HEIDENHAIN

User's Manual HEIDENHAIN Conversational Programming

TNC 360

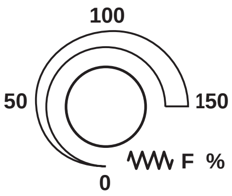
May 1994

Keys and Controls on the TNC 360

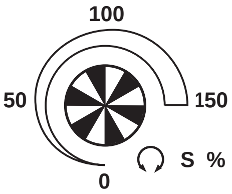
Controls on the Visual Display Unit

☼	Brightness

Override Knobs



Feed rate



Spindle speed

Machine Operating Modes

	MANUAL OPERATION
	ELECTRONIC HANDWHEEL
	POSITIONING WITH MANUAL DATA INPUT
	PROGRAM RUN, SINGLE BLOCK
	PROGRAM RUN, FULL SEQUENCE

Programming Modes

	PROGRAMMING AND EDITING
	TEST RUN

Program and File Management

PGM NR	Select programs and files
CL PGM	Delete programs and files
PGM CALL	Enter program call in a program
EXT	External data transfer
MOD	Supplementary modes

Cursor and GOTO keys

← ↑ → ↓	Move cursor (highlight)
GOTO □	Go directly to blocks, cycles and parameter functions

Graphics

MOD	Graphic operating modes
BLK FORM	Define blank form, reset blank form
MAGN	Magnify detail
START	Start graphic simulation

Programmable Contours

L	Straight line
CC	Circle center / Pole for polar coordinates
C	Circle with center point
CR	Circle with radius
CT	Circle with tangential transition
RND	Corner rounding

Tool Functions

TOOL DEF TOOL CALL	Enter or call tool length and radius
R+ R-	Activate tool radius compensation

Cycles, Subprograms and Program Section Repeats

CYCL DEF CYCL CALL	Define and call cycles
LBL SET LBL CALL	Enter and call labels for subprogramming and program section repeats
STOP	Abort an interrupted program run or enter a program stop in a program
TOUCH PROBE	Set a datum with the 3D touch probe or enter touch probe functions in a program

Entering Numbers and Coordinate Axes, Editing

X ... IV	Select or enter coordinate axes in a program
0 ... 9	Numbers
.	Decimal point
+/–	Algebraic sign
P	Polar coordinates
I	Incremental values
Q Q DEF	Q parameters for part families or in mathematical functions
✛	Actual position capture
NO ENT	Ignore dialog queries, delete words
ENT	Confirm entry and resume dialog
END □	Conclude block
CE	Clear numerical entry or TNC message
DEL □	Abort dialog; delete program sections

TNC Guideline:
From workpiece drawing to program-controlled machining

Step	Task	TNC operating mode	Refer to Section
	Preparation		
1	Select tools	——	——
2	Set workpiece datum for coordinate system	——	——
3	Determine spindle speeds and feed rates	——	12.4
4	Switch on machine	——	1.3
5	Traverse reference marks	✋ or ☸	1.3, 2.1
6	Clamp workpiece	——	——
7	Set the datum / Reset position display ...		
7a	... **with** the 3D touch probe	✋ or ☸	2.5
7b	... **without** the 3D touch probe	✋ or ☸	2.3
	Entering and testing part programs		
8	Enter part program or download over external data interface	◈ or EXT	5 to 8 or 10
9	Test part program for errors	⇥	3.1
10	Test run: Run program block by block without tool	⊡	3.2
11	If necessary: Optimize part program	◈	5 to 8
	Machining the workpiece		
12	Insert tool and run part program	⇥	3.2

Sequence of Program Steps

Milling an outside contour

Program step	Key	Refer to Section
1 Create or select program Input: Program number Unit of measure for programming	PGM NR	4.4
2 Define workpiece blank	BLK FORM	4.4
3 Define tools Input: Tool number Tool length Tool radius	TOOL DEF	4.2
4 Call tool data Input: Tool number Spindle axis Spindle speed	TOOL CALL	4.2
5 Tool change Input: Coordinates of the tool change position Radius compensation Feed rate (rapid traverse) Miscellaneous function (tool change)	L	e.g. 5.4
6 Move to starting position Input: Coordinates of the starting position Radius compensation (R0) Feed rate (rapid traverse) Miscellaneous function (spindle on, clockwise)	L	5.2/5.4
7 Move tool to (first) working depth Input: Coordinate of the (first) working depth Feed rate (rapid traverse)	L	5.4
8 Move to first contour point Input: Coordinates of the first contour point Radius compensation for machining Machining feed rate if desired, with smooth approach: RND after this block	L	5.2/5.4
9 Machining to last contour point Input: Enter all necessary values for each contour element		5 to 8
10 Move to end position Input: Coordinates of the end position Radius compensation (R0) Miscellaneous function (spindle stop) if desired, with smooth departure: RND after this block	L	5.2/5.4
11 Retract tool in spindle axis Input: Coordinates above the workpiece Feed rate (rapid traverse) Miscellaneous function (end of program)	L	5.2/5.4
12 End of program		

How to use this manual

This manual describes functions and features available on the TNC 360 from NC software number 259 900 11.

This manual describes all available TNC functions. However, since the machine builder has modified (with machine parameters) the available range of TNC functions to interface the control to his specific machine, this manual may describe some functions which are not available on your TNC.

TNC functions which are not available on every machine are, for example:

- Probing functions for the 3D touch probe system
- Digitizing
- Rigid tapping

If in doubt, please contact the machine tool builder.

TNC programming courses are offered by many machine tool builders as well as by HEIDENHAIN. We recommend these courses as an effective way of improving your programming skill and sharing information and ideas with other TNC users.

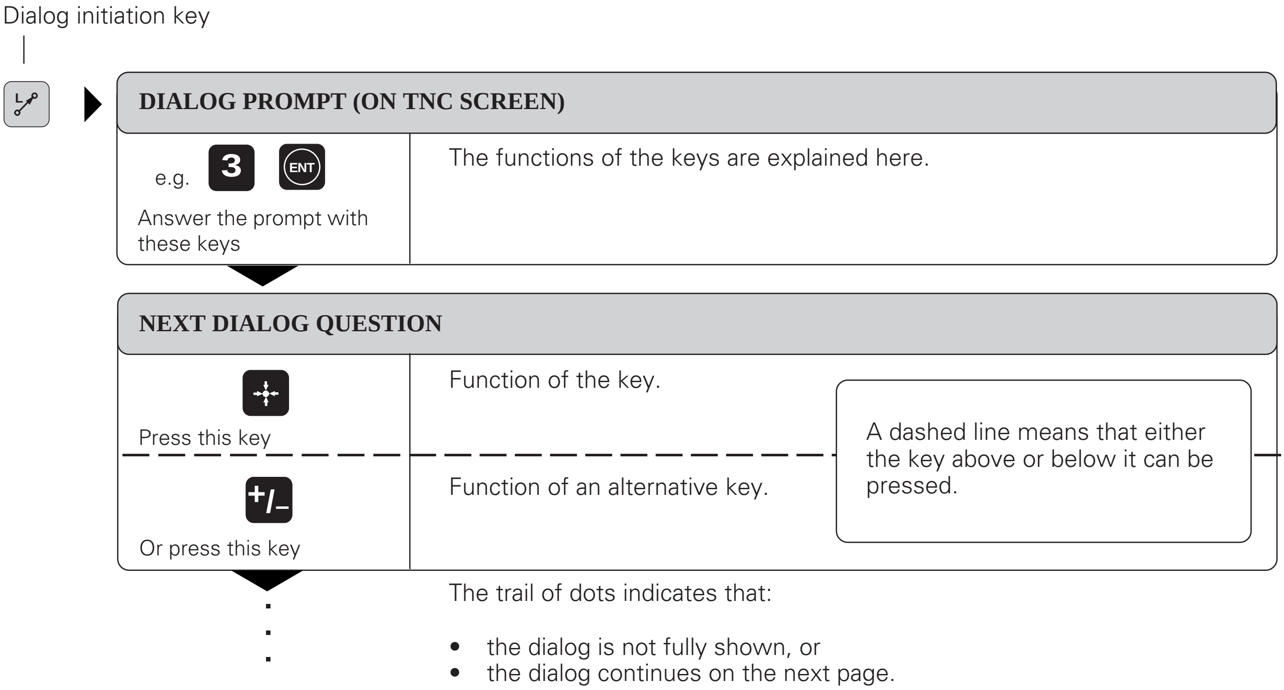
The **TNC beginner** can use the manual as a workbook. The first part of the manual deals with the basics of NC technology and describes the TNC functions. It then introduces the techniques of conversational programming. Each new function is thoroughly described when it is first introduced, and the numerous examples can be tried out directly on the TNC. The TNC beginner should work through this manual from beginning to end to ensure that he is capable of fully exploiting the features of this powerful tool.

For the **TNC expert**, this manual serves as a comprehensive reference work. The table of contents and cross references enable him to quickly find the topics and information he needs. Easy-to-read dialog flowcharts show him how to enter the required data for each function.

The dialog flow charts consist of sequentially arranged instruction boxes. Each key is illustrated next to an explanation of its function to aid the beginner when he is performing the operation for the first time. The experienced user can use the key sequences illustrated in the left part of the flowchart as a quick overview. The TNC dialogs in the instruction boxes are always presented on a gray background.

Note: Placeholders in the program on the screen for entries which are not always programmed (such as the abbreviations R, F, M and REP) are not indicated in the programming examples.

Layout of the dialog flowcharts



Dialog initiation key

DIALOG PROMPT (ON TNC SCREEN)

e.g. 3 ENT

Answer the prompt with these keys

The functions of the keys are explained here.

NEXT DIALOG QUESTION

Press this key

Function of the key.

Or press this key

Function of an alternative key.

A dashed line means that either the key above or below it can be pressed.

The trail of dots indicates that:

- the dialog is not fully shown, or
- the dialog continues on the next page.

Contents User's Manual TNC 360 (from 259 900-xx)

1 Introduction

2 Manual Operation and Setup

3 Test Run and Program Run

4 Programming

5 Programming Tool Movements

6 Subprograms and Program Section Repeats

7 Programming with Q Parameters

8 Cycles

9 Digitizing 3D Surfaces

10 External Data Transfer

11 MOD Functions

12 Tables, Overviews, Diagrams

1.1 The TNC 360

Control

The TNC 360 is a shop-floor programmable contouring control for milling machines, boring machines and machining centers with up to four axes. The spindle can be rotated to a given angular stop position (oriented spindle stop).

Visual display unit and operating panel

The monochrome screen clearly displays all information necessary for operating the TNC. In addition to the CRT monitor (BE 212), the TNC 360 can also be used with a flat luminescent screen (BF 110). The keys on the operating panel are grouped according to their functions. This simplifies programming and the application of the TNC functions.

Programming

The TNC 360 is programmed directly at the machine with the easy to understand HEIDENHAIN plain language dialog format. Programming in ISO or in DNC mode is also possible.

Graphics

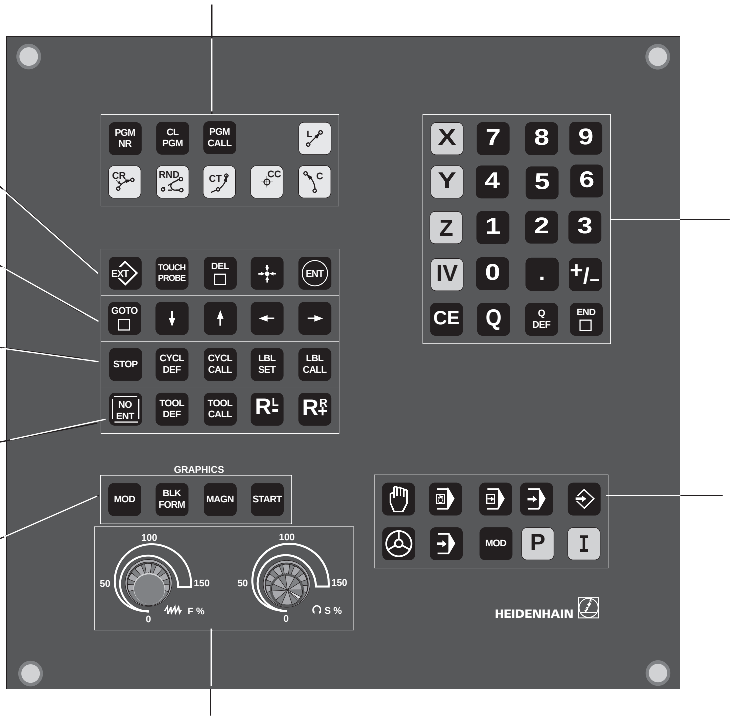
The graphic simulation feature allows programs to be tested before actual machining. Various types of graphic representation can be selected.

Compatibility

Any part program can be run on the TNC 360 as long as the commands in the program are within the functional scope of the TNC 360.

The Operating Panel

The keys on the TNC operating panel are identified with easy-to-remember abbreviations and symbols. The keys are grouped according to function:



- Program selection
- Path function keys

- External data transfer
- Probing functions
- Editing functions

- GOTO statement
- Arrow keys

- STOP key
- Programming of cycles, program section repeats and subprograms

- NO ENT key
- Tool-related entries

Graphic operating modes

- Numerical entries
- Axis selection
- Q parameter programming

- Operating modes
- Incremental and polar coordinates

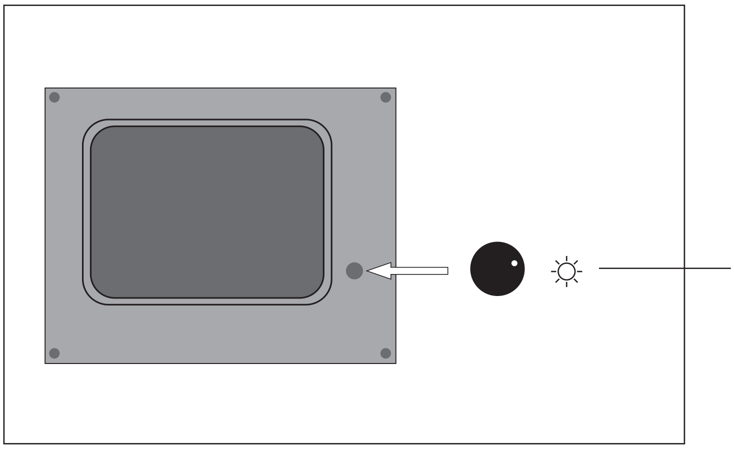
Override controls for spindle speed and feed rate

The functions of the individual keys are described on the inside front cover.

The machine operating buttons, such as I for NC start, are described in the manual for your machine tool.

In this manual they are shown in gray.

The Screen



Brightness control (BE 212 only)

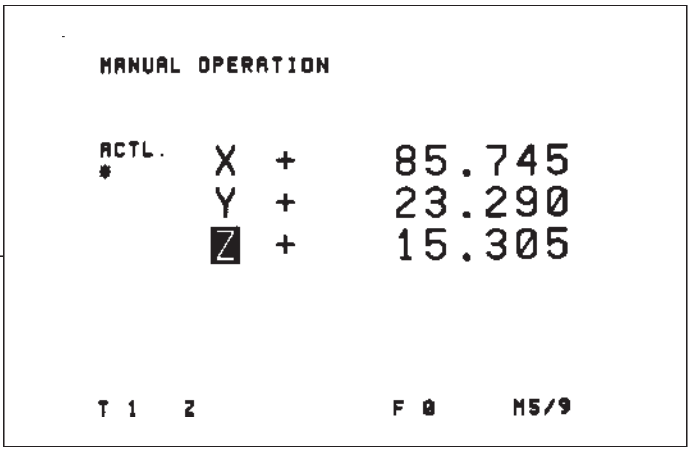
Header

The header of the screen shows the selected operating mode. Dialog questions and TNC messages also appear there.

Screen Layout

MANUAL and EL. HANDWHEEL operating modes:

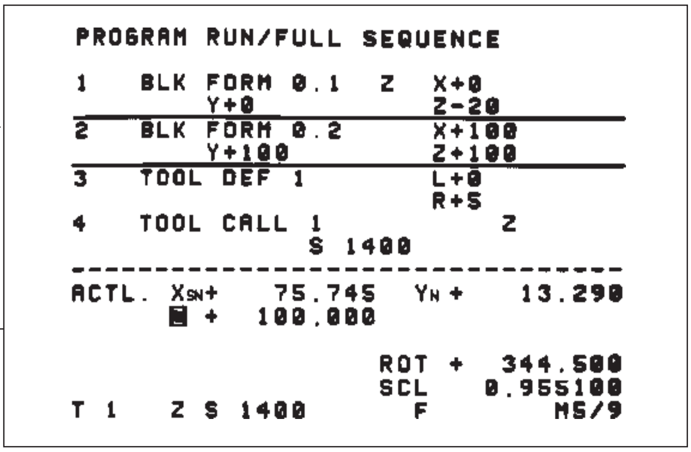
A machine operating mode has been selected



```
MANUAL OPERATION

ACTL.  X  +    85.745
*      Y  +    23.290
       Z  +    15.305




T 1   Z            F 0      M5/9
```

- Coordinates
- Selected axis
- * means: control is in operation
- Status display, e.g. feed rate F, miscellaneous function M

A program run operating mode has been selected



```
PROGRAM RUN/FULL SEQUENCE
1    BLK FORM 0.1   Z   X+0
         Y+0            Z-20
2    BLK FORM 0.2       X+100
         Y+100          Z+100
3    TOOL DEF 1         L+0
                        R+5
4    TOOL CALL 1            Z
               S 1400
---------------------------------
ACTL. XSN+    75.745   YH +    13.290
      ■ +   100.000

                    ROT +   344.500
                    SCL    0.955100
T 1    Z S 1400       F         M5/9
```

Section of selected program

Status display

The screen layout is the same in the operating modes PROGRAM RUN, PROGRAMMING AND EDITING and TEST RUN. The current block is surrounded by two horizontal lines.

TNC Accessories

3D Probe Systems

The TNC features the following functions for the HEIDENHAIN 3D touch probe systems:

- Automatic workpiece alignment (compensation of workpiece misalignment)
- Datum setting
- Measurements of the workpiece can be performed during program run
- Digitizing 3D forms (optional)

The TS 120 touch probe system is connected to the control via cable, while the TS 510 communicates by means of infrared light.

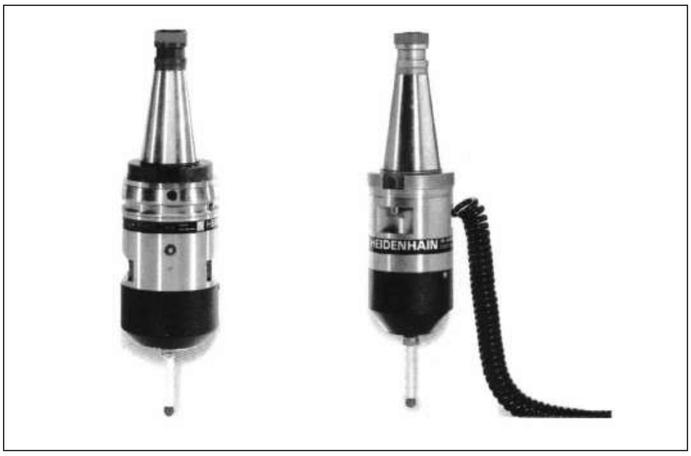


Fig. 1.5: HEIDENHAIN 3D Probe Systems TS 120 and TS 511

Floppy Disk Unit

The HEIDENHAIN FE 401 floppy disk unit serves as an external memory for the TNC, allowing you to store your programs externally on diskette.

The FE 401 can also be used to transfer programs that were written on a PC into the TNC. Extremely long programs which exceed the TNC's memory capacity are “drip fed” block by block. The machine executes the transferred blocks and erases them immediately, freeing memory for further blocks from the FE.

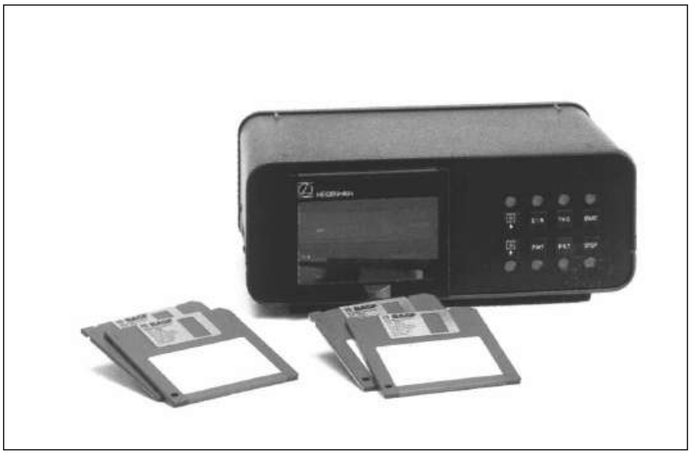


Fig. 1.6: HEIDENHAIN FE 401 Floppy Disk Unit

Electronic Handwheels

Electronic handwheels provide precise manual control of the axis slides. As on conventional machines, turning the handwheel moves the axis by a defined amount. The traverse distance per revolution of the handwheel can be adjusted over a wide range.

Portable handwheels, such as the HR 330, are connected to the TNC by cable. Built-in handwheels, such as the HR 130, are built into the machine operating panel.

An adapter allows up to three handwheels to be connected simultaneously. Your machine manufacturer can tell you more about the handwheel configuration of your machine.

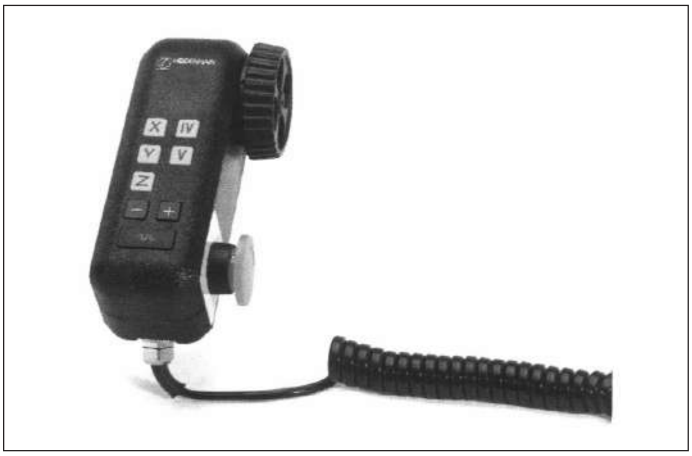


Fig. 1.7: The HR 330 Electronic Handwheel

1.2 Fundamentals of Numerical Control (NC)

Introduction

This chapter addresses the following topics:

- What is NC?
- The part program
- Conversational programming
- Cartesian coordinate system
- Additional axes
- Polar coordinates
- Setting a pole at a circle center (CC)
- Datum setting
- Absolute workpiece positions
- Programming tool movements
- Position encoders
- Reference mark evaluation

What is NC?

NC stands for Numerical Control. Simply put, numerical control is the operation of a machine by means of coded instructions. Modern controls such as the HEIDENHAIN TNCs have a built-in computer for this purpose. Such a control is therefore also called a CNC (Computer Numerical Control).

The part program

A part program is a complete list of instructions for machining a workpiece. It contains such information as the target position of a tool movement, the tool path — i.e. how the tool should move towards the target position — and the feed rate. The program must also contain information on the radius and length of the tools, the spindle speed and the tool axis.

Conversational programming

Conversational programming is a particularly easy way of writing and editing part programs. From the very beginning, HEIDENHAIN numerical controls were designed for the machinist who keys in his programs directly at the machine. This is why they are called TNCs, or "Touch Numerical Controls."

You begin programming each machining step by simply pressing a key. The control then asks for all further information required to execute the step. You can also program the TNC in ISO format or download programs from a central host computer for DNC operation.

Reference system

In order to define positions, one needs a reference system. For example, positions on the earth's surface can be defined "absolutely" by their geographic coordinates of longitude and latitude. The term "coordinate" comes from the Latin word for "that which is arranged". The network of horizontal and vertical lines around the globe constitute an "absolute reference system" — in contrast to the "relative" definition of a position that is referenced, for example, to some other, known location.

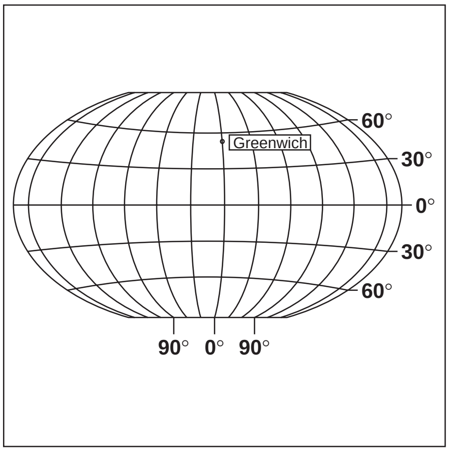


Fig. 1.9: The geographic coordinate system is an absolute reference system

Cartesian coordinate system

A workpiece is normally machined on a TNC controlled milling machine according to a workpiece-reference Cartesian coordinate system (a rectangular coordinate system named after the French mathematician and philosopher Renatus Cartesius; 1596 to 1650). The Cartesian coordinate system is based on three coordinate axes X, Y and Z, which are parallel to the machine guideways. The figure to the right illustrates the "right hand rule" for remembering the three axis directions: the middle finger is pointing in the positive direction of the tool axis from the workpiece toward the tool (the Z axis), the thumb is pointing in the positive X direction, and the index finger in the positive Y direction.

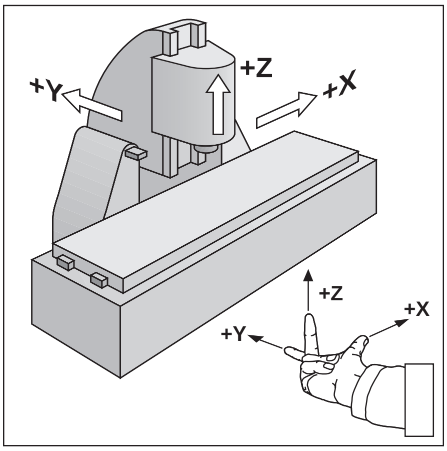


Fig. 1.10: Designations and directions of the axes on a milling machine

Additional axes

The TNC can control machines which have more than three axes. **U**, **V** and **W** are secondary linear axes parallel to the main axes X, Y and Z, respectively (see illustration). **Rotary axes** are also possible. They are designated as axes **A**, **B** and **C**.

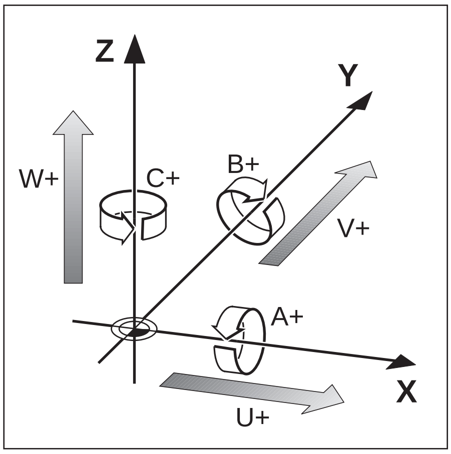


Fig. 1.11: Arrangement and designation of the auxiliary axes

Polar coordinates

The Cartesian coordinate system is especially useful for parts whose dimensions are mutually perpendicular. But when workpieces contain circular arcs, or when dimensions are given in degrees, it is often easier to use polar coordinates. In contrast to Cartesian coordinates, which are three-dimensional, polar coordinates can only describe positions in a plane.

The datum for polar coordinates is the **circle center CC**. To describe a position in polar coordinates, think of a scale whose datum point is rigidly connected to the pole but which can be freely rotated in a plane around the pole.

Positions in this plane are defined by:

- **Polar Radius (PR):** The distance from circle center CC to the defined position.

- **Polar Angle (PA):** The angle between the reference axis and the scale.

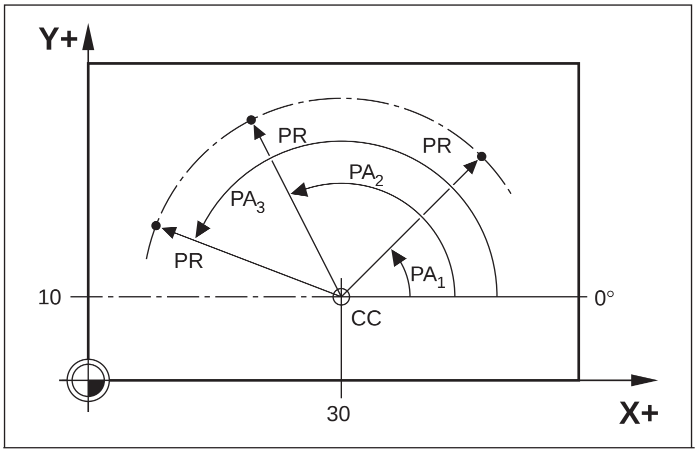


Fig. 1.12: Positions on an arc with polar coordinates

Setting a pole at circle center CC

The pole (circle center) is defined by setting two Cartesian coordinates. These two coordinates also determine the reference axis for the polar angle PA.

Coordinates of the pole	Reference axis of the angle
X Y	+X
Y Z	+Y
Z X	+Z

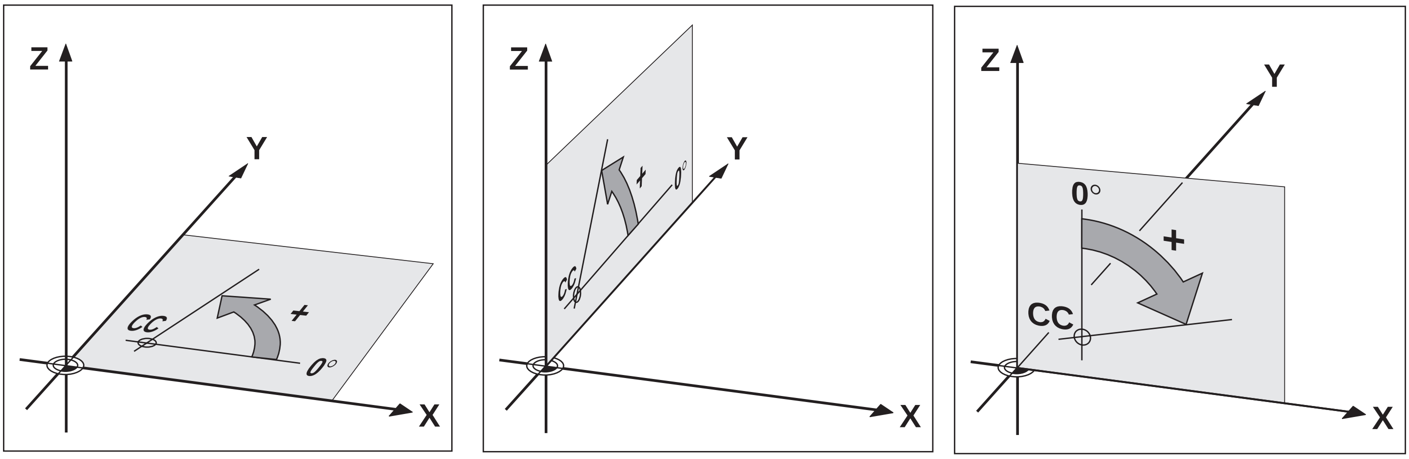


Fig. 1.13: Polar coordinates and their associated reference axes

Setting the datum

The workpiece drawing identifies a certain prominent point on the workpiece (usually a corner) as the "absolute datum" and perhaps one or more other points as relative datums. The process of datum setting establishes these points as the origin of the absolute or relative coordi-nate systems: The workpiece, which is aligned with the machine axes, is moved to a certain position relative to the tool and the display is set either to zero or to another appropriate position value (e.g. to compen-sate the tool radius).

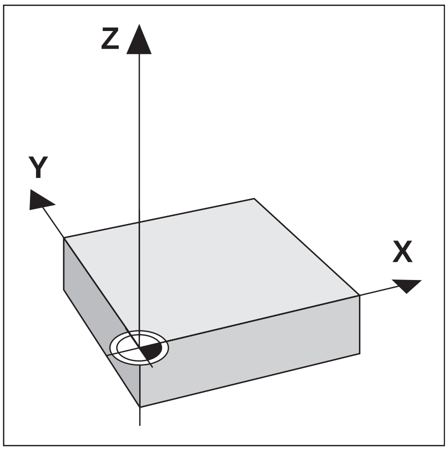
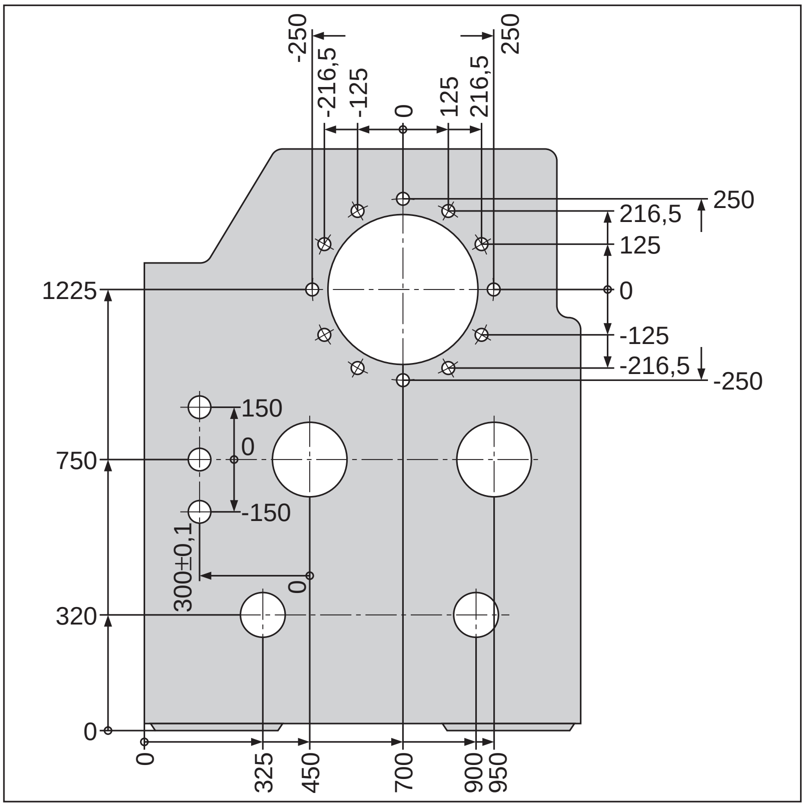


Fig. 1.14: The workpiece datum serves as the origin of the Cartesian coordinate system

Example:

Drawings with several relative datums
(according to ISO 129 or DIN 406, Part 11; Figure 171)



Example:

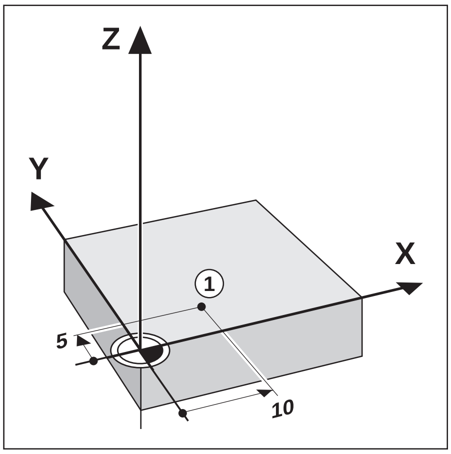


Fig. 1.16: Point 1 defines the coordinate system.

