the protection level that was active at the time for the NC control.
Reaction:	Alarm display.
Remedy:	Transfer the required program into the NCK memory or check the name of the directory (workpiece overview) and of the program (program overview) and correct these. Increase the present protection level to at least the level of the program being executed (by password input). Clear alarm with the Cancel key. No further operator action necessary.
<b>14015</b>	<b>Channel %1: no access permission for file</b>
Explanation:	%1 = Channel number A program is to be executed for which the current protection level is too low. When this program was generated, it received the protection level that was active at the time for the NC control.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Increase the present protection level to at least the level of the program being executed (by password input). Clear alarm with RESET key. Restart part program.
<b>14020</b>	<b>Channel %1 block %2 wrong number of arguments on function or procedure call</b>
Explanation:	%1 = Channel number %2 = Block number, label When a predefined function or procedure (subroutine) was called, the number of actual parameters was either • programmed <b>basically incorrectly</b> , e.g. in frames an odd number of parameters (except when mirroring), or • <b>too few parameters</b> were transferred. (Too many parameters are already recognized in the compiler, which then triggers alarm 11 039: "Channel %1 block %2 parameter number too large").
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct the number of transfer parameters in the call in the NC block. Clear alarm with RESET key. Restart part program.
<b>14021</b>	<b>Channel %1 block %2 wrong number of arguments on function or procedure call</b>
Explanation:	%1 = Channel number %2 = Block number, label In a function or procedure call, an impermissible number of actual parameters has been programmed.



Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable. Modify part program. Clear alarm with RESET key. Restart part program.
<b>14040</b>	<b>Channel %1 block %2 error in end point of circle</b>
Explanation:	%1 = Channel number %2 = Block number, label In circular interpolation, either the circle radii for the initial point and the end point are further apart, or the circle center points are further apart, than specified in the machine data. 1. In circle radius programming the starting and end points are identical, thus the circle position is not determined by starting and end points. 2. <b>Radii:</b> The NCK calculates from the present start point and the other programmed circle parameters the radii for the start and the end point. An alarm message is issued if the difference between the circle radii is either greater than the value in the MD 21000 <b>CIRCLE_ERROR_CONST</b> (for small radii, if the programmed radius is smaller than the quotient of the machine data <b>CIRCLE_ERROR_CONST</b> divided by 21010 <b>CIRCLE_ERROR_FACTOR</b> ), or greater than the programmed radius multiplied by the MD <b>CIRCLE_ERROR_FACTOR</b> (for large radii, if the programmed radius is greater than the quotient of the machine data <b>CIRCLE_ERROR_CONST</b> divided by <b>CIRCLE_ERROR_FACTOR</b> ). 3. <b>Center points:</b> A new circle center is calculated using the circle radius at the starting position. It lies on the mid-perpendicular positioned on the connecting straight line from the starting point to the end point of the circle. The angle in the radian measure between both straight lines from the starting point to the center calculated/programmed as such must be lower than the root of 0.001 (corresponding to approx. 1.8 degrees).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Check MD 21000 <b>CIRCLE_ERROR_CONST</b> and 21010 <b>CIRCLE_ERROR_FACTOR</b> . If the values are within reasonable limits, the circle end point or the circle mid-point of the part program block must be programmed with greater accuracy. Clear alarm with RESET key. Restart part program.
<b>14045</b>	<b>Channel %1 block %2 error in tangent circle programming</b>
Explanation:	%1 = Channel number %2 = Block number, label The alarm may have the following causes: - The tangent direction is not defined for tangent circle / e.g. because no other travel block has been programmed before the current block. - No circle can be formed from start and end point as well as tangent direction because - seen from the start point - the end point is located in the opposite direction to that indicated by the tangent. - It is not possible to form a tangent circle since the tangent is located vertically to the active plane. - In the special case in which the tangent circle changes to a straight line, several complete circular revolutions were programmed with TURN.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable. NC Stop when alarm at block end.
Remedy:	Modify part program. Clear alarm with NC Start and continue program.
<b>14050</b>	<b>Channel %1 block %2 nesting depth for arithmetic operations exceeded</b>
Explanation:	%1 = Channel number %2 = Block number, label For calculating arithmetic expressions in NC blocks, an operand stack with a fixed set size is used. In very complex expressions, this stack can overflow.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Divide up complex arithmetic expressions into several simpler arithmetic blocks. Clear alarm with RESET key. Restart part program.
<b>14051</b>	<b>Channel %1 block %2 arithmetic error in part program</b>
Explanation:	%1 = Channel number %2 = Block number, label • In calculating an arithmetic expression, an overflow has occurred (e.g. division by zero). • In a data type, the representable value range has been exceeded
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized.
Remedy:	Analyze the program and correct the defective point in the program. Clear alarm with NC Start and continue program.
<b>14060</b>	<b>Channel %1 block %2 invalid skip level with differential block skip</b>
Explanation:	%1 = Channel number %2 = Block number, label With „Differential block skip“, a skip level greater than 7 has been specified (in packet 1 specification of a value for the skip level is rejected by the converter as a <b>syntax error</b> , i.e. the only possibility is a "Suppress block" ON/OFF on one level).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Enter a skip level (number behind the slash) less than 8. Clear alarm with RESET key. Restart part program.
<b>14070</b>	<b>Channel %1 block %2 memory for variables not sufficient for subroutine call</b>
Explanation:	%1 = Channel number %2 = Block number, label A called subroutine cannot be processed (opened), either because the internal data memory to be created for

general purposes is not large enough, or because the available memory for the local program variables is too small. The alarm can only occur in MDA mode.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Abschnitt des Teileprogramms analysieren:

1. Has the most useful data type always been selected in the variable definitions? (For example REAL for data bits is poor; BOOL would be better)
  2. Can local variables be replaced by global variables?
- Clear alarm with RESET key. Restart part program.

#### 14080 Channel %1 block %2 jump destination not found

Explanation: %1 = Channel number

%2 = Block number, label

In conditional and unconditional jumps, the jump destination within the program must be a block with a **label** (symbolic name instead of block number). If no jump destination has been found with the given label when searching **in the programmed direction**, an alarm is output.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Check NC part program for the following possible errors:

1. Check whether the target designation is identical with the label.
  2. Is the jump direction correct?
  3. Has the label been terminated with a colon?
- Clear alarm with RESET key. Restart part program.

#### 14090 Channel %1 block %2 invalid D number

Explanation: %1 = Channel number

%2 = Block number, label

A value less than zero has been programmed under address D.

A set of parameters with 25 correction values has been automatically assigned to each active tool. Each tool can have 9 sets of parameters (D1 - D9, initial setting is D1). When the D number changes, the new parameter set is active (D0 is used for deselecting the correction values).

N10 G.. X... Y... T15 Parameter set D1 of T15 active

N50 G.. X... D3 M.. Parameter set D3 of T15 active

N60 G.. X.. T20 Parameter set D1 of T20 active

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Program D numbers in the permissible value range (D0, D1 to D9).

Clear alarm with RESET key. Restart part program.

#### 14091 Channel %1 block %2 invalid function, index %3

Explanation: %1 = Channel number

%2 = Block number, label

Programming RET in the 1st program level.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Select G functions in keeping with the possibilities provided by the NCK.

Clear alarm with RESET key. Restart part program.

#### 14092 Channel %1 block %2 axis %3 has wrong axis type

Explanation: %1 = Channel number

%2 = Block number, label

%3 = Axis name, spindle number

One of the following three programming errors has occurred:

1. The keyword **WAITP(x)** "Wait with block change until the specified positioning axis has reached its end point" has been used for an axis that is not a positioning axis.
2. **G74** "Reference point approach from the program" has been programmed for a spindle. (Only axis addresses are permitted.)
3. The keyword **POS/POSA** has been used for a spindle. (The keywords SPOS and SPOSA must be programmed for the spindle positions.)

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Correct the part program depending on which of the above errors is involved.

Clear alarm with RESET key. Restart part program.

#### 14093 Channel %1 block %2 path interval zero or negative with polynomial interpolation

Explanation: %1 = Channel number

%2 = Block number, label

In the polynomial interpolation POLY, a negative value or zero has been programmed under the keyword for the polynomial length **PL=.....**

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. Correct the value given in **PL = ....**

Clear alarm with RESET key. Restart part program.

#### 14094 Channel %1 block %2 polynomial degree greater than 3 programmed for polynomial interpolation

Explanation: %1 = Channel number

%2 = Block number, label

The polynomial degree in the polynomial interpolation is based on the number of programmed coefficients for an axis. The maximum possible polynomial degree is 3, i.e. the axes are according to the function:

$$f(p) = a_0 + a_1 p + a_2 p^2 + a_3 p^3$$

The coefficient  $a_0$  is the actual position at the start of interpolation and is not programmed!

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Reduce the number of coefficients. The polynomial block may have a form no greater than the following:  
 N1 POLY PO[X]=(1.11, 2.22, 3.33) PO[Y]=(1.11, 2.22, 3.33)  
 N1 PO[n]=... PL=44  
 n ... axis identifier, max. 8 path axes per block  
 Clear alarm with RESET key. Restart part program.

**14095 Channel %1 block %2 circle programmed with zero radius**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 The radius entered for radius programming is too small, i.e. the programmed radius is smaller than half of the distance between start and end point.

Reaction: Alarm display. Interface signals are set. Correction block is reorganized.

Remedy: Modify part program  
 Clear alarm with NC Start and continue program.

**14096 Channel %1 block %2 type conversion not possible**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 During the program run, a variable value assignment or an arithmetic operation has caused data to be processed in such a way that they have to be converted to another type. This would lead to the value range being exceeded.

Variable type	Property	Value range
REAL	Fractional numbers with dec. pt.	$\pm(2^{-1022} - 2^{1023})$
INT	Integers with signs	$\pm(2^{31} - 1)O$
BOOL	Truth value TRUE, FALSE	0,1
CHAR	1 ASCII character	0 - 255
STRING	Character string (max. 100 values)	0 - 255
AXIS	Axis addresses	Axis names only
FRAME	Geometric information	As for axis paths

from - to	REAL	INT	BOOL	CHAR	STRING
REAL		yes*	yes	yes**	-
INT	yes		yes	yes**	-
BOOL	yes	yes		yes	-
CHAR	yes	yes	yes**		yes
STRING	-	-	yes	yes***	

\* Value <>0 corresponds to TRUE, value ==0 corresponds to FALSE.  
 \*\* String length 0 => FALSE, otherwise TRUE  
 \*\*\* If only one character

It is not possible to convert from type AXIS and FRAME nor into type AXIS and FRAME.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Modify the program section such that the value range is not exceeded, e.g. by a modified variable definition.  
 Clear alarm with RESET key. Restart part program.

**14097 Channel %1 block %2 string cannot be converted to AXIS type**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 The called function AXNAME - conversion of the transferred parameters of the STRING type to an axis name (return value) of the AXIS type - has not found this axis identifier in the machine data.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Check the transferred parameters (axis name) of the function AXNAME to determine whether a geometry, channel or machine axis of this name has been configured by means of the machine data:  
 10 000: AXCONF\_MACHAX\_NAME\_TAB  
 20 070: AXCONF\_GEOAX\_NAME\_TAB  
 20 080: AXCONF\_CHANAX\_NAME\_TAB  
 Select the transfer string in accordance with the axis name and change the axis name in the machine data if necessary. (If a change of name is to take place via the NC part program, this change must first be validated by means of a „Power On“.)  
 Clear alarm with RESET key. Restart part program.

**14098 Channel %1 block %2 conversion error: not a number**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 The string is not a valid INT or REAL number.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Modify part program. If an input is concerned, it is possible to test whether the string represents a number by means of the predefined function ISNUMBER (with the same parameter).  
 Clear alarm with RESET key. Restart part program.

**14099 Channel %1 block %2 result in string concatenation too long**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 The result of string chaining returns a string which is greater than the maximum string length laid down by the system.

Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Teileprogramm anpassen. Adapt part program. With the function STRLEN, it is also possible to test the size of the sum string before performing the chaining operation. Clear alarm with RESET key. Restart part program.
<b>14100</b>	<b>Channel %1 block %2 orientation transformation not available</b>
Explanation:	%1 = Channel number %2 = Block number, label Four transformation groupings (transformation types) can be set for each channel via machine data. If a transformation grouping is addressed by means of the keyword <b>TRAORI(n)</b> (n ... number of transformation grouping) but for which the machine data have no default values, then an alarm message is issued.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Press the NC Stop key and select the function „Correction block“ with the softkey PROGRAM CORRECT. The correction pointer positions on the incorrect block. • Check the number of the transformation grouping when calling the part program with the keyword <b>TRAORI(n)</b> (n ... number of the transformation grouping). • Enter the machine data for this transformation grouping and then activate by “Power On“. Clear alarm with RESET key. Restart part program.
<b>14115</b>	<b>Channel %1 block %2 illegal definition of part surface</b>
Explanation:	%1 = Channel number %2 = Block number, label The surface normal vectors programmed at the beginning of block and at the end of block point in opposite directions.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program Clear alarm with RESET key. Restart part program.
<b>14130</b>	<b>Channel %1 block %2 too many initialization values given</b>
Explanation:	%1 = Channel number %2 = Block number, label On assigning an array by means of SET, more initialization values than existing array elements have been specified in the program run.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Reduce the number of initialization values. Clear alarm with RESET key. Restart part program.
<b>14150</b>	<b>Channel %1 block %2 illegal tool carrier number programmed or declared (MD)</b>
Explanation:	%1 = Channel number %2 = Block number, label A toolholder number was programmed which is negative or greater than the machine data MC_MM_NUM_TOOL_CARRIER.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable
Remedy:	Program valid toolholder number or adapt machine data MC_MM_NUM_TOOL_CARRIER. Mit Reset-Taste Alarm löschen.
<b>14200</b>	<b>Channel %1 block %2 polar radius negative</b>
Explanation:	%1 = Channel number %2 = Block number, label In the endpoint specification of a traversing block with G00, G01, G02 or G03 in polar coordinates, the polar radius entered for the keyword <b>RP=...</b> is negative. Definition of terms: • <b>Specification of end of block point</b> with polar angle and polar radius, referring to the current pole (preparatory functions: G00/G01/G02/G03). • <b>New definition of the pole</b> with polar angle and pole radius, referring to the reference point selected with the G function. G110 ... last programmed point in the plane G111 ... zero point in the actual WCS G112 ... last pole
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct NC part program - permissible inputs for the pole radius are only positive absolute values that specify the distance between the current pole and the block end point (the direction is defined by the polar angle <b>AP=...</b> ). Clear alarm with RESET key. Restart part program.
<b>14210</b>	<b>Channel %1 block %2 polar radius too large</b>
Explanation:	%1 = Channel number %2 = Block number, label In specifying the endpoints in a traversing block with G00, G01, G02 or G03 in polar coordinates, the value range of the polar angle programmed under the keyword <b>AP=...</b> has been exceeded. It covers the range from -360 to +360 degrees with a resolution of 0.0 01 degrees. Definition of terms: • <b>Specification of end of block point</b> with polar angle and polar radius, referring to the current pole (preparatory functions: G00/G01/G02/G03).