Coordinates of the point 1:

X = 10 mm
Y = 5 mm
Z = 0 mm

The datum of the Cartesian coordinate system is located 10 mm away from point 1 on the X axis and 5 mm on the Y axis.

The 3D Touch Probe System from HEIDENHAIN is an especially convenient and efficient way to find and set datums.

Absolute workpiece positions

Each position on the workpiece is clearly defined by its absolute coordinates.

Example: Absolute coordinates of the position ①:
X = 20 mm
Y = 10 mm
Z = 15 mm

If you are drilling or milling a workpiece according to a workpiece drawing with absolute coordinates, you are moving the tool **to** the coordinates.

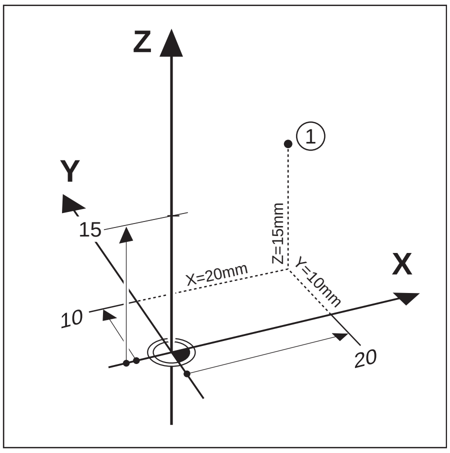


Fig. 1.17: Position definition through absolute coordinates

Incremental workpiece positions

A position can be referenced to the previous nominal position: i.e. the relative datum is always the last programmed position. Such coordinates are referred to as **incremental coordinates** (increment = growth), or also incremental or chain dimensions (since the positions are defined as a chain of dimensions). Incremental coordinates are designated with the prefix I.

Example: Incremental coordinates of the position ③ referenced to position ②

Absolute coordinates of the position ② :
X = 10 mm
Y = 5 mm
Z = 20 mm

Incremental coordinates of the position ③ :
IX = 10 mm
IY = 10 mm
IZ = –15 mm

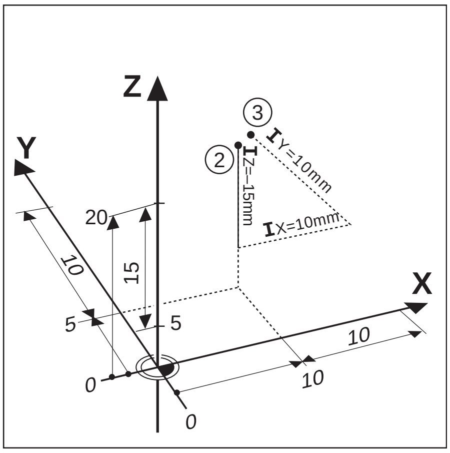


Fig. 1.18: Position definition through incremental coordinates

If you are drilling or milling a workpiece according to a workpiece drawing with incremental coordinates, you are moving the tool **by** the coordinates.

An incremental position definition is therefore intended as an immediately relative definition. This is also the case when a position is defined by the **distance-to-go** to the target position (here the relative datum is located at the target position). The distance-to-go has a negative algebraic sign if the target position lies in the negative axis direction from the actual position.

The polar coordinate system can also express both types of dimensions:

- **Absolute polar coordinates** always refer to the pole (CC) and the reference axis.
- **Incremental polar coordinates** always refer to the last programmed nominal position of the tool.

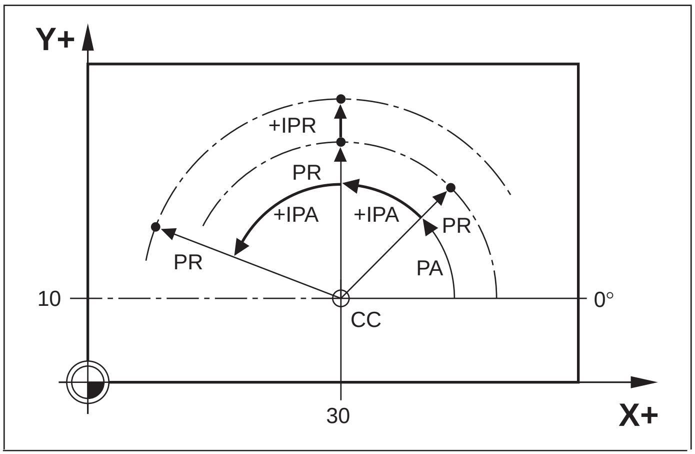
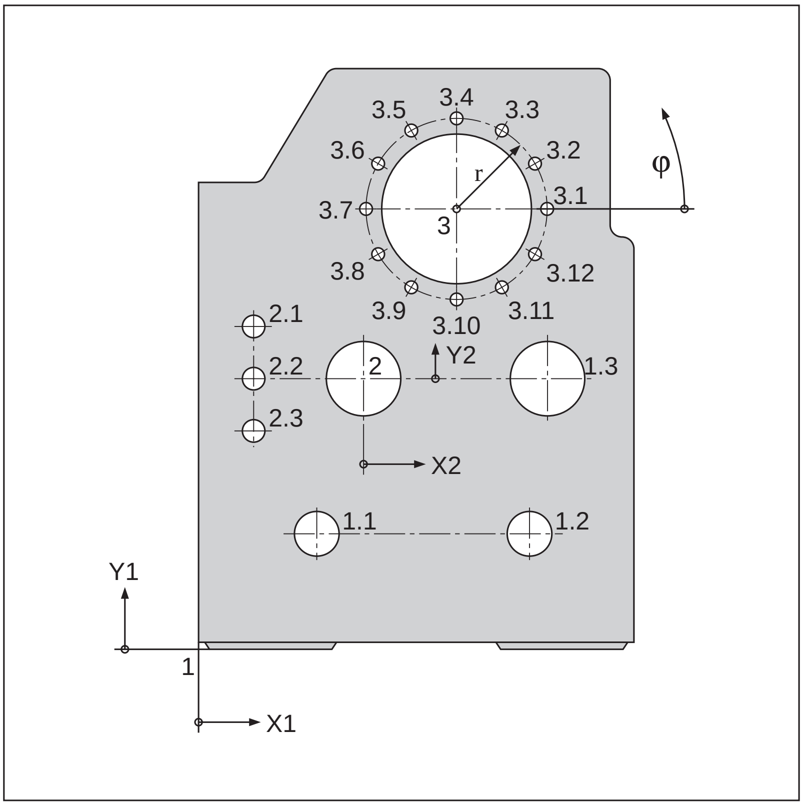


Fig. 1.19: Incremental dimensions in polar coordinates (designated with an "I")

Example:

Workpiece drawing with coordinate dimensioning (according to ISO 129 or DIN 406, Part 11; Figure 179)



Coordinate origin	Pos.	Dimensions in mm				
		Coordinates				
		X1 X2	Y1 Y2	r	φ	d
1	1	0	0			-
1	1.1	325	320			Ø 120 H7
1	1.2	900	320			Ø 120 H7
1	1.3	950	750			Ø 200 H7
1	2	450	750			Ø 200 H7
1	3	700	1225			Ø 400 H8
2	2.1	–300	150			Ø 50 H11
2	2.2	–300	0			Ø 50 H11
2	2.3	–300	–150			Ø 50 H11
3	3.1			250	0°	Ø 26
3	3.2			250	30°	Ø 26
3	3.3			250	60°	Ø 26
3	3.4			250	90°	Ø 26
3	3.5			250	120°	Ø 26
3	3.6			250	150°	Ø 26
3	3.7			250	180°	Ø 26
3	3.8			250	210°	Ø 26
3	3.9			250	240°	Ø 26
3	3.10			250	270°	Ø 26
3	3.11			250	300°	Ø 26
3	3.12			250	330°	Ø 26

Programming tool movements

During workpiece machining, an axis position is changed either by moving the tool or by moving the machine table on which the workpiece is fixed.

> You always program as if the tool is moving and the workpiece is stationary.

If the machine table moves, the axis is designated on the machine operating panel with a prime mark (e.g. X’, Y’). Whether an axis designation has a prime mark or not, the programmed direction of axis movement is always the direction of tool movement relative to the workpiece.

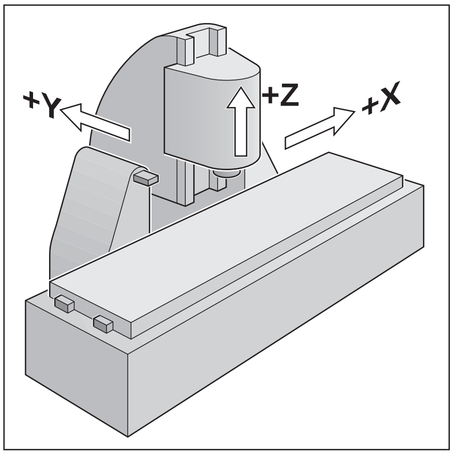


Fig. 1.21: On this machine the tool moves in the Y and Z axes; the workpiece moves in the X axis.

Position encoders

The position encoders – linear encoders for linear axes, angle encoders for rotary axes – convert the movement of the machine axes into electrical signals. The control evaluates these signals and constantly calculates the actual position of the machine axes.

If there is an interruption in power, the calculated position will no longer correspond to the actual position. When power is returned, the TNC can re-establish this relationship.

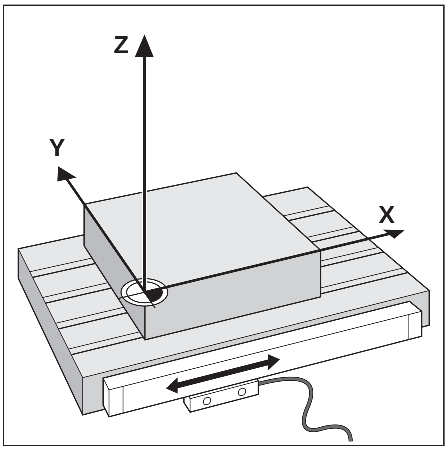


Fig. 1.22: Linear position encoder, here for the X axis

Reference marks

The scales of the position encoders contain one or more reference marks. When a reference mark is passed over, it generates a signal which identifies that position as the machine axis reference point.
With the aid of this reference mark the TNC can re-establish the assignment of displayed positions to machine axis positions.

If the position encoders feature **distance-coded** reference marks, each axis need only move a maximum of 20 mm (0.8 in.) for linear encoders, and 20° for angle encoders.

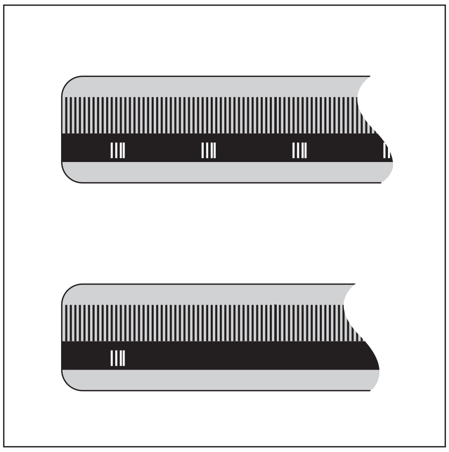


Fig. 1.23: Linear scales: above with distance-coded-reference marks, below with one reference mark

1.3 Switch-On

Switch on the power supply for the TNC and machine. The TNC then begins the following dialog:

MEMORY TEST

The TNC memory is automatically checked.

POWER INTERRUPTED

Message from the TNC indicating that the power was interrupted.
Clear the message with the CE key. ▶ CE

TRANSLATE PLC PROGRAM

The PLC program of the TNC is automatically translated.

RELAY EXT. DC VOLTAGE MISSING

Switch on the control voltage.
The TNC checks the functioning of the EMERGENCY STOP circuit. ▶ I

MANUAL OPERATION

TRAVERSE REFERENCE POINTS

To cross over the reference marks in the displayed sequence:
Press the START button for each axis. ▶ I

To cross over the reference marks in any sequence:
For each axis, press and hold down the axis direction button
until the reference mark has been crossed over. ▶ X , Y , ...

The TNC is now ready for operation. The operating mode MANUAL OPERATION is active.

1.4 Graphics and Status Display

The TNC features various graphic display modes for testing programs. To be able to use this feature, you must select a program run operating mode.

Workpiece machining is simulated graphically in the display modes:

- Plan view
- Projection in three planes
- 3D view

With the fast internal image generation, the TNC calculates the contour and displays a graphic only of the completed part.

Select display mode

GRAPHICS 2 x MOD	Select display mode menu.
↑ / ↓	Select desired display mode.
ENT	Confirm selection.

Start graphic display

GRAPHICS START	Start graphic simulation in the selected display mode.

The START key repeats a graphic simulation as often as desired.

Rotary axis movements cannot be graphically simulated.
An attempted test run will result in an error message.

Plan view

In this mode, contour height is symbolized by image brightness.
The deeper the contour, the darker the image.

Number of depth levels: 7

This is the fastest of the three display modes.

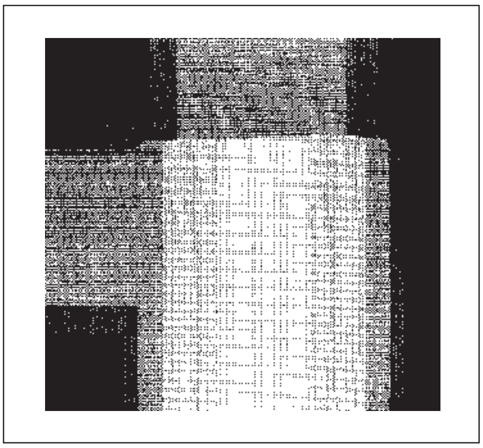


Fig. 1.18: TNC graphics, plan view

Projection in three planes

Here the program is displayed as in a technical drawing, with a plan view and two orthographic sections. A conical symbol near the graphic indicates whether the display is in first angle or third angle projection according to ISO 6433. The type of projection can be selected with MP 7310.

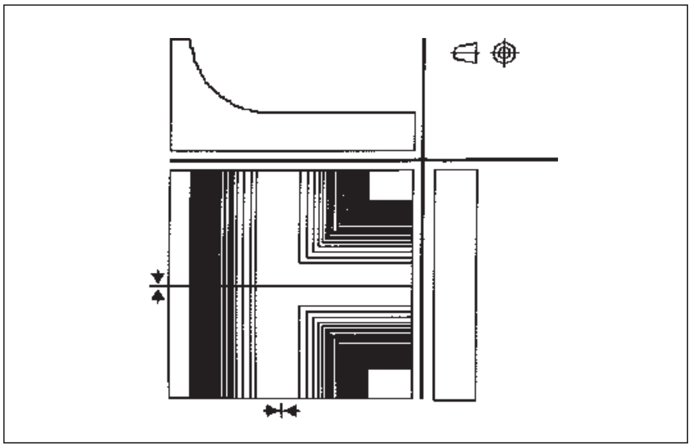


Fig. 1.19: Projection in three planes

Moving the sectional plane

The sectional planes can moved to any position with the arrow keys. The position of the sectional plane is displayed on the screen while it is being moved.

3D view

This mode displays the simulated workpiece in three-dimensional space.

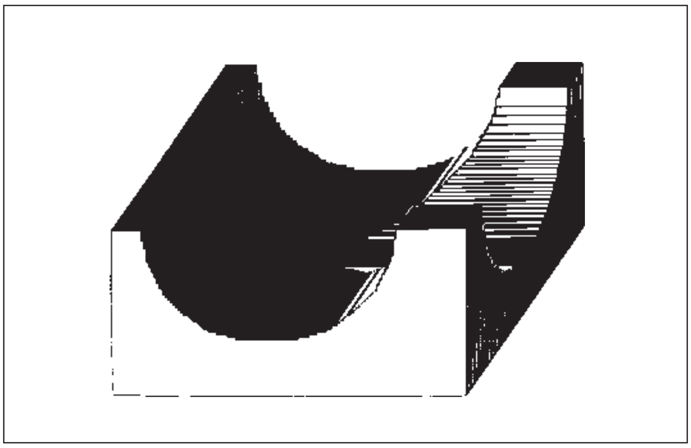


Fig. 1.20: 3D view

Rotating the 3D view

In the 3D view, the image can be rotated around the vertical axis with the horizontal arrow keys. The angle of orientation is indicated with a special symbol:

└ 0^0 rotation

┘ 90^0 rotation

┐ 180^0 rotation

┌ 270^0 rotation

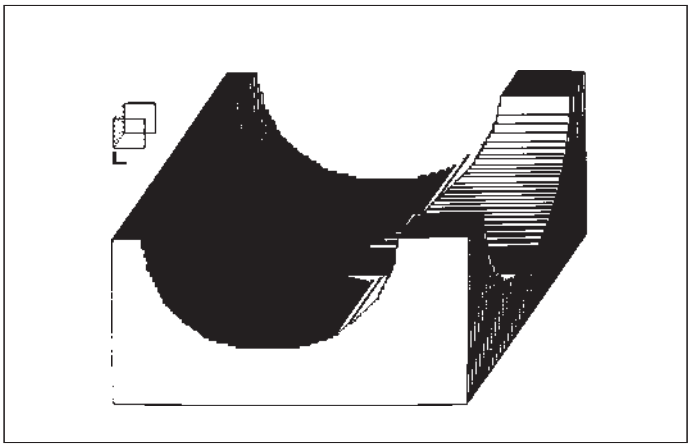


Fig. 1.21: Rotated 3D view

3D view, not true to scale

If the height-to-side ratio is between 0.5 and 50, a non-scaled 3D view can be selected with the vertical arrow keys. This view improves the resolution of the shorter workpiece side.

The dimensions of the angle orientation symbol change to indicate the disproportion.

Detail magnification of a 3D graphic

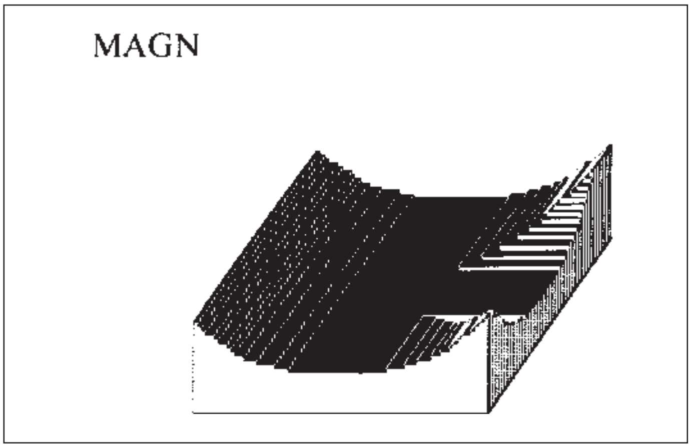


Fig. 1.22: Detail magnification of a 3D graphic

GRAPHICS MAGN	Select function for detail magnification.
↑ / ↓	Select sectional plane.
← / →	Set / reset section.
↓	If desired: switch dialog for transfer of detail.
TRANSFER DETAIL = ENT	
ENT	Magnify detail.

Details can be magnified in any display mode. The abbreviation MAGN appears on the screen to indicate that the image is magnified.

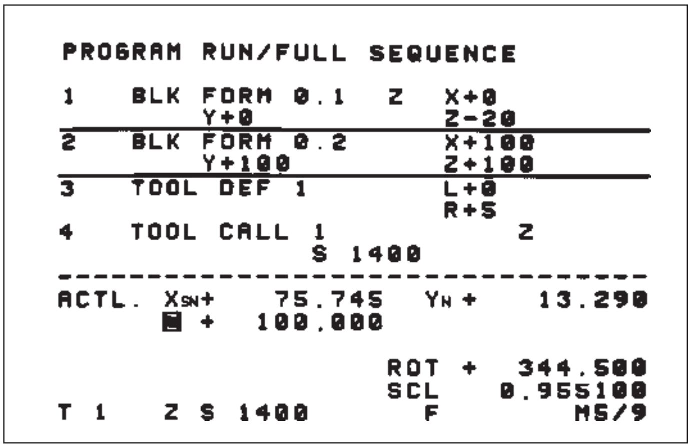
Return to non-magnified view

GRAPHICS BLK FORM	Press BLK FORM to display the workpiece in its programmed size.

Status Display

The status display in a program run operating mode shows the current coordinates as well as the following information:

- Type of position display (ACTL, NOML, ...)
- Axis locked (→|← in front of the axis)
- Number of current tool T
- Tool axis
- Spindle speed S
- Feed rate F
- Active miscellaneous function M
- TNC is in operation (indicated by ❊)
- Machines with gear ranges:
  Gear range following "/" character
  (depends on machine parameter)



```
PROGRAM RUN/FULL SEQUENCE

1    BLK FORM 0.1   Z   X+0
         Y+0            Z-20
2    BLK FORM 0.2       X+100
         Y+100          Z+100
3    TOOL DEF 1         L+0
                        R+5
4    TOOL CALL 1              Z
               S 1400
---------------------------------
ACTL.  XSN+    75.745   YN +    13.290
       ▣ +  100.000

                     ROT +   344.500
                     SCL    0.955100
T 1    Z S 1400        F        M5/9
```

Fig. 1.23: Status display in a program run operating mode

> Bar graphs can be used to indicate analog quantities such as spindle speed and feed rate. These bar graphs must be activated by the machine tool builder.

1.5 Programs

The TNC 360 can store up to 32 part programs at once. The programs can be written in HEIDENHAIN plain language dialog or according to ISO. ISO programs are indicated with "ISO".

Each program is identified by a number with up to eight characters.

Program directory

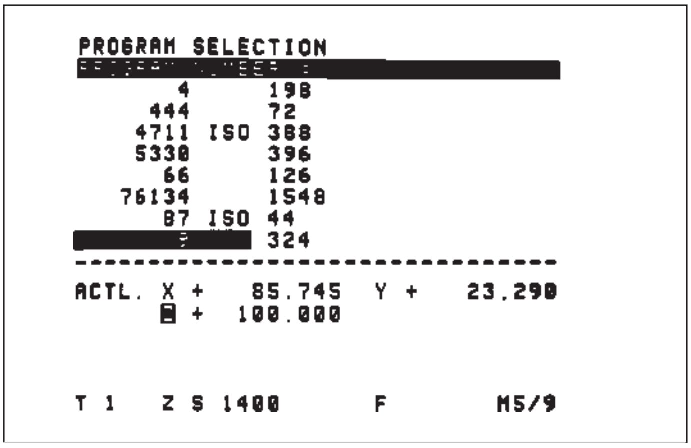
The program directory is called with the PGM NR key. To erase programs in TNC memory, press the CL PGM key.

Action	Mode of operation	Call program directory with ...
Create (a program)	⇨	... PGM NR
Edit	⇨	... PGM NR
Erase	⇨	... CL PGM
Test	⇨	... PGM NR
Execute	⇨ ⇨	... PGM NR

Fig. 1.24: Program management functions

The program directory provides the following information:

- Program number
- Program type (HEIDENHAIN or ISO)
- Program size in bytes, where one byte is the equivalent of one character.



```
PROGRAM SELECTION
PROGRAM NUMBER :
       4      198
     444      72
    4711  ISO 388
    5330      396
      66      126
   76134      1548
      87  ISO 44
       9      324
---------------------------------
ACTL.  X +   85.745   Y +   23.290
       ■ +  100.000


T 1   Z S 1400       F       M5/9
```

Fig. 1.25: Program directory on the TNC screen

Selecting, erasing and protecting programs

To select a program:

PGM NR	Call the program directory.

PROGRAM NUMBER ?

↑ or ↓	Use the arrow keys to highlight the program.
1 5	Enter the desired program number, for example 15.
ENT	Confirm your selection.

To erase a program:

CL PGM	Call the program directory.

ERASE = ENT / END = NO ENT

↑ or ↓	Use the arrow keys to highlight the program.
ENT or NO ENT	Erase the program or abort.

To protect a program:

PGM NR	Call the program directory.

PROGRAM NUMBER = ?

e.g. 5 ENT	Enter the number of the program to be protected.

0 BEGIN 5 MM

→ repeatedly	Press the key until the dialog prompt "PGM PROTECTION?" appears.

PGM PROTECTION ?

ENT	Protect the program.

The letter "P" for protected appears at the end of the first and last program blocks.

To remove edit protection:

Select the protected program, for example 5.

0 BEGIN 5 MM P	
MOD	Select MOD functions.

VACANT BYTES =	
↓ repeatedly	Activate the CODE NUMBER function.

CODE NUMBER	
8 6 3 5 7	Enter the code number 86357: Edit protection is removed, the "P" disappears.

2.1 Moving the Machine Axes

Traversing with the machine axis direction buttons:

✋ ▶ **MANUAL OPERATION**

e.g. X	Press the machine axis direction button and hold it for as long as you wish the axis to move.

You can move several axes at once in this way.

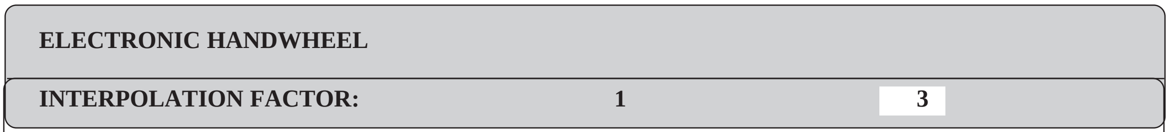
For continuing movement:

✋ ▶ **MANUAL OPERATION**

e.g. Y I together	Press and hold the machine axis direction button, then press the machine start button. The axis continues to move after you release the key.
○	To stop the axis, press the machine STOP button.

You can only move one axis at a time with this method.

Travesing with the electronic handwheel:



ELECTRONIC HANDWHEEL

INTERPOLATION FACTOR: 1 3

e.g. 3 ENT

Enter the desired interpolation factor (see table below).

e.g. X

Select the axis that you wish to move:
for portable handwheels, at the handwheel;
for integral handwheels, at the TNC keyboard.

Now move the selected axis with the electronic handwheel. If you are using the portable handwheel, first press the enabling switch on its back.

Interpolation factor	Traverse in mm per revolution
0	20.000
1	10.000
2	5.000
3	2.500
4	1.250
5	0.625
6	0.312
7	0.156
8	0.078
9	0.039
10	0.019

Fig. 2.1: Interpolation factors for handwheel speed

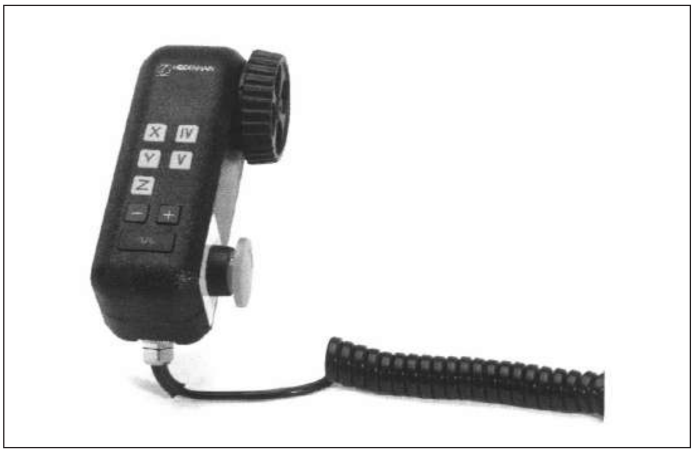


Fig. 2.2: HR 330 Electronic Handwheel

The smallest programable interpolation factor depends on the individual machine tool.
Positioning with the electronic handwheel can also be carried out in the operating mode PROGRAMMING (depending on MP7641).

Working with the HR 330 Electronic Handwheel

Attach the electronic handwheel to a steel surface with the mounting magnets such that it cannot be operated unintentionally.

Be sure not to press the axis direction keys unintentionally when you remove the handwheel from its position as long as the enabling switch (between the magnets) is depressed.

If you are using the handwheel for machine setup, press the enabling switch. Only then can you move the axes with the axis direction keys.

Incremental jog positioning

With incremental jog positioning, a machine axis will move by a prese increment each time you press the corresponding machine axis direction button.

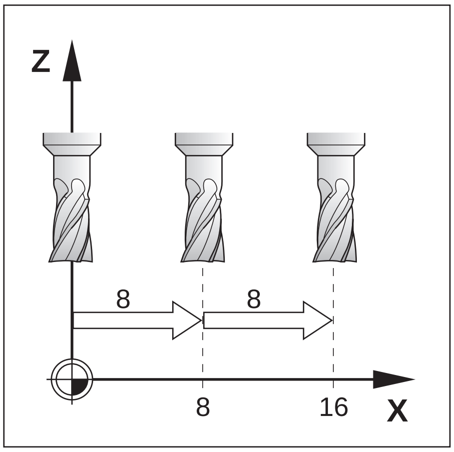
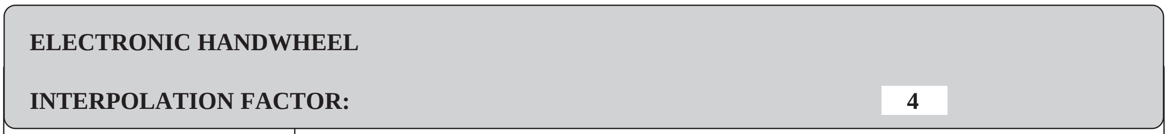


Fig. 2.3: Incremental jog positioning in the X axis

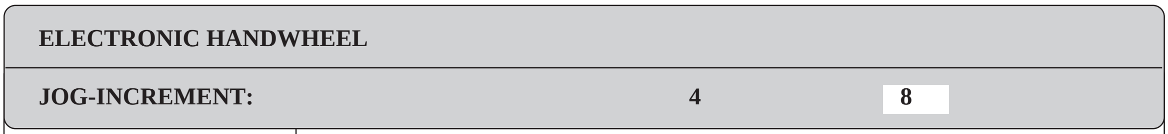
⊛ ▶


ELECTRONIC HANDWHEEL

INTERPOLATION FACTOR: 4

I	Select incremental jog positioning.
⊛	Select incremental jog positioning by pressing the handwheel mode key again.

▼


ELECTRONIC HANDWHEEL

JOG-INCREMENT: 4 8

e.g. 8 ENT e.g. X	Enter the jog increment (here 8 mm). Press the machine axis direction button as often as desired.

Incremental jog positioning must be enabled by the machine tool manufacturer.

Positioning with manual data input (MDI)

Page 5-40 describes positioning by manually entering the target coordinates for the tool.

2.2 Spindle Speed S, Feed Rate F and Miscellaneous Functions M

The following values can be entered and changed in the MANUAL OPERATION AND ELECTRONIC HANDWHEEL modes of operation:

- Miscellaneous function M
- Spindle speed S
- Feed rate F (can be changed but not entered)

For part programs these functions are entered or edited directly in the PROGRAMMING AND EDITING operating mode.

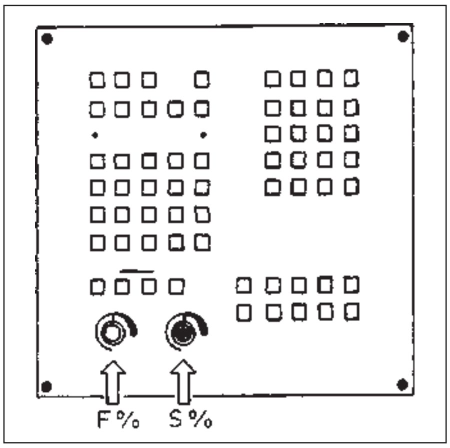


Fig. 2.4: Knobs for spindle speed and feed rate overrides

To enter the spindle speed S:

TOOL CALL	Initiate the dialog with the TOOL CALL key.
SPINDLE SPEED S RPM ?	
e.g. 1 0 0 0 ENT	Enter the spindle speed S, for example 1000 rpm.
I	Confirm the spindle speed S with the machine START button.

A miscellaneous function M starts spindle rotation at the entered speed S.

To enter the miscellaneous function M:

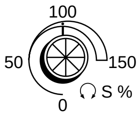
STOP	Select the STOP function.

MISCELLANEOUS FUNCTION M ?

e.g. 6 ENT	Enter the desired miscellaneous function M.
I	Activate the miscellaneous function M with the machine START key.

Chapter 12 provides an overview of the miscellaneous functions.

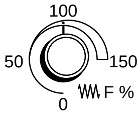
To change the spindle speed S:

	Turn the spindle speed override knob: Adjust the spindle speed S to between 0% and 150% of the last entered value.

The spindle speed override will function only if your machine tool is equipped with a stepless spindle drive.

To change the feed rate F:

In the MANUAL OPERATION mode the feed rate is set through a machine parameter.

	Turn the feed rate override knob: Adjust the feed rate to between 0% and 150% of the last entered value.

2.3 Setting the Datum Without a 3D Touch Probe

You fix a datum by setting the TNC position display to the coordinates of a known point on the workpiece. The fastest, easiest and most accurate way of setting the datum is by using a 3D touch probe system from HEIDENHAIN (see p. 2-14).

To prepare the TNC:

Clamp and align the workpiece.

Insert the zero tool with known radius into the spindle.

	Select the MANUAL OPERATION mode.

Ensure that the TNC is showing actual position values (see p. 11-4).

Setting the datum in the tool axis

Fragile workpiece?
If the workpiece surface must not be scratched, you can lay a metal shim of known thickness d on it. Then enter a tool axis datum value that is larger than the desired datum by the value d.

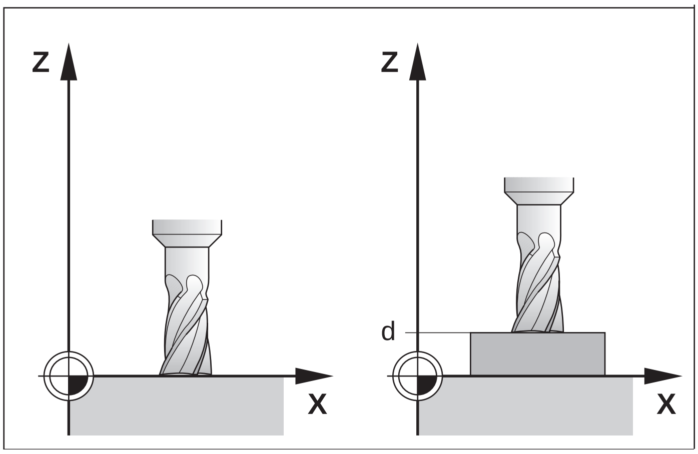


Fig. 2.5: Datum setting in the tool axis; right with protective shim

Move the tool until it touches with workpiece surface.

e.g. Z	Select the tool axis.

DATUM SET Z =

e.g. 0 ENT	For a zero tool: Set the display to Z = 0 or enter thickness d of the shim.
e.g. 5 0 ENT	For a preset tool: Set the display to the length L of the tool, for example Z=50 mm, or enter the sum Z=L+d.

To set the datum in the working plane:

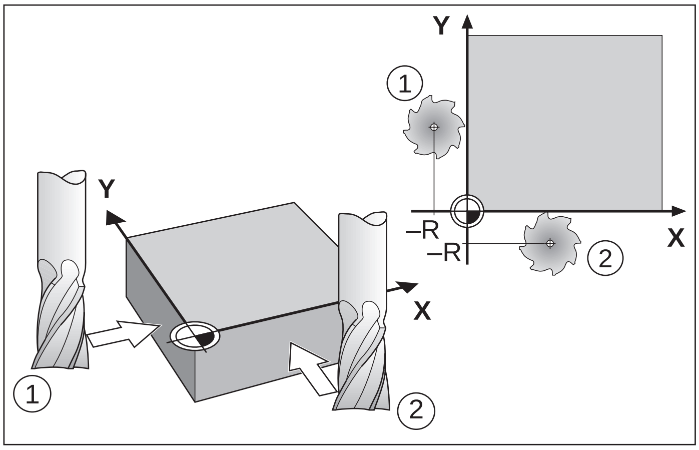


Fig. 2.6: Setting the datum in the working plane; plan view (upper right)

Move the zero tool until it touches the side of the workpiece.

e.g. X	Select the axis.
e.g. +/– 5 ENT	Enter the position of the tool center (here X = –5 mm) in the selected axis.

Repeat the process for all axes in the working plane.

2.4 3D Touch Probe Systems

3D Touch probe applications

The TNC provides touch functions for application of a HEIDENHAIN 3D touch probe. Typical applications for the touch probe systems are:

- Compensating workpiece misalignment (basic rotation)
- Datum setting
- Measuring:
  - Lengths and positions on the workpiece
  - Angles
  - Circle radii
  - Circle centers
- Measurements under program control
- Digitizing 3D surfaces (option)

Fig. 2.7: HEIDENHAIN TS 120 three-dimensional touch probe

> The TNC must be specially prepared by the machine tool builder for the use of a 3D touch probe.

After you press the machine START button, the touch probe begins executing the selected probe function. The machine manufacturer sets the feed rate at which the probe approaches the workpiece. When the 3D touch probe contracts the workpiece, it

- transmits a signal to the TNC, which stores the coordinates of the probed position
- stops moving
- returns to its starting position in rapid traverse

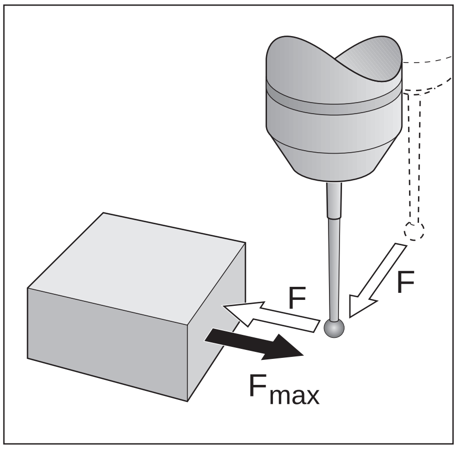


Fig. 2.8: Feed rates during probing

To select the touch probe menu:

MANUAL OPERATION

or

ELECTRONIC HANDWHEEL

TOUCH PROBE	Select the menu of touch probe functions.

CALIBRATION EFFECTIVE LENGTH
CALIBRATION EFFECTIVE RADIUS
BASIC ROTATION
SURFACE = DATUM
CORNER = DATUM
CIRCLE CENTER = DATUM

Calibrating the 3D Touch Probe

The touch probe system must be calibrated

- for commissioning
- after a stylus breaks
- when the stylus is changed
- when the probe feed rate is changed
- in case of irregularities, such as those resulting from machine heating.

During calibration, the TNC finds the "effective" length of the stylus and the "effective" radius of the ball tip. To calibrate the 3D touch probe, clamp a ring gauge with known height and known internal radius to the machine table.

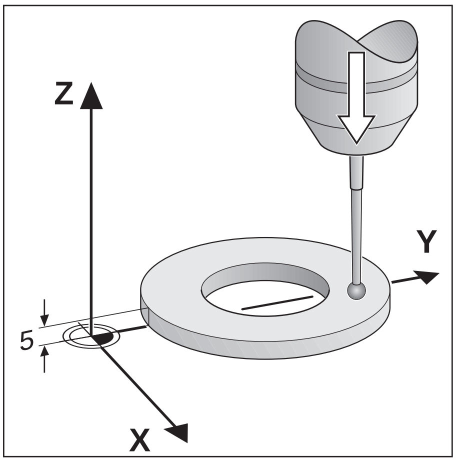


Fig. 2.9: Calibrating the touch probe length

To calibrate the effective length

Set the datum in the tool axis such that for the machine tool table, Z=0.

TOUCH PROBE ▶ **SURFACE = DATUM**

↓ / ↑ ENT	Select the calibration function for the touch probe length.

CALIBRATION EFFECTIVE LENGTH

Z+ Z–

TOOL AXIS = Z

e.g. Z	If necessary, enter the tool axis, for example Z.
↓	Move the highlight to DATUM.
e.g. 5	Enter the height of the ring gauge, for example 5 mm.

Move the touch probe to a position just above the ring gauge.

← or →	If necessary, change the displayed traverse direction.

I	The 3D touch probe contacts the upper surface of the ring gauge.

To calibrate the effective radius

Position the ball tip in the bore hole of the ring gauge.

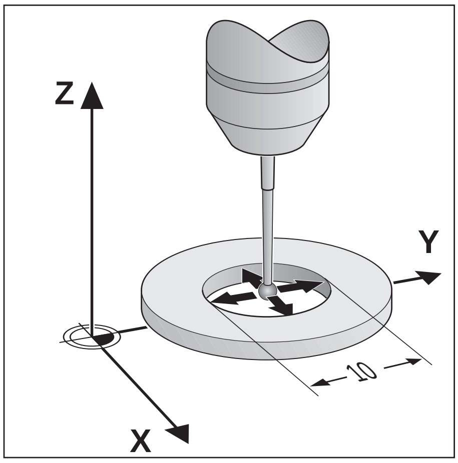


Fig. 2.10: Calibrating the touch probe radius

TOUCH PROBE ▶

SURFACE = DATUM

↓ / ↑ ENT	Select the calibration function for the ball-tip radius.

CALIBRATION EFFECTIVE RADIUS

X+ X– Y+ Y–

↓	Select RADIUS RING GAUGE.

RADIUS RING GAUGE = 0

5 ENT	Enter the radius of the ring gauge, here 5 mm.
4 x I	The 3D touch probe contacts one position on the bore for each axis direction.

Displaying calibration values

The effective length and radius of the 3D touch probe are stored in the TNC for use whenever the touch probe is needed again. The stored values are displayed the next time the calibration function is called.

Compensating workpiece misalignment

The TNC electronically compensates workpiece misalignment by computing a "basic rotation."
Set the ROTATION ANGLE to the angle at which a workpiece surface should be oriented with respect to the angle reference axis (see p. 1-12).

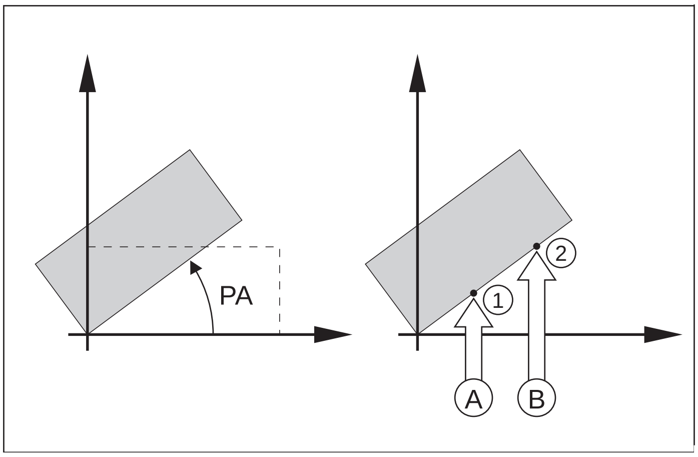


Fig. 2.11: Basic rotation of a workpiece, probing procedure for compensation (right). The dashed line is the nominal position; the angle PA is being compensated.

TOUCH PROBE ▶

SURFACE = DATUM

↓ / ↑ ENT	Select the BASIC ROTATION probe function.

BASIC ROTATION

X+ X- Y+ Y–

ROTATION ANGLE =

e.g. 0 ENT	Enter the nominal value of the rotation angle.

Move the ball tip to a starting position Ⓐ near the first touch point ①.

X + X – Y + Y –

← or →	Select the probe direction.
I	Probe the workpiece.

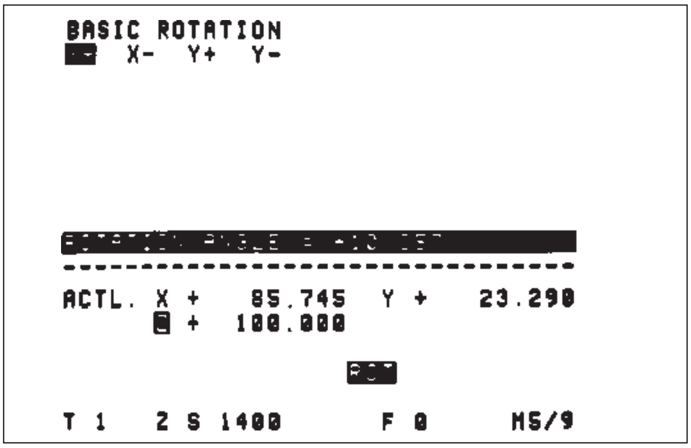
Move the ball tip to a starting position Ⓑ near the second touch point ②.

I	Probe the workpiece.

A basic rotation is kept in non-volatile storage and is effective for all subsequent program runs and graphic simulations.

Displaying basic rotation

The angle of the basic rotation is shown in the rotation angle display. When a basic rotation is active the abbreviation ROT is highlighted in the status display.



```
BASIC ROTATION
X+  X-  Y+  Y-




ROTATION ANGLE = [illegible]
------------------------------
ACTL. X +   85.745  Y +   23.290
      Z + 100.000

               ROT

T 1   Z S 1400      F 0      M5/9
```

Fig. 2.12: Displaying the angle of an active basic rotation

To cancel a basic rotation:

Select BASIC ROTATION again.

ROTATION ANGLE =

0 ENT	Set the rotation angle to 0.
END □	Terminate the probing function.

2.5 Setting the Datum with the 3D Touch Probe System

The following functions are listed for datum setting in the TCH PROBE menu:

- Datum setting in any axis with SURFACE = DATUM
- Setting a corner as datum with CORNER = DATUM
- Setting the datum at a circle center with CIRCLE CENTER = DATUM

To set the datum in a specific axis:

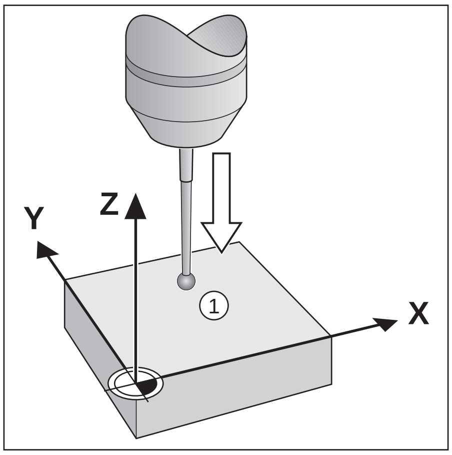


Fig. 2.13: Probing for the datum in the Z axis

Select the probe function SURFACE = DATUM.

Move the touch probe to a position near the touch point.

SURFACE = DATUM

X + X – Y + Y – Z + Z –

← or →	Select the probe direction and axis in which you wish to set the datum, for example Z in the Z– direction.
I	Probe the workpiece.
e.g. 0 ENT	Enter the nominal coordinate of the datum.

Corner as datum

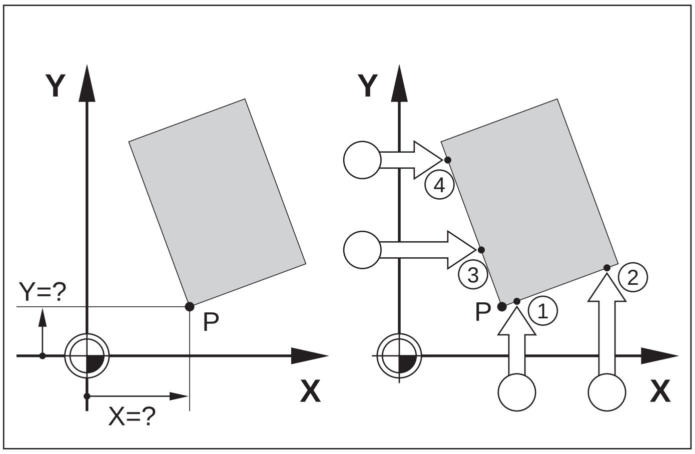


Fig. 2.14: Probing procedure for finding the coordinates of the corner P

Select the CORNER = DATUM probe function.

To use the points that just probed for a basic rotation:

TOUCH POINTS OF BASIC ROTATION?

ENT	Transfer the touch point coordinates to memory.

Move the touch probe to a starting position near the first touch point on the side that was not probed for basic rotation.

CORNER = DATUM

X +   X –   Y +   Y –

← or →	Select the probing direction.
I	Probe the workpiece.

Move the touch probe to a starting position near the second touch point on the same side.

I	Probe the workpiece.

DATUM X =

e.g. 0 ENT	Enter the first coordinate of the datum point, here for the X axis.

⋮

.
.
.

↓	Select the second coordinate.

DATUM Y =

e.g. 0 ENT	Enter the second coordinate of the datum, here in the Y axis.
END □	Terminate the probe function.

If you do not wish to use points that just probed for a basic rotation:

TOUCH POINTS OF BASIC ROTATION?

NO ENT	Ignore the dialog prompt.

Probe both workpiece sides twice.

Enter the datum coordinates.

Circle center as datum

With this function you can set the datum at the center of bore holes, circular pockets, cylinders, journals, circular islands etc.

Inside circle

The TNC automatically probes the inside wall in all four coordinate axis directions.

For incomplete circles (circular arcs) you can choose the appropriate probing directions.

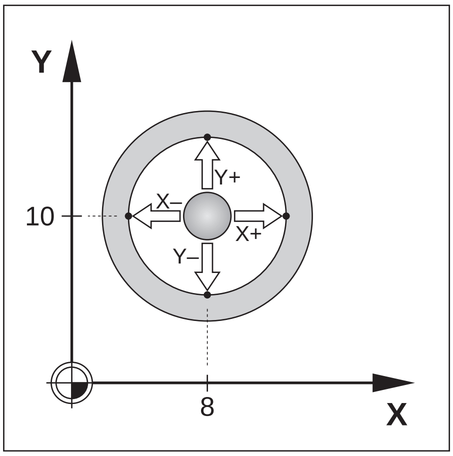


Fig. 2.15: Probing an inside cylindrical surface to find the center

Select the CIRCLE CENTER = DATUM function.

Move the touch probe to a position approximately in the center of the circle.

CIRCLE CENTER = DATUM

X + **X –** **Y +** **Y –**

4 x I	The probe touches four points on the inside of the circle.

DATUM X =

e.g. 8 ENT	Enter the first coordinate of the datum, here in the X axis.
↓	Select the second coordinate.

DATUM Y =

e.g. 1 0 ENT	Enter the second coordinate of the datum, here in the Y axis.
END □	Terminate the probe function.

Outside circle

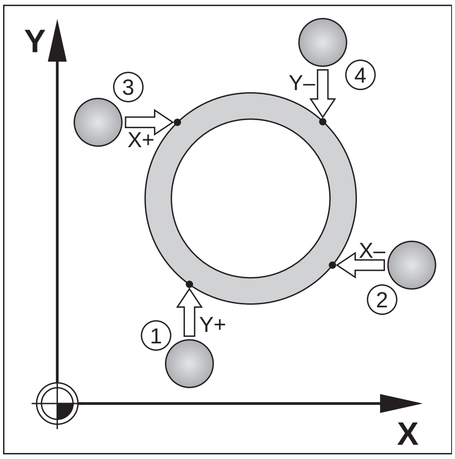


Fig. 2.16: Probing an outside cylindrical surface to find the center

Select the CIRCLE CENTER = DATUM probe function.

Move the touch probe to a starting position ① near the first touch point outside of the circle.

CIRCLE CENTER = DATUM

X + **X –** **Y +** **Y –**

← or → Select the probing direction.

I Probe the workpiece.

Repeat the probing process for points ②, ③ and ④ (see Fig. 2.16).

Enter the coordinates of the circle center.

After the probing procedure is completed, the TNC displays the coordinates of the circle center and the circle radius PR.

2.6 Measuring with the 3D Touch Probe System

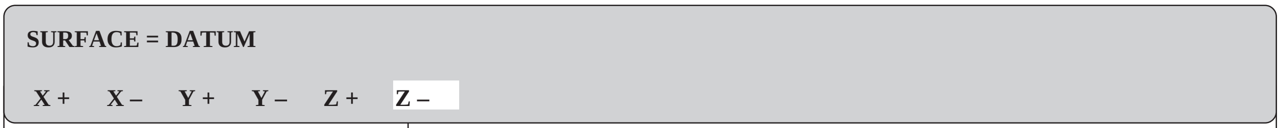
With the 3D touch probe system you can determine

- Position coordinates, and from them,
- dimensions and angles on the workpiece.

Finding the coordinate of a position on an aligned workpiece

Select the SURFACE = DATUM probe function.

Move the probe to a starting position near the touch point.



SURFACE = DATUM

X + X – Y + Y – Z + Z –

← or →	Select the direction and axis in which you wish to find the coordinate.
I	Probe the workpiece.

The TNC displays the coordinates of the touch point as DATUM.

Finding the coordinates of a corner in the working plane

Find the coordinates of the corner point as described under "Corner as datum." The TNC displays the coordinates of the probed corner as DATUM.

Measuring workpiece dimensions

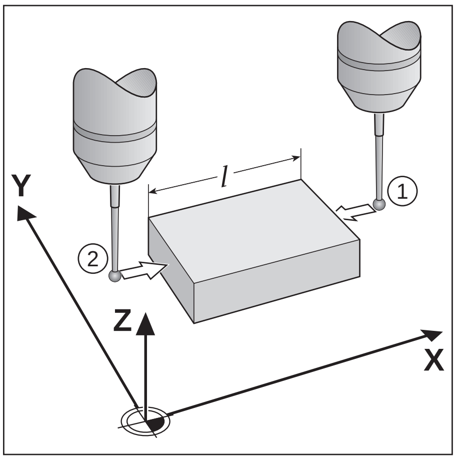


Fig. 2.17: Measuring lengths with the 3D touch probe

Select the SURFACE = DATUM probe function.

Move the probe to a starting position ① near the first touch point.

SURFACE = DATUM

X + X – Y + Y – Z + Z –

← or →	Select the probing axis and direction.
I	Probe the workpiece.

If you will need the current datum later, write down the value that appears in the DATUM display.

DATUM X =

0 ENT	Set the DATUM to 0.
END □	Terminate the dialog.

Re-select the SURFACE = DATUM probe function.

Move the touch probe to a starting position ② near the second touch point.

.
.
.

.
.
.

SURFACE = DATUM

X + X – Y + Y – Z + Z –

← or →	Select the axis direction with the cursor keys — same axis as for ①.
I	Probe the workpiece.

The value displayed as DATUM is the distance between the two points.

To return to the datum that was active before the length measurement:

Select the SURFACE = DATUM probe function.

Probe the first touch point again.

Set the datum to the value that you wrote down previously.

END □	Terminate the dialog.

Measuring angles

You can also use the 3D touch probe system to measure angles in the working plane. You can measure

- the angle between the angle reference axis and a workpiece side, or
- the angle between two sides.

The measured angle is displayed as a value of maximum 90°.

To find the angle between the angle reference axis and a side of the workpiece:

Select the BASIC ROTATION probe function.

ROTATION ANGLE =

If you will need the current basic rotation later, write down the value that appears under ROTATION ANGLE.

Make a basic rotation with the side of the workpiece (see Section “Compensating workpiece misalignment”).

.
.
.

.
.
.

The angle between the angle reference axis and the side of the workpiece appears as the ROTATION ANGLE in the BASIC ROTATION function.

Cancel the basic rotation.

Restore the previous basic rotation by setting the ROTATION ANGLE to the value that you wrote down previously.

To measure the angle between two sides of a workpiece:

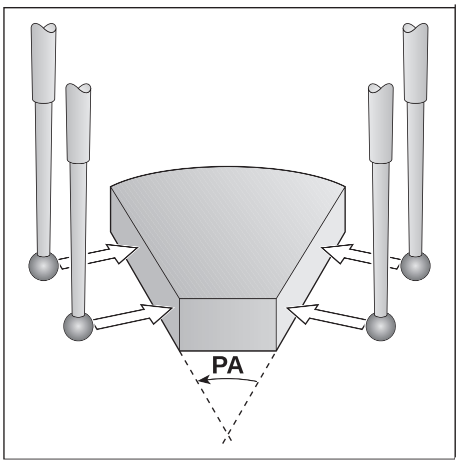


Fig. 2.18: Measuring the angle between two sides of a workpiece

Select the BASIC ROTATION probe function.

ROTATION ANGLE =

If you will need the current basic rotation later, write down the value that appears under ROTATION ANGLE.

Make a basic rotation for the first side (see “Compensating workpiece misalignment“).

Probe the second side as for a basic rotation, but do not set the ROTATION ANGLE to zero!

The angle PA between the workpiece sides appears as the ROTATION ANGLE in the BASIC ROTATION function.

Cancel the basic rotation.

Restore the previous basic rotation by setting the ROTATION ANGLE to the value that you wrote down previously.

3.1 Test Run

In the TEST RUN mode of operation the TNC checks programs and program sections for the following errors without moving the machine axes:

- Geometrical incompatibility
- Missing data
- Impossible jumps

The following TNC functions can be used in the TEST RUN operating mode:

- Test interruption at any block
- Optional block skip

To do a test run:

▶ **TEST RUN**

TO BLOCK NUMBER =

NO ENT	Test the entire program.
e.g. 1 0 ENT	Test the program up to the entered block.

Test run functions

Function	Key
• Interrupt the test run	STOP
• Continue test run after interruption	NO ENT

3.2 Program Run

In the PROGRAM RUN / FULL SEQUENCE mode of operation the TNC executes a part program continuously to its end or up to a program stop.

In the PROGRAM RUN /SINGLE BLOCK mode of operation you execute each block separately by pressing the machine START button.

The following TNC functions can be used during a program run:

- Interrupt program run
- Start program run from a certain block
- Blockwise transfer of very long programs from external storage
- Checking/changing Q parameters
- Graphic simulation of a program run

To run a part program:

- Clamp the workpiece to the machine table.
- Set the datum
- Select the program.

PROGRAM RUN / SINGLE BLOCK

or

PROGRAM RUN / FULL SEQUENCE

Select the part program.

GOTO □ 0 ENT	Go to the first block of the program.
I	Run the part program.
Only in mode PROGRAM RUN / SINGLE BLOCK I repeatedly	Run each block of the part program separately.

The feed rate and spindle speed can be changed with the override knobs.

Interrupting machining

There are various ways to interrupt a program run:

- Programmed interruptions
- External STOP key
- Switching to PROGRAM RUN / SINGLE BLOCK
- EMERGENCY STOP button

If the TNC registers an error during program run, it automatically interrupts machining.

Programmed interruptions

Interruptions can be programmed directly in the part program. The part program is interrupted at a block containing one of the following entries:

- STOP
- Miscellaneous functions M0, M02 or M30
- Miscellaneous function M06, if the machine tool builder has assigned it a stop function

To interrupt or abort machining immediately:

The block which the TNC is currently executing is not completed.

◎	Interrupt machining.

The ❊ sign in the status display blinks.

The part program can be aborted with the STOP key.

STOP	Abort program run.

The ❊ sign disappears from the status display.

To interrupt machining at the end of the current block:

You can interrupt the program run at the end of the current block by switching to the PROGRAM RUN / SINGLE BLOCK.

⊡	Select PROGRAM RUN / SINGLE BLOCK.

Resuming program run after an interruption

When a program run is interrupted the TNC stores:

- The data of the last called tool
- Active coordinate transformations
- The coordinates of the last defined circle center
- The count of a running program section repetition
- The number of the last CALL LBL block

Resuming program run with the START button

You can resume program run by pressing the START button if the program was interrupted in one of the following ways:

- Pressing the machine STOP button
- A programmed interruption
- Pressing the EMERGENCY STOP button (machine-dependent function).

Resuming program run after an error

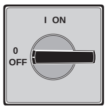
- If the error message is not blinking:

Remove the cause of the error.

CE — Clear the error message from the screen.

Restart the program.

- If the error message is blinking:



Switch off the TNC and the machine.

Remove the cause of the error.

Restart the program.

- If you cannot correct the error:

Write down the error message and contact your repair service agency.

3.3 Blockwise Transfer: Executing Long Programs

Part programs that occupy more memory than the TNC provides can be “drip fed” block by block from an external storage device.

During program run, the TNC transfers program blocks from a floppy disk unit or PC through its data interface, and erases them after execution. This frees memory space for new blocks.

To prepare for blockwise transfer:

- Prepare the data interface.
- Configure the data interface with the MOD function (see page 11-3).
- If you wish to transfer a part program from a PC, adapt the TNC and PC to each other (see pages 10-4 and 12-2).
- Ensure that the transferred program meets the following requirements:
  - The highest block number must not exceed 65534. However, the block numbers can repeat themselves as often as necessary.
  - All programs called from the transferred program must be present in TNC memory
  - The transferred program must not contain:
    Subprograms
    Program section repetitions
    Digitizing cycles (TOUCH PROBE 5.0 to 7.0)
    The function FN 15:PRINT
  - The TNC can store up to 20 TOOL DEF blocks.

PROGRAM RUN / SINGLE BLOCK

or

TEST RUN

EXT	Select the function for blockwise transfer.

PROGRAM NUMBER

e.g. 1 0 ENT	Enter the program number and start data transfer.
I	Execute the program blocks.

If the data transfer is interrupted, press the START key again.

Jumping over blocks

The TNC can jump to any desired block in the program to begin transfer.
The preceding blocks are ignored during a program run.

Select the program and start transfer.

GOTO □ e.g. 1 5 0 ENT	Go to the block number at which you wish to begin data transfer, for example 150.
I	Execute the transferred blocks, starting with the block number that you entered.

In the PROGRAMMING AND EDITING mode of operation (see page 1-19) you can

- create,
- add to,
- edit, and
- erase files.

This chapter describes basic functions and programming input that do not cause machine axis movement. The entry of geometry for workpiece machining is described in the next chapter.

4.1 Editing part programs

Layout of a program

A part program consists of individual program blocks. The TNC numbers the blocks in ascending order. Program blocks contain units of information called “words.”

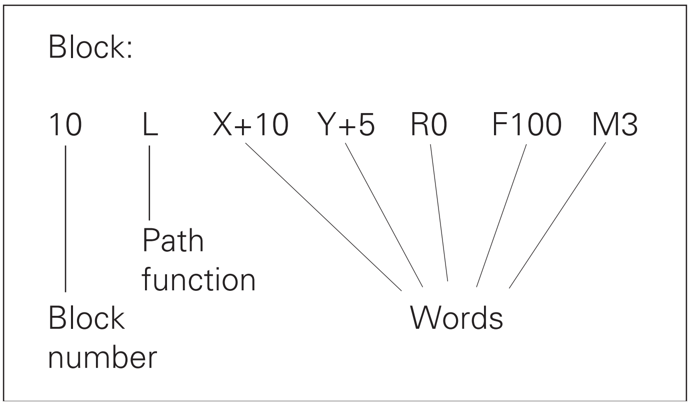


Fig. 4 1: Program blocks contain words of specific information

Plain language dialog

You initiate a dialog for conversational programming by pressing a function key (see inside front cover). The TNC then asks you for all the information necessary to program the desired function. After you have answered all the questions, the TNC automatically ends the dialog.

You can shorten the dialog by skipping over words that need not be programmed or ending the block immediately after entering the necessary information.

Function	Key
• Continue the dialog	ENT
• Ignore the dialog question	NO ENT
• End the dialog immediately	END □
• Abort the dialog and erase the block	DEL □

Editing functions

Editing means entering, adding to or changing commands for the TNC.

The TNC enables you to

- Enter data with the keyboard
- Select desired blocks and words
- Insert and erase blocks and words
- Correct erroneously entered values and commands
- Easily clear TNC messages from the screen

Types of input

Numbers, coordinate axes and radius compensation are entered directly by keyboard. You can set the algebraic sign either before, during or after a numerical entry.

Selecting blocks and words

- To call a block with a certain block number:

GOTO □ e.g. 1 0 ENT	The entered block is shown between two horizontal lines.

- To move one block forward or backward:

↑ or ↓	Press the vertical cursor keys.

- To select individual words in a block:

← or →	Press the horizontal cursor keys.

- To find the same word in other blocks:

← or →	Select the word in the block.
↑ or ↓	Jump to the same word in other blocks.

Inserting blocks

Additional program blocks can be inserted behind any existing block (except the PGM END block).

↑ or ↓ / GOTO □	Select the block in front of the desired insertion.

Program the new block.

The block numbers of all subsequent blocks automatically increase by one.

Editing and inserting words

Highlighted words can be changed as desired: simply overwrite the old value with the new one. Plain language dialog indicates the type of information required. After entering the new information, press a horizontal cursor key or the END key to confirm the change.

In addition to changing the existing words in a block, you can also add new words with the aid of the plain language dialog.

Erasing blocks and words

Function	Key
• Set the selected number to 0	CE
• Erase an incorrect number	CE
• Clear a non-blinking error message	CE
• Delete the selected word	NO ENT
• Delete the selected block	DEL □
• Erase cycles and program sections: First select the last block of the cycle or program section to be erased.	DEL □

4.2 Tools

Each tool is identified by a number.
The tool data, consisting of the:

- length L, and
- radius R

are assigned to the tool number.

The tool data can be entered:

- into the individual part program in a TOOL DEF block, or
- once for each tool into a common tool table that is stored as program 0.

Once a tool is defined, the TNC then associates its dimensions with the tool number and accounts for them when executing positioning blocks.

Determining tool data

Tool number

Each tool is designated with a number between 0 and 254.

The tool with the number 0 is defined as having length L = 0 and radius R = 0. In tool tables, T0 should also be defined with L = 0 and R = 0.

Tool radius R

The radius of the tool is entered directly.

Tool length L

The compensation value for the tool length is measured

- as the difference in length between the tool and a zero tool, or
- with a tool pre-setter.

A tool pre-setter eliminates the need to define a tool in terms of the difference between its length and that of another tool.

Determining tool length with a zero tool

For the sign of the tool length L:

$L > L_0$ A positive value means the tool is longer than the zero tool.

$L < L_0$ A negative value means the tool is shorter than the zero tool.

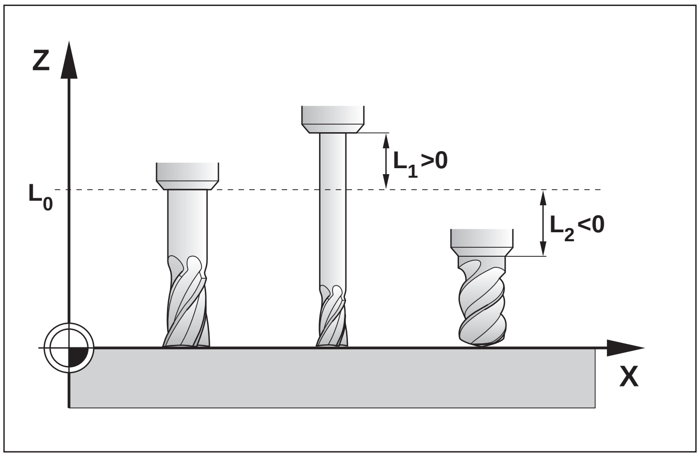


Fig. 4.2: Tool lengths can be given as the difference from the zero tool

Move the zero tool to the reference position in the tool axis (e.g. workpiece surface with Z = 0).

If necessary, set the datum in the tool axis to 0.

Change tools.

Move the new tool to the same reference position as the zero tool.

The TNC displays the compensation value for the length L.

Write the value down and enter it later.

Enter the display value by using the “actual position capture” function (see page 4-19).

Entering tool data into the program

The following data can be entered for each tool in the part program:

- Tool number
- Tool length compensation value L
- Tool radius R

To enter tool data in the program block:

TOOL DEF ▶

TOOL NUMBER?

e.g. 5 ENT	Designate the tool with a number, for example 5.

TOOL LENGTH L?

e.g. 1 0 ENT	Enter the compensation value for the tool length, for example L = 10 mm.

TOOL RADIUS R?

e.g. 5 ENT	Enter the tool radius, e.g. R = 5 mm.

Resulting NC block: TOOL DEF 5 L+10 R+5

You can enter the tool length L directly in the tool definition by using the "actual position capture" function (see page 4-19).

Entering tool data in program 0

The data for all tools can be entered in a common tool table. The number of tools in the table is selected through the machine parameter MP 7260.

If your machine uses an automatic tool changer, the tool data must be stored in the tool table.

Editing the tool table (program 0)

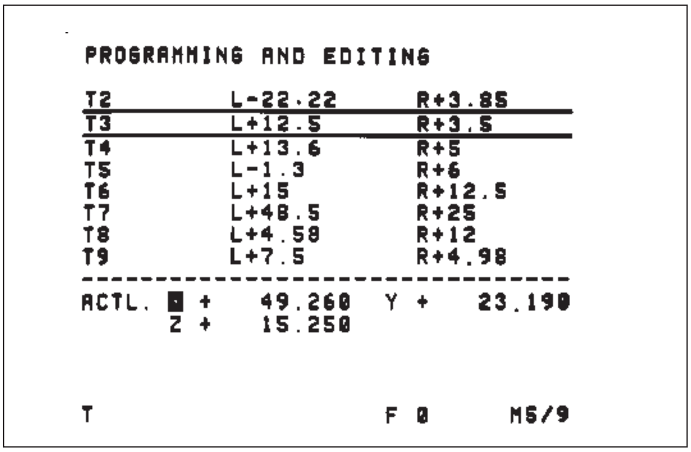
PROGRAMMING AND EDITING	
PGM NR	Call the program directory.

PROGRAM NUMBER =	
0 ENT	Select the tool table.

In the ELECTRONIC HANDWHEEL and MANUAL modes of operation, you can call the tool table at any time by simply pressing ENT.

Data in the tool table

The tool table contains further information in addition to the tool dimensions.



```
PROGRAMMING AND EDITING

T2          L-22.22       R+3.85
T3          L+12.5        R+3.5
T4          L+13.6        R+5
T5          L-1.3         R+6
T6          L+15          R+12.5
T7          L+48.5        R+25
T8          L+4.58        R+12
T9          L+7.5         R+4.98
----------------------------------
ACTL. ■ +    49.260   Y +   23.190
      Z +    15.250


T                     F 0     M5/9
```

Fig. 4.3: Tool table

Abbreviation	Input	Dialog
T	**T**ool number: the number with which the tool is called from the part program	–
S	**S**pecial tool with large radius requiring more than one pocket in the tool magazine. A certain number of pockets is kept vacant on each side of the special tool. The letter S then appears in front of the tool number.	SPECIAL TOOL? YES = ENT / NO = NO ENT
P	**P**ocket number of the tool in the magazine	POCKET NUMBER?
L	Compensation value for the **L**ength of the tool	TOOL LENGTH L?
R	**R**adius of the tool	TOOL RADIUS R?