	<ul style="list-style-type: none"> <li>• <b>New definition of the pole</b> with polar angle and pole radius, referring to the reference point selected with the G function.</li> </ul> <p>G110 ... referred to the last programmed point in the plane  G111 ... referred to the zero point of the current workpiece coordinate system (WCS)  G112 ... referred to the last pole</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct NC part program. The permissible input range for the polar angle is between the values -360 degrees and +360 degrees with a resolution of 0.001 degrees. Clear alarm with RESET key. Restart part program.
<b>14250</b>	<b>Channel %1 block %2 pole radius negative</b>
Explanation:	%1 = Channel number %2 = Block number, label in redefining the pole with G110, G111 or G112 in polar coordinates, the pole radius specified under keyword <b>RP=...</b> is negative. Only positive absolute values are permitted. Definition of terms: <ul style="list-style-type: none"> <li>• <b>Specification of end of block point</b> with polar angle and polar radius, referring to the current pole (preparatory functions: G00/G01/G02/G03).</li> <li>• <b>New definition of the pole</b> with polar angle and pole radius, referring to the reference point selected with the G condition.</li> </ul> <p>G110 ... last programmed point in the plane  G111 ... zero point of the current workpiece coordinate system (WCS)  G112 ... last pole</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct the NC part program. Permissible inputs for the pole radius are only positive, absolute values that specify the distance between the reference point and the new pole (the direction is defined with the pole angle <b>AP=...</b> ). Clear alarm with RESET key. Restart part program.
<b>14260</b>	<b>Channel %1 block %2 pole angle too large</b>
Explanation:	%1 = Channel number %2 = Block number, label In redefining the pole with G110, G111 or G112 in polar coordinates, the value range of the pole angle specified under keyword <b>AP=...</b> has been exceeded. It covers the range from -360 to +360 degrees with a resolution of 0.001 degrees. Definition of terms: <ul style="list-style-type: none"> <li>• <b>Specification of end of block point</b> with pole angle and pole radius, referring to the current pole (preparatory functions: G00/G01/G02/G03).</li> <li>• <b>New definition of the pole</b> with pole angle and pole radius, referring to the reference point selected with the G function.</li> </ul> <p>G110 ... last programmed point in the plane  G111 ... zero point of the current workpiece coordinate system (WCS)  G112 ... last pole</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct NC part program. The permissible input range for the polar angle is between the values -360 degrees and +360 degrees with a resolution of 0.001 degrees. Clear alarm with RESET key. Restart part program.
<b>14270</b>	<b>Channel %1 block %2 pole programmed incorrectly</b>
Explanation:	%1 = Channel number %2 = Block number, label When defining the pole, an axis was programmed that does not belong to the selected processing level. Programming in polar coordinates always refers to the plane activated with G17 to G19. This also applies to the definition of a new pole with G110, G111 or G112.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct the NC part program. Only the two geometry axes may be programmed that establish the current machining plane. Clear alarm with RESET key. Restart part program.
<b>14280</b>	<b>Channel %1 block %2 polar coordinates programmed incorrectly</b>
Explanation:	%1 = Channel number %2 = Block number, label The end point of the displayed block has been programmed both in the polar coordinate system (with <b>AP=...</b> , <b>RP=...</b> ) and in the Cartesian coordinate system (axis addresses X, Y,...).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	NC Correct the NC part program - the axis motion may be specified in <b>one coordinate system only</b> . Clear alarm with RESET key. Restart part program.
<b>14300</b>	<b>Channel %1 block %2 overlaid handwheel motion activated incorrectly</b>
Explanation:	%1 = Channel number %2 = Block number, label Handwheel override has been called up incorrectly: 1. For positioning axes: - Handwheel override programmed for indexing axes, - No position programmed, - FA and FDA programmed for the same axis in the block. 2. For contouring axes:

	<ul style="list-style-type: none"> <li>- No position programmed,</li> <li>- G60 not active,</li> <li>- 1st G group incorrect (only G01 to CIP)</li> </ul>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Clear alarm with RESET key. Restart part program.
<b>14310</b>	<b>Handwheel %1 configuration not correct or inactive</b>
Explanation:	%1 = handwheel number <ul style="list-style-type: none"> <li>• The inputs are using a drive with a drive number that does not exist or</li> <li>• an inactive drive for assignment of the handwheel (ENC_HANDWHEEL_MODULE_NR), or</li> <li>• an axis is using a measuring circuit which does not exist for the drive hardware.</li> </ul>
Reaction:	Alarm display. Interface signals are set. NC Start disable
Remedy:	Check input configuration (machine data) and/or drive hardware. Runup is interrupted. Switch control OFF - ON.
<b>14400</b>	<b>Channel %1 block %2 tool radius compensation active at transformation switchover</b>
Explanation:	%1 = Channel number %2 = Block number, label A change of transformation is not allowed when tool radius compensation is active.
Reaction:	Perform tool radius compensation in the NC part program with <b>G40</b> (in a block with G00 or G01) before performing a transformation change. Clear alarm with RESET key. Restart part program.
<b>14401</b>	<b>Channel %1 block %2 transformation not available</b>
Explanation:	%1 = Channel number %2 = Block number, label The required transformation is not available. Example: This was programmed: N220 TRAORI(3); 5-axis transform. no. 3 ON but only transformation 1 and 2 exist
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	<ul style="list-style-type: none"> <li>• Modify part program, program defined transformations only.</li> <li>• Check MD 24100 TRAFO_TYPE_n (assigns the transformation to part program instructions).</li> </ul> Clear alarm with RESET key. Restart part program.
<b>14403</b>	<b>Channel %1 block %2 preparation might not be synchronized with interpolation</b>
Explanation:	%1 = Channel number %2 = Block number, label Positioning axis runs cannot be accurately calculated beforehand. Consequently, the position in the MCS is not known exactly. It might therefore be possible that a change in the multiple significance of the transformation has been performed in the main run although no provision was made for this in the preprocessing run.
Reaction:	Alarm display.
Remedy:	Modify part program. Synchronize preprocessing run and main run. Clear alarm with the Cancel key. No further operator action necessary.
<b>14404</b>	<b>Channel %1 block %2 invalid argument in selection of transformation</b>
Explanation:	%1 = Channel number %2 = Block number, label Error has occurred when selecting transformation. Possible causes of error: <ul style="list-style-type: none"> <li>• An axis traversed by the transformation has not been enabled: · is being used by another channel (-&gt; enable)</li> <li>• is in spindle mode (-&gt; enable with SPOS)</li> <li>• is in POSA mode (-&gt; enable with WAITP)</li> <li>• is competing Pos axis (enable with -&gt; WAITP)</li> <li>• Parameterization via machine data has an error</li> <li>• Axis or geometry axis assignment to the transformation has an error,</li> <li>• Machine data has an error (-&gt; modify machine data, cold restart)</li> </ul> Note: Any axes that have not been enabled might be signaled via EXINAL_ILLEGAL_AXIS = 14092 or BSAL_SYSERRCHAN_RESET = 1011 instead of EXINAL_TRANSFORM_PARAMETER = 14404. Transformation-dependent error causes can be in: TRAORI: -TRANSMIT: <ul style="list-style-type: none"> <li>• The current machine axis position is unsuitable for selection (e.g. selection in the pole) (-&gt; change position slightly)</li> <li>• Parameterization via machine data has an error</li> <li>• Special requirement with respect to the machine axis has not been satisfied (e.g. rotary axis is not a modulo axis) (-&gt; modify machine data, cold restart)</li> </ul> TRACYL: <ul style="list-style-type: none"> <li>• The programmed parameter is not allowed when transformation is selected.</li> </ul> TRAANG:



	<ul style="list-style-type: none"> <li>• The programmed parameter is not allowed when transformation is selected.</li> <li>• Parameterization via machine data has an error</li> <li>• Parameter has an error (e.g. TRAANG: unfavorable angular value) (-&gt; modify machine data, cold restart)</li> </ul>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program or machine data. Clear alarm with RESET key. Restart part program.
<b>14411</b>	<b>Channel %1 block %2 tool radius compensation active at change of geoaxis</b>
Explanation:	%1 = Channel number %2 = Block number, label It is not permissible to change the assignment of geometry axes to channel axes when tool radius compensation is active.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Clear alarm with RESET key. Restart part program.
<b>14412:</b>	<b>Channel %1 block %2 transformation active at change of geoaxis</b>
Explanation:	%1 = Channel number %2 = Block number, label It is not permissible to change the assignment of geometry axes to channel axes when transformation is active.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Clear alarm with RESET key. Restart part program.
<b>14413</b>	<b>Channel %1 block %2 fine tool correction: changeover geometry / channel</b>
Explanation:	%1 = Channel number %2 = Block number, label It is not permissible to change the assignment of geometry axes to channel axes during active tool fine compensation.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable
Remedy:	Modify part program Clear alarm with RESET key.
<b>14414</b>	<b>Channel %1 block %2 function GEOAX: incorrect call</b>
Explanation:	%1 = Channel number %2 = Block number, label The parameters for the GEOAX(...) call are incorrect. Possible causes are: - Uneven number of parameters. - More than 6 parameters were specified. - A geometry axis number was programmed which was smaller than 0 or greater than 3. - A geometry number was programmed more than once. - An axis identifier was programmed more than once. - An attempt was made to assign a channel axis to a geometry axis which has the same name as one of the channel axes. - An attempt was made to remove a geometry axis from the geometry axis grouping and the geometry axis has the same name as one of the channel axes.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable
Remedy:	Modify part program or correction block Cancel alarm with the Cancel key. No further operator action necessary.
<b>14420</b>	<b>Channel %1 block %2 index axis %3 frame not allowed</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = axis The axis is to be traversed as an indexing axis, but a frame is active. This is not allowed by machine data FRAME_OR_CORRPOS_NOTALLOWED.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable
Remedy:	Modify part program, change machine data CORR_FOR_AXIS_NOT_ALLOWED Clear alarm with RESET key. Restart part program.
<b>14500</b>	<b>Channel %1 block %2 illegal DEF or PROC statement within part program</b>
Explanation:	%1 = Channel number %2 = Block number, label NC part programs with high-level language elements are divided into a preceding <b>definition part</b> followed by a <b>program part</b> . The transition is not marked specifically; a definition statement is not allowed to follow the first program command.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Put definition and PROFC statements at the beginning of the program. Clear alarm with RESET key. Restart part program.
<b>14510</b>	<b>Channel %1 block %2 PROC statement missing on subroutine call</b>
Explanation:	%1 = Channel number %2 = Block number, label subroutine calls with parameter transfer („call-by-value“ or „call-by-reference“) the called subroutine must begin with a PROC statement.

Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Define the subroutine in accordance with the type used. 1.. Conventional subroutine structure (without parameter transfer): % SPF 123456 : M17 2.. Subroutine structure with keyword and subroutine name (without parameter transfer): PROC UPNAME : M17 ENDPROC 3. Subroutine structure with keyword and subroutine name (with parameter transfer "call-by-value"): PROC UPNAME (VARNAME1, VARNAME2, ...) : M17 ENDPROC 4. Subroutine structure with keyword and subroutine name (with parameter transfer "call-by-reference"): PROC UPNAME (Typ1 VARNAME1, Typ2 VARNAME2, ...) : M17 ENDPROC Clear alarm with RESET key. Restart part program.
<b>14520</b>	<b>Channel %1 block %2 illegal PROC statement in data definition section</b>
Explanation:	%1 = Channel number %2 = Block number, label The PROC statement may only be programmed at the beginning of the subroutine.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify NC part program appropriately. Clear alarm with RESET key. Restart part program.
<b>14530</b>	<b>Channel %1 block %2 EXTERN and PROC statement do not correspond</b>
Explanation:	%1 = Channel number %2 = Block number, label Subroutines with parameter transfer must be known <b>before</b> they are called in the program. If the subroutines are always available (fixed cycles) the control establishes the call interfaces at the time of system power-up. Otherwise an EXTERN statement must be programmed in the calling program. Example: N123 EXTERN UPNAME (TYPE1, TYPE2, TYPE3, ...) The type of the variable must definitely correspond to the type given in the definition (PROC statements) or it must be compatible with it. The name can be different.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Check the variable types in the EXTERN and the PROC statements for correspondence and correct. Clear alarm with RESET key. Restart part program.
<b>14610</b>	<b>Channel %1 block %2 compensation block not possible</b>
Explanation:	%1 = Channel number %2 = Block number, label An alarm was output which could be eliminated basically via program correction. Since the error occurred in a program which is processed from external, a compensation block/program correction is not possible.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	- Abort program with reset. - Correct program on MMC or PC. - Restart reloading (possibly with block search and interrupt location). Clear alarm with RESET key. Restart part program.
<b>14660</b>	<b>Channel %1 block %2 SETINT instruction uses with invalid input to trigger ASUP</b>
Explanation:	%1 = Channel number %2 = Block number, label Asynchronous subroutines are subroutines that are executed following a hardware input (interrupt routine started by a rapid NCK input). The number of the NCK input must be between 1 and 8. It is provided with the keyword PRIO = ... with a priority of 1 - 128 (1 is the highest priority) in the SETINT statement. Example: If NCK input 5 changes to „1“ the subroutine LIFT_Z should be started with the highest priority. N100 SETINT (5) PRIO = 1 ABHEB_Z
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
REMEDY:	Program the NCK input of the SETINT statement with a value of not less than 1 or greater than 128. Clear alarm with RESET key. Restart part program.
<b>14750</b>	<b>Channel %1 block %2 too many auxiliary functions programmed</b>
Explanation:	%1 = Channel number %2 = Block number, label More than 10 auxiliary functions have been programmed in an NC block.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized.
Remedy:	Check whether all auxiliary functions are necessary in one block - modal functions need not be repeated. Create



separate auxiliary function block or divide the auxiliary functions over several blocks.  
Clear alarm with RESET key. Restart part program.

**14760**

Explanation:

**Channel %1 block %2 auxiliary function of a group programmed repeatedly**

%1 = Channel number

%2 = Block number, label

The M and H functions can be divided up as required over machine data in groups in any variation. Auxiliary functions are thus put into groups that mutually preclude several individual functions of one group. Within one group only one auxiliary function is advisable and permissible.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Program only one auxiliary function per auxiliary function group (group allocations: refer to the machine manufacturer's programming guide).

Clear alarm with RESET key. Restart part program.

**14770**

Explanation:

**Channel %1 block %2 auxiliary function programmed incorrectly**

%1 = Channel number

%2 = Block number, label

The permissible number of programmed auxiliary functions per NC block has been exceeded or more than one auxiliary function of the same auxiliary function group has been programmed (M and S function). In the user-defined auxiliary functions, the maximum number of auxiliary functions per group in the NCK system settings has been defined for all auxiliary functions by means of the machine data 11100 AUXFU\_MAXNUM\_GROUP\_ASSIGN (default: 1).

For each user-defined auxiliary function to be assigned to a group, the assignment is effected through 4 channel-specific machine data.

22010 AUXFU\_ASSIGN\_TYPE: type of auxiliary function, e.g. M

22000 AUXFU\_ASSIGN\_GROUP: required group

22020 AUXFU\_ASSIGN\_EXTENSION: any required extension

22030 AUXFU\_ASSIGN\_VALUE: function value

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Correct the part program - max. 16 auxiliary functions, max. 5 M functions per NC block, max. 1 auxiliary function per group.