Calling tool data

The following data can be programmed in the TOOL CALL block:

- Tool number
- Spindle axis
- Spindle speed in rpm

To call the tool data:

TOOL CALL ▶

TOOL NUMBER?

e.g. 5 ENT	Enter the number of the tool as it was defined in a tool table or in a “TOOL DEF” block, for example 5.

WORKING SPINDLE AXIS X/Y/Z?

e.g. Z	Enter the spindle axis, for example Z.

SPINDLE SPEED S IN RPM?

e.g. 5 0 0 ENT	Enter the desired spindle speed, such as S = 500 rpm.

Resulting NC block: TOOL CALL 5 Z S500

Tool pre-selection with tool tables

If you are using tool tables, you can indicate which tool you will next need by entering a TOOL DEF block. Simply enter the tool number.

Tool change

The TNC can work with either automatic or manual tool change.

Automatic tool change

If your machine is built for automatic tool changing, the TNC controls the replacement of the inserted tool by another from the tool magazine. The program run is not interrupted.

Manual tool change

To change the tool manually, stop the spindle and move the tool to the tool change position. Sequence of action:

- Interrupt program run (see page 3-4)
- Move to the tool change position (under program control, if desired)
- Change the tool
- Continue the program run (see page 3-5)

Tool change position

A tool change position must lie next to or above the workpiece to prevent tool collision. With the miscellaneous functions M91 and M92 (see page 5-39) you can enter machine-referenced rather than workpiece-referenced coordinates for the tool change position.

If TOOL CALL 0 is programmed before the first tool call, the TNC moves the spindle to an uncompensated position.

4.3 Tool Compensation Values

For each tool, the TNC adjusts the spindle path in the tool axis by the compensation value for the tool length. In the working plane it compensates the tool radius.

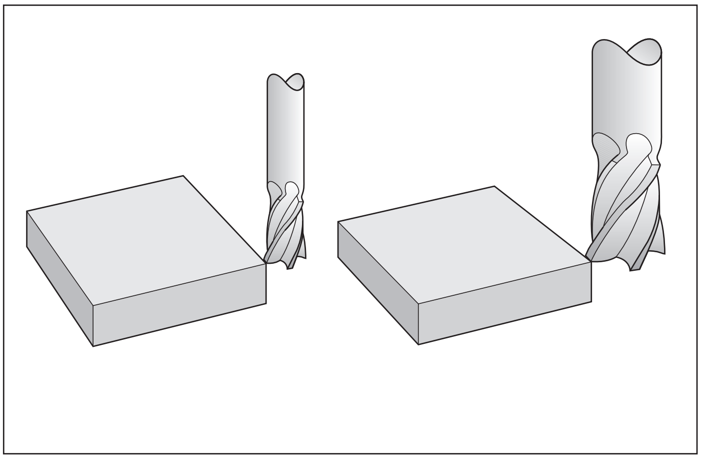


Fig. 4.4 : The TNC must compensate the length and radius of the tool

Effect of tool compensation values

Tool length

Length compensation becomes effective automatically as soon as a tool is called and the tool axis moves.

To cancel length compensation, call a tool with the length L = 0.

If a positive length compensation was in effect before TOOL CALL 0, the clearance to the workpiece is reduced.
If the tool axis is moved immediately after a TOOL CALL, the difference in length between the old and new tools is added to the programmed value.

Tool radius

Radius compensation becomes effective as soon as a tool is called and is moved in the working plane with RL or RR.

To cancel radius compensation, program a positioning block with R0.

Tool radius compensation

Tool traverse can be programmed:

- Without radius compensation: R0
- With radius compensation: RL or RR
- As single-axis movements with R+ or R–

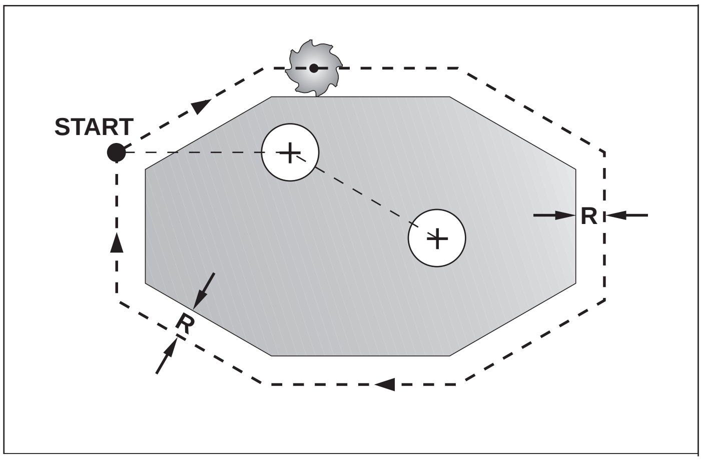


Fig. 4.5: Programmed contour (–, +) and the path of the tool center (- - -)

Traverse without radius compensation: R0

The tool center moves to the programmed coordinates.

Applications:

- Drilling and boring
- Pre-positioning

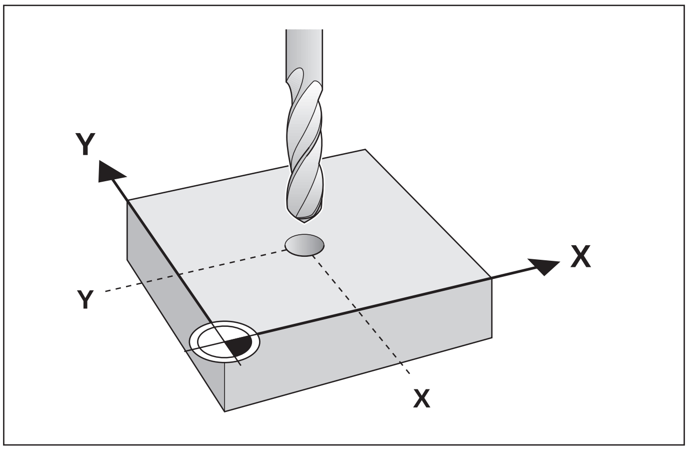


Fig. 4.6: These drilling positions are entered without radius compensation

To position without radius compensation:

TOOL RADIUS COMP.: RL/RR/NO COMP.?	
ENT	Select tool movement without radius compensation.

.
.
.

Traverse with radius compensation RR, RL

The tool center moves to the left (RL) or to the right (RR) of the programmed contour at a distance equal to the tool radius. "Right" or "left" is meant as seen in the direction of tool movement as if the workpiece were stationary.

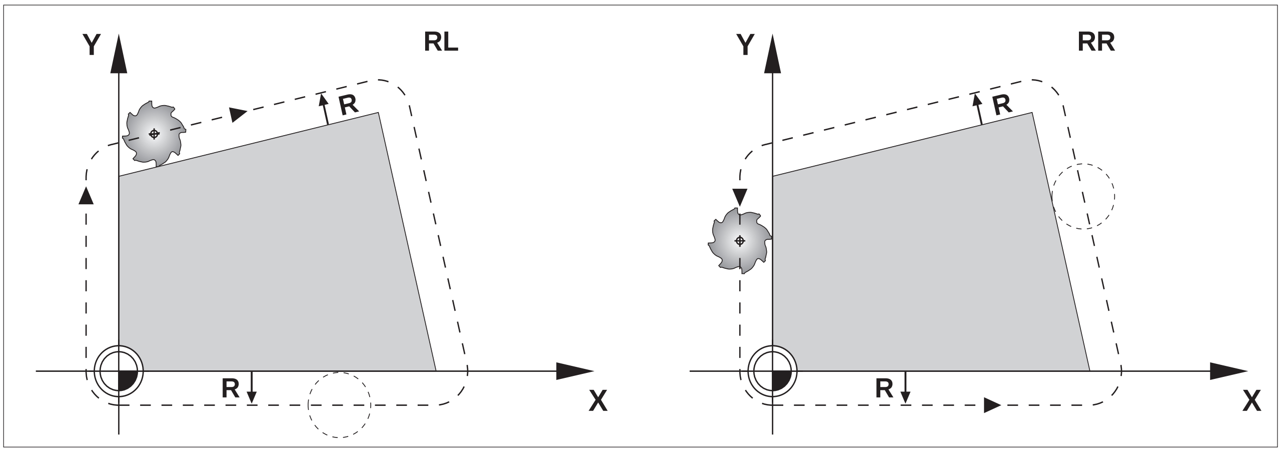


Fig. 4.7: The tool moves to the left (RL) or to the right (RR) of the workpiece during milling

To position with radius compensation:

.
.
.

TOOL RADIUS COMP.: RL/RR/NO COMP.?	
R^{L}_{-}	Select tool movement to the **left** of the programmed contour.
R^{R}_{+}	Select tool movement to the **right** of the programmed contour.

Radius compensation RR/RL is not in effect until the end of the block in which it is first programmed.

Between two program blocks with differing radius compensation you must program at least one block without radius compensation (that is, with R0).

Shortening or lengthening single-axis movements R+, R–

This type of radius compensation is possible only for single-axis movements in the working plane: The programmed tool path is shortened (R-) or lengthened (R+) by the tool radius.

Applications:

- Single-axis machining
- Occasionally for pre-positioning the tool, such as for the SLOT MILLING cycle.

R+ and R– are activated by opening a positioning block with an orange axis key.

Machining corners

Outside corners

The TNC moves the tool in a transitional arc around outside corners. The tool “rolls around” the corner point.

If necessary, the feed rate F is automatically reduced at outside corners to reduce machine strain, for example for very sharp changes in direction.

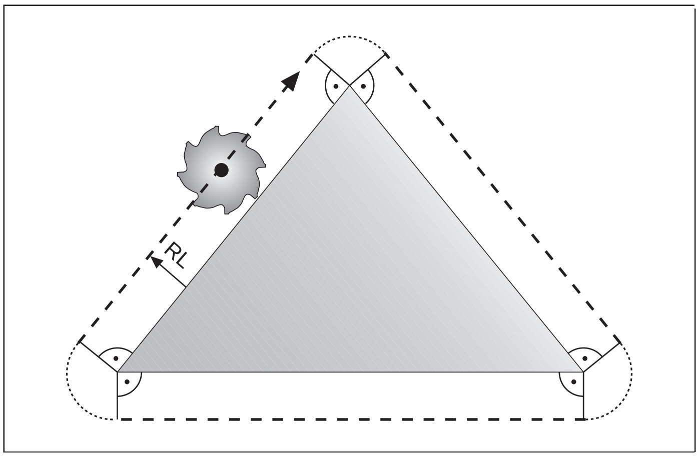


Fig. 4.8: The tool “rolls around” outside corners

If you work without radius compensation, you can influence the machining of outside corners with M90 (see page 5-36).

Inside corners

The TNC calculates the intersection of the tool center paths at inside corners. From this point it then starts the next contour element. This prevents damage to the workpiece at inside corners.

When two or more inside corners adjoin, the chosen tool radius must be small enough to fit in the programmed contour.

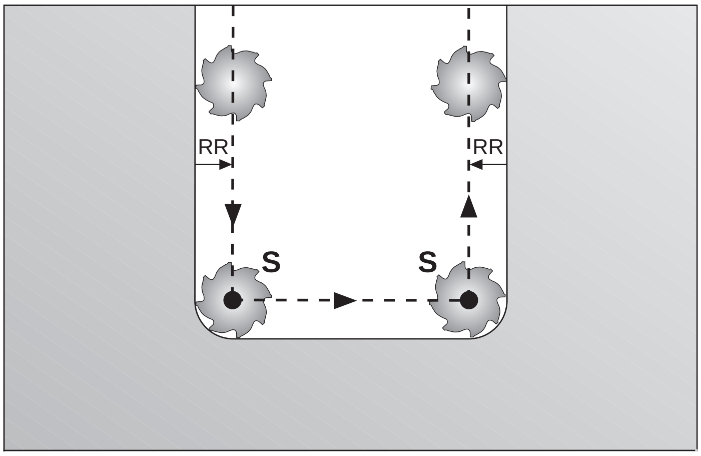


Fig. 4.9: Tool path for inside corners

4.4 Program Creation

To create a new part program:

PGM NR	Call the file directory.

Select any program.

PROGRAM NUMBER =

e.g. 7 4 3 2 ENT	Enter the number of the new program, for example 7432.

MM = ENT / INCH = NO ENT

ENT or NO ENT	Indicate whether the dimensions will be entered in millimeter or in inches.

Two program blocks then appear in the TNC screen.

0 BEGIN PGM 7432 MM

Block 0: Program beginning, name, unit of measure.

1 END PGM 7432 MM

Block 1: Program end, name, unit of measure.

The TNC generates the block numbers and the BEGIN and END blocks automatically. The unit of measure used in the program appears behind the program name.

Defining the blank form – BLK FORM

If you wish to use the TNC's graphic workpiece simulation you must first define a rectangular workpiece blank. Its sides lie parallel to the X, Y and Z axes and can be up to 30 000 mm long.

To start the dialog for blank form definition, press the BLK FORM key.

MIN and MAX points

The blank form is defined by two of its corner points:

- The MIN point — the smallest X, Y and Z coordinates of the blank form, entered as absolute values.
- The MAX point — the largest X, Y and Z coordinates of the blank form, entered as absolute or incremental values.

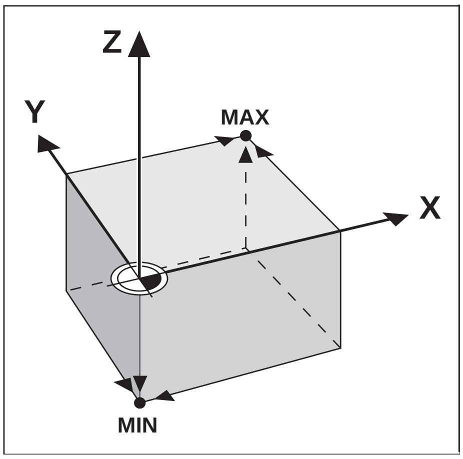


Fig. 4.10: The MIN and MAX points define the blank form

The ratio of the blank-form side lengths must be less than 84:1.

4.5 Entering Tool-Related Data

Besides the tool data and compensation, you must also enter the following information:

- Feed rate F
- Spindle speed S
- Miscellaneous functions M

The tool-related data can be determined with the aid of diagrams (see page 12-15).

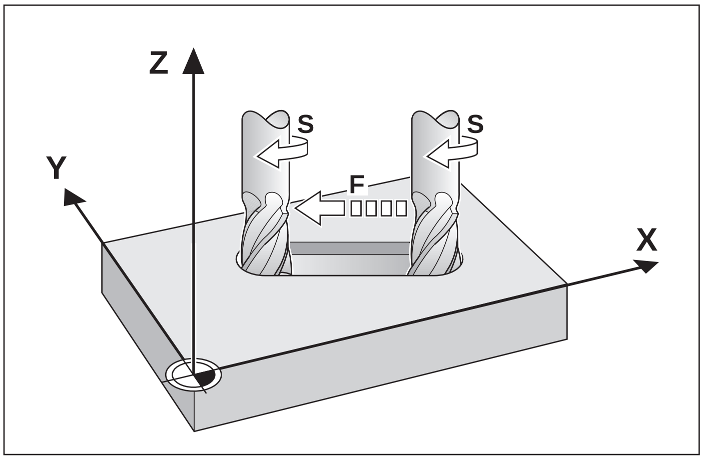


Fig. 4.11 : Feed rate F and spindle speed S of the tool

Feed Rate F

The feed rate is the speed in (mm/min or inch/min) with which the tool center moves.

Input range:
F = 0 to 29 999 mm/min (1181 inch/min)

The maximum feed rate is set in machine parameters individually for each axis.

To set the feed rate:

Answer the following dialog question in the positioning block:

FEED RATE F = ? / F MAX = ENT	
e.g. 1 0 0 ENT	Enter the feed rate F, for example F = 100 mm/min.

The question does not always appear with F MAX.

Rapid traverse

If you wish to program rapid traverse, press ENT for FMAX. If you know the maximum traverse speed, you can also program it directly. FMAX is effective only for the block in which it is programmed.

Duration of feed rate F

A feed rate that is entered as a numerical value remains in effect until the control executes a block in which another feed rate has been programmed.

If the new feed rate is FMAX, after that block is executed the feed rate returns to the last numerically entered feed rate.

Changing the feed rate F

You can vary the feed rate by turning the knob for feed rate override on the TNC keyboard (see page 2-5).

Spindle speed S

Enter the spindle speed S in revolutions per minute (rpm) in the TOOL CALL block.

Input range:
S = 0 to 99 999 rpm

To change the spindle speed S in the part program:

TOOL CALL	Press the TOOL CALL key.

TOOL NUMBER?

NO ENT	Ignore the prompt for the tool number.

WORKING SPINDLE AXIS X / Y / Z?

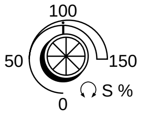
NO ENT	Ignore the prompt for the tool axis.

SPINDLE SPEED S?

e.g. 1 0 0 0 ENT	Enter the spindle speed S, for example 1000 rpm.

Resulting NC block: TOOL CALL S1000

To change the spindle speed S during program run:

	You can vary the spindle speed S on machines with stepless ballscrew drives by turning the spindle speed override knob on the TNC keyboard.

4.6 Entering Miscellaneous Functions and STOP

The M functions (M for miscellaneous) affect:

- Program run
- Machine functions
- Tool behavior

On the inside back cover of this manual you will find a list of M functions that are predetermined for the TNC. The list indicates whether an M function begins at the start or at the end of the block in which it is programmed.

Answer the following prompts in a positioning block:

.

.

.

MISCELLANEOUS FUNCTION M?	
e.g. 3 ENT	Enter the miscellaneous function, for example M3 (spindle on, clockwise rotation).

.

.

.

To enter an M function in a STOP block:

MISCELLANEOUS FUNCTION M?	
e.g. 5 ENT	Enter the miscellaneous function, for example M5 (spindle stop).

Resulting NC block: STOP M5

If the M function was programmed in a STOP block, program run will be interrupted at that block.

> Some M functions are not effective on certain machines. The machine tool builder may also add some of his own M functions.

A program run or test run is interrupted when it reaches a block containing the STOP function.

An M function can be programmed in a STOP block.

If you wish to interrupt the program run or program test for a certain duration, use the cycle 9: DWELL TIME (see page 8-38).

To enter a STOP function:

STOP ▶ Press the STOP key.

MISCELLANEOUS FUNCTION M ?	
e.g. 6 ENT	Enter an M function, if desired, for example M6 (tool change).

Resulting NC block: STOP M6

4.7 Actual Position Capture

Sometimes you may want to enter the actual position of the tool in a specific axis as a coordinate in a part program. Instead of reading the actual position values and entering them with the numeric keypad, you can simply press the “actual position capture” key.

A machine parameter determines whether the coordinates are written into an existing L block or a new block is generated (see also page 11-5).

This feature can be used, for example, to enter the tool length (see page 4-7).

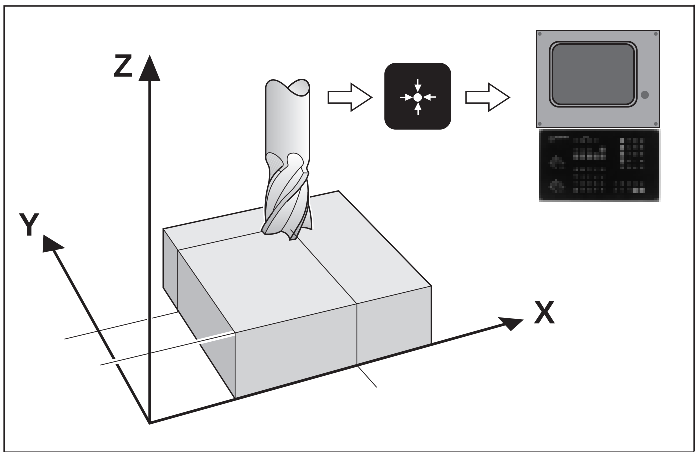


Fig. 4.12: Storing the actual position in the TNC

To capture the actual position:

▶ **MANUAL OPERATION**

Move the tool to the position that you wish to capture.

▶ **PROGRAMMING AND EDITING**

Select or create the block in which you wish to enter the actual position of the tool.

COORDINATES?

e.g. X	Select the axis in which you wish to capture a coordinate, for example X.
⌖	Transfer the actual position coordinate to the program.

Enter the radius compensation according the position of the tool relative to the workpiece.

Generating a new L block with the actual position coordinates

PROGRAMMING AND EDITING

In the PROGRAMMING AND EDITING mode, select the block behind which the L-block should be added.

MANUAL OPERATION

Move the tool to the position that you wish to capture.

⌖	The coordinates of the actual position are written into an L block.

The generated L block is inserted after the block selected in the PROGRAMMING AND EDITING mode. The L block has no tool radius compensation, feed rate, or M function. These must be added if needed.

> You can use the MOD function to define which axis coordinates are placed in the new L block (see page 11-5).
> The machine and TNC must be prepared by the machine tool builder for this feature.

5.1 General Information on Programming Tool Movements

A tool movement is always programmed as if the tool is moving and the workpiece is stationary.

Always pre-position the tool at the beginning of a part program to prevent the possibility of damaging the tool or workpiece.

Path functions

Each element of the workpiece contour is entered separately using path functions. The various path functions produce:

- Straight lines
- Circular arcs

You can also program a combination of the two (helical paths).

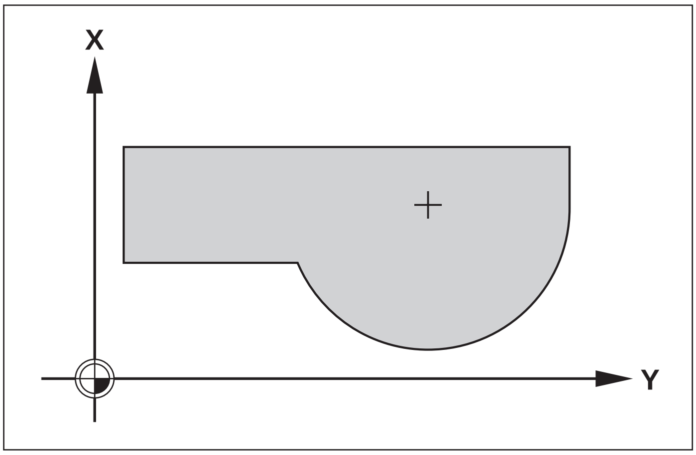


Fig. 5.1: A contour consists of a combination of straight lines and circular arcs

The contour elements are executed in sequence to machine the programmed contour.

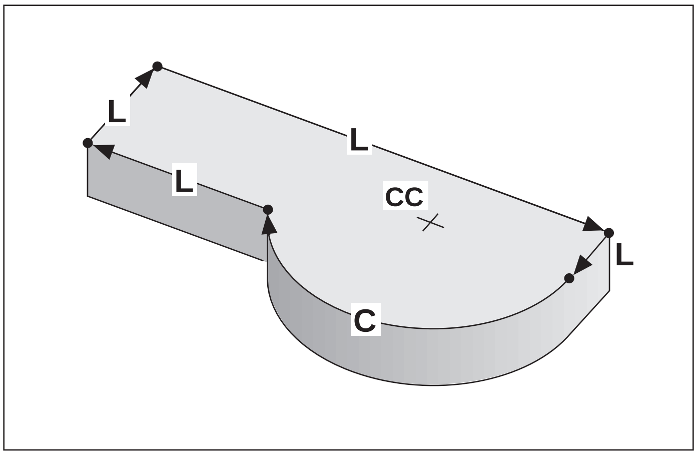


Fig. 5.2: Contour elements are programmed and executed in sequence

Subprograms and program section repeats

If a machining sequence repeats itself in a program, you can save time and reduce the chance of programming error by entering the sequence once and defining it as a subprogram or program section repeat.

Programming possibilities:

- To repeat a machining routine immediately after it is executed (program section repeat)
- To insert a machining routine at certain locations in a program (subprogram)
- To call a separate program for execution or test run within the main program (program call)

Cycles

Common machining routines are delivered with the control as standard cycles. The TNC features fixed cycles for:

- Peck drilling
- Tapping
- Slot milling
- Pocket and island milling

Coordinate transformation cycles can be used to change the coordinates of a machining sequence in a defined way, i.e.:

- Datum shift
- Mirroring
- Basic rotation
- Enlarging and reducing

Parameter programming

Instead of numerical values you enter markers in the program, so-called parameters, which are defined through mathematical functions or logical comparisons. You can use parametric programming for:

- Conditional and unconditional jumps
- Measurements with the 3D touch probe during program run
- Output of values and measurements
- Transferring values to and from memory

The following mathematical functions are available:

- Assign
- Addition/Subtraction
- Multiplication/Division
- Angle measurement/Trigonometry

etc.

5.2 Contour Approach and Departure

An especially convenient way to approach and depart a workpiece is on a tangential arc. This is done with the "corner rounding" function (RND) (see page 5-25).

Starting and end positions

Starting position

The tool moves from the starting position to the first contour point. The starting position is programmed without radius compensation.

The starting position must be:

- approachable without collision
- near the first contour point
- located to prevent contour damage during workpiece approach

If you choose a starting position within the hatch marked area of Figure 5.3 the tool will damage the contour as it approaches the first contour point.

The best starting position Ⓢ lies on the extension of the tool path for machining the first contour element.

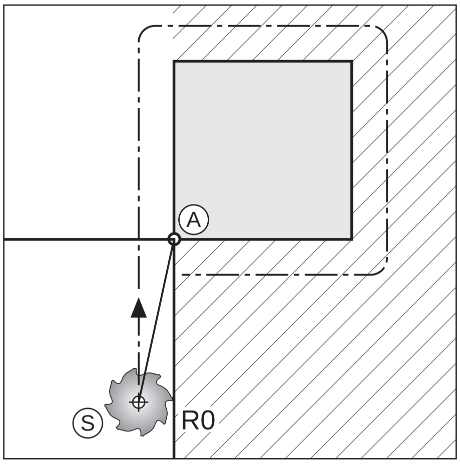


Fig. 5.3: Starting position Ⓢ for contour approach

First contour point

Workpiece machining starts at the first contour point. The tool moves on a radius-compensated path to this point.

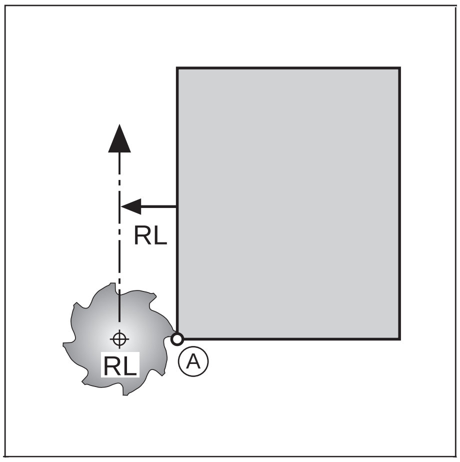


Fig. 5.4: First contour point Ⓐ for machining

Approaching the starting point in the spindle axis.

The spindle moves to its working depth as it approaches the starting position Ⓢ.

If there is any danger of collision, move the spindle axis separately to the starting position.

Example:	L	X ... Y ...	Positioning in X/Y
	L	Z–10	Positioning in Z

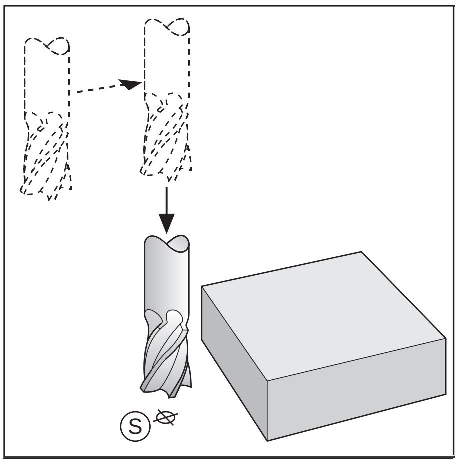


Fig. 5.5: Move the spindle axis separately if there is any danger of collision

End position

The end position, like the starting point, must be

- approachable without collision
- near the last contour point
- located to prevent contour damage during workpiece departure

The best end position Ⓔ lies on the extension of the tool path. The end position can be located anywhere outside of the hatch marked area in Fig. 5.6. It is approached without radius compensation.

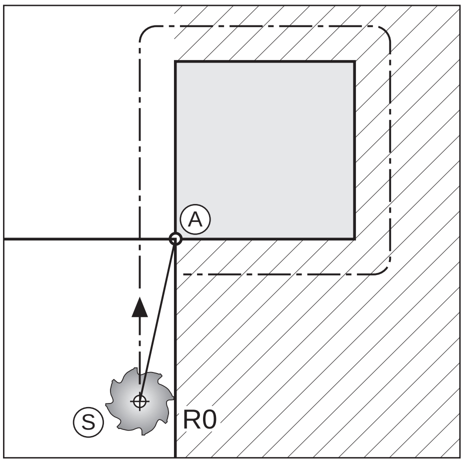


Fig. 5.6: End position Ⓔ after machining

Departing the end position in the spindle axis

The spindle axis is moved separately when the end position is departed.

```
Example:  L    X ... Y ... R0    approaching the end position
          L    Z+50              retracting the tool
```

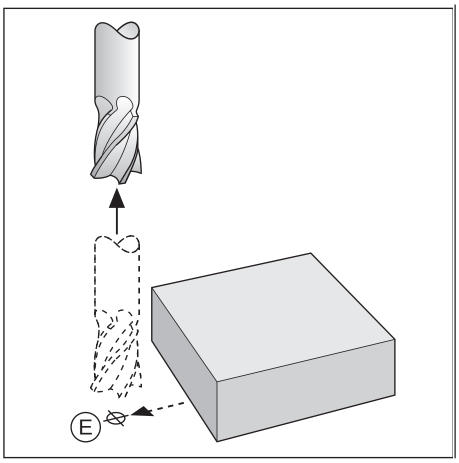


Fig. 5.7: Retract separately in the spindle axis

Common starting and end position

A common starting and end position SE can be located outside of the hatch marked area in the figures.

The best common starting and end position lies exactly between the extensions of the tool paths for machining the first and last contour elements.

A common starting and end position is approached without radius compensation.

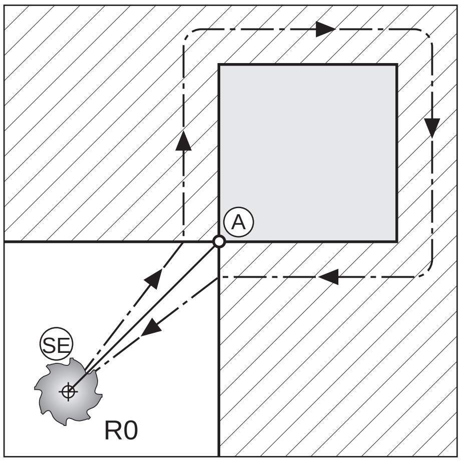


Fig. 5.8: Common starting and end position

Smooth approach and departure

With the RND function the tool approaches and departs the workpiece at a tangent. This prevents dwell marks on the workpiece surface.

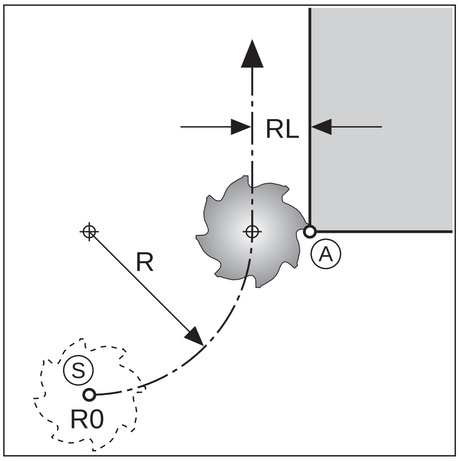


Fig. 5.9: Smooth approach onto a contour

Starting and end position

The starting Ⓢ and end Ⓔ positions of machining lie outside of the workpiece and near the first and last contour elements, respectively.

The tool path to the starting and end positions are programmed without radius compensation.

Input

The RND function is entered at the following locations in the program:

- During contour approach: after the block in which the first contour point is programmed, i.e. after the first RL/RR radius-compensated block.
- During contour departure: after the block in which the last contour point is programmed, i.e. after the last RL/RR radius-compensated block.

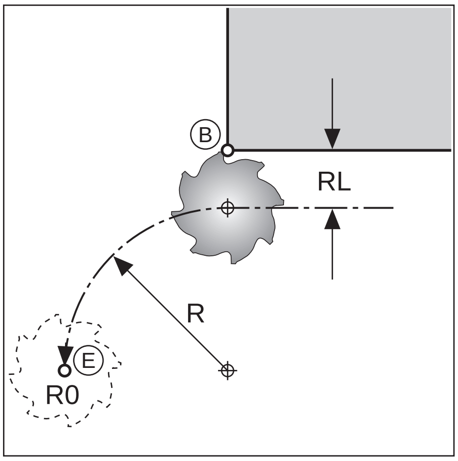


Fig. 5.10: Smooth departure from a contour

Program example

```
.
.
.
L     X ... Y ... R0 ... ..................................................... Starting position Ⓢ
L     X ... Y ... RL ... ..................................................... First contour point Ⓐ
RND  R ... .................................................................... Smooth approach
.
.
.
Contour elements
.
.
.
L     X ... Y ... RL ... ..................................................... Last contour point Ⓑ
RND  R ... .................................................................... Smooth departure
L     X ... Y ... R0 ... ..................................................... End position Ⓔ
```

For proper execution of an RND function, a radius must be chosen such that the arc can connect the start or end point with the contour point.

5.3 Path Functions

General information

Part program input

To create a part program you enter the dimensional information given on the workpiece drawing. The workpiece coordinates are entered as relative or absolute values.

You normally program a contour element by entering its end point.

The TNC automatically calculates the tool path from the tool data and the radius compensation.

Machine axis movement under program control

All axes programmed in a single block are moved simultaneously.

Paraxial movement

Paraxial movement means that the tool path is parallel to the programmed axis.

Number of axes programmed in the NC block: 1

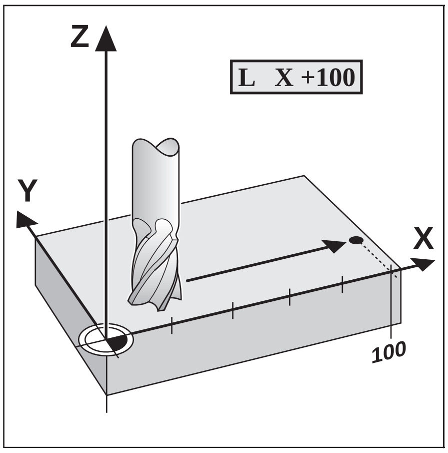


Fig. 5.11: Paraxial movement

Movement in the main planes

With this type of movement the tool moves to the programmed position on a straight line or circular arc in a “working plane.”

Number of axes programmed in the NC block: 2

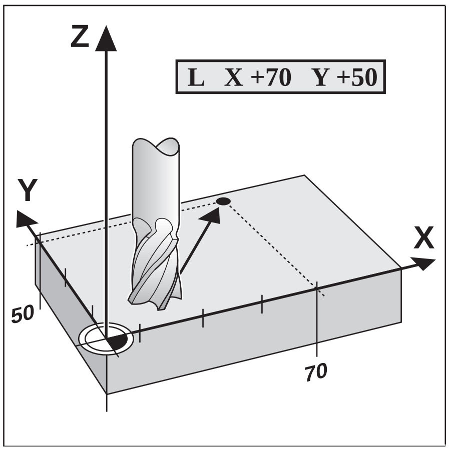


Fig. 5.12: Movement in a main plane (X/Y plane)

Movement of three machine axes (3D movement)

The tool moves in a straight line to the programmed position.