Clear alarm with RESET key. Restart part program.

**14820**

Explanation:

**Channel %1 block %2 negative value for maximum spindle speed programmed with constant cutting speed**

%1 = Channel number

%2 = Block number, label

For the function "Constant cutting speed G96" a maximum spindle speed can be programmed with the keyword LIMS=.... The values are in the range 0.1 -999 999.9 [rev/min].

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Program the maximum spindle speed for the constant cutting speed within the limits given above. The keyword LIMS is modal and can either be placed in front of or within the block that selects the constant cutting speed.

Clear alarm with RESET key. Restart part program.

**14830**

Explanation:

**Channel %1 block %2 wrong feed type selected**

%1 = Channel number

%2 = Block number, label

Im G97 has been programmed in the displayed block although G96 was not (or G97 already) active previously.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Remove G97 from the displayed block and program the correct feed type (G93, G94, G95 or G96) for the machining section which follows.

Clear alarm with RESET key. Restart part program.

**14840**

Explanation:

**Channel %1 block %2 value for constant cutting speed out of range**

%1 = Channel number

%2 = Block number, label

The programmed cutting speed is not within the input range

Input range metric: 0.01 to 9 999.99 [m/min] Input range inch: 0.1 to 99 999.99 [inch/min]

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Program cutting speed under address S within the permissible range of values.

Clear alarm with RESET key. Restart part program.

**14900**

Explanation:

**Channel %1 block %2 use either center point or end point programming**

%1 = Channel number

%2 = Block number, label

When programming a circle by means of the opening angle, the circle center point was programmed together with the circle end point. This is too much information for the circle. Only one of the two points is allowed.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Select the programming variant guaranteeing that the dimensions are definitely taken over from the workpiece drawing (avoidance of calculation errors).

Clear alarm with RESET key. Restart part program.

**14910**

Explanation:

**Channel %1 block %2 invalid angle of aperture for programmed circle**

%1 = Channel number

%2 = Block number, label

When programming a circle by means of the opening angle, a negative opening angle or an opening angle greater than or equal to 360 degrees has been programmed.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:	Program opening angle within the allowed range of values between 0.0001 and 359.9999 [degrees]. Clear alarm with RESET key. Restart part program.
<b>14920</b>	<b>Channel %1 block %2 intermediate point of circle incorrect</b>
Explanation:	%1 = Channel number %2 = Block number, label When programming a circle by means of an intermediate point (CIP) all 3 points (initial, end and intermediate points) are on a straight line and the intermediate point (programmed by means of interpolation parameters I, J, K) is not located between the initial and end points. If the circle is the component of a helix, the specified number of turns (keyword TURN=...) determines further block processing: • TURN>0: alarm display because the circle radius is infinitely great. • TURN=0 and CIP specified between initial and end points. A straight line is generated between the initial and end points (without alarm message).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Locate the position of the intermediate point with the parameters I, J and K in such a way that it actually is located between the initial and end points of the circle or do not make use of this type of circle programming and instead program the circle with radius or opening angle or center point parameters. Clear alarm with RESET key. Restart part program.
<b>15010</b>	<b>Channel %1 block %2 channel-sync instruction using illegal mark</b>
Explanation:	%1 = Channel number %2 = Block number, label A WAITM/WAITMC/SETM/CLEARM instruction was programmed with a marker number
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct the instruction accordingly. Clear alarm with RESET key. Restart part program.
<b>15180</b>	<b>Channel %1 block %2 program %3 cannot be executed as INI file</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = string Errors occurred when reading in as INI file. The error message which is then displayed refers to the program specified here.
Reaction:	Alarm display.
Remedy:	Correct the part program. Clear alarm with the Cancel key. No further operator action necessary.
<b>15185</b>	<b>Channel %1 %2 errors in INI file</b>
Explanation:	%1 = Channel number %2 = Number of detected errors An error was found when processing an INI file
Reaction:	Alarm display. Interface signals are set. NC Start disable.
Remedy:	Correct the INI file or correct the MD and create a new INI file (via „Upload“). Switch control OFF-ON.
<b>15300</b>	<b>Channel %1 block %2 invalid number-of-passed blocks during block search</b>
Explanation:	%1 = Channel number %2 = Block number, label In the function „Block search with calculation“ a negative number of passes has been entered in column P (number of passes). The permissible range of values is P 1 - P 9 999.
Reaction:	Alarm display.
Remedy:	Enter only positive number of passes within the range of values. Clear alarm with Cancel key. No further operator action necessary.
<b>15310</b>	<b>Channel %1 block %2 file requested during block search is not loaded</b>
Explanation:	%1 = Channel number %2 = Block number, label During block search, a target has been specified with a program that has not been loaded
Reaction:	Alarm display.
Remedy:	Correct the specified search target accordingly or reload the file Clear alarm with the Cancel key. No further operator action necessary.
<b>15320</b>	<b>Channel %1 block %2 invalid block search command</b>
Explanation:	%1 = Channel number %2 = Block number, label The block search command (type of search target) is smaller than 1 or greater than 5. It is entered in column <b>type</b> of the block search window. The following block search orders are allowed. Type Meaning 1 Search for block number 2 Search for label 3 Search for string 4 Search for program name 5 Search for line number in a file
Reaction:	Alarm display.
Remedy:	Modify the block search command. Clear alarm with the Cancel key. No further operator action necessary.

<b>15330</b>	<b>Channel %1 block %2 invalid block number as target of block search</b>
Explanation:	%1 = Channel number %2 = Block number, label Syntax error! Positive integers are allowed as block numbers. Block numbers must be preceded by „:“ and subblocks by an „N“.
Reaction:	Alarm display.
Remedy:	Repeat the input with corrected block number. Clear alarm with the Cancel key. No further operator action necessary.
<b>15340</b>	<b>Channel %1 block %2 invalid label as target of block search</b>
Explanation:	%1 = Channel number %2 = Block number, label Syntax error! A label must have at least 2 but no more than 32 characters, and the first two characters must be alphabetic or underscore characters. Labels must be concluded with a colon.
Reaction:	Alarm display.
Remedy:	Repeat the input with corrected label. Clear alarm with the Cancel key. No further operator action necessary.
<b>15350</b>	<b>Channel %1 block %2 target of block search not found</b>
Explanation:	%1 = Channel number %2 = Block number, label The specified program has been searched to the end of the program without the selected search target having been found.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Check the part program, change the block search (typing error in the part program) and restart the search. Clear alarm with RESET key. Restart part program.
<b>15360</b>	<b>Channel %1 invalid target of block search (syntax error)</b>
Explanation:	%1 = Channel number The specified search target (block number, label or string) is not allowed in block search.
Reaction:	Alarm display.
Remedy:	Correct object of block search. Clear alarm with the Cancel key. No further operator action necessary.
<b>15370</b>	<b>Channel %1 target of block search not found</b>
Explanation:	%1 = Channel number In a block search, an impermissible search target has been specified (e.g. negative block number).
Reaction:	Alarm display.
Remedy:	Check the specified block number, label or character string. Repeat entry with correct search target. Clear alarm with the Cancel key. No further operator action necessary.
<b>15400</b>	<b>Channel %1 block %2 selected initial ini file does not exist</b>
Explanation:	%1 = Channel number %2 = Block number, label The operator has selected an INI block for a read, write or execution function which: 1. Does not exist in the NCK range or 2. Does not have the necessary protection level required for performing the function
Reaction:	Alarm display.
Remedy:	Check whether the selected INI block is contained in the file system of the NCK. The present protection level must be selected to be at least equal to (or greater than) the protection level that has been defined for the read, write or execution function at the time of creating the file. Clear alarm with RESET key. Restart part program.
<b>15410</b>	<b>Channel %1 block %2 initialization file contains invalid M function</b>
Explanation:	%1 = Channel number %2 = Block number, label The only M function allowed in an Init block is the M02, M17 or M30 end-of-program function.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Remove all M functions from the Init block except for the end identifier. An Init block may contain value assignments only (and global data definitions if they are not defined again in a program that can be executed later) but no motion or synchronous actions. Clear alarm with RESET key. Restart part program.
<b>15420</b>	<b>Channel %1 block %2 instruction not accepted in current mode</b>
Explanation:	%1 = Channel number %2 = Block number, label In executing an Init block, the interpreter encountered an impermissible statement (e.g. a traversing statement).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Remove all motion actions and auxiliary functions from the Init block except for the end identifier. An Init block may contain value assignments only (and global data definitions if they are not defined again in a program that can be executed later) but no motion or synchronous actions. Clear alarm with RESET key. Restart part program.
<b>15460</b>	<b>Channel %1 block %2 syntax conflict with modal G functions</b>
Explanation:	%1 = Channel number %2 = Block number, label The addresses programmed in the block are not compatible with the modal syntax-determining G function.

	Example: N100 G01 ... I .. J.. K.. LF
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct the displayed block and ensure that the G functions and addresses in the block are in agreement. Clear alarm with RESET key. Restart part program.
<b>15800</b>	<b>Channel %1 block %2 wrong starting condition for CONTPRON</b>
Explanation:	%1 = Channel number %2 = Block number, label The start conditions for contour preprocessing (keyword CONTPRON) are not correct: • G40 (deselection of the tool radius compensation) is not active • Spline or polynomial interpolation has been selected
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Deselect spline or polynomial interpolation and/or tool radius compensation with G40. Clear alarm with RESET key. Restart part program.
<b>15810</b>	<b>Channel %1 block %2 wrong array dimension for CONTPRON</b>
Explanation:	%1 = Channel number %2 = Block number, label The number of columns in a contour table is a fixed quantity. The value required here must be taken from the relevant technology programming guide.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct the array definition for the contour table. The number of rows is freely definable and corresponds to the number of contour elements (circles, straight lines). The number of columns is fixed (release 6/94: column number = 11). Example: N100 DEF REAL KONTAB_1 [30, 11] Clear alarm with RESET key. Restart part program.
<b>15900</b>	<b>Channel %1 block %2 touch probe not available</b>
<b>15910</b>	<b>Channel %1 block %2 touch probe not available</b>
Explanation:	%1 = Channel number %2 = Block number, label Alarm no.: 15 900 ... Measure with deletion of distance-to-go Alarm no.: 15 910 ... Measure without deletion of distance-to-go In the part program, an illegal probe has been programmed with the command MEAS (measure with deletion of distance-to-go) or MEAW (measure without distance-to-go). The probe numbers 0 ... no probe 1 ... probe 1 2 ... probe 2 are allowed, whether the probe is actually connected or not. Example: N10 MEAS=2 G01 X100 Y200 Z300 F1000; Probe 2 with deletion of distance-to-go
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Include a probe number within the limits given above in the keyword MEAS=... or MEAW=... This must correspond to the hardware connection of the probe. Clear alarm with RESET key. Restart part program.
<b>15950</b>	<b>Channel %1 block %2 no traverse motion programmed</b>
<b>15960</b>	<b>Channel %1 block %2 no traverse motion programmed</b>
Explanation:	%1 = Channel number %2 = Block number, label Alarm no.: 15 950 ... Measure with deletion of distance-to-go Alarm no.: 15 960 ... Measure without deletion of distance-to-go In the part program, no axis or a traversing path of zero has been programmed with the command MEAS (measure with deletion of distance-to-go) or MEAW (measure without deletion of distance-to-go).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Correct the part program and add the axis address or the traversing path to the measurements block. Clear alarm with RESET key. Restart part program.
<b>16000</b>	<b>Channel %1 block %2 invalid value for lifting direction</b>
Explanation:	%1 = Channel number %2 = Block number, label In „Rapid lift from the contour“ (keyword: LIFTFAST) a code value has been programmed for the direction of lift (keyword: ALF=...) that is outside of the permissible range (permitted range of values: 0 to 8). With active cutter radius compensation: Code numbers 2, 3 and 4 cannot be used in G41 Code numbers 6, 7 and 8 cannot be used in G42 because they code the direction to the contour.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Program the lifting direction under ALF=... within the permissible limits. Clear alarm with RESET key. Restart part program.
<b>16005</b>	<b>Channel %1 block %2 invalid value for lifting distance</b>
Explanation:	%1 = Channel number %2 = Block number, label Mistake in the programming: the value for the lifting path must not be negative.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable

Remedy:	Modify part program. Clear alarm with RESET key.
<b>16020</b>	<b>Channel %1 repositioning in block %2 is not possible.</b>
Explanation:	%1 = Channel number %2 = Block number, label Programming or operator action incorrect: A block is to be approached again for which there is no repositioning information (e.g. REPOS programmed but no REORG performed, REPOS with A spline or B spline).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Change part program if necessary. Clear alarm with RESET key. Restart part program.
<b>16100</b>	<b>Channel %1 block %2 spindle %3 not available in channel</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = String Mistake in programming: This channel does not recognize the spindle number. The alarm can occur together with a dwell or SPI function.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Check the <b>part program</b> to determine whether the programmed spindle number is correct and whether the program is run in the correct channel. Check MD 35000 <b>SPIND_ASSIGN_TO_MACHAX</b> for all machine axes to see whether one of them contains the programmed spindle number. This machine axis number must be entered in a channel axis of the channel-specific machine data 20070 <b>AXCONF_MACHAX_USED</b> . Clear alarm with RESET key. Restart part program.
<b>16110</b>	<b>Channel %1 block %2 spindle %3 for dwell time not in speed control mode</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis, spindle The spindle can be in the positioning mode, oscillating mode and control mode. With the M command M70 it can be changed from a spindle to an axis. The control mode is divided into the speed-controlled and position-controlled mode, and it is possible to alternate between these with the keywords SPCON and SPCOF. Positioning mode: Position control (spindle position under SPOS/SPOSA) Oscillating mode: Speed control (M41 - M45 or M40 and S...) Control mode: Speed control (spindle speed under S..., M3/M4/M5) Position control (SPCON/SPCOF, spindle speed under S..., M3/M4/M5) Axis mode: Position control (M70/M3, M4, M5, axis position under user-selectable axis name)
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Check part program for correct spindle number. With M3, M4 or M5 put the required spindle into control mode before calling the dwell time. Clear alarm with RESET key. Restart part program.
<b>16120</b>	<b>Channel %1 block %2 invalid index for online tool compensation</b>
Explanation:	%1 = Channel number %2 = Block number, label Mistake in programming: The 2nd parameter in the PUTFTOC command indicates for which tool parameter the value is to be corrected (1 - 3 tool lengths, 4 tool radius). The programmed value is beyond the permitted range. Permissible values are 1 - 4 if on-line tool radius compensation is allowed (see machine data ONLINE_CUTCOM_ENABLE), otherwise values 1 - 3.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program: Length 1 - 3 or 4 permissible for radius Clear alarm with RESET key. Restart part program.
<b>16130</b>	<b>Channel %1 block %2 instruction not allowed with active FTOCON</b>
Explanation:	%1 = Channel number %2 = Block number, label Case 1: Change of plane is not allowed if the modal G function FTOCON: „Tool fine compensation on“ is active. Case 2: Transformation selection is allowed only for zero transformation or transformation inclined axis, Transmit or Tracyl if FTOCON is active. Case 3: Tool change is not allowed with M06 if FTOCON has been active since the last tool change. Case 4: Orientable tool holder is active.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Deselect fine tool compensation with FTOCOF Clear alarm with RESET key. Restart part program.
<b>16140</b>	<b>Channel %1 block %2 FTOCON not allowed</b>
Explanation:	%1 = Channel number %2 = Block number, label The tool fine compensation (FTOC) is not compatible with the currently active transformation.



Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Deselect tool fine compensation with FTOCOF Clear alarm with RESET key. Restart part program.
<b>16150</b>	<b>Channel %1 block %2 invalid spindle no. with PUTFTOCF</b>
Explanation:	%1 = Channel number %2 = Block number, label The spindle number programmed for PUTFTOC or PUTFTOCF is beyond the permitted range for the spindle numbers.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Is the programmed spindle number available? Clear alarm with RESET key. Restart part program.
<b>16410</b>	<b>Channel %1 block %2 axis %3 is not a geometry axis</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number A geometry axis has been programmed that cannot be imaged on any machine axis in the current transformation (possibly there is no transformation active at the moment). Example: Without transformation: Polar coordinate system with X, Z, and C axis With transformation: Cartesian coordinate system with X, Y, and Z e.g. with TRANSMIT.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Activate transformation type with TRAORI (n) or do not program geometry axes that do not participate in the transformation grouping. Clear alarm with RESET key. Restart part program.
<b>16420</b>	<b>Channel %1 block %2 axis %3 repeatedly programmed</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number It is not allowed to program an axis more than once.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Delete the axis addresses that have been programmed more than once. Clear alarm with RESET key. Restart part program.
<b>16430</b>	<b>Channel %1 block %2 geometry axis %3 cannot traverse as positioning axis in rotated coordinate system</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number In the rotated coordinate system, traversing of a geometry axis as positioning axis (i.e. along its axis vector in the rotated coordinate system) would mean traversing of several machine axes. This is in conflict with the positioning axis concept, however, in which <b>one</b> axis interpolator runs in addition to the path interpolator!
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Traverse geometry axes as positioning axes only with rotation deactivated. Deactivate rotation: Keyword ROT without further specification of axis and angle. Example: N100 ROT Clear alarm with RESET key. Restart part program.
<b>16500</b>	<b>Channel %1 block %2 chamfer or radius negative</b>
Explanation:	%1 = Channel number %2 = Block number, label A negative chamfer or rounding has been programmed under the keywords CHF= ..., RND=... or RNDM=...
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Values for chamfers, roundings and modal roundings must be programmed with positive values only. Clear alarm with RESET key. Restart part program.
<b>16510</b>	<b>Channel %1 block %2 facing axis is not defined</b>
Explanation:	%1 = Channel number %2 = Block number, label Diameter programming has been activated with the keyword DIAMON although no facing axis has been programmed in this NC block. If the diameter axis is not a geometry axis, in the initial setting „DIAMON“ the alarm appears as soon as the control is switched on.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Activate the modal G function DIAMON only in NC blocks containing a facing axis or deactivate diameter program with DIAMOF. In machine data 20150 GCODE_RESET_VALUES[28] select „DIAMOF“ for the initial setting. Clear alarm with RESET key. Restart part program.
<b>16700</b>	<b>Channel %1 block %2 axis %3 invalid feed type</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number At a thread cutting operation the feed was programmed in a wrong unit.

1. **G33 (thread with constant lead)** and the feed have not been programmed with G94 or G95.
2. **G33 (thread with constant lead)** is active (modal) and G63 is programmed additionally in a following block conflict situation! -> (G63 is in the 2nd G group, G33, G331 and G332 are in the 1st G group).
3. **G331 or G332 (rigid tapping)** and the feed have not been programmed with G94.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.  
 Remedy: Use only the feed type G94 or G95 in the thread cutting functions.  
 After G33 and before G63, deselect the thread cutting function with G01.  
 Clear alarm with RESET key. Restart part program.

**16710 Channel %1 block %2 axis %3 master spindle not programmed**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Axis name, spindle number  
 A master spindle function has been programmed (G33, G331, G95, G96) but the speed or the direction of rotation of the master spindle is missing.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.  
 Remedy: Add S value or direction of rotation for the master spindle in the displayed block.  
 Clear alarm with RESET key. Restart part program.

**16715 Channel %1 block %2 axis %3 master spindle not in standstill**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Spindle number  
 In the applied function (G74, reference point approach), the spindle must be stationary.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.  
 Remedy: Program M5 or SPOS/SPOSA in front of the defective block in the part program.  
 Clear alarm with RESET key. Restart part program.

**16720 Channel %1 block %2 axis %3 thread lead is zero**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Axis name, spindle number  
 No lead was programmed in a thread block with **G33** (thread with constant lead) or **G331** (rigid tapping).

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.  
 Remedy: The thread lead must be programmed for the specified geometry axis under the associated interpolation parameters. X -> I, Y -> J, Z -> K  
 Clear alarm with RESET key. Restart part program.

**16730 Channel %1 block %2 axis %3 wrong parameter for thread cutting**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Axis name, spindle number  
 In **G33** (tapping with constant lead) the lead parameter was not assigned to the axis that determines the velocity. For longitudinal and face threads, the thread lead for the specified geometry axis must be programmed under the associated interpolation parameter. X -> I, Y -> J, Z -> K  
 For taper threads, the address I, J, K depends on the axis with the longer path (thread length). A 2nd lead for the other axis is, however, **not specified**.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.  
 Remedy: Assign lead parameters to the axis that determines the velocity.  
 Clear alarm with RESET key. Restart part program.

**16740 Channel %1 block %2 geometry axis must be programmed**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 No geometry axis was programmed for tapping (G33) or for rigid tapping (G331, G332). The geometry axis is, however, essential if an interpolation parameter has been specified.  
 Example:

```
N100 G33 Z400 K2 ; thread lead 2mm, thread: end Z=400mm
N200 SPOS=0 ; position spindle in axis mode
N201 G90 G331 Z-50 K-2 ; tapping to Z=-50, counterclockwise
N202 G332 Z5 ; retraction, direction reversal automatic
N203 S500 M03 ; spindle again in spindle mode
```

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.  
 Remedy: Specify geometry axis and corresponding interpolation parameters.  
 Clear alarm with RESET key. Restart part program.

**16750 Channel %1 block %2 axis %3 SPCON not programmed**

Explanation: %1 = Channel number  
 %2 = Block number, label  
 %3 = Axis name, spindle number  
 For the programmed function (rotary axis, positioning axis), the spindle must be in position control mode.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.  
 Remedy: Program position control of the spindle with SPCON in the previous block.  
 Clear alarm with RESET key. Restart part program.

<b>16751</b>	<b>Channel %1 block %2 spindle/axis %3 SPCOF.</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number For the programmed function, the spindle must be in the open-loop control mode. In the positioning or axis mode, the position control must not be deselected.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Put the spindle into open-loop control mode in the preceding block. This can be done with M3, M4 or M5 for the relevant spindle. Clear alarm with RESET key. Restart part program.
<b>16755</b>	<b>Channel %1 block %2 no wait needed</b>
Explanation:	%1 = Channel number %2 = Block number, label No Stop is needed for the programmed function. A Stop is necessary after SPOSA or after M5 if the next block is to be applied only after the spindle has come to a stop.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Do not write instruction. Clear alarm with RESET key. Restart part program.
<b>16760</b>	<b>Channel %1 block %2 axis %3 S value missing</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number No spindle speed has been given for rigid tapping (G331 or G332).
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Program the spindle speed under address S in [rpm] (in spite of axis mode); the direction of rotation is given by the sign of the spindle lead. Positive thread lead: Rotational direction as M03 Negative thread lead: Rotational direction as M04 Clear alarm with RESET key. Restart part program.
<b>16761</b>	<b>Channel %1 block %2 axis/spindle %3 not programmable in channel</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number Mistake in the programming: The axis / spindle can not be programmed in the channel at this time. This alarm can occur when the axis / spindle is being used by another channel or by the PLC.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program, use „GET()“. Clear alarm with RESET key. Restart part program.
<b>16762</b>	<b>Channel %1 block %2 spindle %3 function of thread or drill is active</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Spindle number Mistake in programming: The spindle function cannot be executed at the present time. This alarm occurs when the spindle (master spindle) is linked with the axes by an interpolation function.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Modify part program. Deselect thread cutting or tapping. Clear alarm with RESET key. Restart part program.
<b>16763</b>	<b>Channel %1 block %2 axis %3 programmed speed is illegal (zero or negative)</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number A spindle speed (S value) was programmed with the value zero or with a negative value.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable
Remedy:	The programmed spindle speed (S value) must be positive. Depending on the application case, the value zero can be accepted (e.g. G25 S0). Clear alarm with RESET key.
<b>16770</b>	<b>Channel %1 block %2 axis %3 encoder missing</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number SPCON, SPOS or SPOSA has been programmed. These functions require at least one measuring system. According to MD: NUM_ENC5 the machine axis/spindle has no measuring system.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Retrofit a measuring system. Clear alarm with RESET key. Restart part program.
<b>16783</b>	<b>Channel %1 block %2 slave axis/spindle %3 currently not available</b>



Explanation:	<p>%1 = Channel number          %2 = Block number, label          %3 = Axis name, spindle number</p> <p>A coupling has been switched on in which the slave spindle/axis is currently not available. Possible causes are:</p> <ul style="list-style-type: none"> <li>• The spindle/axis is active in the other channel.</li> <li>• The spindle/axis has been operated from the PLC and has not yet been enabled.</li> </ul>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Abhilfe:	Put the master spindle/axis with spindle/axis exchange into the necessary channel or release by the PLC. Clear alarm with RESET key. Restart part program.
<b>16785</b>	
Explanation:	<p><b>Channel %1 block %2 master and slave axis/spindle %3 are identical</b></p> <p>%1 = Channel number          %2 = Block number, label          %3 = Axis name, spindle number</p> <p>A coupling has been switched on in which the following spindle/axis is identical to the master spindle/axis</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	<ul style="list-style-type: none"> <li>• Configure link accordingly in MD (channel MD: COUPLE_AXIS_n)</li> <li>• or correct the part program.</li> </ul> <p>Clear alarm with RESET key. Restart part program.</p>
<b>16800</b>	
Explanation:	<p><b>Channel %1 block %2 traverse instruction DC/CDC for axis %3 not allowed</b></p> <p>%1 = Channel number          %2 = Block number, label          %3 = Axis name, spindle number</p> <p>The keyword DC (Direct Coordinate) can only be used for rotary axes. This causes approach of the programmed <b>absolute position</b> along the shortest path.</p> <p>Example:          N100 C=DC(315)</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	<p>Replace the keyword DC in the displayed NC block by specifying <b>AC (Absolute Coordinate)</b>.</p> <p>If the alarm display is the result of an error in the axis definition, the axis can be declared as a rotary axis by means of the axis-specific MD 30 300 <b>IS_ROT_AX</b>.</p> <p>Corresponding machine data:          MD 30 310: <b>ROT_IS_MODULO</b>          MD 30 320: <b>DISPLAY_IS_MODULO</b></p> <p>Clear alarm with RESET key. Restart part program.</p>
<b>16810</b>	
Explanation:	<p><b>Channel %1 block %2 traverse instruction ACP for axis %3 not allowed</b></p> <p>%1 = Channel number          %2 = Block number, label          %3 = Axis name, spindle number</p> <p>The keyword <b>ACP (Absolute Coordinate Positive)</b> is only allowed for "modulo axes". It causes approach of the programmed <b>absolute position</b> in the specified direction.</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	<p>In the displayed NC block, replace the keyword ACP by specifying <b>AC (Absolute Coordinate)</b>. If the alarm display is based on an incorrect axis definition, the axis with the axis-specific MD 30 300: <b>IS_ROT_AX</b> and MD 30 310: <b>ROT_IS_MODULO</b> can be declared a rotary axis with modulo change.</p> <p>Corresponding machine data:          MD 30 320: <b>DISPLAY_IS_MODULO</b></p> <p>Clear alarm with RESET key. Restart part program.</p>
<b>16820</b>	
Explanation:	<p><b>Channel %1 block %2 traverse instruction ACN for axis %3 not allowed</b></p> <p>%1 = Channel number          %2 = Block number, label          %3 = Axis name, spindle number</p> <p>The keyword <b>ACN (Absolute Coordinate Negative)</b> is only allowed for "modulo axes". It causes approach of the programmed <b>absolute position</b> in the specified direction.</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	<p>In the displayed NC block, replace the keyword ACN by specifying <b>AC (Absolute Coordinate)</b>. If the alarm display is based on an incorrect axis definition, the axis with the axis-specific MD 30 300: <b>IS_ROT_AX</b> and MD 30 310: <b>ROT_IS_MODULO</b> can be declared a rotary axis with modulo change.</p> <p>Corresponding machine data: MD 30 320: <b>DISPLAY_IS_MODULO</b></p> <p>Clear alarm with RESET key. Restart part program.</p>
<b>16830</b>	
Explanation:	<p><b>Channel %1 block %2 invalid position for axis/spindle %3 programmed</b></p> <p>%1 = Channel number          %2 = Block number, label          %3 = Axis name, spindle number</p> <p>A position beyond the range of 0 - 359.999 has been programmed for a modulo axis.</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	<p>Program position in the range 0 - 359.999.</p> <p>Clear alarm with RESET key. Restart part program.</p>
<b>16903</b>	
Explanation:	<p><b>Channel %1 program control: action %2 not allowed in current state</b></p>

Explanation:	%1 = Channel number %2 = Action number/action name The relevant action cannot be processed now. This can occur, for instance, during read-in of machine data.
Reaction:	Alarm display
Remedy:	Wait until the procedure is terminated or abort with Reset and repeat the operation. Clear alarm with the Cancel key. No further operator action necessary.
<b>16904</b>	<b>Channel %1 program control: action %2 not allowed in current state</b>
Explanation:	%1 = Channel number %2 = Action number/action name The operation (program, JOG, block search, reference point, etc.) cannot be started or continued in the current status.
Reaction:	Alarm display
Remedy:	Check the program status and channel status. Clear alarm with the Cancel key. No further operator action necessary.
<b>16905</b>	<b>Channel %1 program control: action %2 not allowed</b>
Explanation:	%1 = Channel number %2 = Action number/action name Operation cannot be started or continued. A start is only accepted when an NCK function can be started. Example: A start is accepted in JOG mode when, for example, the function generator is active or a JOG movement has first been stopped with the Stop key.
Reaction:	Alarm display depending on MD 11411 ENABLE_ALARM_MASK
Remedy:	Check the program status and channel status. Clear alarm with the Cancel key. No further operator action necessary.
<b>16906</b>	<b>Channel %1 program control: action %2 is aborted because of an active alarm</b>
Explanation:	%1 = Channel number %2 = Action number/action name The action was aborted due to an alarm.
Reaction:	Alarm display
Remedy:	Remedy the error and acknowledge the alarm. Then repeat the operation. Clear alarm with the Cancel key. No further operator action necessary.
<b>16907</b>	<b>Channel %1 action %2 only possible in stop</b>
Explanation:	%1 = Channel number %2 = Action number/action name This action may only be performed in Stop state
Reaction:	Alarm display
Remedy:	Check the program status and channel status. Clear alarm with the Cancel key. No further operator action necessary.
<b>16908</b>	<b>Channel %1 action %2 only possible in reset or at the block end</b>
Explanation:	%1 = Channel number %2 = Action number/action name This action may only be performed in Reset state or at end of block.
Reaction:	Alarm display
Remedy:	Check the program status and channel status Clear alarm with the Cancel key. No further operator action necessary.
<b>16909</b>	<b>Channel %1 action %2 is not allowed in current mode</b>
Explanation:	%1 = Channel number %2 = Action number / action name You have to activate a different operating mode for the function to be activated.
Reaction:	Alarm display
Remedy:	Check operation and operating state. Clear alarm with the Cancel key. No further operator action necessary.
<b>16911</b>	<b>Channel %1 mode change is not allowed</b>
Explanation:	%1 = Channel number The change from overstoreing in another operating mode is not allowed.
Reaction:	Alarm display
Remedy:	After overstoreing is terminated, it is possible to change to another operating state again. Clear alarm with the Cancel key. No further operator action necessary.
<b>16912</b>	<b>Channel %1 program control: action %2 only possible in reset</b>
Explanation:	%1 = Channel number %2 = Action number / action name This action can only be performed in Reset state. Example: Program selection through MMC or channel communication (INIT) can only be performed in Reset state.
Reaction:	Alarm display
Remedy:	Reset or wait until processing is terminated.