Number of axes programmed in the NC block: 3

Exception: A helical path is created by combining a circular movement in a plane with a linear movement perpendicular to the plane.

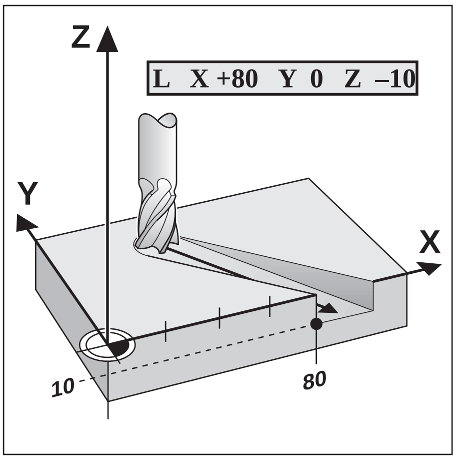


Fig. 5.13: Three-dimensional tool movement

Overview of path functions

The path function keys determine the type of contour element and initiate the plain language dialog.

Function	Key	Tool movement
Line	L	Straight line
Circle **C**enter	CC	Coordinates of a circle center or pole
Circle	C	Circular arc around a circle center CC to an arc end point
Circle by **R**adius	CR	Circular arc with a certain radius
Circle, **T**angential	CT	Circular arc with a tangential connection to the previous contour element
Rou**ND**ing of corners	RND	Circular arc with tangential connection to the previous and subsequent contour elements

5.4 Path Contours – Cartesian Coordinates

Straight line

To program a straight line, you enter:

- The coordinates of the end point Ⓔ
- If necessary:
  Radius compensation, feed rate, miscellaneous function

The tool moves in a straight line from its starting position Ⓢ to the end point Ⓔ. The starting position was reached in the previous block.

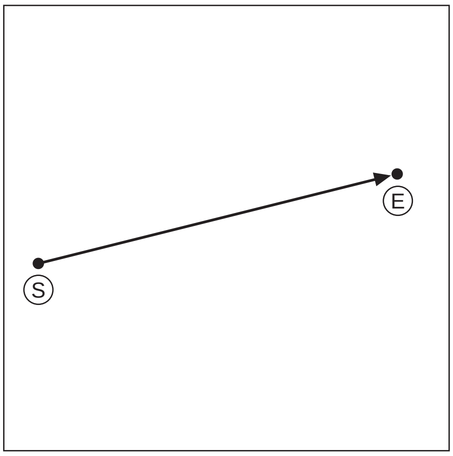


Fig. 5.14: A linear movement

To program a straight line:

COORDINATES?	
If necessary I	Identify coordinates as relative values.
e.g. X	Press the orange axis-selection key, for example X.
e.g. 5 0	Enter the coordinate of the end point, for example 50 mm.
If necessary +/–	If a coordinate is negative, press the +/- key.
e.g. Y . . . e.g. Z	Enter all further coordinates of the end point.
ENT	After entering all coordinates, close the dialog with the ENT key.

.
.
.

⋮

TOOL RADIUS COMP.: RL / RR / NO COMP. ?	
RL	The tool must move to the **left** of the programmed contour to compensate its own radius.
RR	The tool must move to the **right** of the programmed contour to compensate its own radius.
ENT	The tool moves **directly** to the end point.

FEED RATE F = ? / F MAX = ENT	
e.g. 1 0 0 ENT	Enter the feed rate of the tool on the straight line, for example 100 mm/min.
ENT	Enter rapid tool traverse, F = FMAX.

MISCELLANEOUS FUNCTION M?	
e.g. 3 ENT	Enter a miscellaneous function, if appropriate, for example M3 (spindle on, clockwise rotation).

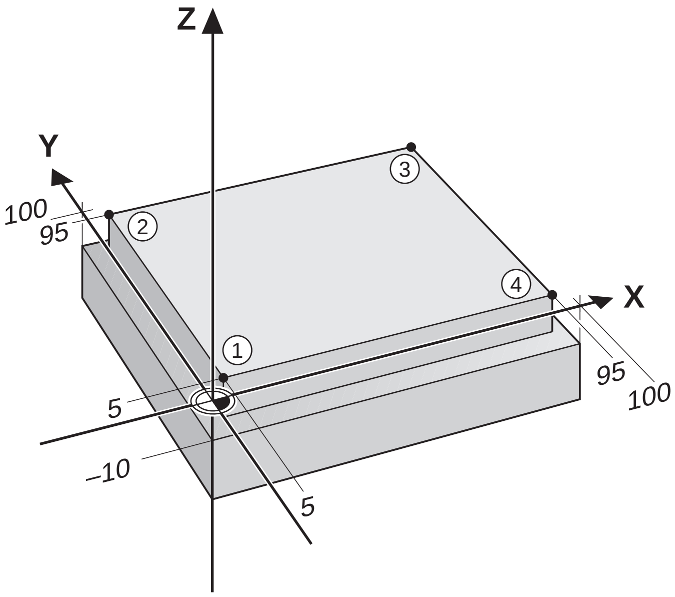
Resulting NC block: L IX–50 Y+10 Z–20 RR F100 M3

Example for exercise: Milling a rectangle

Coordinates of the corner points:

①	X	=	5 mm	Y	=	5 mm
②	X	=	5 mm	Y	=	95 mm
③	X	=	95 mm	Y	=	95 mm
④	X	=	95 mm	Y	=	5 mm

Milling depth: Z = –10 mm



Part program

```
0    BEGIN PGM 360511 MM .................................. Begin program; program number 360511; dimensions in
                                                              millimeters
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0 ..................... Define blank form for graphic workpiece simulation
                                                              (MIN and MAX point)
3    TOOL DEF 1 L+0 R+5
4    TOOL CALL 1 Z S1000 ...................................... Define tool in the program; call tool in the spindle axis Z;
                                                              spindle speed S = 1000 rpm
5    L Z+100 R0 FMAX M6 ...................................... Retract in the spindle axis; rapid traverse; insert tool
6    L X–10 Y–10 FMAX.............................................. Pre-position in X and Y; rapid traverse
7    L Z–10 FMAX M3 ............................................... Move to working depth; rapid traverse; spindle on with
                                                              clockwise rotation
8    L X+5 Y+5 RL F100 ........................................... Move to first contour point – corner point 1 – with radius
                                                              compensation (RL) and reduced feed rate (F 100)
9    L Y+95 ................................................................ Move to second contour point – corner point 2: all values that
                                                              remain the same as in block 8 need not be re-programmed
10   L X+95 ................................................................ Move to third contour point – corner point 3
11   L Y+5 .................................................................. Move to fourth contour point – corner point 4
12   L X+5 Y+5 .......................................................... Conclude milling, return to first contour point
13   L X–10 Y–10 R0 FMAX ...................................... For safety reasons, retract in X and Y; rapid traverse
14   L Z+100 FMAX M2 ............................................. Move tool to setup clearance; rapid traverse;
                                                              spindle off, coolant off, program stop,
                                                              Return jump to block 1
15   END PGM 360511 MM ..................................... End of program
```

Chamfer

The chamfer function permits you to cut off corners at the intersection of two straight lines.

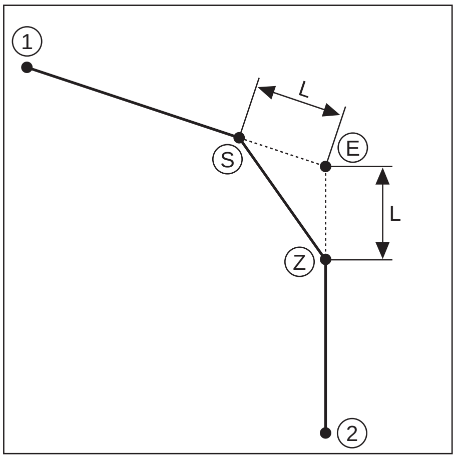


Fig. 5.15: Chamfer from Ⓢ to Ⓩ

You enter the length to be removed from each side of the corner.

Prerequisites:

- The blocks before and after the chamfer block must be in the same working plane.
- The radius compensation before and after the chamfer block must bethe same.
- An inside chamfer must be large enough to accommodate the currenttool.

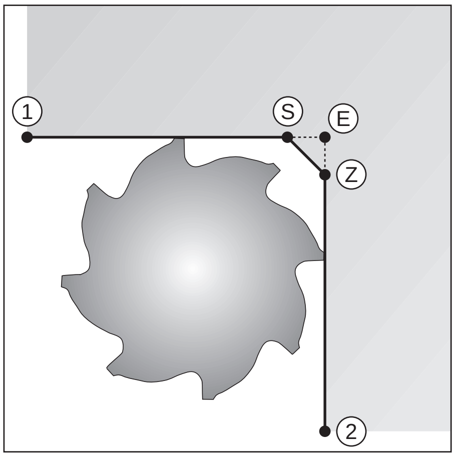


Fig. 5.16: Tool radius too large

- You cannot start a contour with a chamfer block.
- A chamfer is only possible in the working plane.
- The feed rate for chamfering is taken from the previous block.
- The corner point Ⓔ is cut off by the chamfer and is not part of the resulting contour.

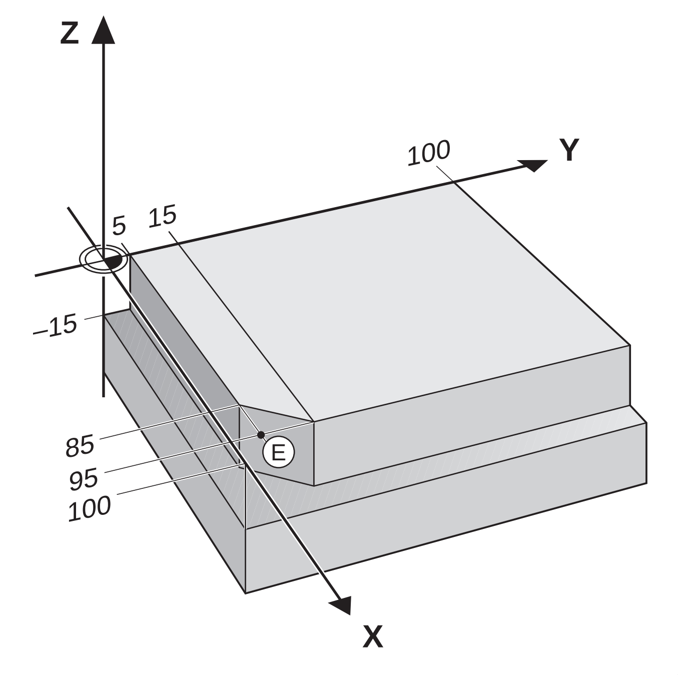
To program a chamfer:

L	Select the “straight line” function.
COORDINATES?	
e.g. 5 ENT	Enter the length to be removed from each side of the corner, for example 5 mm.

Resulting NC block: L5

Example for exercise: Chamfering a corner

Coordinates of the corner points Ⓔ:	X	=	95	mm
	Y	=	5	mm
Chamfer length:	L	=	10	mm
Milling depth:	Z	=	–15	mm
Tool radius:	R	=	+10	mm



Part program

```
0    BEGIN PGM 360513 MM ................................... Begin program
1    BLK FORM 0.1 Z X+0 Y+0 Z–20 ........................ Workpiece blank MIN point
2    BLK FORM X+100 Y+100 Z+0 ........................... Workpiece blank MAX point
3    TOOL DEF 5 L+5 R+10 ...................................... Tool definition
4    TOOL CALL 5 Z S500 ......................................... Tool call
5    L Z+100 R0 FMAX M6 ....................................... Retract spindle and insert tool
6    L X–10 Y–5 FMAX............................................... Pre-position in X, Y
7    L Z–15 FMAX M3 ............................................... Pre-position to the working depth
8    L X+0 Y+5 RR F200........................................... Move with radius compensation (RR) and reduced feed (F200)
                                                              to the first contour point
9    L X+95 Y+5 ........................................................ Program the first straight line for corner Ⓔ
10   L 10...................................................................... Chamfer block: inserts a chamfer with L = 10 mm
11   L X+95 Y+100 ..................................................... Program the second straight line for corner Ⓔ
12   L X+110 Y+110 R0 FMAX ................................. Retract the tool in X, Y (12) and Z (13); return to block 1 (13)
                                                              and end program
13   L Z+100 FMAX M2
14   END PGM 360513 MM
```

Circle and circular arcs

The TNC can control two machine axes simultaneously to move the tool in a circular path.

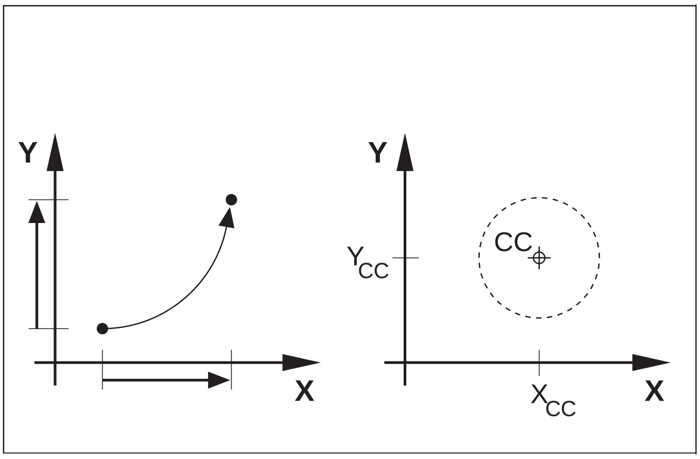


Fig. 5-17: Circular arc and circle center

Circle Center CC

You can define a circular movement by entering its center CC.

A circle center can also serve as reference (pole) for polar coordinates.

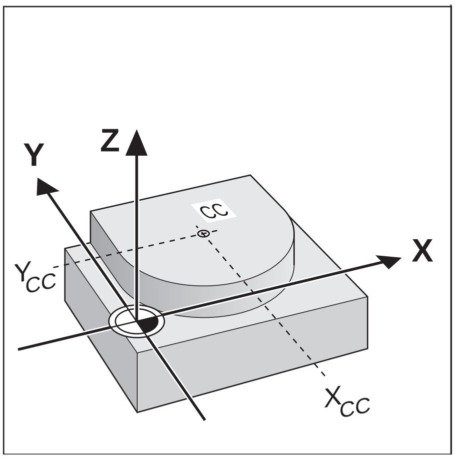


Fig. 5.18: Circle center coordinates

Direction of Rotation DR

When there is no tangential transition to another contour element, enter the mathematical direction of rotation DR, where

- a clockwise direction of rotation is mathematically negative: DR-
- a counterclockwise direction of rotation is mathematically positive: DR+

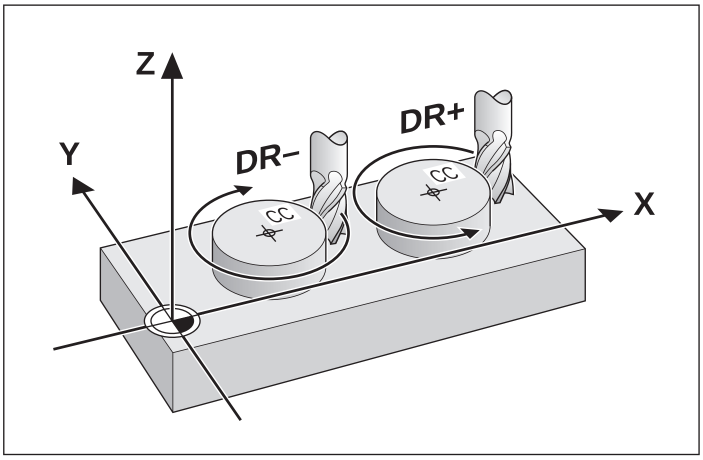


Fig. 5.19: Direction of rotation for circular movements

Radius compensation in circular paths

You cannot begin radius compensation in a circle block. It must be activated beforehand in a line block.

Circles in the main planes

When you program a circle, the TNC assigns it to one of the main planes. This plane is automatically defined when you set the spindle axis during TOOL CALL.

Spindle axis	Main plane
Z	XY
Y	ZX
X	YZ

Fig. 5.20: Defining the spindle axis also defines the main plane

You can program circles that do not lie parallel to a main plane by using Q parameters. See Chapter 7.

Circle Center CC

If you program an arc using the C path function key, you must first define the circle center CC by:

- entering the Cartesian coordinates of the circle center
- using the circle center defined in an earlier block
- capturing the actual position

You can define the last programmed position as circle center CC by entering an empty CC block.

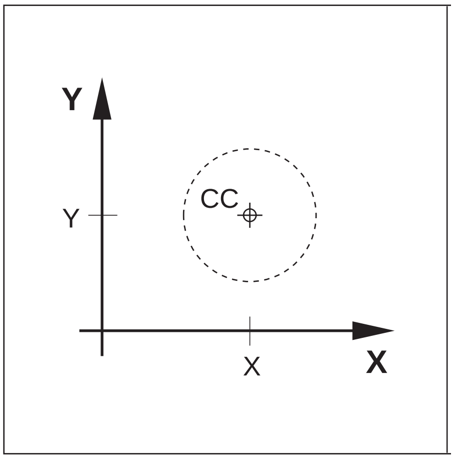


Fig. 5.21: Circle center CC

Duration of a circle center definition

A circle center definition remains effective until a new circle center is defined.

Entering CC in relative values

If you enter the circle center with relative coordinates, you have defined it relative to the last programmed tool position.

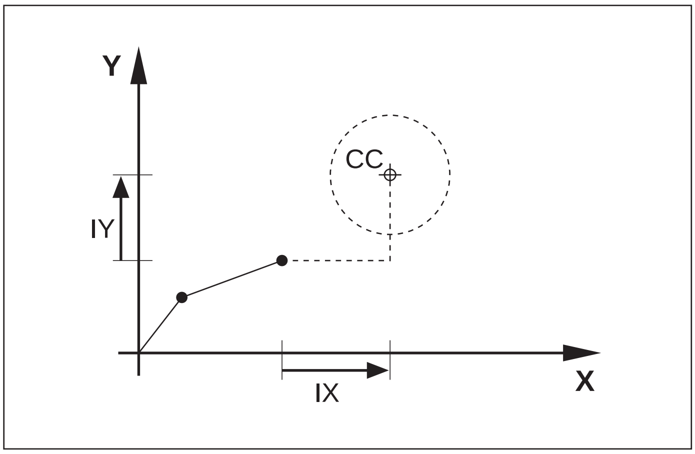


Fig. 5.22: Incremental circle center coordinates

- The circle center CC also serves as pole for polar coordinates
- CC defines a position as a circle center. The resulting contour is located on the circle, not on the circle center.

To program a circle center (pole):

CC ▶

COORDINATES?	
e.g. X	Select the coordinate axis, for example X.
e.g. 2 0 ENT	Enter the coordinate for the circle center in this axis, for example X = 20 mm.
e.g. Y	Select the second coordinate axis, for example Y.
e.g. 1 0 +/– ENT	Enter the coordinate of the circle center, for example Y = –10 mm.

Resulting NC block: CC X+20 Y-10

Circular Path C Around the Center Circle CC

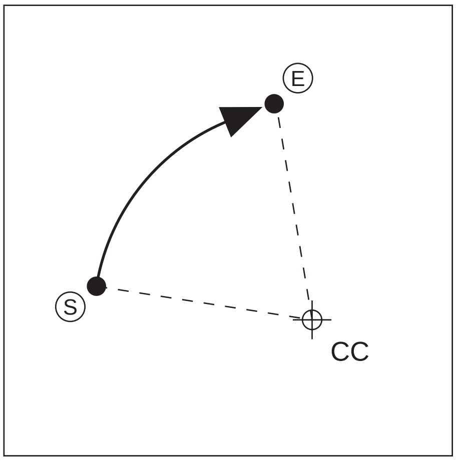


Fig. 5.23: A circular arc from Ⓢ to Ⓔ around CC

Prerequisites

The circle center CC must have been previously defined in the program. The tool is located at the arc starting point Ⓢ.

Input

- Arc end point
- Direction of rotation (DR)

The starting and end points of the arc must lie on the circle.
Input tolerance: up to 0,016 mm.

- To program a full circle, enter the same point for the end point as for the start point in a C-block.

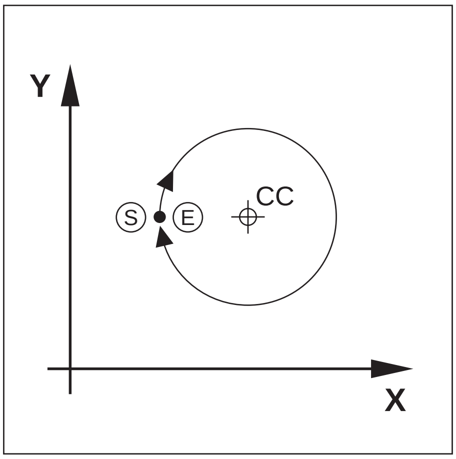


Fig. 5.24: Full circle around CC with a C-block

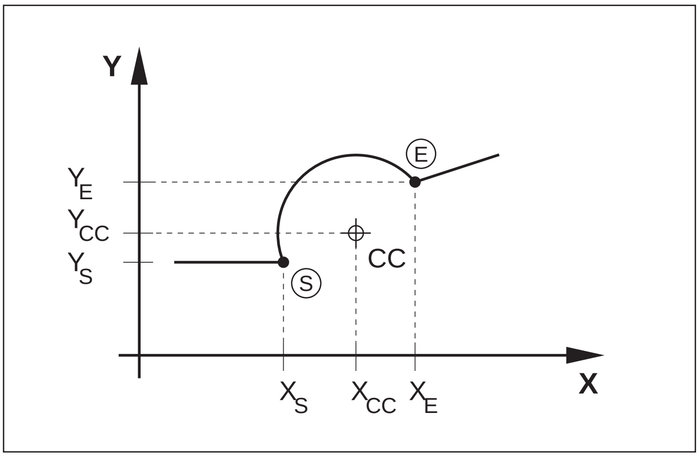


Fig. 5.25: Coordinates of a circular arc

To program a circular arc C around a circle center CC:

C ▶ **COORDINATES?**

e.g. I X 5	Enter the first coordinate of the arc end point, for example IX = 5 mm.
e.g. Y +/– 5	Enter the second coordinate of the arc end point, for example Y = – 5 mm.
ENT	Terminate coordinate entry.

ROTATION CLOCKWISE: DR-?

1 x +/– or 2 x +/– ENT	Select negative (DR-) or positive (DR+) direction of rotation.

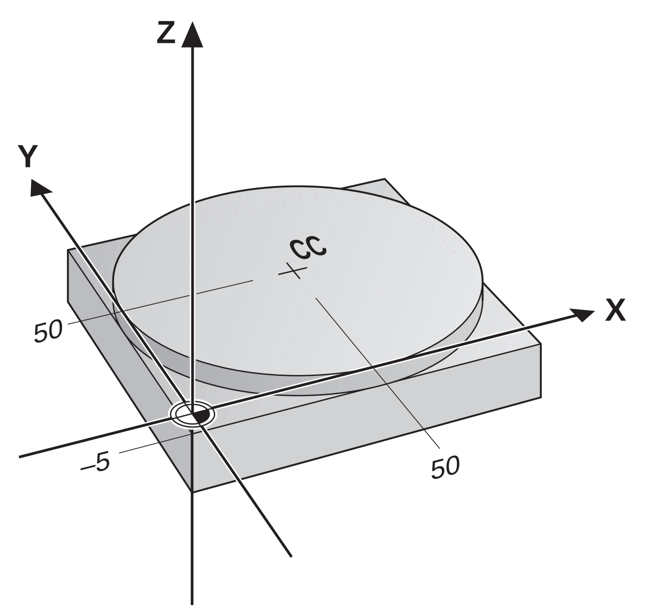
If necessary, enter also:

- Radius compensation
- Feed rate
- Miscellaneous function

Resulting NC block: C IX+5 Y-5 DR–

Example for exercise: Milling a full circle in one block

Circle center CC:	X	=	50 mm
	Y	=	50 mm
Beginning and end of a circle center C:	X	=	50 mm
	Y	=	0 mm
Milling depth:	Z	=	–5 mm
Tool radius:	R	=	15 mm



Part program

```
0    BEGIN 360519 MM .......................................... Begin program
1    BLK FORM 0.1 Z X+0 Y+0 Z–20 ....................... Define workpiece blank
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 6 L+0 R+15 .................................... Define tool
4    TOOL CALL 6 Z S500 ....................................... Call tool
5    CC X+50 Y+50 ................................................. Coordinates of the circle center CC
6    L Z+100 R0 FMAX M6 ..................................... Insert tool
7    L X+50 Y–40 FMAX .......................................... Pre-position the tool
8    L Z–5 FMAX M3
9    L X+50 Y+0 RL F100 ........................................ Move under radius compensation to the first contour point
10   RND R10 ........................................................... Smooth approach
11   C X+50 Y+0 DR– .............................................. Mill circular arc C around circle center CC; end point coordi-
                                                                   nates X = +50 mm and Y = 0; negative direction of rotation
12   RND R10 ........................................................... Smooth departure
13   L X+50 Y–40 R0 FMAX
14   L Z+100 FMAX M2
15   END PGM 360519 MM .................................... Retract tool and end program
```

Circular path CR with defined radius

The tool moves on a circular path with the radius R.

Input

- Coordinates of the arc end point
- Arc radius R
- Direction of rotation DR

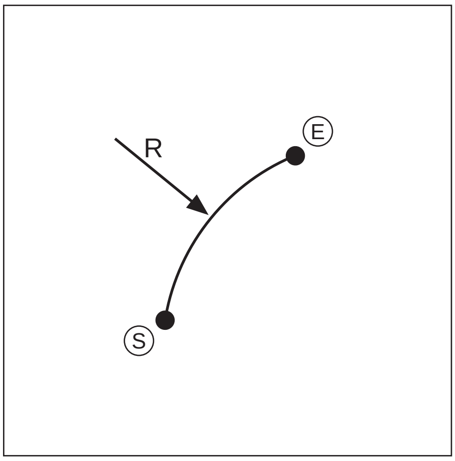


Fig. 5.26: Circular path from Ⓢ to Ⓔ with radius R

- To program a full circle with CR you must enter two successive CR-blocks.
- The distance from the starting point to the end point cannot be larger than the diameter of the circle.
- The maximum permissible radius is 30 m (9.8 ft).

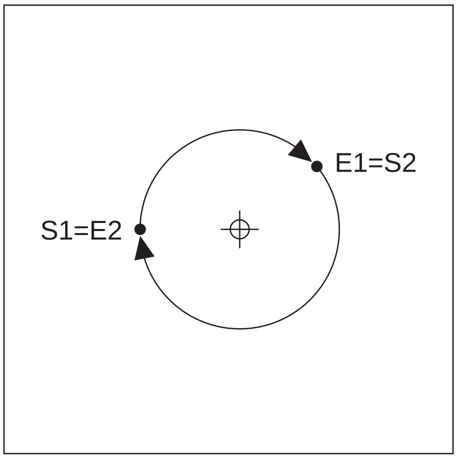


Fig. 5.27: Full circle with two CR-blocks

Arc radius R

Starting point Ⓢ and end point Ⓔ can be connected by four different arcs with the same radius. The arcs differ in their curvatures and lengths.

To program **a large semicircle** enter the radius R with a **negative** sign (R<0).

To program a arc **small semicircle**: enter the radius R with a **positive** sign (R>0).

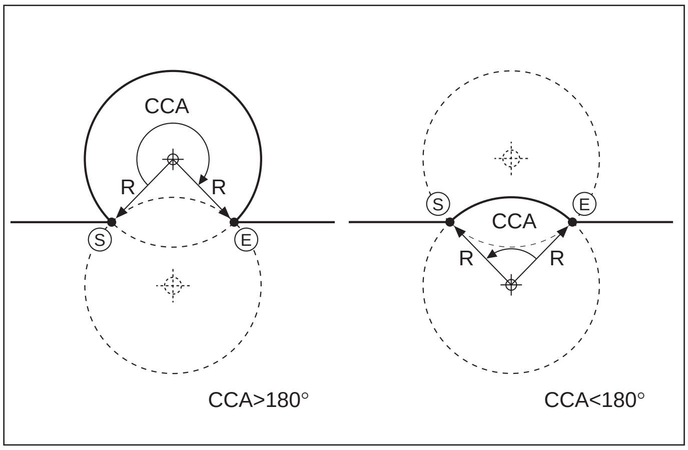


Fig. 5.28: Circular arcs with central angles greater than and less than 180°

Direction of rotation DR and arc shape

This direction of rotation determines whether the arc is

- convex (curved outward) or

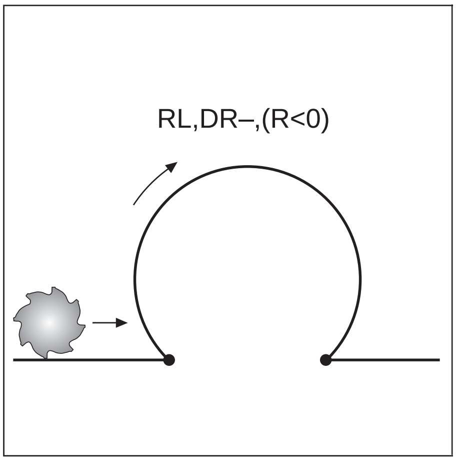


Fig. 5.29: Convex path

- concave (curved inward)

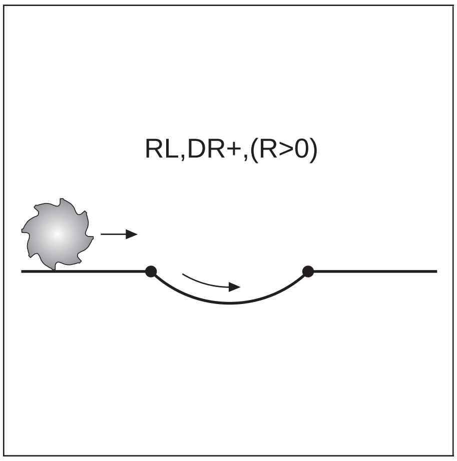


Fig. 5.30: Concave path

To program a circular arc with defined radius:

CR ▶ **COORDINATES?**

e.g. X 1 0 Y 2 ENT — Enter the coordinates of the arc end point, for example X = 10 mm, Y = 2 mm.

CIRCLE RADIUS (SIGN)?

e.g. 5 +/– ENT — Enter the arc radius, for example R = 5 mm; and determine the size of the arc using the sign, here the negative sign.

ROTATION CLOCKWISE: DR-?

1x +/– or 2x +/– ENT — Define the circular arc with a negative (DR–) or positive direction of rotation (DR+).

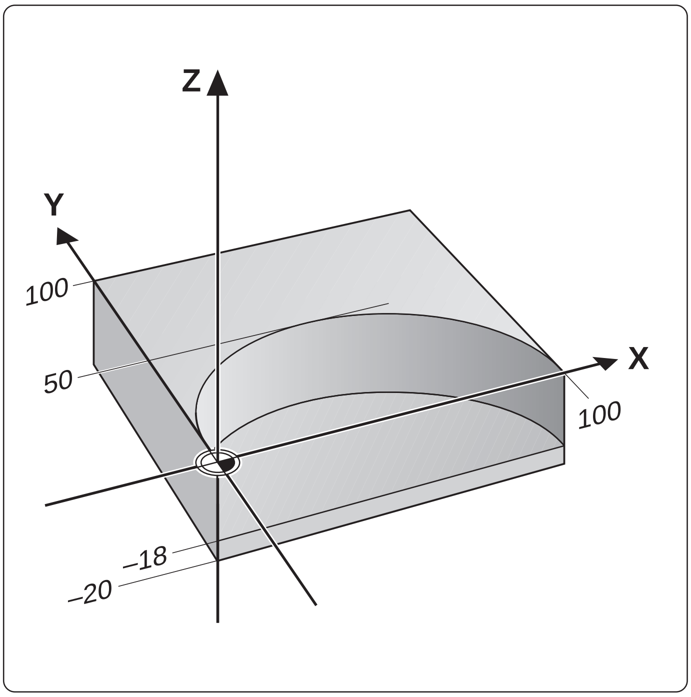
If necessary, enter also:

- Radius compensation
- Feed rate
- Miscellaneous function

Resulting NC block: CR X+10 Y+2 R–5 DR– RL

Example for exercise: Milling a concave semicircle

Semicircle radius:	R	=	50 mm
Coordinates of the arc starting point:	X	=	0
	Y	=	0
Coordinates of the arc end point:	X	=	100 mm
	Y	=	0
Tool radius:	R	=	25 mm
Milling depth:	Z	=	18 mm



Part program

```
0    BEGIN PGM 360522 M ..................................... Begin program
1    BLK FORM 0.1 Z X+0 Y+0 Z–20 ........................ Define the workpiece blank
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 2 L+0 R+25 ..................................... Define the tool
4    TOOL CALL 2 Z S2000 ...................................... Call the tool
5    L Z+100 R0 FMAX M6 ...................................... Insert and pre-position the tool
6    L X+25 Y–30 FMAX
7    L Z–18 FMAX M3
8    L X+0 Y+0 RR F100 .......................................... First contour point
9    CR X+100 Y+0 R+50 DR– ................................. Mill circular arc CR to the end point X = 100 mm, Y = 0; radius
                                                              R = 50 mm, negative direction of rotation
10   L X+70 Y–30 R0 FMAX
11   L Z+100 FMAX M2
12   END PGM 360522 MM ..................................... Retract the tool and end the program
```

Circular path CT with tangential connection

The tool moves in an arc that starts at a tangent with the previously programmed contour element.

A transition between two contour elements is called tangential when one contour element makes a smooth and continuous transition to the next. There is no visible corner at the intersection.