	Clear alarm with the Cancel key. No further operator action necessary.
<b>16913</b>	<b>Mode group %1 channel %2 mode change: action %3 not allowed</b>
Explanation:	%1 = Channel number %2 = Mode group number %3 = Action number / action name The change to the desired mode is not permitted. The change can only take place in the Reset state. Example: Program processing is halted in AUTO mode by NC Stop. Then there is a mode change to JOG mode (program status interrupted). From this operating mode it is only possible to change to AUTO mode and not to MDA mode!
Reaction:	Alarm display
Remedy:	Either activate the Reset key to reset program processing, or activate the mode in which the program was being processed previously. Clear alarm with the Cancel key. No further operator action necessary.
<b>16914</b>	<b>Mode group %1 channel %2 mode change: action %3 not allowed</b>
Explanation:	%1 = Channel number %2 = Mode group number %3 = Action number / action name Incorrect mode change, e.g.: Auto->MDAREF
Reaction:	Alarm display
Remedy:	Check operation or selected mode. Clear alarm with the Cancel key. No further operator action necessary.
<b>16915</b>	<b>Channel %1 action %2 in the current block not allowed</b>
Explanation:	%1 = Channel number %2 = Action number / action name If traversing blocks are interrupted by asynchronous subroutines, then it must be possible for the interrupted program to continue (reorganization of block processing) after termination of the asynchronous subroutine. The 2nd parameter describes which action wanted to interrupt block processing.
Reaction:	Alarm display
Remedy:	Let the program continue to a reorganized NC block or modify part program. Clear alarm with the Cancel key. No further operator action necessary.
<b>16916</b>	<b>Channel %1 reposition: action %2 not allowed in the current state</b>
Explanation:	%1 = Channel number %2 = Action number / action name Repositioning of block processing presently not possible. In certain cases this can prevent a mode change from taking place. The 2nd parameter describes which action should be used to perform repositioning.
Reaction:	Alarm display
Remedy:	Let the program continue to a repositioned NC block or modify part program. Clear alarm with the Cancel key. No further operator action necessary.
<b>16918</b>	<b>Channel %1 for action %2 needs reset in all channels</b>
Explanation:	%1 = Channel number %2 = Action number / action name All channels must be in the initial setting in order to carry out the action! (For example, for machine data loading)
Reaction:	Alarm display
Remedy:	Either wait until the channel status is aborted or press the Reset key. Clear alarm with the Cancel key. No further operator action necessary.
<b>16919</b>	<b>Channel %1 action %2 is not allowed, because of an alarm</b>
Explanation:	%1 = Channel number %2 = Action number / action name This action cannot be performed due to an alarm, or the channel is in Fail
Reaction:	Alarm display
Remedy:	Press RESET key Clear alarm with the Cancel key. No further operator action necessary.
<b>16920</b>	<b>Channel %1 action %2 is already enabled</b>
Explanation:	%1 = Channel number %2 = Action number / action name An identical action is still active.
Reaction:	Alarm display
Remedy:	Wait until the first procedure is terminated or abort with Reset and repeat the operation. Clear alarm with Cancel key. No further operator action necessary.
<b>16923</b>	<b>Channel %1 program control: action %2 not allowed in the current state</b>
Explanation:	%1 = Channel number %2 = Action number / action name The current processing cannot be stopped, due to an active preprocessing process. This applies to, for example, loading machine data and block searches until the search object is found.
Reaction:	Alarm display. Interface signals are set
Remedy:	Abort by pressing Reset! Clear alarm with Cancel key. No further operator action necessary.
<b>16924</b>	<b>Channel %1 caution: program test will of change the tool data</b>
Explanation:	%1 = Channel number

Tool management data is changed during program testing. It is not possible to automatically rectify the data after termination of the program testing. This error message prompts the user to make a backup copy of the data or to reimport the data after the operation is terminated.

Reaction: Alarm display

Remedy: Save tool data on MMC and reimport data after „ProgtestOff“.  
Clear alarm with Cancel key. No further operator action necessary.

**16925 Channel %1 program control: action %2 not allowed in the current state action %3 active**

Explanation: %1 = Channel number

%2 = Action number / action name

%3 = Action number / action name

The action has been refused since a mode or sub-mode change (change to automatic mode, MDA, JOG, oversteering, digitizing, etc.) is taking place.

Example: This alarm message is output if the Start key is pressed during a mode or sub-mode change from, for example, automatic to MDA, before the NCK has confirmed selection of the mode.

Reaction: Alarm display

Remedy: Repeat action. Clear alarm with Cancel key. No further operator action necessary.

**16930 Channel %1: Predecessor and current block %2 must be separated by an executable block**

Explanation: %1 = Channel number

%2 = Block number

The language functions WAITMC, SETM, CLEARM and MSG must be packed in separate NC blocks due to the language definition. To avoid velocity drops, these blocks are attached to the following NC block internally in the NCK (for WAITMC to the previous NC\_block). For this reason, there must always be an executable block (no calculation block) between the NC blocks. An executable NC block includes always e.g. travel movements, a help function, Stopre, dwell time etc.

Reaction: Alarm display. Interface signals are set. Interpreter stop Correction block is reorganized

Remedy: Program an executable NC block between the previous and the current NC block.

Example:

N10 SETM

N15 STOPRE; insert executable NC block

N20 CLEARM

Clear alarm with NC Start. Restart part program.

**17020 Channel %1 block %2 1st array index out of range**

Explanation: %1 = Channel number

%2 = Block number, label

A read or write access has been programmed to an array variable with invalid 1st array index. The valid array indices must be contained within the defined array size and the absolute limits (0 - 32 766).

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Correct the specification of array elements in the access instruction to match the defined size.

Clear alarm with RESET key. Restart part program.

**17030 Channel %1 block %2 2nd array index out of range**

Explanation: %1 = Channel number

%2 = Block number, label

A read or write access has been programmed to an array variable with invalid 2nd array index. The valid array indices must be contained within the defined array size and the absolute limits (0 - 32 766).

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Correct the specification of array elements in the access instruction to match the defined size.

Clear alarm with RESET key. Restart part program.

**17040 Channel %1 block %2 illegal axis index**

Explanation: %1 = Channel number

%2 = Block number, label

A read or write access has been programmed to an axial variable in which the axis name cannot be unambiguously imaged on a machine axis.

Example:

Writing of an axial machine data.

\$MA\_... [X]= ... ; but geometry axis X cannot be imaged on a machine axis; because of a transformation

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Deselect transformation before writing the axial data (keyword: TRAF00F) or use the machine axis name as axis index.

Clear alarm with RESET key. Restart part program.

**17050 Channel %1 block %2 illegal value**

Explanation: %1 = Channel number

%2 = Block number, label

On accessing an individual frame element, a frame component other than TRANS, ROT, SCALE or MIRROR was addressed or the function CSCALE has been given a negative scale factor.

Example:

\$P\_UIFR[5] = CSCALE (X, -2.123)

The frame components are either selected by means of the keywords

TR for translation (TRANS, internal 0)

RT for rotation (ROT, internal 1)

SC for scaling and (SCALE, internal 3)

**MI** for mirroring (MIRROR, internal 4)  
or they are specified directly as an integral value 0, 1, 3, 4.  
Example:

Access to the rotation around the X axis of the currently settable frame.  
R10=\$P\_UIFR[\$AC\_IFRNUM, X, RT] can also be programmed as:  
R10=\$P\_UIFR[\$AC\_IFRNUM, X, 1]

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Address frame components only with the keywords provided; program the scale factor between the limits of 0.000 01 to 999.999 99.

Clear alarm with RESET key. Restart part program.

**17070 Channel %1 block %2 data is write-protected**

Explanation: %1 = Channel number

%2 = Block number, label

An attempt was made to write a write-protected variable (e.g. a system variable) or a machine data for which a higher protection level has been declared than the one currently active.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Remove write access to write-protected system variables from the NC program. Increase the current protection level for writing the machine data.

Clear alarm with RESET key. Restart part program.

**17160 Channel %1 block %2 tool is not selected**

Explanation: %1 = Channel number

%2 = Block number, label

An attempt has been made to access the current tool offset data via the system variables:

\$P\_AD [n] Contents of the parameter (n: 1 - 25)

\$P\_TOOL Active D number (tool edge number)

\$P\_TOOLL [n] Active tool length (n: 1 - 3)

\$P\_TOOLR Active tool radius

although no tool had been selected previously.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Program or activate a tool offset in the NC program before using the system variables.

Example:

N100 G.. ... T5 D1 ... LF

With the channel-specific machine data:

MD 22 550:TOOL\_CHANGE\_MODE

New tool offset for M function

MD 22 560:TOOL\_CHANGE\_M\_CODE

M function with tool change

It is established whether a tool offset is activated in the block with the T word or whether the new offset values are allowed for only when the M word for tool change occurs.

Clear alarm with RESET key. Restart part program.

**17170 Channel %1 block %2 too many symbols defined**

Explanation: %1 = Channel number

%2 = Block number, label

The predefined symbols could not be read in during power-up.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Clear alarm with RESET key. Restart part program.

**170180 Channel %1 block %2 illegal D number**

Explanation: %1 = Channel number

%2 = Block number, label

In the displayed block, access is made to a D number (tool edge number) that is not initialized and therefore is not available.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Check tool call in the NC part program:

- Correct tool edge number D.. programmed? If no tool edge number is specified, then D1 is automatically active.
- Tool parameters P1 - P25 defined?

The dimensions of the tool edge must have been entered previously either through the operator panel or through the V.24 interface.

Description of the system variables \$P\_DP x [n, m]

n ... Associated tool number T

m ... Tool edge number D

x ... Parameter number P

Clear alarm with RESET key. Restart part program.

**17190 Channel %1 block %2 illegal T number**

Explanation: %1 = Channel number

%2 = Block number, label

In the displayed block, access is made to a T number (tool number) that is not initialized and therefore not available.

Reaction: Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy: Check tool call in the NC part program:

- Correct tool number T.. programmed?
- Tool parameters P1 - P25 defined?

The dimensions of the tool edge must have been previously entered either through the operator panel or through the V.24 interface.

Description of the system variables \$P\_DP x [n, m]

n ... Associated tool number T

m ... Tool edge number D

x ... Parameter number P

Clear alarm with RESET key. Restart part program.

**17200**

Explanation:

**Channel %1 block %2 cannot delete an active tool**

%1 = Channel number

%2 = Block number, label

An attempt has been made to delete from the part program the tool data for a tool currently being processed. Tool data for tools involved in the current machining operation may not be deleted. This applies both for the tool preselected with T or that has been changed in place of another, and also for tools for which the constant grinding wheel peripheral speed or tool monitoring is active.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Check access to tool offset memory by means of \$TC\_DP1[t,d] = 0 or deselect tool

Clear alarm with RESET key. Restart part program.

**17220**

Explanation:

**Channel %1 block %2 tool not available**

%1 = Channel number

%2 = Block number, label

If an attempt is made to access a tool via a T no. that has not (yet) been defined. For example, when tools are to be put into magazine locations by programming \$TC\_MPP6 = 'toolNo'. This is possible only when both the magazine location and the tool given by 'toolNo' have been defined.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Correct the NC program. Clear alarm with RESET key. Restart part program.

**17230**

Explanation:

**Channel %1 block %2 Duplo no. already disposed**

%1 = Channel number

%2 = Block number, label

If an attempt is made to write a tool Duplo number to the name of which another tool (another T number) already exists with the same Duplo number.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Correct the NC program.

Clear alarm with RESET key. Restart part program.

**17240**

Explanation:

**Channel %1 block %2 invalid definition of tool**

%1 = Channel number

%2 = Block number, label

If an attempt is made to modify a tool data that would subsequently damage the data consistency or lead to a conflicting definition, this alarm will appear.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Correct the NC program.

Clear alarm with RESET key. Restart part program.

**17250**

Explanation:

**Channel %1 block %2 invalid definition of magazine**

%1 = Channel number

%2 = Block number, label

If an attempt is made to modify a magazine data that would subsequently damage the data consistency or lead to a conflicting definition, this alarm will appear

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Correct the NC program.

Clear alarm with RESET key. Restart part program.

**17260**

Explanation:

**Channel %1 block %2 invalid definition of magazine location**

%1 = Channel number

%2 = Block number, label

If an attempt is made to modify a magazine location data that would subsequently damage the data consistency or lead to a conflicting definition, this alarm will appear

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Correct the NC program.

Clear alarm with RESET key. Restart part program.

**17270**

Explanation:

**Channel %1 block %2 call-by-reference: illegal variable**

%1 = Channel number

%2 = Block number, label

Machine data and system variables must not be transferred as call-by-reference parameters.

Reaction:

Alarm display. Interface signals are set. Interpreter stop. NC Start disable.

Remedy:

Modify NC program: Assign the value of the machine data or of the system variable to a program-local variable and transfer this as parameter.

Clear alarm with RESET key. Restart part program.

**17500**

Explanation:

**Channel %1 block %2 axis %3 is not an indexing axis**

%1 = Channel number

%2 = Block number, label

%3 = Axis name, spindle number



Reaction:	An indexing axis position has been programmed for an axis with the keywords CIC, CAC or CDC that has not been defined as indexing axis in the machine data.
Remedy:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable. Remove programming instruction for indexing axis positions (CIC, CAC, CDC) from the NC part program or declare the relevant axis to be an indexing axis. Indexing axis declaration: MD 30 500: <b>INDEX_AX_ASSIGN_POS_TAB</b> (indexing axis assignment) The axis becomes an indexing axis if an assignment to an indexing position table has been made in the specified MD. Two tables are possible (input value 1 or 2). MD 10 900: <b>INDEX_AX_LENGTH_POS_TAB_1</b> MD 10 920: <b>INDEX_AX_LENGTH_POS_TAB_2</b> (Number of positions for 1st/2nd indexing axis) Standard value: 0 Maximum value: 60 MD 10 910: <b>INDEX_AX_POS_TAB_1 [n]</b> MD 10 930: <b>INDEX_AX_POS_TAB_2 [n]</b> (Positions of the 1st indexing axis) The absolute axis positions are entered. (The list length is defined via MD 10 900). Clear alarm with RESET key. Restart part program.
<b>17502</b>	<b>Channel %1 block %2 indexing axis %3 with Hirth tooth system Stop delayed</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name For the indexing axis, the 'Hirth tooth system' function is activated and the override has been set to 0 or another stop condition (e.g. VDI interface signal) is active. Since it is possible to stop only on indexing axes, the next possible indexing position is approached. The alarm is displayed until this position is reached or the stop condition is deactivated.
Reaction:	Alarm display.
Remedy:	Wait until the next possible indexing position is reached or set override > 0 or deactivate another stop condition. Alarm display disappears with alarm cause. No further operation necessary.
<b>17510</b>	<b>Channel %1 block %2 invalid index for indexing axis %3</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number The programmed index for the indexing axis is beyond the position table range. Example: Perform an absolute approach of the 56th position in the list allocated via the axis-specific machine data 30 500 <b>INDEX_AX_ASSIGN_POS_TAB</b> with the 1st positioning axis, the number of positions is e.g. only 40 (MD 10 900 <b>INDEX_AX_LENGTH_POS_TAB_1</b> = 40). N100 G. U=CAC (56)
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Program the indexing axis position in the NC part program in accordance with the length of the current position table, or add the required value to the position table and adjust the length of the list
<b>17600</b>	<b>Channel %1 block %2 preset on transformed axis %3 not possible</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number The displayed axis is involved in the current transformation. This means that it is not possible to set the actual value memory (preset) for this axis. Example: The machine axis A should be set to the new actual value A 100 at the absolute position A 300. : N100 G90 G00 A=300 N101 PRESETON A=100 :
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Avoid preset actual value memory for axes, which are participating in a transformation, or deselect the transformation with the keyword <b>TRAFOOF</b> . Clear alarm with RESET key. Restart part program.
<b>17610</b>	<b>Channel %1 block %2 positioning axis %3 cannot participate in transformation</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number The axis addressed with the keyword POS or POSA is involved in the active transformation. Therefore, it cannot be traversed as a positioning axis.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Remove the POS or POSA instruction from the part program block or previously deselect transformation with <b>TRAFOOF</b> . Clear alarm with RESET key. Restart part program.
<b>17620</b>	<b>Channel %1 block %2 fixpoint cannot be approached for transformed axis %3</b>
Explanation:	%1 = Channel number %2 = Block number, label

	<p>%3 = Axis name, spindle number In the displayed block, an axis is programmed for the fixed point approach (G75) that is involved in the active transformation. Fixed point approach is not performed with this axis</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
	Remove G75 instruction from the part program block or previously deselect transformation with TRAFOOF Clear alarm with RESET key. Restart part program.
<b>17630</b>	<b>Channel %1 block %2 referencing not possible for transformed axis %3</b>
Explanation:	<p>%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number In the displayed block, an axis is programmed for reference point approach (G74) that is involved in the active transformation. Reference point approach is not performed with this axis!</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Remove G74 instruction, or the machine axes involved in transformation, from the part program block or previously deselect the transformation with TRAFOOF. Clear alarm with RESET key. Restart part program.
<b>17640</b>	<b>Channel %1 block %2 spindle cannot be used as transformed axis %3</b>
Explanation:	<p>%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number The axis programmed for the spindle operation is involved in the current transformation as geometry axis. This is not allowed.</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	First switch off the transformation function. Clear alarm with RESET key. Restart part program.
<b>17800</b>	<b>Channel %1 block %2 illegal fixed-stop end point programmed</b>
Explanation:	<p>%1 = Channel number %2 = Block number, label The position number n specified with the keyword FP=n is not permissible. Two absolute axis positions can be defined as fixed points via the axis-specific MD 30 600 FIX_POINT_POS [n].</p>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Program keyword FP with machine fixed points 1 or 2. Example: Approach fixed point 2 with machine axes X1 and Z2. N100 G75 FP=2 X1=0 Z2=0 Clear alarm with RESET key. Restart part program.
<b>17900</b>	<b>Channel %1 block %2 axis %3 use machine axis identifier</b>
Explanation:	<p>%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number At this point, the block context calls for a machine axis. This is the case with:</p> <ul style="list-style-type: none"> <li>• G74 (reference point approach)</li> <li>• G75 (fixed point approach). If a geometry or additional axis identifier is used, then it must also be allowed as machine axis identifier. (MD: 10000 AXCONF_MACHAX_NAME_TAB).</li> </ul>
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Use machine axis identifier when programming. Clear alarm with RESET key. Restart part program.
<b>18001</b>	<b>Channel %1 block %2 wrong definition of global protection area %3, error code %4</b>
Explanation:	<p>%1 = Channel number %2 = Block number, label %3 = Number of global protection zone %4 = Error specification There is an error in the definition of the protection area. The error numbers indicate the specific reason for the alarm. The following meanings apply:</p> <ol style="list-style-type: none"> <li>1: Incomplete or conflicting contour definition</li> <li>2: Contour encompasses more than one surface area</li> <li>3: Tool-related protection zone is not convex.</li> <li>4: If both boundaries are active in the 3rd dimension of the protection zone and both limits have the same value.</li> <li>5: The number of the protection area does not exist (negative number, zero or greater than the maximum number of protection zones)</li> <li>6: Protection zone definition consists of more than 10 contour elements</li> <li>7: Tool-related protection zone is defined as inside protection zone.</li> <li>8: Incorrect parameter used.</li> <li>9: Protection zone to be activated is not defined</li> <li>10: Incorrect modal G code used for protection zone definition.</li> <li>11: Contour definition incorrect or frame activated.</li> <li>12: Other errors not specified further</li> </ol>
Reaction:	Alarm display. Interface signals are set. NC Start disable.
Remedy:	Modify definition of the protection zone and check MD. Clear alarm with Cancel key. No further operator action



	necessary.
<b>18003</b>	<b>Channel %1 block %2 channel-specific protection area %3 cannot be activated, error code %4</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Number of the channel-specific protection zone %4 = Error specification An error has occurred on activating the protection zone. The error number gives the specific reason for the alarm. The following meanings apply: 1: Incomplete or conflicting contour definition. 2: Contour encompasses more than one surface area. 3: Tool-related protection zone is not convex. 4: If both boundaries are active in the 3rd dimension of the protection zone and both limits have the same value. 5: The number of the protection area does not exist (negative number, zero or greater than the maximum number of protection zones). 6: Protection zone definition consists of more than 10 contour elements. 7: Tool-related protection zone is defined as inside protection zone. 8: Incorrect parameter used. 9: Protection zone to be activated is not defined. 10: Error in internal structure of the protection zones. 11: Other errors not specified further. 12: The number of protection zones simultaneously active exceeds the maximum number (channel-specific machine data). 13,14: Contour element for protection zones cannot be created. 15,16: No more memory space for the protection zones. 17: No more memory space for the contour elements.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. Interpreter stop. NC Start disable.
Remedy:	1. Reduce the number of simultaneously active protection zones (MD). 2. Modify part program. • Delete other protection zones. • Preprocessing stop. Clear alarm with NC Start and continue program.
<b>18006</b>	<b>Channel %1 block %2 serious error in definition of channel-specific protection area %3.</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Protection zone number The protection zone definition must be terminated with EXECUTE before a preprocessing stop is performed. This also applies to any that are initiated implicitly such as with G74, M30, M17.
Reaction:	Alarm display. Interface signals are set. Correction block is reorganized. NC Start disable.
Remedy:	Modify part program. Clear alarm with NC Start and continue program.
<b>18100</b>	<b>Channel %1 block %2 invalid argument passed to FXS</b>
Explanation:	%1 = Channel number %2 = Block number, label The following values are valid at the present time: 0: „Deselect traverse against fixed stop“ 1: „Select traverse against fixed stop“.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Clear alarm with RESET key. Restart part program.
<b>18101</b>	<b>Channel %1 block %2 invalid argument passed to FXST</b>
Explanation:	%1 = Channel number %2 = Block number, label Only the range 0.0 - 100.0 is valid at the present time.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Clear alarm with RESET key. Restart part program.
<b>18102</b>	<b>Channel %1 block %2 invalid argument passed to FXSW</b>
Explanation:	%1 = Channel number %2 = Block number, label Only positive values including zero are valid at the present time.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable.
Remedy:	Clear alarm with RESET key. Restart part program.
<b>18300</b>	<b>Channel %1 block %2 frame: Fine shift not possible</b>
Explanation:	%1 = Channel number %2 = Block number, label Allocation of a fine shift to settable frames or the basic frame is not possible since MD \$MN_FRAME_FINE_TRANS is unequal to 1.
Reaction:	Alarm display. Interface signals are set. Interpreter stop. NC Start disable
Remedy:	Modify program or set MD \$MN_FRAME_FINE_TRANS to 1. Clear alarm with NC Start and continue program.
<b>20000</b>	<b>Channel %1 axis %2 reference cam not reached</b>

Explanation:	<p>%1 = Channel number          %2 = Axis name, spindle number          After starting the reference point approach, the rising edge of the reduction cam must be reached within the section defined in the MD 34030 <b>REFP_MAX_CAM_DIST</b> (phase 1 of referencing). (This error occurs <b>only with incremental encoders</b>).</p>
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	<p>There are 3 possible causes of error:</p> <ol style="list-style-type: none"> <li>1. The value entered in MD 34030 <b>REFP_MAX_CAM_DIST</b> is too small. Determine the maximum possible distance from the beginning of reference motion up to the reduction cam and compare with the value in the MD: <b>REFP_MAX_CAM_DIST</b>, increase the value in the MD if necessary.</li> <li>2. The cam signal is not received by the PLC input module. Operate the reference point switch by hand and check the input signal on the NC/PLC interface (route: switch!connector!cable! PLC input!user program).</li> <li>3. The reference point switch is not operated by the cam. Check the vertical distance between reduction cam and activating switch.</li> </ol> <p>Clear alarm with RESET key. Restart part program.</p>
<b>20001</b>	<b>Channel %1 axis %2 cam signal missing</b>
Explanation:	<p>%1 = Channel number          %2 = Axis name, spindle number          At the beginning of phase 2 of reference point approach, the signal from the reduction cam is no longer available. Phase 2 of reference point approach begins when the axis remains stationary after deceleration to the reduction cam. The axis then starts in the opposite direction in order to select the next zero marker of the measuring system on leaving the reduction cam or approaching it again (negative/positive edge).</p>
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	<p>Check whether the <b>deceleration path</b> after the approach velocity is greater than the distance to reference point cam - in which case the axis cannot stop until it is beyond the cam. Use longer cam or reduce the approach velocity in machine data 34020 <b>REFP_VELO_SEARCH_CAM</b>. When the axis has stopped at the cam, it must be checked whether the signal „DECELERATION REFERENCE POINT APPROACH“ is still available at the interface to the NCK (DB 31 - 48, DBX 12.7).</p> <ul style="list-style-type: none"> <li>• Hardware: Wire break? Short circuit?</li> <li>• Software: User program?</li> </ul> <p>Clear alarm with RESET key. Restart part program.</p>
<b>20002</b>	<b>Channel %1 axis %2 zero reference mark not found</b>
Explanation:	<p>%1 = Channel number          %2 = Axis name, spindle number          The zero marker of the <b>incremental encoder</b> is not within a defined section. Phase 2 of reference point approach ends when the zero marker of the encoder has been detected after the rising/falling edge of the PLC interface signal „DECELERATION REFERENCE POINT APPROACH“ (DB 31 - 48, DBX 12.7) has given the trigger start. The maximum distance between the trigger start and the zero marker that follows is defined in the machine data 34060 <b>REFP_MAX_MARKER_DIST</b>. The monitor prevents a zero marker signal from being overtraveled and the next being evaluated as reference point signal. (Faulty cam adjustment or excessive delay by the PLC user program.)</p>
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	<p>Check the cam adjustment and make sure that the distance is sufficient between the end of the cam and the zero marker signal that follows. The path must be greater than the axis can cover in the PLC cycle time. Increase the machine data 34060 <b>REFP_MAX_MARKER_DIST</b>, but do not select a value greater than the distance between the 2 zero markers. This might result in the monitor being switched off.</p> <p>Clear alarm with the RESET key. Restart part program.</p>
<b>20003</b>	<b>Channel %1 axis %2 encoder error</b>
Explanation:	<p>%1 = Channel number          %2 = Axis name, spindle number          In a measuring system with distance-coded reference marks, the distance between two adjacent markers has been found to be more than twice the distance entered in the machine data 34300 <b>ENC_REFP_MARKER_DIST</b>. The control issues the alarm after having made a second attempt in reverse direction with half the traversing velocity and detecting that the distance is too large again.</p>
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	<p>Determine the distance between 2 odd reference point markers (reference point marker interval). This value (which is 20.00 mm on Heidenhain scales) must be entered in the machine data 34300 <b>ENC_REFP_MARKER_DIST</b>. Check the <b>reference point track</b> of the scale including the electronics for the evaluation.</p> <p>Clear alarm with RESET key. Restart part program.</p>
<b>20004</b>	<b>Channel %1 axis %2 reference mark missing</b>
Explanation:	<p>%1 = Channel number          %2 = Axis name, spindle number          In the <b>distance-coded length measurement system</b> two reference marks were not found within the defined searching distance (axis-specific MD: 34060 <b>REFP_MAX_MARKER_DIST</b>). No reduction cam is required for distance-coded scales (but an existing cam will be evaluated). The conventional direction key determines the direction of search. The searching distance 34060 <b>REFP_MAX_MARKER_DIST</b>, within which the two reference point markers are expected is counted commencing at the start point.</p>
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	Determine the <b>distance</b> between 2 odd reference point markers (reference point marker interval). This value (which

is 20.00 mm on Heidenhain scales) must be entered in the machine data 34060 **REFP\_MAX\_MARKER\_DIST**. Check the **reference point track** of the scale including the electronics for the evaluation. Clear alarm with RESET key. Restart part program.

**20005****Channel %1 axis %2 reference point approach aborted**

Explanation:

%1 = Channel number

%2 = Axis name, spindle number

Channel-specific referencing could not be completed for all specified axes (e.g. termination because of missing encoder enable, measuring system switchover, release of direction key, etc.).

Reaction:

Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.

Remedy:

Check the possible reasons for termination:

- Servo enable missing (DB 21 - 28, DBX 2.1)
- Measuring system switchover (DB 21 - 28, DBX 1.5 and DBX 1.6)
- Traversing key + or - missing (DB 21 - 28, DBX 8.6 and DBX 8.7)
- Feed override = 0

The axis-specific MD 34110 **REFP\_CYCLE\_NR** determines which axes are involved in the channel-specific referencing.

-1: No channel-specific referencing, NC Start **without** referencing.

0: No channel-specific referencing, NC Start **with** referencing.

1-8: Channel-specific referencing. The number entered here corresponds to the referencing sequence. (When all axes with contents 1 have reached the reference point, then the axes with contents 2 start, etc.).

Clear alarm with RESET key. Restart part program.