Input

Coordinates of the arc end point

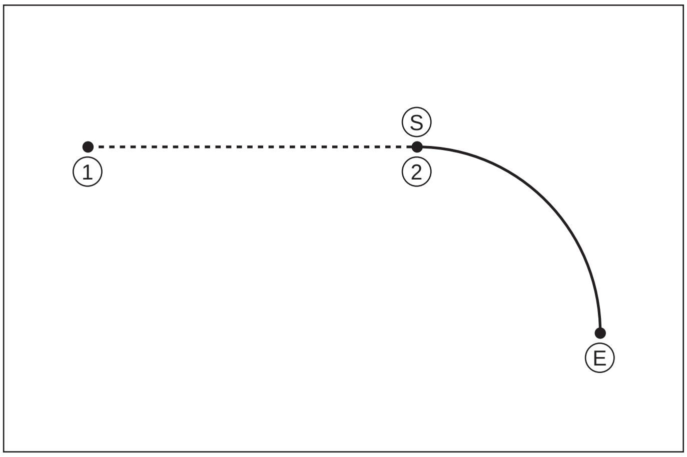


Fig. 5.31: The straight line ① - ② is connected tangentially to the circular arc Ⓢ - Ⓔ

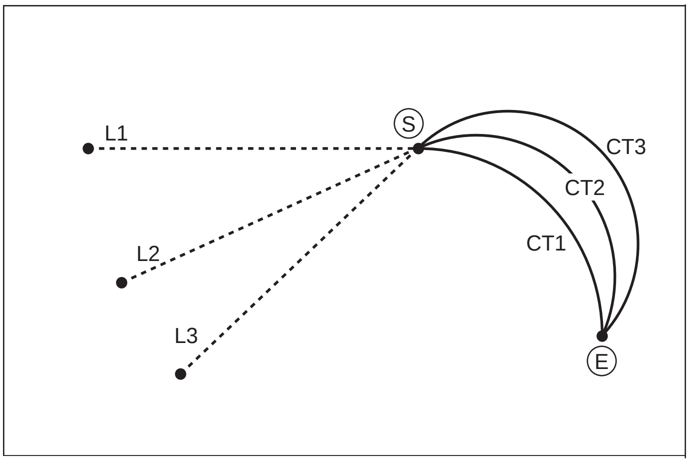


Fig. 5.32: The path of a tangential arc depends on the preceding contour element

Prerequisites:

- The contour element to which the tangential arc connects must be programmed immediately before the CT block.
- There must be at least two positioning blocks defining the tangentially connected contour element before the CT block.

A tangential arc is a two-dimensional operation: the coordinates in the CT block and in the positioning block before it should be in the plane of the arc.

To program a circular path CT with tangential connection:

CT ▶ **COORDINATES?**

e.g. I X 5 0 I Y +/– 1 0 ENT	Enter the coordinates of the arc end point, for example IX = 50 mm, IY = –10 mm.

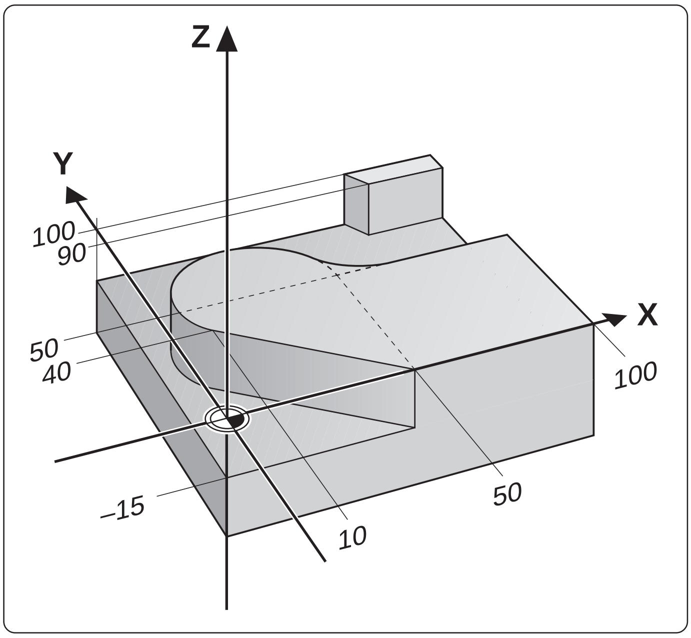
If necessary, enter also:

- Radius compensation
- Feed rate
- Miscellaneous function

Resulting NC block: CT IX+50 IY-10 RR

Example for exercise: Circular arc connecting to a straight line

Coordinates of the transition point from the line to the arc:	X	=	10 mm
	Y	=	40 mm
Coordinates of the arc end point:	X	=	50 mm
	Y	=	50 mm
Milling depth:	Z	=	–15 mm
Tool radius:	R	=	20 mm



Part program

```
0   BEGIN PGM 360524 MM ................................. Begin program
1   BLK FORM 0.1 Z X+0 Y+0 Z–20 ....................... Define the workpiece blank
2   BLK FORM 0.2 X+100 Y+100 Z+0
3   TOOL DEF 2 L+0 R+20 ..................................... Define the tool
4   TOOL CALL 2 Z S 1000 ..................................... Call the tool
5   L Z+100 R0 FMAX M6 ...................................... Insert the tool
6   L X+30 Y–30 FMAX .......................................... Pre-position the tool
7   L Z–15 FMAX M3
8   L X+50 Y+0 RL F100 ........................................ First contour point
9   L X+10 Y+40 .................................................... Straight line connecting tangentially to the arc
10  CT X +50 Y +50 ................................................ Arc to end point with coordinates X = 50 mm and Y =
                                                                  50 mm; Connects tangentially to the straight line in block 9
11  L X+100 ........................................................... End of contour
12  L X+130 Y+70 R0 FMAX
13  L Z+100 FMAX M2 ............................................ Retract tool and end program
14  END PGM 360524 MM
```

Corner rounding RND

The tool moves in an arc that connects tangentially both with the preceding and the subsequent elements. The RND function is useful for:

- Rounding corners

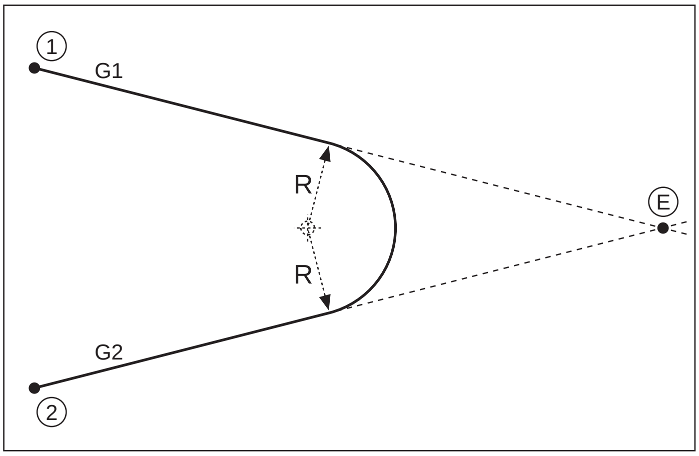


Fig. 5.33: Rounding radius R between G1 and G2

- Approaching and departing contours on a tangent

Input

- Radius of the arc
- Feed rate for RND

Prerequisite

On inside corners, the rounding arc must be large enough to accommodate the tool.

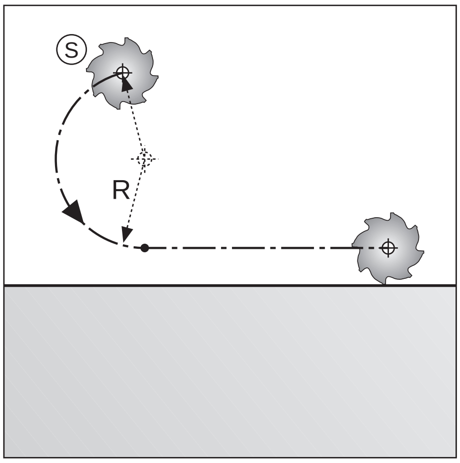


Fig. 5.34: Smooth approach with RND

- In the preceding and subsequent blocks both coordinates should lie in the plane of the arc.
- The corner point Ⓔ is cut off by the rounding arc and is not part of the contour.
- A feed rate programmed in the RND block is effective only in that block. After the RND block the previous feed rate becomes effective again.

To program a tangential arc between two contour elements:

RND

ROUNDING OFF RADIUS R?

e.g. 1 0 ENT

Enter the rounding radius, for example R = 10 mm.

FEED RATE? F=

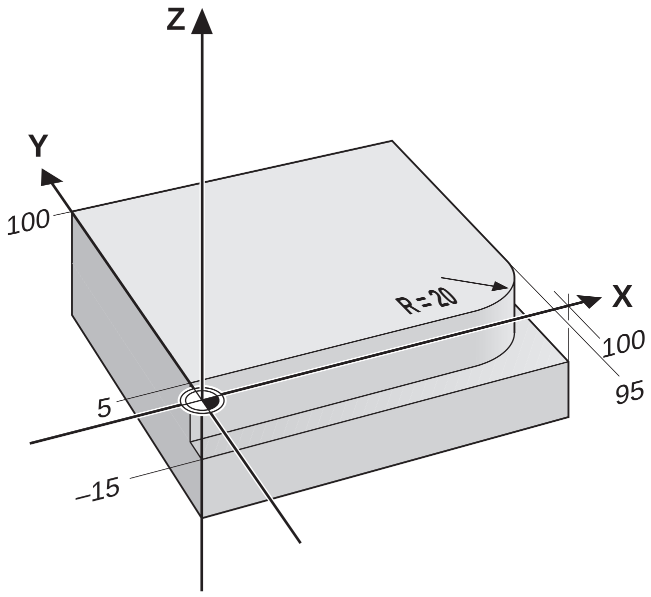
e.g. 1 0 0 ENT

Enter the feed rate for the rounding radius, here F = 100 mm/min.

Resulting NC block: RND 10 F 100

Example for exercise: Rounding a corner

Coordinates of the corner point:	X	=	95 mm
	Y	=	5 mm
Rounding radius:	R	=	20 mm
Milling depth:	Z	=	–15 mm
Tool radius:	R	=	10 mm



Part program

```
0    BEGIN PGM 360526 MM ................................. Begin program
1    BLK FORM 0.1 Z X+0 Y+0 Z–20 ....................... Define the workpiece blank
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+10 .................................... Define the tool
4    TOOL CALL 1 Z S1500 ..................................... Call the tool
5    L Z+100 R0 FMAX M6 ..................................... Insert the tool
6    L X–10 Y–5 FMAX ............................................. Pre-position the workpiece
7    L Z–15 FMAX M3
8    L X+0 Y+5 RR F100 .......................................... First contour element
9    L X+95 .............................................................. First straight line for the corner
10   RND R20 ........................................................... Round the corner with a tangential arc with radius
                                                                         R = 20 mm between the two sides
11   L Y+100 ............................................................ Second straight line for the corner
12   L X+120 Y+120 R0 FMAX ................................ Retract the tool and end program
13   L Z+100 R0 FMAX M2
14   END PGM 360526 MM
```

5.5 Path Contours – Polar Coordinates

Polar coordinates are useful for programming:

- Positions on circular arcs
- Positions from workpiece drawings showing angular dimensions

Section 1.2 "Fundamentals of NC" provides a detailed description of polar coordinates.

Polar coordinates are marked with a P.

Polar coordinate origin: Pole CC

You can define the pole anywhere in the program before the blocks containing polar coordinates. Enter the pole in Cartesian coordinates as a circle center in a CC block.

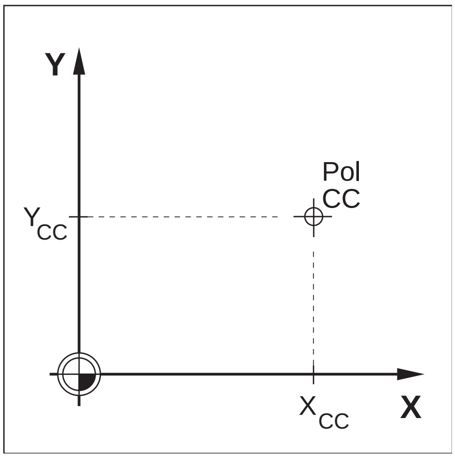


Fig. 5.35: The pole is entered as CC

Straight line LP

- You can enter any value from –360° to +360° for PA.
- Enter the algebraic sign for PA relative to the angle reference axis:
  For an angle from the reference axis **counterclockwise** to PR: PA>0
  For an angle from the reference axis **clockwise** to PR: PA<0

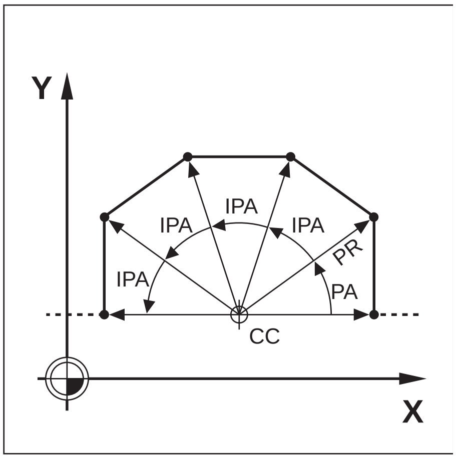


Fig. 5.36: Contour consisting of straight lines with polar coordinates

L ▶ **COORDINATES?**

P	Select polar coordinates.

POLAR COORDINATES RADIUS PR?

e.g. 5 ENT	Enter the radius from the pole to the straight line end point, for example PR = 5 mm.

POLAR COORDINATES ANGLE PA?

e.g. 3 0 ENT	Enter the angle from the reference axis to PR, for example PA = 30°.

If necessary, enter also:

Radius compensation
Feed rate
Miscellaneous function

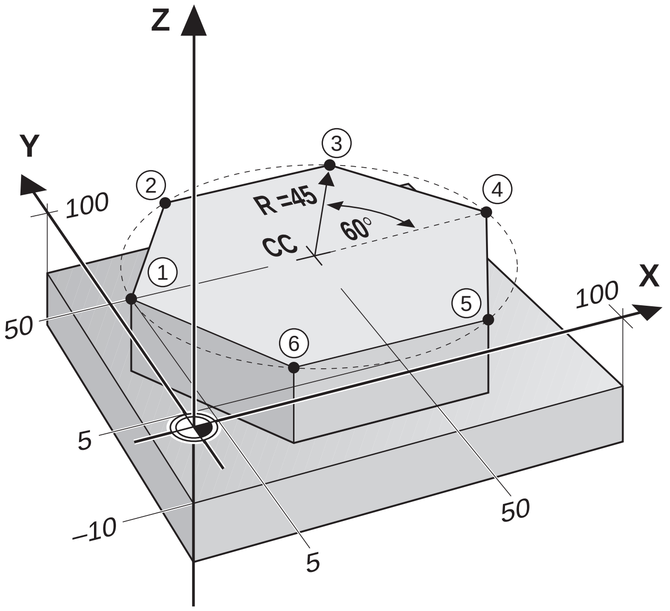
Resulting NC block: LP PR+5 PA+30

Example for exercise: Milling a hexagon

Corner point coordinates:

①	PA = 180°	PR = 45 mm
②	PA = 120°	PR = 45 mm
③	PA = 60°	PR = 45 mm
④	PA = 0°	PR = 45 mm
⑤	PA = 300°	PR = 45 mm
⑥	PA = 240°	PR = 45 mm

Milling depth:	Z	= –10 mm
Tool radius:	R	= 5 mm



Part program

```
0    BEGIN PGM 360529 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+5
4    TOOL CALL 1 Z S 1000
5    CC X+50 Y+50                      General data and first contour point (corner point 1)
6    L Z+100 R0 FMAX M6
7    LP PR+60 PA–190 R0 FMAX
8    L Z–10 FMAX M3
9    LP PR+45 PA+180 RL F100

10   LP PA+120
11   LP PA+60
12   LP IPA–60                         Corner points ② to ⑥ and last contour point at ①;
13   LP PA–60                          absolute and incremental programming
14   LP PA+240
15   LP PA+180

16   LP PR+60 PA+170 R0 FMAX ............................ Retract the tool and end the program
17   L Z+100 FMAX M2
18   END PGM 360529 MM
```

Circular path CP around pole CC

The polar coordinate radius is also the radius of the arc. It is already defined by the distance from the pole CC to the starting point Ⓢ.

Input

- Polar coordinate angle PA for arc end point
- Direction of rotation DR

- For incremental values, enter the same sign for DR and PA.
- You can enter values from –5400° to +5400° for PA.

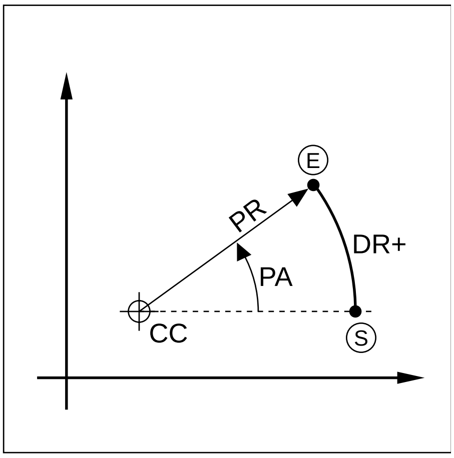


Fig. 5.37: Circular path around a pole

COORDINATES?	
P	Select polar coordinates.

POLAR COORDINATES ANGLE PA?	
e.g. 1 0 ENT	Enter the angle PA of the arc end point PA = 10°.

ROTATION CLOCKWISE: DR–?	
+/– ENT	Set the direction of rotation for the tool path, for example negative for clockwise rotation.

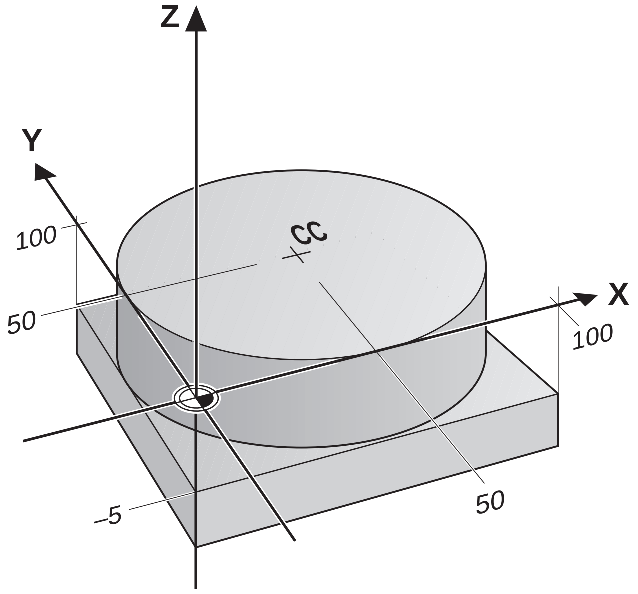
If necessary, enter also:

Radius compensation
Feed rate
Miscellaneous function

Resulting NC block: CP PA+10 DR–

Example exercise: Milling a full circle

Circle radius:	50			mm
Circle center coordinates:	X	=	50	mm
	Y	=	50	mm
Milling depth:	Z	=	– 5	mm
Tool radius:	R	=	15	mm



Part program

```
0    BEGIN PGM 360531 MM           ⎫
1    BLK FORM 0.1 Z X+0 Y+0 Z–20   ⎪
2    BLK FORM 0.2 X+100 Y+100 Z+0  ⎪
3    TOOL DEF 1 L+0 R+15           ⎪
4    TOOL CALL 1 Z S1000           ⎬ General data and first contour point
5    CC X+50 Y+50                  ⎪
6    L Z+100 R0 FMAX M6            ⎪
7    LP PR+70 PA+280 FMAX          ⎪
8    L Z–5 FMAX M3                 ⎪
9    LP PR+50 PA–90 RL F100        ⎭

10   RND R10 ......................... Smooth approach
11   CP PA+270 DR– ................... Circle to end point PA = 270°, negative direction of rotation
12   RND R10 ......................... Smooth departure

13   LP PR+70 PA–110 R0 FMAX       ⎫
14   L Z+100 FMAX M2               ⎬ Retract tool and end program
15   END PGM 360531 MM             ⎭
```

Circular path CTP with tangential connection

The tool moves on a circular path, starting tangentially (at ②) from a preceding contour element (① to ②).

Input:

- Polar coordinate angle PA of the arc end point Ⓔ
- Polar coordinate radius PR of the arc end point Ⓔ

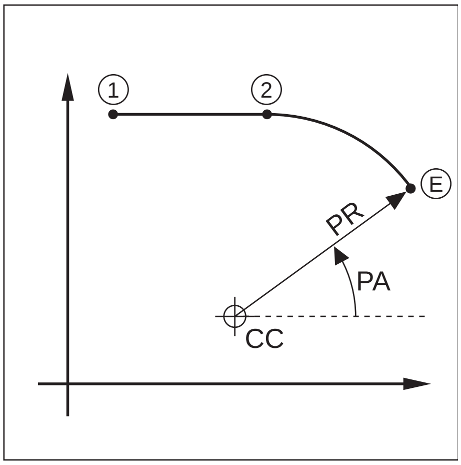


Fig. 5.38: Circular path around a pole, tangential connection

- The transition points must be defined exactly.
- The pole is not the center of the contour arc.

CT

COORDINATES ?	
P	Select polar coordinates.

POLAR COORDINATES RADIUS PR?	
1 0 ENT	Enter the distance from the pole to the arc end point, for example PR = 10 mm.

POLAR COORDINATES ANGLE PA ?	
8 0 ENT	Enter the angle from the reference axis to PR, for example PA = 80°.

If necessary, enter also:

Radius compensation
Feed rate
Miscellaneous function

Resulting NC block: CTP PR +10 PA +80

Helical interpolation

A helix is the combination of a circular movement in a main plane and a linear movement perpendicular to the plane.

A helix is programmed only in polar coordinates.

Applications:

You can use helical interpolation with form cutters to machine:

- Large-diameter internal and external threads
- Lubrication grooves

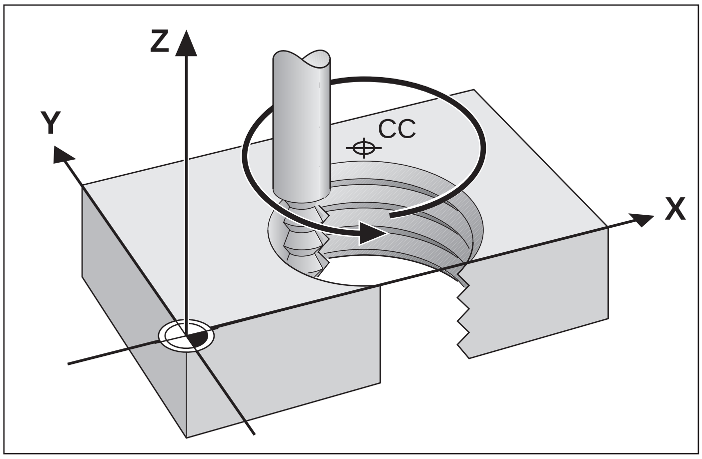


Fig. 5. 39: Helix: a combination of circular and linear paths

Input

- Total incremental angle of tool traverse on the helix
- Total height of the helix

Input angle

Calculate the incremental polar coordinate angle IPA as follows:
IPA = n · 360°; where
n = number of revolutions of the helical path.
For IPA you can enter any value from –5400° to +5400° (n = ±15).

Input height

Enter the helix height H in the tool axis. The height is calculated as:
H = n x P,
n = number of thread revolutions
P = thread pitch

Radius compensation

Enter the radius compensation for the helix according to the table at right.

Internal thread	Work direction	Rotation	Radius comp.
Right-hand	Z+	DR+	RL
Left-hand	Z+	DR-	RR
Right-hand	Z-	DR-	RR
Left-hand	Z-	DR+	RL

External thread	Work direction	Rotation	Radius comp.
Right-hand	Z+	DR+	RR
Left-hand	Z+	DR-	RL
Right-hand	Z-	DR-	RL
Left-hand	Z-	DR+	RR

Fig. 5.40: The shape of the helix determines the direction of rotation and the radius compensation

To program a helix:

COORDINATES ?	
P	Select polar coordinates.

POLAR COORDINATES ANGLE PA ?	
I	Enter PA incrementally.
e.g. 1 0 8 0	Enter the total angle of tool traverse along the helix, for example PA = 1080°.

e.g. Z	Enter the tool axis; for example Z.

COORDINATES ?	
If necessary I	Identify the height entry as incremental.
e.g. 5	Enter the height H of the helix, for example 5 mm.
ENT	Terminate coordinate input.

ROTATION CLOCKWISE: DR-?	
1 x +/– or 2 x +/– ENT	Clockwise helix: DR– or counterclockwise: DR+

RADIUS COMP.: RL/RR/NO COMP.?	
RL or RR	Enter radius compensation according to the table.

If necessary, enter also:

Feed rate
Miscellaneous function

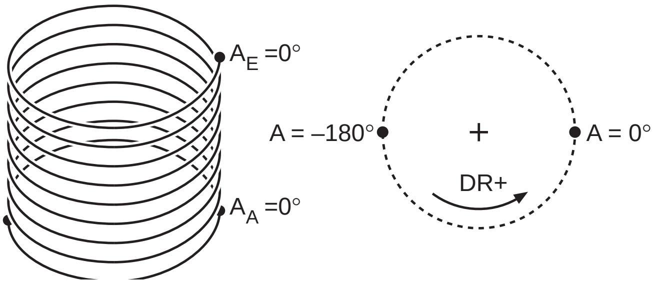
Resulting NC block: CP IPA+1080 IZ+5 DR–RL

Example for exercise: Tapping

Given Data

Thread:
Right-hand internal thread M64 x 1.5

Pitch P:	1.5 mm
Start angle A_S:	0°
End angle A_E:	360° = 0° at Z_E = 0
Thread revolutions n_T:	8
Thread overrun	
• at start of thread n_S:	0.5
• at end of thread n_E:	0.5
Number of cuts:	1



Calculating the input values

- Total height H: $H = P \cdot n$
  $P = 1.5$ mm
  $n = n_T + n_S + n_E = 8 + 0.5 + 0.5 = 9$
  $H = 1.5 \text{ mm} \cdot 9 = 13.5$ mm
- Incremental polar coordinate angle IPA:
  $IPA = n \cdot 360°$
  n = 9 (see total height H)
  $IPA = 360° \cdot 9 = 3240°$
- Start angle A_S with thread overrun n_S
  $n_S = 0.5$
  n = 1 = 360°, n = 0.5 = 180° (half a revolution)
  The starting angle of the helix is advanced by 180°. With positive rotation this means that A_S with $n_S = A_S - 180° = -180°$
- Starting coordinate: $Z = P \cdot (n_T + n_S) = -1.5 \cdot 8.5 \text{ mm} = -12.75$

Note:

- The thread is being cut in an upward direction towards Z_E = 0; therefore Z_S is negative.

Part program

```
 0  BEGIN PGM 360535 MM
 1  BLK FORM 0.1 Z X+0 Y+0 Z–20
 2  BLK FORM 0.2 X+100 Y+100 Z+0
 3  TOOL DEF 1 L+0 R+5
 4  TOOL CALL 1 Z S 1500
 5  L Z+100 R0 FMAX M6
 6  L X+50 Y+50 FMAX
 7  CC
 8  L Z–12.75 R0 FMAX M3
 9  LP PR+32 PA-180 RL F100
10  CP IPA +3240 IZ+13.5 DR+ RL F200
11  L X+50 Y+50 R0
12  L Z+100 FMAX M2
13  END PGM 360535 MM
```

5.6 M-Functions for Contouring Behavior and Coordinate Data

The following miscellaneous functions enable you to change the TNC's standard contouring behavior in certain situations, such as:

- Smoothing corners
- Machining small contour steps
- Machining open contour corners
- Entering machine-referenced coordinates

Smoothing corners: M90

Standard behavior – without M90

At angular transitions such as internal corners and contours without radius compensation (i.e. with R0), the TNC stops the axes briefly.

Advantages:

- Reduced wear on the machine
- High definition of corners

Note:

In program blocks with radius compensation (RR/RL), at external corners the TNC automatically inserts a transition arc.

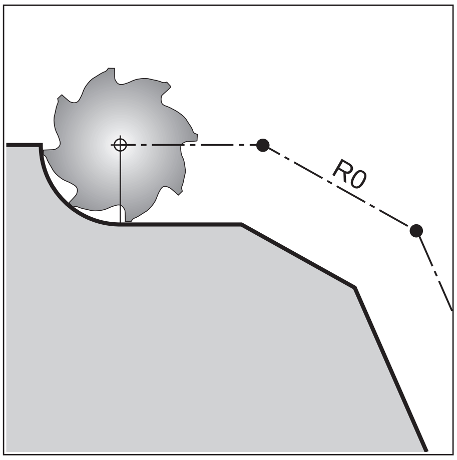


Fig. 5.41: Standard contouring behavior with R0 and without M90

Smoothing corners with M90

The tool moves around corners at constant speed.

Advantages:

- Provides a smoother, more continuous surface
- Reduces machining time

Example application:
Surfaces consisting of several straight line elements.

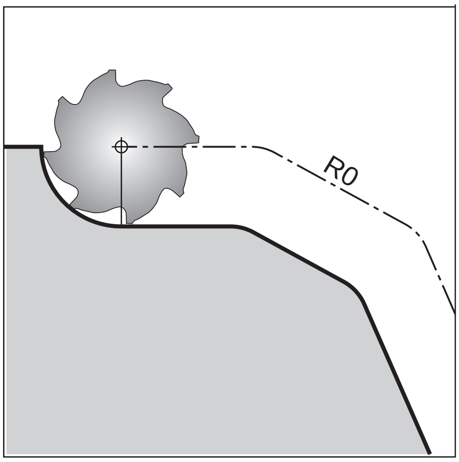


Fig. 5.42: Contouring behavior with R0 and M90

Duration of effect

The miscellaneous function M90 is effective only in the blocks in which it is programmed. Operation with servo lag must be active.

A limit value can be set in machine parameter MP7460 (see page 12-9) below which the tool will move at constant leed rate (valid for operation both with servo lag and with feed precontrol). This value is valid regardiess of M90.

Machining small contour steps: M97

Standard behavior – without M97

The TNC inserts a transition arc at outside corners. At very short contour steps this would cause the tool to damage the contour. In such cases the TNC interrupts the program run and shows the error message TOOL RADIUS TOO LARGE.

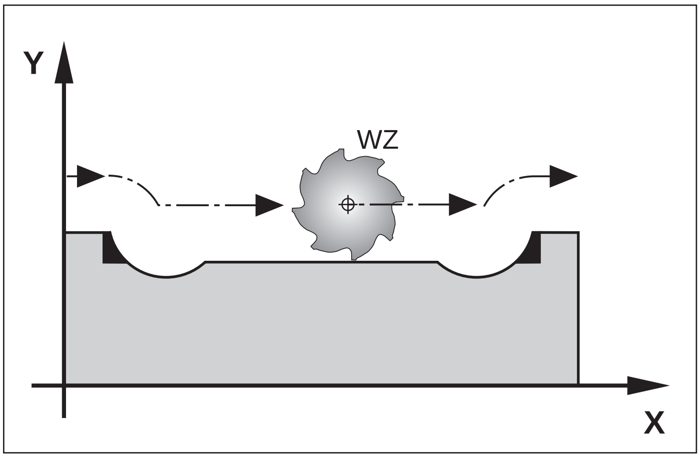


Fig. 5.43: Standard behavior without M97 if the block were to be executed as programmed

Machining contour steps with M97

The TNC calculates the contour intersection Ⓢ (see figure) of the contour elements – as at inside corners – and moves the tool over this point. M97 is programmed in the same block as the outside corner point.

Duration of effect

The miscellaneous function M97 is effective only in the blocks in which it is programmed.

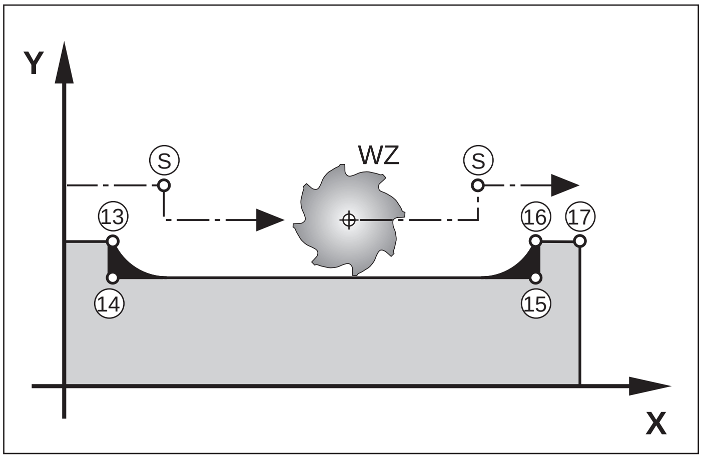


Fig. 5.44: Contouring behavior with M97

A contour machined with M97 is less complete than one without. You may wish to rework the contour with a smaller tool.

Program example

```
.
.
.
5     TOOL DEF L ... R+20 ......................................... Large tool radius
.
.
.
13    L X ... Y ... R.. F .. M97 ...................................... Move to contour point 13
14    L IY–0.5 .... R .. F.. ............................................ Machine the small contour step 13 - 14
15    L IX+100 ... ......................................................... Move to contour point 15
16    L IY+0.5 ... R .. F.. M97 ..................................... Machine the small contour step 15 - 16
17    L X .. Y ... ........................................................... Move to contour point 17
.
.
.
```

The outer corners are programmed in blocks 13 and 16: these are the blocks in which you program M97.

Machining open contours: M98

Standard behavior – without M98

The TNC calculates the intersections Ⓢ of the radius-compensated tool paths and changes traverse direction at these points. If the corners are open on one side, however, machining is incomplete.

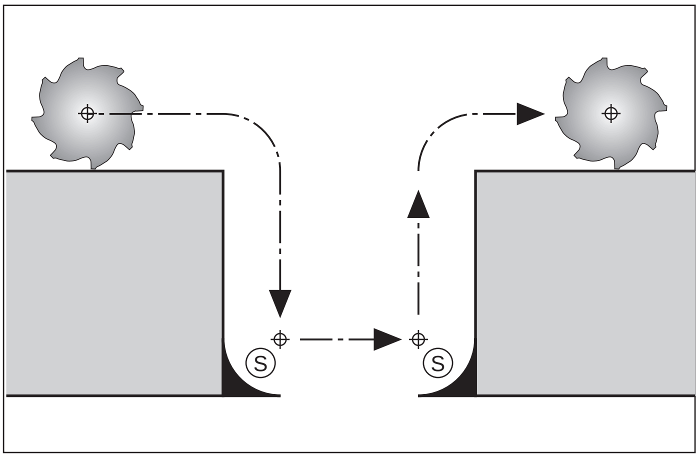


Fig. 5.45: Tool path without M98

Machining open corners with M98

With the miscellaneous function M98 the TNC temporarily suspends radius compensation to ensure that both corners are completely machined.

Duration of effect

The miscellaneous function M98 is effective only in the blocks in which it is programmed.

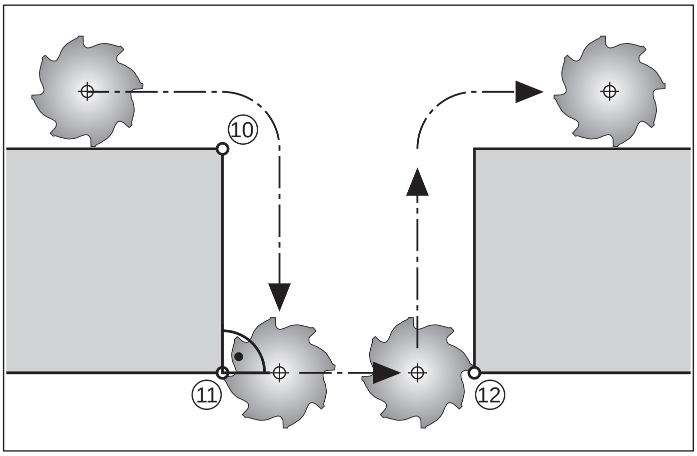


Fig. 5.46: Tool path with M98

Programming example

```
 .
 .
 .
10   L X ... Y ... RL F .. ............................................. Move to contour point 10
11   L X .. IY–.. ... M98 ............................................ Move to contour point 11
12   L IX + .. ... ....................................................... Move to contour point 12
 .
 .
 .
```

Programming machine-reference coordinates: M91/M92

Standard behavior

Coordinates are referenced to the workpiece datum (see page 1-7).