**20006****Channel %1 axis %2 reference point creep velocity not reached**

Explanation:

%1 = Channel number

%2 = Axis name, spindle number

In phase 2 of reference point approach (wait for zero mark), the cam end was reached but the reference point approach velocity was not within the tolerance window. (This can occur when the axis is already at the end of the cam at the beginning of reference point approach. This means that phase 1 has already been concluded and will not be started.) Phase 2 is terminated (this time in front of the cam) and reference point approach is started automatically once again with phase 1. If the approach velocity is not reached during the 2nd attempt, then referencing is aborted and the alarm is output.

Approach velocity: 34040 **REFP\_VELO\_SEARCH\_MARKER**

Velocity tolerance: 35150 **SPIND\_DES\_VELO\_TOL**

Reaction:

Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.

Remedy:

Reduce the MD for the approach velocity 34040 **REFP\_VELO\_SEARCH\_MARKER** and/or increase the MD for the velocity tolerance 35150 **SPIND\_DES\_VELO\_TOL**.

Clear alarm with RESET key. Restart part program.

**20007****Channel %1 axis %2 reference point approach needs 2 encoders**

Explanation:

%1 = Channel number

%2 = Axis name, spindle number

Bei der Einstellung 34200 **ENC\_REFP\_MODE** = 6 werden 2 Geber benötigt!

Reaction:

Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.

Remedy:

Referiermodus 34200 **ENC\_REFP\_MODE** ändern o. zweiten Geber einbauen und konfigurieren

Clear alarm with RESET key. Restart part program.

**20008****Channel %1 axis %2 Referenzpunktfahren benoetigt zweites referiertes Messsystem**

Explanation:

%1 = Channel number

%2 = Axis name, spindle number

2 encoders are needed for setting 34200 **ENC\_REFP\_MODE** = 6!

Reaction:

Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.

Remedy:

Modify reference mode 34200 **ENC\_REFP\_MODE** or install and configure a second encoder

Clear alarm with RESET key. Restart part program.

**20050****Channel %1 axis %2 handwheel mode active**

Explanation:

%1 = Channel number

%2 = Axis name, spindle number

The axes cannot be traversed in JOG mode using the traversing keys because traversing is still taking place via the The axes cannot be traversed in JOG mode using the traversing keys because traversing is still taking place via the handwheel/handwheel.

Reaction:

Alarm display.

Remedy:

Decide whether the axis is to be traversed by means of the jog keys or via the handwheel. End handwheel travel and delete the axial distance-to-go if necessary (interface signal DB 31 - 48, DBX 2.2).

Alarm display showing cause of alarm disappears. No further operator action.

**20051****Channel %1 axis %2 handwheel mode not possible**

Explanation:

%1 = Channel number

%2 = Axis name, spindle number

The axis is already traveling via the traversing keys, so handwheel mode is no longer possible.

Reaction:

Alarm display.

Remedy:

Decide whether the axis is to be traversed by means of the jog keys or via the handwheel.

Alarm display showing cause of alarm disappears. No further operator action.

**20052****Channel %1 axis %2 already active**

Explanation:

%1 = Channel number

%2 = Axis name, spindle number

	<p>The axis is to traverse as machine axis in JOG mode via the jog keys on the machine control panel. However, this is not possible because:</p> <ol style="list-style-type: none"> <li>1. It is already traversing as geometry axis (through the channel-specific interface DB 21 - 28, DBX 12.6, DBX 12.7, DBX 16.6, DBX 16.7 or DBX 20.6 and DBX 20.7) or</li> <li>2. It is already traversing as machine axis (through the axis-specific interface DB 31 - 48, DBX 8.6 and DBX 8.7) or</li> <li>3. A frame is valid for a rotated coordinate system and another geometry axis involved in this is already traversing in JOG mode by means of the direction keys.</li> </ol>
Reaction:	Alarm display.
Remedy:	Stop traversing through the channel or axis interface or stop the other geometry axis. Clear alarm with Cancel key. No further operator action necessary.
<b>20053</b>	<b>Channel %1 axis %2 DRF, FTOCON, external setting of offset not possible</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number The axis is traversed in a mode (e.g. referencing) that allows no additional overlaid interpolation.
Reaction:	Alarm display.
Remedy:	Wait until the axis has reached its reference position or terminate reference point approach with „Reset“ and start DRF once again. Clear alarm with Cancel key. No further operator action necessary.
<b>20054</b>	<b>Channel %1 axis %2 wrong index for indexing axis in JOG mode</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number 1. The displayed indexing axis is to be traversed incrementally in JOG mode (by 1 indexing position). However, no further indexing position is available in the selected direction. 2. The axis is stationary at the last indexing position. In incremental traversing the working area limitation or the software limit switch is reached without an indexing position being located in front of it at which a stop could be made.
Reaction:	Alarm display.
Remedy:	Correct (add to) the list of indexing positions by means of the machine data MD 10 900: INDEX_AX_LENGTH_POS_TAB_1 MD 10 910: INDEX_AX_POS_TAB_1 MD 10 920: INDEX_AX_LENGTH_POS_TAB_2 MD 10 930: INDEX_AX_POS_TAB_2 or set the working area limits or the software limit switches to other values. Clear alarm with Cancel key. No further operator action necessary.
<b>20055</b>	<b>Channel %1 master spindle not available in JOG mode</b>
Explanation:	%1 = Channel number The displayed axis is to be traversed as machine axis in JOG mode with revolutional feed, but no master spindle has been defined from which the actual speed could have been derived.
Reaction:	Alarm display. Interface signals are set.
Remedy:	If the revolutional feed is also to be active in JOG mode, then a master spindle must be declared via the channel-specific machine data 20090 SPIND_DEF_MASTER_SPIND. In this case you have to open a screen in the PARAMETER operating area with the softkeys "SETTINGDATA" and "JOG DATA" and preselect the G function G95 there. The JOG feedrate can then be entered in [mm/rev]. (If 0 mm/rev is set as JOG feed, the control takes the value assigned in the axis-specific MD 32050 JOG_REV_VELO or in the case of rapid traverse overlay 32040 JOG_REV_VELO_RAPID). The revolutional feed in JOG mode is deactivated by changing the G function from G95 to G94. Clear alarm with Cancel key. No further operator action necessary.
<b>20056</b>	<b>Channel %1 axis %2 no revolutional feedrate possible. Axis/spindle %3 stationary</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Axis name, spindle number An axis is to travel in JOG with revolutional feed, but the spindle/axis the feed is to be derived from is 0.
Reaction:	Alarm display
Remedy:	Traverse the spindle/axis from which the feed is to be derived. Alarm display showing cause of alarm disappears. No further operator action required.
<b>20057</b>	<b>Channel %1 block %2 revolution velocity of axis/spindle %3 is less or equal zero.</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number Revolutional feed has been programmed for an axis/spindle, but the velocity was not programmed or the programmed value is smaller than or equal to zero.
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable. LOCALREACTION. COMPBLOCKWITHREORG. Channel processing not ready
Remedy:	• Correct the part program or • Specify the correct feed for PLC axes at the VDI interface, or • Specify feed for oscillating axes in the setting data \$SA_OSCILL_VELO. Clear alarm with the RESET key. Restart part program.
<b>20060</b>	<b>Channel %1 axis %2 cannot move as geometry axis</b>
Explanation:	%1 = Channel number

%2 = Achsname

The axis is currently not in „Geometry axis“ state. Therefore, it cannot be traversed in JOG mode as geometry axis. If the abbreviation **WCS** (workpiece coordinate system) is displayed in the "Position" screen, then **only the geometry axes** can be traversed by means of the direction keys! (**MCS** ... Machine coordinate system; **all machine axes** can now be traversed by using the direction keys on the machine control panel).

Reaction:

Alarm display.

Remedy:

Check the operating steps to establish whether geometry axes really must be traversed, otherwise switch over to the machine axes by activating the "WCS/MCS" key on the machine control panel.

Clear alarm with Cancel key. No further operator action necessary.

#### 20062

Explanation:

##### Channel %1 axis %2 already active

%1 = Channel number

%2 = Axis name, spindle number

The displayed axis is already traversing as machine axis. Therefore, it cannot be operated as a geometry axis. Traversing of an axis can take place in JOG mode through 2 different interfaces.

1. As geometry axis: Through the channel-specific interface DB 21 - DB 28, DBX12.6 or DBX12.7

2. As machine axis: Through the axis-specific interface DB 31 - DB 48 DBX8.6 or DBX8.7

With the standard machine control panel, it is not possible to operate an axis as machine axis and geometry axis at the same time!

Reaction:

Alarm display.

Remedy:

Do not start the geometry axis until the traversing motion as machine axis has been concluded.

Clear alarm with Cancel key. No further operator action necessary.

#### 20065

Explanation:

##### Channel %1 master spindle not defined for geometry axes in JOG mode

%1 = Channel number

The displayed axis is to be traversed as geometry axis in JOG mode with rotary feed, but no master spindle has been defined from which the actual speed could be derived.

Reaction:

Alarm display. Interface signals are set.

Remedy:

If the revolutionary feed is also to be active in JOG mode, then a master spindle must be declared via the channel-specific machine data 20090 SPIND\_DEF\_MASTER\_SPIND. In this case you have to open a screen in the PARAMETER operating area with the softkeys "SETTINGDATA" and "JOG DATA" and preselect the G function **G95** there. The JOG feedrate can then be entered in [mm/rev]. (If 0 mm/rev is set as JOG feed, the control takes the value assigned in the axis-specific MD 32050 JOG\_REV\_VELO or in the case of rapid traverse overlay 32040 JOG\_REV\_VELO\_RAPID).

The revolutionary feed in JOG mode is deactivated by changing the G function from **G95** to **G94**.

Clear alarm with Cancel key. No further operator action necessary.

#### 20070

Explanation:

##### Channel %1 axis %2 programmed end position is beyond software limit %3

%1 = Channel number

%2 = Axis number

%3 = „+“ or „-“

The axis is traversed as competing positioning axis and the target position is situated behind the corresponding software limit switch. The axis does not traverse.

Reaction:

Alarm display.

Remedy:

Specify smaller target position. Modify MD for SW limit switch. Possibly activate another SW limit switch.

Alarm display showing cause of alarm disappears. No further operator action necessary.

#### 20071

Explanation:

##### Channel %1 axis %2 programmed end position is beyond working area limit %3

%1 = Channel number

%2 = Axis number

%3 = „+“ or „-“

The displayed axis is operated as a competing positioning axis. Its target position is behind the preset working area limitation.

Reaction:

Alarm display.

Remedy:

Parameterize target position within the permissible traversing range (parameter POS of FC ?) or correct position of software limit switch (activate 2nd software limit switch).

Alarm display showing cause of alarm disappears. No further operator action necessary.

#### 20072

Explanation:

##### Channel %1 axis %2 is not an indexing axis

%1 = Channel number

%2 = Axis number

The displayed axis is operated as a competing positioning axis. Its target position is parameterized in the FC INDEX-AXIS as indexing position number, but the axis is not an indexing axis.

Reaction:

Alarm display.

Remedy:

The FC POS-AXIS for linear and rotary axes should be used or the axis should be declared as an indexing axis.

Corresponding machine data for indexing axis declaration:

MD 30 500: INDEX\_AX\_ASSIGN\_POS\_TAB

MD 10 900: INDEX\_AX\_LENGTH\_POS\_TAB\_1

MD 10 910: INDEX\_AX\_POS\_TAB\_1

MD 10 920: INDEX\_AX\_LENGTH\_POS\_TAB\_2

MD 10 930: INDEX\_AX\_POS\_TAB\_2

Alarm display showing cause of alarm disappears. No further operator action necessary.



<b>20073</b>	<b>Channel %1 axis %2 cannot be repositioned</b>
Explanation:	%1 = Channel number %2 = Axis number The competing positioning axis cannot be positioned because it has already been restarted via the VDI interface and is still active. No repositioning motion takes place and the motion initiated by the VDI interface is not affected.
Reaction:	Alarm display.
Remedy:	None. Clear alarm with Cancel key. No further operator action necessary.
<b>20074</b>	<b>Channel %1 axis %2 wrong index position</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number For a competing positioning axis declared as indexing axis, the PLC has given an index number that is not available in the table.
Reaction:	Alarm display.
Remedy:	Check the indexing axis number given by the PLC and correct this if necessary. If the indexing axis number is correct and the alarm results from an indexing position table that has been set too short, check the machine data for indexing axis declaration. MD 30 500: INDEX_AX_ASSIGN_POS_TAB MD 10 900: INDEX_AX_LENGTH_POS_TAB_1 MD 10 910: INDEX_AX_POS_TAB_1 MD 10 920: INDEX_AX_LENGTH_POS_TAB_2 MD 10 930: INDEX_AX_POS_TAB_2 Alarm display showing cause of alarm disappears. No further operator action necessary.
<b>20075</b>	<b>Channel %1 axis %2 oscillating currently not possible</b>
Explanation:	%1 = Channel number %2 = Axis number The axis cannot perform an oscillating movement now because it is already being traversed, e.g. in JOG mode.
Reaction:	Alarm display.
Remedy:	End the other traversing motion. Clear alarm with Cancel key. No further operator action necessary.
<b>20076</b>	<b>Channel %1 axis %2 change of operation mode not possible during oscillation</b>
Explanation:	%1 = Channel number %2 = Axis number The axis is performing an oscillating movement. Mode change is not possible because oscillation is not allowed in the selected mode.
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	Do not initiate mode change. Cause the PLC to check the axis and make sure in the PLC program that the axis ends oscillation if such mode changes take place. Clear alarm with RESET key. Restart part program.
<b>20077</b>	<b>Channel %1 axis %2 programmed position is beyond software limit %3</b>
Explanation:	%1 = Channel number %2 = Axis number %3 = „+“ or „-“ The axis is traversed as oscillating axis and the target position (reversal position or end position) is located behind the corresponding software limit switch. The axis does not traverse.
Reaction:	Alarm display. Interface signals are set. NC Start disable. NC Stop when alarm.
Remedy:	Specify smaller target position. Modify MD for SW limit switch. Possibly activate another SW limit switch. Clear alarm with RESET key. Restart part program.
<b>20078</b>	<b>Channel %1 axis %2 programmed position is beyond working area limit %3</b>
Explanation:	%1 = Channel number %2 = Axis number %3 = „+“ or „-“ The axis is traversed as oscillating axis and the target position (reversal position or end position) is located behind the corresponding valid working area limitation. The axis does not traverse.
Reaction:	Alarm display. Interface signals are set. NC Start disable. NC Stop when alarm.
Remedy:	Specify smaller target position. Deactivate working area limitation. Set working area limitation differential. Clear alarm with RESET key. Restart part program.
<b>20080</b>	<b>Channel %1 axis %2 handwheel not assigned for overlaid handwheel motion</b>
Explanation:	%1 = Channel number %2 = Axis number No handwheel has been assigned for this axis after handwheel overlay has been started in automatic mode.
Reaction:	Alarm display.
Remedy:	If handwheel control is required, a handwheel must be activated. Alarm display showing cause of alarm disappears. No further operator action necessary.
<b>20085</b>	<b>Channel %1 contour handwheel: traverse direction or overtravel not allowed from beginning of block</b>
Explanation:	%1 = Channel number Travel takes place on the path with the contour handwheel in the opposite direction to the programmed travel direction and the starting point of the path has been reached at the start of the block.
Reaction:	Alarm display