Scale reference point

The position feedback scales are provided with one or more reference marks. Reference marks are used to indicate the position of the scale reference point. If the scale has only one reference mark, its position is the scale reference point. If the scale has several – distance-coded – reference marks, then the scale reference point is indicated by the left-most reference mark (at the beginning of the measuring range).

Machine zero — miscellaneous function M91

The machine zero point is required for the following tasks:

- Defining the limits of traverse (software limit switches)
- Moving to machine-reference positions (e.g. tool-change position)
- Setting the workpiece datum

Machine zero is identical with the scale reference point.

If you want the coordinates in a positioning block to be reference to the machine zero point, end the block with the miscellaneous function M91.

Coordinates that are referenced to the machine zero point are indicated in the display with REF.

Additional machine datum M92

In addition to the machine zero point, the machine tool builder can define another machine-reference position, the machine datum.

The machine tool builder defines the distance for each axis from the machine zero to the machine datum.

If you want the coordinates in a positioning block to be referenced to the machine datum, end the block with the miscellaneous function M92.

Workpiece datum

The user enters the coordinates of the datum for workpiece machining in the MANUAL OPERATION mode (see page 2-7).

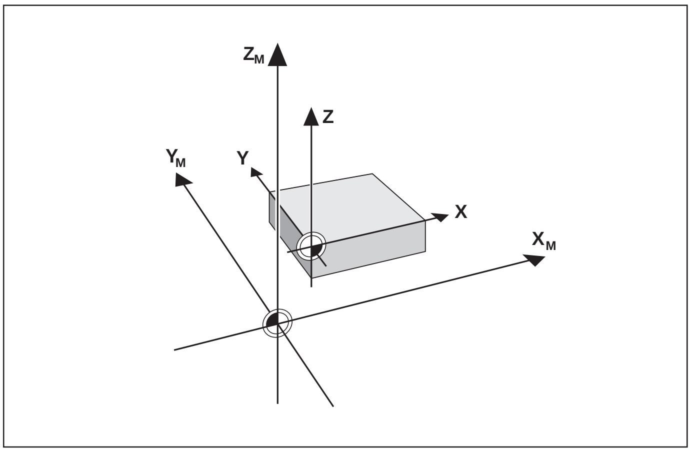


Fig. 5.44: Machine datum ⊕ and workpiece datum ⊕

5.7 Positioning with Manual Data Input (MDI)

In the POSITIONING WITH MANUAL DATA INPUT mode you can enter and execute single-axis positioning blocks. The entered positioning blocks are not stored.

Application examples:

- Pre-positioning
- Face milling

▶ **POSITIONING MANUAL DATA INPUT**

Select the MDI operating mode.

▼

Press an orange axis-selection key and enter a single-axis positioning block.

▼

BLOCK COMPLETE

I	Execute the block.

Application example

To remove workpiece misalignment on a rotary table

Preparation:
Perform a basic rotation with the 3D touch probe system; write down the ROTATION ANGLE and cancel the basic rotation again.

- Switch modes of operation

▶ **POSITIONING WITH MANUAL DATA INPUT**

Select the positioning with MDI mode of operation.

- Program the desired rotation

IV ▶ **COORDINATES?**

- Enter the ROTATION ANGLE that you previously wrote down.
- Enter the FEED RATE.

▼

END □	Conclude the block.

▼

BLOCK COMPLETE

I	The table rotates to correct the misalignment.

Subprograms and program section repeats enable you to program a machining sequence once and then run it as often as you wish.

Labels

Subprograms and program section repeats are marked by labels.

A label carries a number from 0 to 254. Each label number (except 0) can only appear once in a program. Labels are assigned with the command LABEL SET.

LABEL 0 marks the end of a subprogram.

6.1 Subprograms

Principle

The program is executed up to the block in which the subprogram is called with CALL LBL (①).

Then the subprogram is executed from beginning to end (LBL 0) (②).

Finally, the main program is resumed from the block after the subprogram call (③).

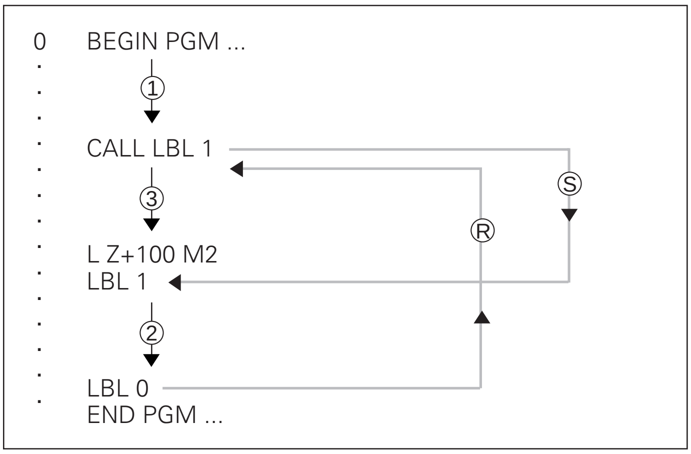


Fig. 6.1: Flow diagram for a subprogram.
Ⓢ = jump, Ⓡ = return jump

Operating limits

- One main program can contain to 254 subprograms.
- Subprograms can be called in any sequence and as often as desired.
- A subprogram cannot call itself.
- Subprograms should be located at the end of the main program (after the block with M2 or M30).
- If subprograms are located in the program before the block with M02 or M30, they will be executed at least once even without being called.

Programming and calling subprograms

To mark the beginning of the subprogram:

LBL SET ▶ **LABEL NUMBER?**

e.g. 5 ENT	The subprogram begins with label number 5.

Resulting NC block: LBL 5

To mark the end of the subprogram:

A subprogram must always end with label number 0.

LBL SET ▶ **LABEL NUMBER?**

0 ENT	End of subprogram.

Resulting NC block: LBL 0

To call the subprogram:

A subprogram is called with its label number.

LBL CALL ▶ **LABEL NUMBER?**

e.g. 5 ENT	Calls the subprogram following LBL 5.

▼

REPEAT REP?

NO ENT	Program section is subprogram: no repetitions.

Resulting NC block: CALL LBL 5

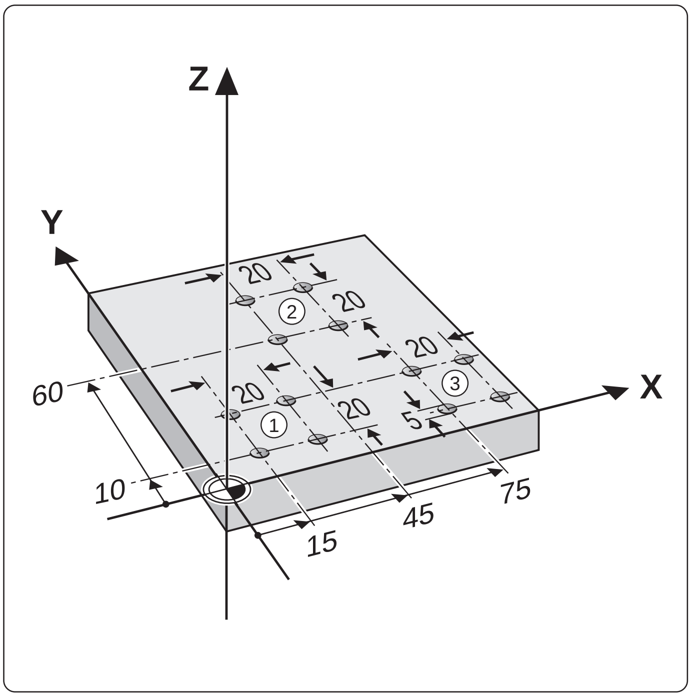
The command CALL LBL 0 is not allowed because label 0 can only be used to mark the end of a subprogram.

Example for exercise: Group of four holes at three different locations

The holes are drilled with cycle 1 PECK DRILLING. You enter the setup clearance, feed rate, drilling feed rate etc. once in the cycle. You can then call the cycle with the miscellaneous function M99 (see page 8-3).

Coordinates of the first hole in each group:

Group 1	X = 15 mm	Y = 10 mm
Group 2	X = 45 mm	Y = 60 mm
Group 3	X = 75 mm	Y = 10 mm
Spacing of holes:		
	IX = 20 mm	IY = 20 mm
Total hole depth (DEPTH):		Z = 10 mm
Hole diameter:	Ø	= 5 mm



Part Program

```
0    BEGIN PGM 360064 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+2.5
4    TOOL CALL 1 Z S1000
5    CYCL DEF 1.0 PECK DRILLING  }
6    CYCL DEF 1.1 SETUP –2       }
7    CYCL DEF 1.2 DEPTH –10      } Cycle definition PECKING (see page 8-5)
8    CYCL DEF 1.3 PECKG –10      }
9    CYCL DEF 1.4 DWELL 0        }
10   CYCL DEF 1.5 F100           }

11   L Z+100 FMAX
12   L X+15 Y+10 R0 FMAX M6 ............................... Move to hole group 1, insert tool
13   L Z+2 FMAX M3 ........................................ Pre-position in the infeed axis
14   CALL LBL 1 ........................................... Subprogram call (with block 14 the subprogram is executed
                                                            once)
15   L X+45 Y+60 FMAX ..................................... Move to hole group 2
16   CALL LBL 1 ........................................... Subprogram call
17   L X+75 Y+10 FMAX ..................................... Move to hole group 3
18   CALL LBL 1 ........................................... Subprogram call
19   L Z+100 FMAX M2 ...................................... Retract tool; return to program (M2):
                                                            The subprogram is entered after M2
20   LBL 1 ................................................ Beginning of subprogram
21   L M99 ................................................ Execute peck drilling cycle for first hole in group
22   L IX+20 FMAX M99 ..................................... Move to position for second hole and drill
23   L IY+20 FMAX M99 ..................................... Move to position for third hole and drill
24   L IX–20 FMAX M99 ..................................... Move to position for fourth hole and drill
25   LBL 0 ................................................ End of subprogram
26   END PGM 360064 MM
```

6.2 Program Section Repeats

As with subprograms, program section repeats are marked with labels.

Principle

The program is executed up to the end of the labelled program section (block with CALL LBL) (①, ②).

Then the program section between the called LBL and the label call is repeated the number of times entered after REP in the CALL LBL command (③, ④).

After the last repetition, the program is resumed (⑤).

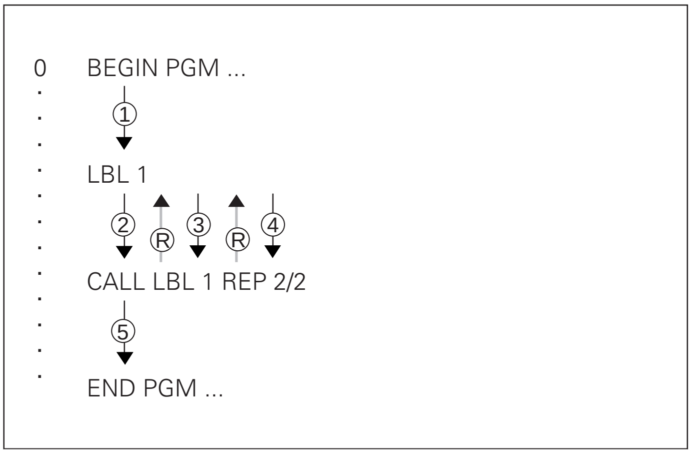


Fig. 6.2: Flow diagram with program section repeats, Ⓡ= return jump

Programming notes

- A program section can be repeated up to 65 534 times in succession.
- The number behind the slash after REP indicates the number of remaining repetitions.
- The total number of times the program section will be carried out is always one more than the programmed number of repetitions.

Programming and calling a program section repeat

Mark the beginning:

LBL SET ▶

LABEL NUMBER?	
e.g. 7 ENT	Repeat the program section beginning with LABEL 7.

Resulting NC block: LBL 7

Number of repetitions

Enter the number of repetitions in the block which calls the label. This block also identifies the end of the program section.

LBL CALL ▶

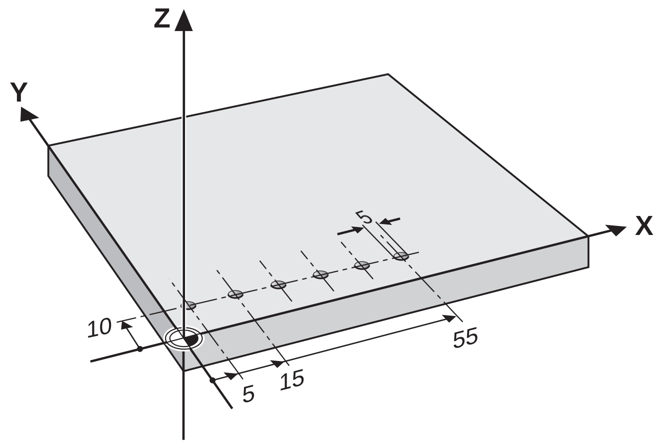
LABEL NUMBER?	
e.g. 7 ENT	Execute the program section beginning with LABEL 7.

REPEAT REP?	
e.g. 1 0 ENT	Repeat the program section from LBL 7 to this block 10 times. The program section will therefore be executed a total of 11 times.

Resulting NC block: CALL LBL 7 REP 10/10

Example for exercise: Row of holes parallel to X-axis

Coordinates of 1st hole:	X = 5 mm	Y =10 mm
Spacing between holes:	IX =15 mm	
No. of holes:	N = 6	
Hole depth:	Z = 10	
Hole diameter:	Ø = 5 mm	



Part Program

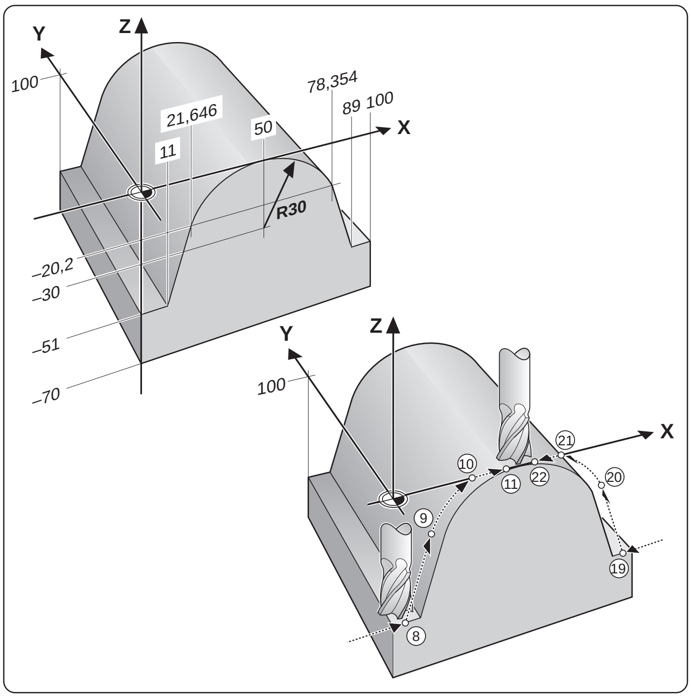
```
0    BEGIN PGM 360066 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2  X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+2,5
4    TOOL CALL 1 Z S1000
5    L Z+100 R0 FMAX M6
6    L X–10 Y+10 Z+2 FMAX M3 .............................. Pre-position in negative X-direction
7    LBL 1 .................................................................. Beginning of program section to be repeated
8    L IX+15 FMAX
9    L Z–10 F100
10   L Z+2 FMAX ....................................................... Move to hole position, drill, retract
11   CALL LBL 1 REP 5/5........................................... Call LABEL 1; repeat program section between blocks 7 and
                                                                  11 five times (for 6 holes!)
12   L Z+100 R0 FMAX M2
13   END PGM 360066 MM
```

Example for exercise: Milling with program section repeat without radius compensation

Machining sequence

- Upward milling direction
- Machine the area from X=0 to 50 mm (program all X-coordinates with the tool radius subtracted) and from Y=0 to 100 mm: LBL 1
- Machine the area from X=50 to X=100 mm (program all X-coordinates with the tool radius added) and from Y=0 to 100 mm: LBL 2
- After each upward pass, the tool is moved by an increment of +2.5 mm in the Y-axis.

The illustration to the right shows the block numbers containing the end points of the corresponding contour elements.



Part Program:

```
0    BEGIN PGM 360067 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–70
2    BLK FORM 0.2 X+100 Y+100 Z+0 .................... Note: the blank form has changed
3    TOOL DEF 1 L+0 R+10
4    TOOL CALL 1 Z S1000
5    L X–20 Y–1 R0 FMAX M3

6    LBL 1
7    L Z–51 FMAX
8    L X+1 F100
9    L X+11.646 Z–20.2                  Program section repeat 1: machining from
10   CT X+40 Z+0                        X=0 to 50 mm and Y=0 to 100 mm
11   L X+41
12   L Z+10 FMAX
13   L X–20 IY+2.5
14   CALL LBL 1 REP40/40

15   L Z+20 FMAX                        Retract, reposition
16   L X+120 Y–1

17   LBL2
18   L Z–51 FMAX
19   L X+99 F100
20   L X+88.354 Z–20.2                  Program section repeat 2: machining from X=50 to
21   CT X+60 Z+0                        100 mm and Y=0 to 100 mm
22   L X+59
23   L Z+10 FMAX
24   L X+120 IY+2.5
25   CALL LBL 2 REP40/40

26   L Z+100 FMAX M2
27   END PGM 360067 MM
```

6.3 Main Program as Subprogram

Principle

A program is executed until another program is called (block with CALL PGM) (①).

The called program is executed from beginning to end (②).

Execution of the program from which the other program was called is then resumed with the block following the CALL PGM block (③).

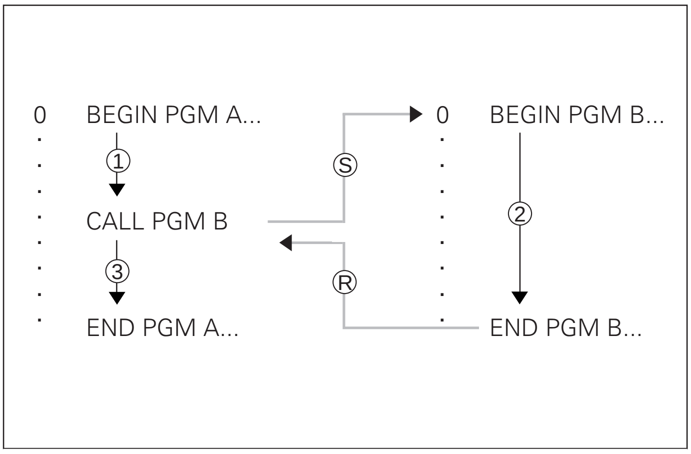


Fig. 6.3: Flow diagram of a main program as subprogram; Ⓢ = jump, Ⓡ = return jump

Operating limits

- Programs called from an external data storage medium (such as a floppy disk) must not contain any subprograms or program section repeats.
- No labels are needed to call main programs as subprograms.
- The called program must not contain the miscellaneous functions M2 or M30.
- The called program must not contain a jump into the calling program.

Calling a main program as a subprogram

PGM CALL ▶ **PROGRAM NUMBER?**

Enter the main program call and the number of the program you want to call.

Resulting NC block: CALL PGM NAME

A main program can also be called with Cycle 12 PGM CALL (see page 8-38).

6.4 Nesting

Subprograms and program section repeats can be nested in the following variations:

- Subprograms in subprograms
- Program section repeats in program section repeats
- Subprograms can be repeated
- Program section repeats can appear in subprograms

Nesting depth

The nesting depth is the number of successive levels for which subprograms or program sections can call further subprograms or program sections.

Maximum nesting depth for subprograms: 8

Maximum nesting depth for calling main programs: 4

Subprogram in a subprogram

Program layout

```
       0   BEGIN PGM 360069 MM
       .     |
       .     ①
       .     ▼
e.g. 17   CALL LBL 1 ............................ Call of subprogram at LBL1
       .     |
       .     ⑤
       .     ▼
e.g. 35   L Z+100 R0 FMAX M2 .................... Last program block of main program (with M2)
      36   LBL 1
       .     |
       .     ②
       .     ▼
e.g. 39   CALL LBL 2
       .     |
       .     ④
       .     ▼
e.g. 45   LBL 0 ................................. End of subprogram 1
      46   LBL 2
       .     |
       .     ③
       .     ▼
e.g. 62   LBL 0 ................................. End of subprogram 2
      63   END PGM 360069 MM ..................... End of main program
```

Subprogram 1 with program call of subprogram 2

Subprogram 2

Sequence of program execution

Step 1: Main program 360069 is executed up to block 17.

Step 2: Subprogram 1 is called and executed up to block 39.

Step 3: Subprogram 2 is called and executed up to block 62. End of subprogram 2 and return to subprogram from which it was called.

Step 4: Subprogram 1 is executed from block 40 to block 45. End of subprogram 1 and return to main program 360069.

Step 5: Main program 360069 is executed from block 18 to block 35. Return jump to block 1 and program end.

A subprogramm ending with LBL 0 must not be nested in another subprogram!

Example for exercise: Group of four holes at three positions (see page 6-4), but with three different tools

Machining sequence:

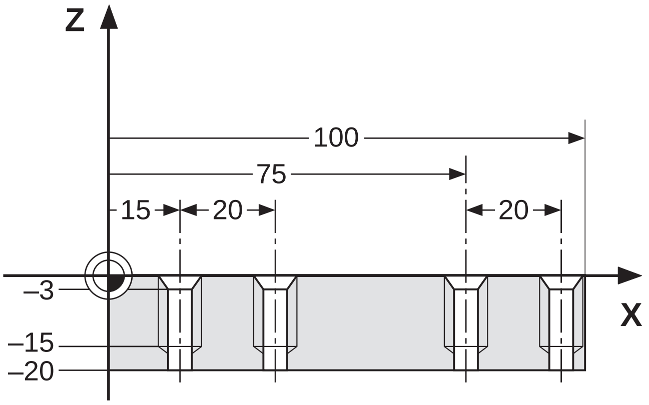
Countersinking – Drilling – Tapping

The drilling operation is programmed with cycle 1: PECK DRILLING (see page 8-5) and cycle 2: TAPPING (see page 8-7). The groups of holes are approached in one subprogram, and the machining is performed in a second subprogram.

Coordinates of the first hole in each group:

1	X =	15 mm	Y =	10 mm
2	X =	45 mm	Y =	60 mm
3	X =	75 mm	Y =	10 mm

Spacing between holes:	IX = 20 mm	IY = 20 mm
Hole data:		
Countersinking	ZS = 3 mm	Ø = 7 mm
Drilling	ZT = 15 mm	Ø = 5 mm
Tapping	ZG = 10 mm	Ø = 6 mm



Part program

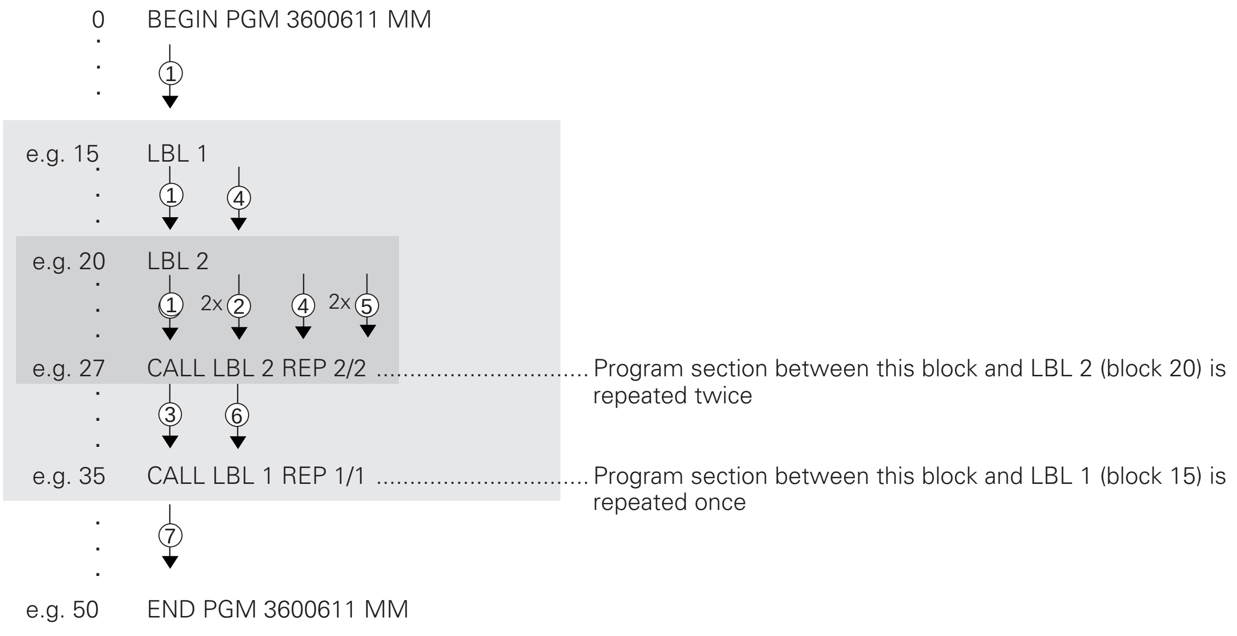
```
0   BEGIN PGM 3600610 MM
1   BLK FORM 0.1 Z X+0 Y+0 Z–20
2   BLK FORM 0.2 X+100 Y+100 Z+0
3   TOOL DEF 25 L+0 R+2,5
4   TOOL DEF 30 L+0 R+3        Tool definition for countersinking (T35), peck drilling (T25) and
5   TOOL DEF 35 L+0 R+3.5      tapping (T30)
6   CYCL DEF 1.0 PECKING
7   CYCL DEF 1.1 SETUP–2
8   CYCL DEF 1.2 DEPTH–3
9   CYCL DEF 1.3 PECKG–3       Cycle definition PECKING for countersinking
10  CYCL DEF 1.4 DWELL0
11  CYCL DEF 1.5 F100
12  TOOL CALL 35 Z S 500
13  CALL LBL 1 ........................................................ Call of subprogram 1
14  CYCL DEF 1.0 PECKING
15  CYCL DEF 1.1 SETUP–2
16  CYCL DEF 1.2 DEPTH–25
17  CYCL DEF 1.3 DEPTH–6       Cycle definition PECKING
18  CYCL DEF 1.4 DWELL0
19  CYCL DEF 1.5 F50
20  TOOL CALL 25 Z S 1000
21  CALL LBL 1 ........................................................ Call of subprogram 1
22  CYCL DEF 2.0 TAPPING
23  CYCL DEF 2.1 SETUP–2
24  CYCL DEF 2.2 DEPTH–15      Cycle definition TAPPING
25  CYCL DEF 2.3 DWELL0
26  CYCL DEF 2.4 F100
27  TOOL CALL 30 Z S 250
28  CALL LBL 1 ........................................................ Call of subprogram 1
29  L Z+100 R0 FMAX M2 ...................................... Last program block, return jump
```

Continued...

```
30  LBL 1
31  L X+15 Y+10 R0 FMAX M3
32  L Z+2 FMAX
33  CALL LBL 2
34  L X+45 Y+60 FMAX              Move to first hole in each group, then call subprogram 2
35  CALL LBL 2
36  L X+75 Y+10 FMAX
37  CALL LBL 2
38  LBL 0

39  LBL 2
40  L M99
41  L IX+20 F9999 M99             Machine first hole, then move to and machine the other holes
42  L IY+20 M99                   using the same cycle
43  L IX–20 M99
44  LBL 0

45  END PGM 3600610 MM
```

Repeating program section repeats

Program layout



0 BEGIN PGM 3600611 MM

e.g. 15 LBL 1

e.g. 20 LBL 2

e.g. 27 CALL LBL 2 REP 2/2 .... Program section between this block and LBL 2 (block 20) is repeated twice

e.g. 35 CALL LBL 1 REP 1/1 .... Program section between this block and LBL 1 (block 15) is repeated once

e.g. 50 END PGM 3600611 MM

Sequence of program execution

Step 1: Main program 3600611 is executed up to block 27.

Step 2: Program section between block 27 and block 20 is repeated twice.

Step 3: Main program 3600611 is executed from block 28 to block 35.

Step 4: Program section between block 35 and block 15 is repeated once.

Step 5: Repetition of step 2 within step ④.

Step 6: Repetition of step 3 within step④.

Step 7: Main program 3600611 is executed from block 36 to block 50. End of program.

Repeating subprograms

Program layout

```
        0   BEGIN PGM 3600612 MM
        .
        .          ①
        .
e.g.   10   LBL 1
       11   CALL LBL 2 ...................2x ③.............. Subprogram call
       12   CALL LBL 1 REP 2/2                        ..... Program section repeat
        .
        .     ④
        .
e.g.   19   L Z+100 R0 FMAX M2 ............................ Last program block of main program with M2
       20   LBL 2 ......................................... Beginning of subprogram
        .
        .     ②
        .
e.g.   28   LBL 0 ......................................... End of subprogram
       29   END PGM 3600612 MM ............................ End of main program
```

Sequence of program execution

Step 1: Main program 3600612 is executed to block 11.

Step 2: Subprogram 2 is called and executed.

Step 3: Program section between block 12 and block 10 is repeated twice: subprogram 2 is repeated twice.

Step 4: Main program 3600612 is executed from block 13 to block 19. End of program.

Q Parameters are used for:

- **Programming families of parts**
- **Defining contours through mathematical functions**

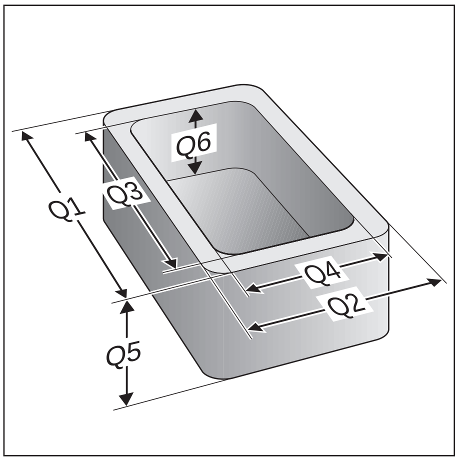


Fig. 7.1: Q parameters as variables

A **family of parts** can be programmed in the TNC in a **single part program**. You do this by entering variables — called Q parameters — instead of numerical values.

Q parameters can represent for example:

- Coordinate values
- Feed rates
- Spindle speeds
- Cycle data

A Q parameter is designated by the letter Q and a number between 0 and 113.

Q parameters also enable you to program **contours** that are defined through **mathematical functions**.

With Q parameters you can make the execution of machining steps dependent on **logical conditions**.

Q parameters and numerical values can also be **mixed** within a program.

The TNC automatically assigns data to some Q parameters. For example, parameter Q108 is assigned the current tool radius. You will find a list of these parameters in chapter 12..

7.1 Part Families — Q Parameters Instead of Numerical Values

The Q parameter function FN0: ASSIGN is used for assigning numerical values to Q parameters.
Example: Q10 = 25

This enables you to enter variable Q parameters in the program instead of numerical values.
Example: L X + Q10 (corresponds to L X + 25)

For part families, the characteristic workpiece dimensions can be programmed as Q parameters. Each of these parameters is then assigned a different value when the parts are machined.

Example

Cylinder with Q parameters

Cylinder radius	R	=	Q1
Cylinder height	H	=	Q2
Cylinder Z1:	Q1	=	+30
	Q2	=	+10
Cylinder Z2:	Q1	=	+10
	Q2	=	+50

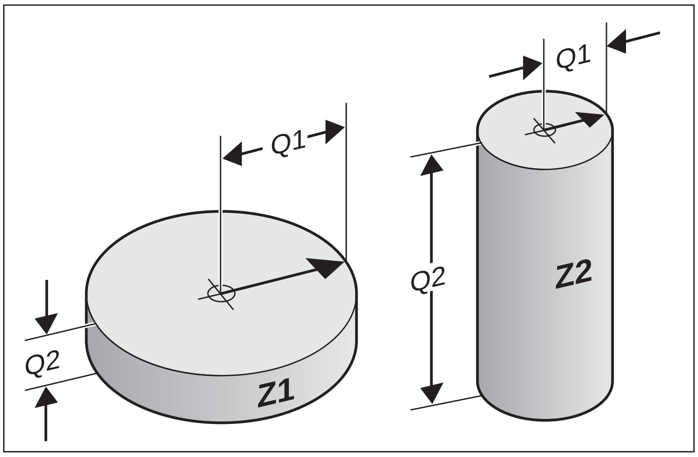


Fig. 7.2: Workpiece dimensions as Q parameters

To assign numerical values to Q parameters:

Q DEF ▶ **FN0: ASSIGN**

ENT	Open a new block with the function FN0: ASSIGN.

PARAMETER NUMBER FOR RESULT?

e.g. 5 ENT	Enter Q parameter number.

FIRST VALUE / PARAMETER?

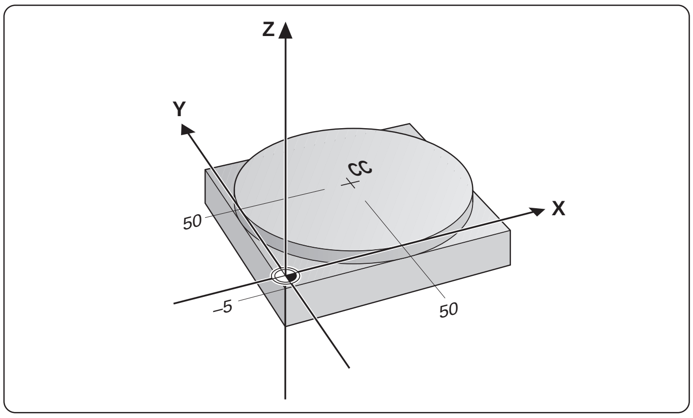
e.g. 6 ENT	Enter value or another Q parameter whose value is to be assigned to Q5.

Resulting NC block: FN0: Q5 = 6

The value to the right of the equal sign is assigned to the Q parameter to the left.