Remedy:	Turn the contour handwheel in the opposite direction Alarm display verschwindet mit Alarmursache. No further operator action necessary.
<b>20090</b>	<b>Axis %1 activation of fixed stop not possible. Check program line and axis parameters.</b>
Explanation:	%1 = Axis name, spindle number 1. The „Traverse against fixed stop“ function has been programmed with FXS[AX]=1 but the axis does not (yet) support this. Check MD 37000. This function is not available for gantry axes and simulated axes. 2. On selection, no movement was programmed for axis AX. AX is a machine axis identifier. 3. It is always necessary to program a traversing movement in the selection block for the axis/spindle for which the „Traverse against fixed stop“ function is activated. The alarm can be reprogrammed in the MD ALARM_REACTION_CHAN_NOREADY (channel not ready).
Reaction:	Mode group not ready. In certain cases, it is possible to switch over for all channels via MD. Channel not ready. NC Start disable. NC Stop when alarm. Alarm display. Interface signals are set.
Remedy:	• Check the axis type • Check MD 37000 • Is a machine axis movement missing in the approach block? Press the Reset key to clear alarm in all channels of this mode group.
<b>20091</b>	<b>Axis %1 has not reached fixed stop</b>
Explanation:	%1 = Axis name, spindle number On attempting to traverse against a fixed stop, the programmed end position has been reached or the traversing movement has been aborted. The alarm can be concealed by means of the machine data \$MA_FIXED_STOP_ALARM_MASK. The alarm can be reprogrammed in the MD ALARM_REACTION_CHAN_NOREADY (channel not ready).
Reaction:	Mode group not ready. In certain cases, it is possible to switch over for all channels via MD. Channel not ready. NC Start disable. NC Stop when alarm. Alarm display. Interface signals are set.
Remedy:	Correct the part program and the settings: • Has the traversing block been aborted? • If the axis position does not correspond to the programmed end position, then correct the end position. • If the programmed end position is in the part, the triggering criterion must be checked. • Has the contour deviation leading to triggering been dimensioned too large? Has the torque limit been set too high? Press the Reset key to clear alarm in all channels of this mode group
<b>20092</b>	<b>Axis %1 fixed stop mode still active</b>
Explanation:	%1 = Axis name, spindle number An attempt has been made to move an axis while it is in fixed stop or while the deselection function has not yet been completed. The alarm can be reprogrammed in the MD ALARM_REACTION_CHAN_NOREADY (channel not ready).
Reaction:	Mode group not ready. In certain cases, it is possible to switch over for all channels via MD. Channel not ready. NC Start disable. NC Stop when alarm. Alarm display. Interface signals are set.
Remedy:	Check the following: • Has the axis at the fixed stop also been moved by a traversing movement of geometry axes? • Is a selection carried out even though the axis is stationary at the stop? • Has the deselection process been interrupted by a RESET? • Has the PLC switched the acknowledgement signals? Press the Reset key to clear alarm in all channels of this mode group.
<b>20200</b>	<b>Channel %1 invalid spindle no. %2 with fine compensation of tool geometry</b>
Explanation:	%1 = Channel number target channel %2 = Spindle number There is no spindle/axis assignment in the target channel for the spindle specified in the PUTFTOC command.
Reaction:	Alarm display. Interpreter stop. Interface signals are set. NC Start disable. NC Stop when alarm.
Remedy:	Modify program in channel that writes the tool fine compensation. Clear alarm with RESET key. Restart part program.
<b>20201</b>	<b>Channel %1 spindle %2 no tool assigned</b>
Explanation:	%1 = Channel number %2 = Spindle number In order to make allowance for the fine tool compensation for the tool currently in the spindle, a spindle/tool assignment must be active. This is not presently the case for the programmed spindle in the target channel of fine tool compensation.
Reaction:	Alarm display. Interpreter stop. Interface signals are set. NC Start disable. NC Stop when alarm.
Remedy:	1. Modify the part program (write the tool fine compensation). 2. Establish spindle/tool assignment by programming: • TMON (tool monitoring). • GWPSON (tool selection). Clear alarm with RESET key. Restart part program.

<b>20203</b>	<b>Channel %1 no tool selected</b>
Explanation:	%1 = Channel number A tool fine compensation has been written for the active tool of channel %1 with PUTFTOC. No tool is active in this channel. Therefore, the compensation cannot be assigned.
Reaction:	Alarm display. Interpreter stop. Interface signals are set. NC Start disable. NC Stop when alarm. Programm korrigieren Clear alarm with RESET key. Restart part program.
<b>20204</b>	<b>Channel %1 instruction PUTFTOC not allowed during FTOCOF</b>
Explanation:	%1 = Channel number A tool fine compensation has been written for channel %1 with PUTFTOC. The tool fine compensation is not active in this channel. FTOCON must be active in the target channel of the PUTFTOC command.
Reaction:	Alarm display. Interpreter stop. Interface signals are set. NC Start disable. NC Stop when alarm.
Remedy:	Correct the program in the machining channel: Select FTOCON so that the channel is ready to receive the PUTFTOC command. Clear alarm with RESET key. Restart part program.
<b>21617</b>	<b>Channel %1 block %2 transformation does not allow to traverse the pole.</b>
Explanation:	%1 = Channel number %2 = Block number, label The given curve passes through the pole or a forbidden area of transformation.
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	Modify part program (if alarm has occurred in AUTO mode). To escape from the alarm position, transformation must be deselected (it is not enough to try a RESET if the transformer remains active when RESET is applied). Clear alarm with RESET key. Restart part program.
<b>21618</b>	<b>Channel %1 as from block %2 transformation active: overlaid motion too great</b>
Explanation:	%1 = Channel number %2 = Block number, label The share of overlaid motion on the transformation-related axes is so high that the path movement planned by the preparation no longer sufficiently corresponds to the actual ratio for the interpolation. Strategy of singularities, monitoring of working range limitation and dynamic Look Ahead are possibly no longer correct.
Reaction:	Alarm display
Remedy:	With overlaid motion it is necessary to keep a sufficiently large path safety distance with regard to poles and working range limitations. Clear alarm with Cancel key. No further operator action necessary.
<b>21619</b>	<b>Channel %1 block %2 transformation active: motion not possible</b>
Explanation:	%1 = Channel number %2 = Block number, label The machine kinematics does not allow the specified motion.
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable
Remedy:	If the working area limitation is violated (see machine position), the part program's working area must be changed such that the possible operating range be adhered to (e.g. modified part settings). If the alarm is output in a pole position, care must be taken that in JOG it is only possible to traverse a pole or retract from it at the same angle at which it was entered. Note: RESET alone is not sufficient if Trafo also remains active after RESET. Clear alarm with RESET key. Restart part program.
<b>21650</b>	<b>Channel %1 axis %2 overlaid motion not allowed</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number An overlaid motion was requested for the axis, however, this is not allowed due to the machine data FRAME_OR_CORRPOS_NOTALLOWED.
Reaction:	Alarm display. Interface signals are set. NC Start disable. NC Stop when alarm.
Remedy:	Deselect the overlaid motion or change machine data FRAME_OR_CORRPOS_NOTALLOWED Clear alarm with RESET key. Restart part program.
<b>21700</b>	<b>Channel %1 block %3 axis %2 touch probe already deflected, edge</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number The probe programmed under the keyword MEAS or MEAW is already deflected and has switched. For a further measuring operation, the probe signal must first be canceled (quiescent state of the probe). The axis display is of no significance at the present time but an axis-specific evaluation has been planned for later stages of development.
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	Verify the start position of the measuring operation or check the probe signals. Are the cables and connectors in good order? Clear alarm with RESET key. Restart part program.
<b>21701</b>	<b>Channel %1 block %3 axis %2 measurement not possible</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number Isn't measurement possible?
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable. Clear alarm with RESET key. Restart part program.



<b>21702</b>	<b>Channel %1 block %3 axis %2 measurement aborted</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number The measurement block has ended (the programmed end position of the axis has been reached) but the activated touch probe has not yet responded.
Reaction:	Alarm display.
Remedy:	Verify the traversing movement in the measurements block. • Is it necessary in all cases for the activated probe to have switched up to the specified axis position? • Are the probe, cable, cable distributor, terminal connections in good order? Clear alarm with Cancel key. No further operator action necessary.
<b>21703</b>	<b>Channel %1 block %3 axis %2 touch probe not deflected, edge polarity not possible</b>
Explanation:	%1 = Channel number %2 = Axis name, spindle number %3 = Block number The selected probe is not (!) deflected and therefore cannot record any measured value from the deflected to the non-deflected state.
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	- Check probe - Check start positioning for measuring - Check program Clear alarm with RESET key. Restart part program.
<b>22000</b>	<b>Channel %1 block %3 spindle %2 change of gear stage not possible</b>
Explanation:	%1 = Channel number %2 = Spindelnummer %3 = Block number, label Automatic gear stage selection has been programmed with M40. The new M word is not in the present gear stage, but the spindle is not in „Open-loop control mode“. For automatic gear stage change (M40 in conjunction with spindle speed in address S) the spindle must be in "Open-loop control mode".
Reaction:	Alarm display. Interface signals are set. NC Stop when alarm. NC Start disable.
Remedy:	Before the S word which requires a gear stage change, change into the open-loop control mode of the spindle: Change to the open-loop control mode is carried out with: • M03, M04, M05 or M41 ... M45 from axis mode and positioning mode • Interface signal "Gear is changed" (DB 31 - 48, DBX 16.3) from oscillation mode Clear alarm with RESET key. Restart part program.
<b>22010</b>	<b>Channel %1 block %3 spindle %2 actual gear stage differs from requested gear stage.</b>
Explanation:	%1 = Channel number %2 = Spindelnummer %3 = Block number, label The requested gear stage change has been concluded. The actual gear stage reported by the PLC as being engaged is not the same as the required gear stage called for by the NC. Note: Wherever possible, the requested gear stage should always be engaged..
Reaction:	Alarm display.
Remedy:	PLC-Programm korrigieren. Clear alarm with Cancel key. No further operator action necessary.
<b>22270</b>	<b>Channel %1 block %2 spindle %3 spindle speed too high for thread cutting</b>
Explanation:	%1 = Channel number %2 = Block number, label %3 = Axis name, spindle number The spindle speed for thread cutting G33 is so high that the maximum axis velocity is exceeded because of the programmed thread lead.
Reaction:	Alarm display.
Remedy:	Program a lower spindle speed or a speed limitation with G26 S or reduce the spindle speed in front of the thread block by means of the setting data 43 220 SPIND_MAX_VELO_G26 or reduce the spindle override.
Remedy:	Clear alarm with Cancel key. No further operator action necessary.

## Cycle Alarms 60000 - 63000

These alarms will be triggered by the machining cycles of the control.

These are the same alarms as they would appear on the original SIEMENS control.

<b>61000</b>	<b>No tool offset active</b>
Cycle:	LONGHOLE, SLOT1, SLOT2, POCKET1, POCKET2, CYCLE90, CYCLE93, CYCLE94, CYCLE95, CYCLE96.
Remedy:	D offset must be programmed before the cycle is called.
<b>61001</b>	<b>Thread pitch wrong</b>
Cycle:	CYCLE84, CYCLE840, CYCLE96, CYCLE97.
Remedy:	Check parameters for thread size and check pitch information (contradict each other).
<b>61002</b>	<b>Machining type incorrectly defined</b>
Cycle:	SLOT1, SLOT2, POCKET1, POCKET2, CYCLE93, CYCLE95, CYCLE97, CYCLE98.
Remedy:	The value assigned to parameter VARI for the machining type is incorrect and must be altered.
<b>61101</b>	<b>Reference plane incorrectly defined</b>
Cycle:	CYCLE 81-90, CYCLE840, SLOT1, SLOT2, POCKET1, POCKET2, LONGHOLE.
Remedy:	Either different values must be entered for the reference plane and the retraction plane if they are relative values or an absolute value must be entered for the depth.
<b>61102</b>	<b>No spindle direction programmed</b>
Cycle:	CYCLE 86, CYCLE87, CYCLE88, CYCLE840
Remedy:	Parameter SDIR (or SDR in CYCLE840) must be programmed.
<b>61103</b>	<b>Number of holes equals zero</b>
Cycle:	HOLES1, HOLES2
Remedy:	No value has been programmed for the number of holes.
<b>61104</b>	<b>Contour violation of the slots/elongated holes</b>
Cycle:	SLOT1, SLOT2, LONGHOLE
Ursache:	Incorrect parameterization of the milling pattern in the parameters that define the position of the slots/elongated holes in the cycle and their shape.
<b>61105</b>	<b>Cutter radius too large</b>
Cycle:	SLOT1, SLOT2, POCKET1, POCKET2, LONGHOLE, CYCLE90
Remedy:	The diameter of the milling cutter being used is too large for the figure that is to be machined; either a tool with a smaller radius must be used or the contour must be changed.
<b>61106</b>	<b>Number of or distance between circular elements</b>
Cycle:	HOLES2, LONGHOLE, SLOT1, SLOT2
Ursache:	Incorrect parameterization of NUM or INDA, the circular elements cannot be arranged in a full circle.
<b>61107</b>	<b>First drilling depth incorrectly defined</b>
Cycle:	CYCLE83
Ursache:	First drilling depth is incompatible with final drilling depth.
<b>61601</b>	<b>Finished part diameter too small</b>
Cycle:	CYCLE94, CYCLE96
Ursache:	A finished part diameter of <3 mm has been programmed
<b>61602</b>	<b>Tool width incorrectly defined</b>
Cycle:	CYCLE93
Ursache:	Grooving tool is larger than the programmed groove width.
<b>61603</b>	<b>Groove form incorrectly defined</b>
Cycle:	CYCLE93
Ursache:	<ul style="list-style-type: none"> <li>• Radii/chamfers at the base of the groove are not compatible with the groove width</li> <li>• Face recess on a contour element that runs parallel to the longitudinal axis is not possible.</li> </ul>
<b>61604</b>	<b>Active tool violates programmed contour</b>
Cycle:	CYCLE95
Remedy:	Contour violation in relief cut elements as a result of the clearance angle of the tool being used, i.e. used a different tool or check the contour subroutine.

<b>61605</b>	<b>Contour incorrectly programmed</b>
Cycle:	CYCLE95
Ursache:	Illegal relief cut element detected.
<b>61606</b>	<b>Error on contour preparation</b>
Cycle:	CYCLE95
Ursache:	An error was detected during contour preparation, this alarm is always output with NCK alarm 10930 ... 10934, 15800 or 15810.
<b>61607</b>	<b>Starting point incorrectly programmed</b>
Cycle:	CYCLE95
Ursache:	The starting point reached before the cycle was called does not lie outside the rectangle described by the contour subroutine.
<b>61608</b>	<b>Wrong tool point direction programmed</b>
Cycle:	CYCLE94, CYCLE96
Remedy:	A tool point direction between 1 ... 4 that matches the undercut form must be programmed.
<b>61109</b>	<b>Form incorrectly programmed</b>
Cycle:	CYCLE94, CYCLE96
Remedy:	Check parameters for the undercut form
<b>61110</b>	<b>No drilling cycle active</b>
Cycle:	HOLES1, HOLES2
Remedy:	No drilling cycle was called modally before the drilling pattern cycle was called.