Example for exercise: Full circle

Circle center CC:	X	=	50 mm
	Y	=	50 mm
Beginning and end of circular arc C:	X	=	50 mm
	Y	=	0 mm
Milling depth:	Z	=	–5 mm
Tool radius:	R	=	15 mm



Part program without Q parameters

```
0     BEGIN 360074 MM ........................................... Start of program
1     BLK FORM 0.1 Z X+0 Y+0 Z–20 ........................ Blank form definition
2     BLK FORM 0.2 X+100 Y+100 Z+0
3     TOOL DEF 6 L+0 R+15 ..................................... Tool definition
4     TOOL CALL 6 Z S500 ....................................... Tool call
5     CC X+50 Y+50 .................................................. Coordinates of circle center CC
6     L Z+100 R0 FMAX M6 ...................................... Insert tool
7     L X+30 Y–20 FMAX .......................................... Pre-position tool
8     L Z–5 FMAX M3
9     L X+50 Y+0 RR F100 ........................................ Move to first compensation point with radius compensation
10    C X+50 Y+0 DR+ .............................................. Mill circular arc C around circle center CC; coordinates of end
                                                                  point: X = +50 mm and Y = 0; positive direction of rotation
11    L X+70 Y–20 R0 FMAX
12    L Z+100 FMAX M2
13    END PGM 360074 MM ..................................... Retract tool and end program
```

Part program with Q parameters

```
0     BEGIN PGM 3600741 MM
1     FN 0: Q1 = +100 ............................................... Clearance height
2     FN 0: Q2 = +30 ................................................. Start pos. X
3     FN 0: Q3 = –20 .................................................. Start-End pos. Y
4     FN 0: Q4 = +70 ................................................. End pos. X
5     FN 0: Q5 = –5 .................................................... Milling depth
6     FN 0: Q6 = +50 ................................................. Center point X
7     FN 0: Q7 = +50 ................................................. Center point Y
8     FN 0: Q8 = +50 ................................................. Circle starting point X
9     FN 0: Q9 = +0 ................................................... Circle starting point Y
10    FN 0: Q10 = +0 ................................................. Tool length L
11    FN 0: Q11 = +15 ............................................... Tool radius R
12    FN 0: Q20 = +100 ............................................. Milling feed rate F
```

Blocks 1 to 12:
Assign numerical values to the Q parameters

```
13    BLK FORM 0.1.Z X+0 Y+0 Z–20
14    BLK FORM 0.2 X+100 Y+100 Z+0
15    TOOL DEF 1 L+Q10 R+Q11
16    TOOL CALL 1 Z S500
17    CC X+Q6 Y+Q7
18    L Z+Q1 R0 FMAX M6
19    L X+Q2 Y+Q3 F MAX
20    L Z+Q5 F MAX M3
21    L X+Q8 Y+Q9 RR FQ20
22    C X+Q8 Y+Q9 DR+
23    L X+Q4 Y+Q3 R0 FMAX
24    L Z+Q1 FMAX M2
25    END PGM 3600741 MM
```

Blocks 13 to 24:
Corresponding to blocks 1 to 12 from program 360074

7.2 Describing Contours Through Mathematical Functions

Overview

The mathematical functions assign the results of one of the following operations to a Q parameter:

FN0: ASSIGN e.g. FN0: Q5 = +60 Assigns a value directly
FN1: ADDITION e.g. FN1: Q1 = –Q2 + –5 Calculates and assigns the sum of two values
FN2: SUBTRACTION e.g. FN2: Q1 = +10 – +5 Calculates and assigns the difference between two values
FN3: MULTIPLICATION e.g. FN3: Q2 = +3 ✻ +3 Calculates and assigns the product of two values
FN4: DIVISION e.g. FN4: Q4 = +8 DIV +Q2 Calculates and assigns the quotient of two values **Note**: Division by 0 is not possible!
FN5: SQUARE ROOT e.g. FN5: Q20 = SQRT 4 Calculates and assigns the square root of a number **Note:** Square root of a negative number is not possible!

The “values” in the overview above can be:

- two numbers
- two Q parameters
- a number and a Q parameter

The Q parameters and numerical values in the equations can be entered with positive or negative signs.

To select a mathematical operation

Q DEF ▶ **FN0: ASSIGN**

GOTO □ 3 ENT or ↓/↑	Select function directly or with arrow keys, e.g. FN3: MULTIPLICATION.

FN3: MULTIPLICATION

ENT	Open a new block with the function FN3: MULTIPLICATION.

Programming example for fundamental operations

Assign the value 10 to parameter Q5, and assign the product of Q5 and 7 to parameter Q12.

Q DEF ▶ **FN 0 : ASSIGN**

ENT	Select Q parameter function FN 0.

PARAMETER NUMBER FOR RESULT?

5 ENT	Enter parameter number, e.g. Q5.

FIRST VALUE / PARAMETER?

1 0 ENT	Assign numerical value to Q5.

Q DEF ▶ **FN 0 : ASSIGN**

GOTO □ 3 ENT	Select Q parameter function FN 0 again.

FN3: MULTIPLICATION

ENT	Select Q parameter function FN3.

PARAMETER NUMBER FOR RESULT?

1 2 ENT	Enter parameter number, for example Q12.

FIRST VALUE / PARAMETER?

Q 5 ENT	Enter Q5 (=10).

SECOND VALUE / PARAMETER?

7 ENT	Enter the value 7.

Resulting NC blocks: FN0: Q5 = +10
*FN3: Q12 = +Q5 * +7*

7.3 Trigonometric Functions

Sine, cosine and tangent are the terms for the ratios of the sides of right triangles. Trigonometric functions simplify many calculations.

For a right triangle,

Sine: $\sin \alpha = a / c$

Cosine: $\cos \alpha = b / c$

Tangent: $\tan \alpha = a / b = \sin \alpha / \cos \alpha$

Where

- c is the side opposite the right angle
- a is the side opposite the angle α
- b is the third side

The angle can be derived from the tangent:

$\alpha = \arctan \alpha = \arctan (a / b) = \arctan (\sin \alpha / \cos \alpha)$

Example:
$a = 10$ mm
$b = 10$ mm
$\alpha = \arctan (a / b) = \arctan 1 = 45°$

Furthermore: $a^2 + b^2 = c^2$ $(a^2 = a \cdot a)$

$c = \sqrt{a^2 + b^2}$

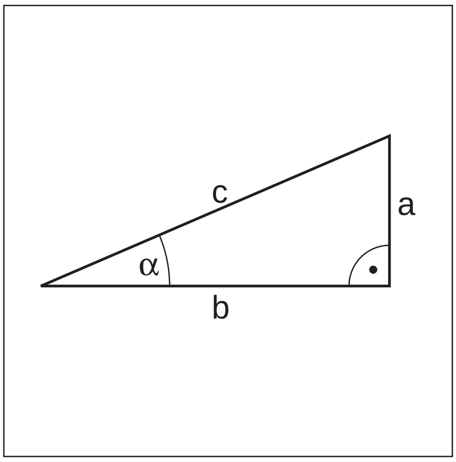


Fig. 7.3: Sides and angles on a right triangle

Overview

FN6: SINE
e.g. FN6: Q20 = SIN –Q5
Calculate sine of an angle in degrees (°) and assign it to a parameter

FN7: COSINE
e.g. FN7: Q21 = COS –Q5
Calculate the cosine of an angle in degrees (°) and assign it to a parameter

FN8: ROOT SUM OF SQUARES
e.g. FN8: Q10 = +5 LEN +4
Take the square root of the sum of two squares, and assign it to a parameter

FN13: ANGLE
e.g. FN13: Q20 = +10 ANG –Q1
Calculate the angle from the arc tangent of two sides or from the sine and cosine of the angle, and assign it to a parameter

7.4 If–Then Operations with Q Parameters

If–Then conditional operations enable the TNC to compare a Q parameter with another Q parameter or with a numerical value.

Jumps

The jump target is specified in the block through a label number. If the programmed condition is true, the TNC continues the program at the specified label; if it is false, the next block is executed.

To jump to another program, you enter a PGM CALL after the block with the target label (see page 6-8).

Abbreviations used:

IF	If
EQU	Equals
NE	Not equal
GT	Greater than
LT	Less than
GOTO	Go to

Overview

FN9: IF EQUAL, JUMP
e.g. FN9: IF +Q1 EQU +Q3 GOTO LBL 5
If the two values or parameters are equal,
jump to the specified label.

FN10: IF NOT EQUAL, JUMP
e.g. FN10: IF +10 NE –Q5 GOTO LBL 10
If the two parameters or values are not equal,
jump to the specified label.

FN11: IF GREATER THAN, JUMP
e.g. FN11: IF +Q1 GT–10 GOTO LBL 5
If the first value or parameter is greater
than the second value or parameter,
jump to the specified label.

FN12: IF LESS THAN, JUMP
e.g. FN12: IF +Q5 LT +0 GOTO LBL 1
If the first value or parameter is less
than the second value or parameter,
jump to the specified label.

Unconditional jumps

Unconditional jumps are jumps which are always executed because the condition is always true.
Example:

FN 9: IF +10 EQU +10 GOTO LBL1

Since it is always true that 10=10, the jump will always be executed.

Program example

When Q5 becomes negative, a jump to program 100 will occur.

```
.
.
.
5    FN0: Q5 = +10 ................................... Assign value (such as 10) to parameter Q5
.
.
.
9    FN 2: Q5 = +Q5–+12 .............................. Reduce the value of Q5
10   FN 12: IF +Q5 LT +0 GOTO LBL 5 .................. If +Q5 is less than 0, jump to label 5
.
.
.
15   LBL 5 ........................................... Label 5
16   PGM CALL 100 .................................... Jump to program 100
.
.
.
```

7.5 Checking and Changing Q Parameters

Q parameters can be checked during program run or during a test run, and changed if necessary.

Preparation:

- A running program must be aborted (e.g. press machine STOP button and STOP key)
- If you are doing a test run, you must interrupt it

To call a Q parameter:

Q ▶

```
Q      =
```

e.g. 1 0 ENT	Select desired parameter (in this example, Q10).

```
Q10 = + 100
```

The TNC displays the current value.

e.g. 0 ENT	Change Q parameter (in this example, Q10 is changed to 0).
ENT	Leave the Q parameter unchanged.

7.6 Output of Q Parameters and Messages

Displaying error messages

With the function FN14:ERROR you can call messages that were preprogrammed by the machine tool builder.

If the TNC encounters a block with FN 14 during a program run or test run, it interrupts the run and displays an error message. The program must then be restarted.

Input example:

FN 14: ERROR = 254

The TNC will display the text of error number 254.

Error number to be entered	Prepared dialog text
0 to 299	ERROR 0 to ERROR 299
300 to 399	PLC ERROR 01 to PLC ERROR 99
400 to 483	DIALOG 1 to 83
484 to 499	USER PARAMETER 15 to 0

Your machine builder may have programmed a text that differs from the above.

Output through an external data interface

The function FN 15: PRINT transmits the values of Q parameters and error messages over the data interface. This enables you to send such data to external devices, for example to a printer.

- FN15: PRINT with numerical values up to 200
  Example: FN15: PRINT 20
  Transmits the corresponding error message (see overview for FN14).
- FN 15: PRINT with Q parameter
  Example: FN15: PRINT Q20
  Transmits the value of the corresponding Q parameter.

Up to six Q parameters and numerical values can be transmitted simultaneously. The TNC separates them with slashes.
Example: FN15: PRINT 1/Q1/2/Q2

Assigning values for the PLC

Function FN19: PLC transmits up to two numerical values for Q parameters to the PLC.

Input increment and unit of measure: 1 μm or 0.001°

Example: FN19: PLC = +10/+Q3

The number 10 corresponds to 10 μm or 0.01°.

7.7 Measuring with the 3D Touch Probe During Program Run

The 3D touch probe can measure positions on a workpiece during program run.

Applications:

- Measuring differences in the height of cast surfaces
- Checking tolerances during machining

To activate the touch probe, press the TOUCH PROBE key. You pre-position the probe, which then automatically probes the specified position. The coordinate measured for the probe point is stored in a Q parameter.

The TNC interrupts the probing process if the probe is not deflected within a certain range (range selected with MP 6130).

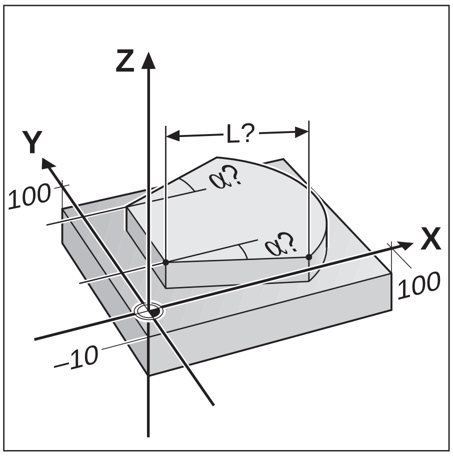


Fig. 7.4: Workpiece dimensions to be measured

To program the use of a touch probe:

TOUCH PROBE

TCH PROBE 0: REF. PLANE	
ENT	Select the touch probe function.

PARAMETER NUMBER FOR RESULT?	
5 ENT	Enter the number of the Q parameter to which the coordinate is to be assigned (in this example, Q5).

PROBING AXIS/PROBING DIRECTION?	
X	Enter the probing axis for the coordinate (in this example, X).
+/– ENT	Select and confirm the probing direction.

POSITION VALUE?	
X 5 Y 0 Z +/– 5	Enter all coordinates of the pre-positioning point values, in this example, X = 5 mm, Y = 0, Z = –5 mm.

ENT	Conclude input.

Resulting NC blocks:

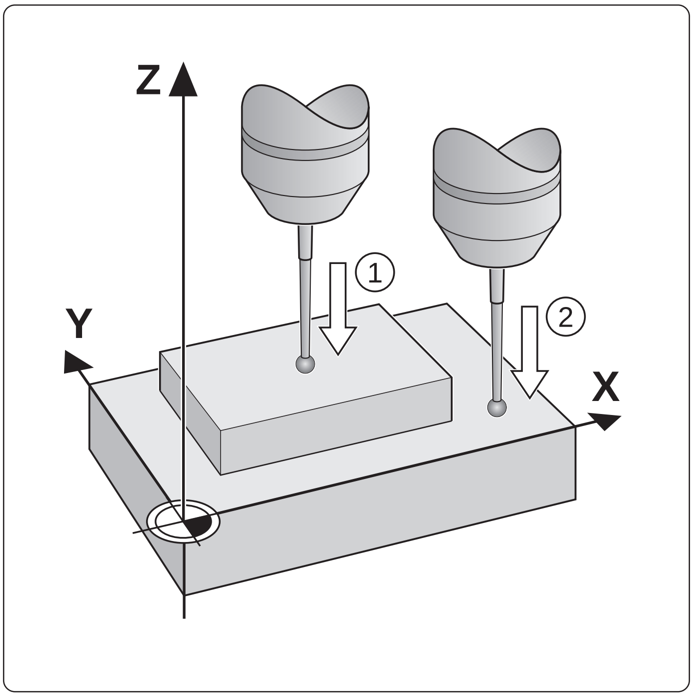
```
TCH   PROBE    0.0     REF. PLANE Q5 X–
TCH   PROBE    0.1     X+5 Y+0 Z–5
```

Pre-position the touch probe manually such that it will not collide with the workpiece when it moves toward the programmed position.

Example for exercise: Measuring the height of an island on a workpiece

Coordinates for pre-positioning the 3D touch probe

Touch point 1:	X	=	+ 20	mm (Q11)
	Y	=	50	mm (Q12)
	Z	=	10	mm (Q13)
Touch point 2:	X	=	+ 50	mm (Q21)
	Y	=	10	mm (Q22)
	Z	=	0	mm (Q23)



Part program:

```
0    BEGIN PGM 3600717 MM          }
1    FN0: Q11 = + 20               }
2    FN0: Q12 = + 50               }  Begin the program; assign the coordinates for pre-
3    FN0: Q13 = + 10               }  positioning the touch probe
4    FN0: Q21 = + 50               }
5    FN0: Q22 = + 10               }
6    FN0: Q23 = + 0                }
7    TOOL CALL 0 Z
8    L Z+100 R0 FMAX M6 ...................................... Insert touch probe
9    TCH PROBE 0.0 REF.PLANE Q10 Z–
10   TCH PROBE 0.1 X+Q11 Y+Q12 Z+Q13 ............ The Z coordinate probed in the negative direction is stored in
                                                  Q10 (1st point)
11   L X+Q21 Y+Q22 ................................................ Auxiliary point for second pre-positioning
12   TCH PROBE 0.0 REF.PLANE Q20 Z–
13   TCH PROBE 0.1 X+Q21 Y+Q22 Z+Q23 ............ The Z coordinate probed in the negative direction is stored in
                                                  Q20 (2nd point)
14   FN2: Q1 = Q20–Q10 ........................................ Measure the height of the island and assign to Q1
15   STOP .................................................................. Q1 can be checked after the program run has been stopped
                                                  (see page 7-14)
16   L Z+100 R0 FMAX M2
17   END PGM 3600717 MM .................................. Retract the tool and end the program
```

7.8 Example for Exercise

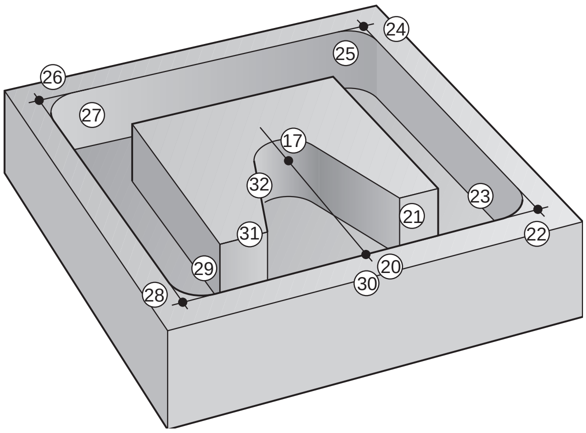
Rectangular pocket with corner rounding and tangential approach

Pocket center coordinates:

X = 50 mm (Q1)
Y = 50 mm (Q2)

Pocket length	X	=	90 mm	(Q3)
Pocket width	Y	=	70 mm	(Q4)
Working depth	Z	=	(–) 15 mm	(–Q5)
Corner radius	R	=	10 mm	(Q6)
Milling feed	F	=	200 mm/min	(Q7)

At the corners 21 and 31 the workpiece will be machined slightly differently than shown in the drawing!



Part program

```
0    BEGIN PGM 360077 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    FN 0: Q1 = +50
4    FN 0: Q2 = +50
5    FN 0: Q3 = +90
6    FN 0: Q4 = +70
7    FN 0: Q5 = +15
8    FN 0: Q6 = +10
9    FN 0: Q7 = +200
10   TOOL DEF 1 L+0 R+5
11   TOOL CALL 1 Z S1000
12   L Z+100 R0 FMAX M6
13   FN4: Q13 = +Q3 DIV+2
14   FN4: Q14 = +Q4 DIV+2
15   FN4: Q16 = +Q6 DIV+4 ..................................... Rounding radius for tangential approach
16   FN4: Q17 = +Q7 DIV+2 ..................................... Feed rate in corners is half the rate for linear movement
17   L X+Q1 Y+Q2 R0 FMAX M3 .............................. Pre-position in X and Y (pocket center), spindle ON
18   L Z+2 FMAX ....................................................... Pre-position over workpiece
19   L Z–Q5 FQ7 ....................................................... Move to working depth Q5 (= –15 mm) with feed rate Q7
                                                                    (=100)
20   L IX+Q13 Y+Q2 RL
21   RND RQ16 FQ17
22   L IY+Q14
23   RND RQ6 FQ17
24   L IX–Q3
25   RND RQ6 FQ17
26   L IY–Q4
27   RND RQ6 FQ17
28   L IX+Q3
29   RND RQ6 FQ17
30   L IY+Q14
31   RND RQ16 FQ17
32   L X+Q1 Y+Q2 R0 FMAX
33   L Z+100 FMAX M2 ............................................ Retract tool
34   END PGM 360077 MM
```

Bolt hole circle

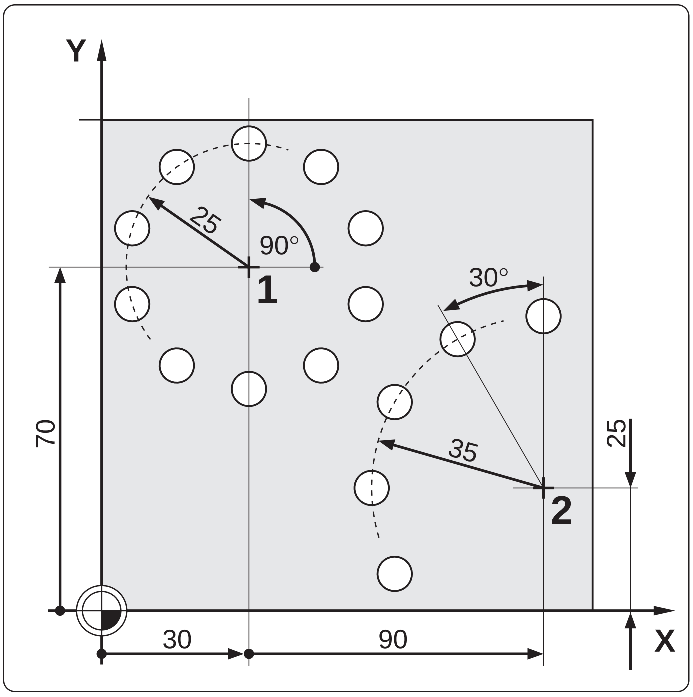
Bore pattern distributed over a full circle:

Entry values are listed below in program blocks 1 - 8.

Movements in the plane are programmed with polar coordinates.

Bore pattern distributed over a circle sector:

Entry values are listed below in lines 20 - 24, Q5, Q7 and Q8 remain the same.



Part program

```
0 BEGIN PGM 3600715 MM ..................................... Load data for bolt hole circle 1:
1 FN 0: Q1 = + 30 .......................................... Circle center X coordinate
2 FN 0: Q2 = +70 ........................................... Circle center Y coordinate
3 FN 0: Q3 = +11 ........................................... Number of holes
4 FN 0: Q4 = +25 ........................................... Circle radius
5 FN 0: Q5 = +90 ........................................... Start angle
6 FN 0: Q6 = +0 ............................................ Hole angle increment (0: distribute hole over 360°)
7 FN 0: Q7 = +2 ............................................ Setup clearance
8 FN 0: Q7 = +15 ........................................... Total hole depth
9 BLK FORM 0.1 Z X+0 Y+0 Z–20
10 BLK FORM 0.2 X+100 Y+100 Z+0
11 TOOL DEF 1 L+0 R+4
12 TOOL CALL 1 Z S2500
13 CYCL DEF 1.0 PECKING ...................................... Definition of the pecking cycle
14 CYCL DEF 1.1 SET UP +Q7 ................................... Setup clearance
15 CYCL DEF 1.2 DEPTH –Q8 .................................... Total hole depth according to the load data
16 CYCL DEF 1.3 PECKG +5
17 CYCL DEF 1.4 DWELL 0
18 CYCL DEF 1.5 F250
19 CALL LBL 1 ................................................ Call bolt hole circle 1, load data for bolt hole circle  2
                                                             (only re-enter changed data)
20 FN 0: Q1 = +90 ............................................ New circle center X coordinate
21 FN 0: Q2 = +25 ............................................ New circle center Y coordinate
22 FN 0: Q3 = +5 ............................................. New number of holes
23 FN 0: Q4 = +35 ............................................ New circle radius
24 FN 0: Q6 = +30 ............................................ New hole angle increment (not a full circle, 5
                                                             holes at 30° intervals)
25 CALL LBL 1 ................................................ Call bolt hole circle 2
26 L Z+200 R0 F MAX M2
```

Continued ...

```
27 LBL 1 ..................................................................Subprogram bolt hole circle
28 FN 0: Q10 = +0 ......................................................Set the counter for finished holes
29 FN 10: IF +Q6 NE +0 GOTO LBL 10 ....................If the hole angle increment has been entered, jump to LBL 10
30 FN 4: Q6 = +360 DIV +Q3 ...................................Calculate the hole angle increment, distribute holes over 360°
31 LBL 10
32 FN 1: Q11 = +Q5 + +Q6......................................Calculate second hole position from the start angle and hole
                                                              angle increment
33 CC X+Q1 Y+Q2 ....................................................Set pole at bolt hole circle center
34 LP PR+Q4 PA+Q5 R0 F MAX M3 ........................Move in the plane to 1st hole
35 L Z+Q7 R0 F MAX M99 ........................................Move in Z to setup clearance, call cycle
36 FN 1: Q10 = +Q10 + +1 ......................................Count finished holes
37 FN 9: IF +Q10 EQU +Q3 GOTO LBL 99 ..............Finished?
38 LBL 2
39 LP PR+Q4 PA+Q11 R0 F MAX M99 ....................Make a second and further holes
40 FN 1: Q10 = +Q10 + +1 ......................................Count finished holes
41 FN 1: Q11 = +Q11 + +Q6....................................Calculate angle for next hole
                                                              (update)
42 FN 12: IF + Q10 LT + Q3 GOTO LBL 2 ................Not finished?
43 LBL 99
44 L Z+200 R0 F MAX ...............................................Retract in Z
45 LBL 0....................................................................End of subprogram
46 END PGM 3600715 MM
```

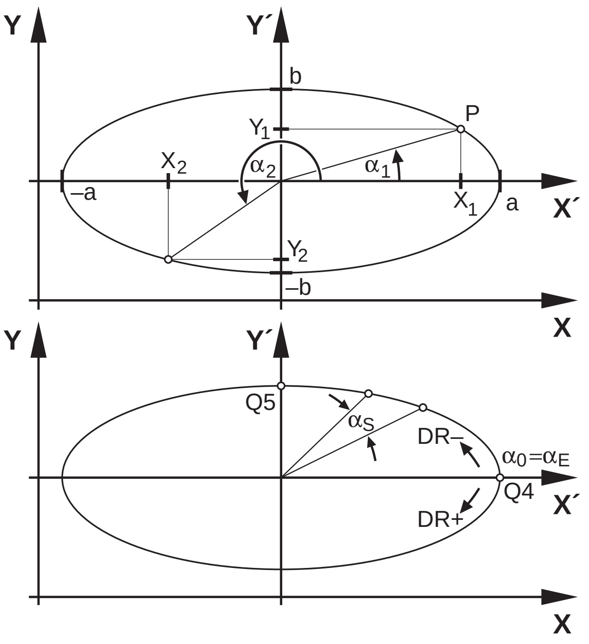
Ellipse

X coordinate calculation: X = a x cos α
Y coordinate calculation: Y = b x sin α

a, b : Semimajor and semiminor axes of the ellipse
α : Angle between the leading axis and the connecting line from P to the center of the ellipse.
X_0, Y_0 : Center of the ellipse

Process:
The points of the ellipse are calculated and connected by many short lines. The more points that are calculated and the shorter the lines between them, the smoother the curve.

The machining direction can be varied by changing the entries for start and end angles. The input parameters are listed below in blocks 1 - 12.



Part program

```
0 BEGIN PGM 360079 MM ..................................... Load data
1 FN 0: Q1 = +50 .......................................... X coordinate for center of ellipse
2 FN 0: Q2 = +50 .......................................... Y coordinate for center of ellipse
3 FN 0: Q3 = +50 .......................................... Semiaxis in X
4 FN 0: Q4 = +20 .......................................... Semiaxis in Y
5 FN 0: Q5 = +0 ........................................... Start angle
6 FN 0: Q6 = +360 ......................................... End angle
7 FN 0: Q7 = +40 .......................................... Number of calculating steps
8 FN 0: Q8 = +0 ........................................... Rotational position
9 FN 0: Q9 = +10 .......................................... Depth
10 FN 0: Q10 = +100 ....................................... Plunging feed rate
11 FN 0: Q11 = +350 ....................................... Milling feed rate
12 FN 0: Q12 = +2 ......................................... Setup clearance Z
13 BLK FORM 0.1 Z X+0 Y+0 Z–20
14 BLK FORM 0.2 X+100 Y+100 Z+0
15 TOOL DEF 1 L+0 R+2.5
16 TOOL CALL 1 Z S2800
17 L Z+2000 R0 F MAX
18 CALL LBL 10 ............................................ Call subprogram ellipse
19 L Z+20 R0 F MAX M02 .................................... Retract in Z, end of main program
```

Continued ...

```
20 LBL 10
21 CYCL DEF 7.0 DATUM SHIFT
22 CYCL DEF 7.1 X+Q1
23 CYCL DEF 7.2 Y+Q2 ........................................... Shift datum to center of ellipse
24 CYCL DEF 10.0 ROTATION
25 CYCL DEF 10.1 ROT +Q8 ................................... Activate rotation, if Q8 is loaded
26 FN2: Q35 = +Q6 – +Q5 ....................................... Calculate angle increment (end angle to start angle
                                                              divided by number of steps)
27 FN4: Q35 = +Q35 DIV +Q7 ................................. Current angle for calculation =
                                                              set start angle
28 FN0: Q36 = +Q5 ................................................... Set counter for milled steps
29 FN0: Q37 = +0 ..................................................... Call subprogram for calculating the points of the ellipse
30 CALL LBL 11 REP ................................................ Call subprogram for calculating the points of the ellipse
31 L X+Q21 Y+Q22 R0 F MAX M03 ......................... Move to start point in the plane
32 L Z+Q12 R0 F MAX M ......................................... Rapid traverse in Z to setup clearance
33 L Z–Q9 R0 FQ10 M .............................................. Plunge to milling depth at plunging feed rate

34 LBL 1
35 FN1: Q36 = +Q36 + +Q35................................... Update the angle
36 FN1: Q37 = +Q37 + +1 ....................................... Update the counter
37 CALL LBL11 REP ................................................. Call subprogram for calculating the points of the ellipse
38 L X+Q21 Y+Q22 R0 FQ11 M ............................... Move to next point
39 FN 12: IF +Q37 LT +Q7 GOTO LBL 1 ................. Unfinished?

40 CYCL DEF 10.0 ROTATION
41 CYCL DEF 10.1 ROT+0 ........................................ Reset rotation
42 CYCL DEF 7.0 DATUM SHIFT
43 CYCL DEF 7.1 X+0
44 CYCL DEF 7.2 Y+0 ................................................ Reset datum shift
45 L Z+Q12 R0 F MAX M ......................................... Move in Z to setup clearance
46 LBL 0 .................................................................... End of subprogram for milling the ellipse

47 LBL 11
48 FN7: Q21 = COS + Q36
49 FN3: Q21 = +Q21* + Q3 ..................................... Calculate X coordinate
50 FN6: Q22 = SIN + Q36
51 FN3: Q22 = +Q22 * +Q4 ..................................... Calculate Y coordinate
52 LBL 0
53 END PGM 360079 MM
```

Three-dimensional machining (machining a hemisphere with an end mill)

Notes on the program:

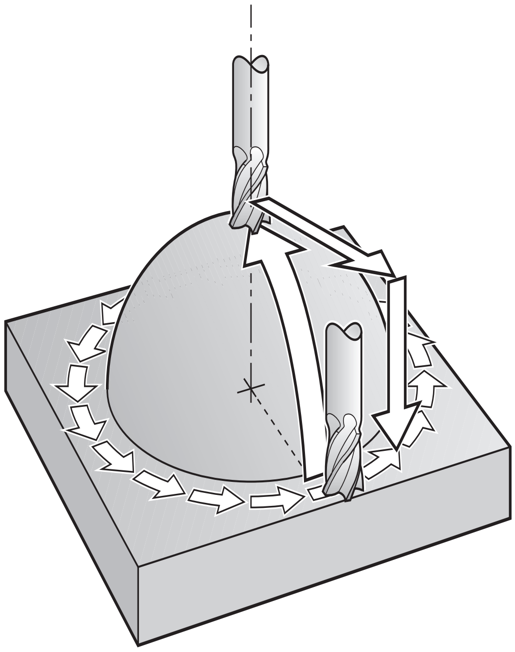
- The tool moves upwards in the ZX plane.
- You can enter an oversize in block 12 (Q12) if you want to machine the contour in several steps.
- The tool radius is automatically compensated with parameter Q108.

The program works with the following values:

• Solid angle:	Start angle	Q1
	End angle	Q2
	Increment	Q3
• Sphere radius		Q4
• Setup clearance		Q5
• Plane angle:	Start angle	Q6
	End angle	Q7
	Increment	Q8
• Center of sphere:	X coordinate	Q9
	Y coordinate	Q10
• Milling feed rate		Q11
• Oversize		Q12

The parameters additionally defined in the program have the following meanings:

- Q15: Setup clearance above the sphere
- Q21: Solid angle during machining
- Q24: Distance from center of sphere to center of tool
- Q26: Plane angle during machining
- Q108: TNC parameter with tool radius



Part program

```
0    BEGIN PGM 360712 MM
1    FN 0:  Q1   =   + 90
2    FN 0:  Q2   =   + 0
3    FN 0:  Q3   =   + 5
4    FN 0:  Q4   =   + 45
5    FN 0:  Q5   =   + 2
6    FN 0:  Q6   =   + 0                    Assign the sphere data to the parameters
7    FN 0:  Q7   =   + 360
8    FN 0:  Q8   =   + 5
9    FN 0:  Q9   =   + 50
10   FN 0:  Q10  =   + 50
11   FN 0:  Q11  =   + 500
12   FN 0:  Q12  =   + 0
13   BLK FORM 0.1 Z X+0 Y+0 Z–50
14   BLK FORM 0.2 X+100 Y+100 Z+0           Workpiece blank; define and insert tool
15   TOOL DEF 1 L+0 R+5
16   TOOL CALL 1 Z S1000
17   L Z+100 R0 FMAX M6
18   CALL LBL 10 .......................... Subprogram call
19   L Z+100 R0 FMAX M2 ................... Retract tool; jump to beginning of program
```

Continued...

```
20  LBL 10
21  FN1: Q15 = + Q5 + + Q4
22  FN0: Q21 = + Q1                      Determine starting and calculation values
23  FN1: Q24 = + Q4 + + Q108
24  FN0: Q26 = + Q6

25  CYCL DEF 7.0 DATUM
26  CYCL DEF 7.1 X+Q9                    Shift datum to center of sphere
27  CYCL DEF 7.2 Y+Q10
28  CYCL DEF 7.3 Z–Q4

29  CYCL DEF 10.0 ROTATION               Rotation for program start (starting plane angle)
30  CYCL DEF 10.1 ROT + Q6

31  CC X+0 Y+0
32  LP PR + Q24 PA + Q6 R0 FQ11 ........ Pre-positioning before machining
33  LBL 1
34  CC Z+0 X+Q108
35  L Y+0 Z+0 FQ11 ..................... Pre-positioning at beginning of each arc
36  LBL 2

37  LP PR+Q4 PA+Q21 R0 FQ11
38  FN2: Q21 = + Q21 – + Q3              Mill the sphere upward until the highest points is reached
39  FN11: IF + Q21 GT + Q2 GOTO LBL2

40  LP PR+Q4 PA+Q2
41  L Z+Q15 R0 F1000                     Mill the highest point and then retract the tool
42  L X+Q24 R0 FMAX

43  FN1: Q26 = + Q26 + + Q8 ............ Prepare the next rotation increment
44  FN0: Q21 = + Q1 .................... Reset solid angle for machining to the starting value

45  CYCL DEF 10.0 ROTATION
46  CYCL DEF 10.1 ROT + Q26              Rotate the coordinate system about the Z axis until
47  FN12: IF + Q26 LT + Q7 GOTO LBL1     plane end angle is reached
48  FN9: IF + Q26 EQU + Q7 GOTO LBL1

49  CYCL DEF 10.0 ROTATION
50  CYCL DEF 10.1 ROT + 0
51  CYCL DEF 7.0 DATUM                   Reset rotation and datum shift
52  CYCL DEF 7.1 X+0
53  CYCL DEF 7.2 Y+0
54  CYCL DEF 7.3 Z+0

55  LBL 0 .............................. End of subprogram
56  END PGM 360712 MM
```

8.1 General Overview of Cycles

Frequently recurring machining sequences which comprise several steps are stored in the TNC as cycles. Coordinate transformations and other special functions are also available as cycles.

The cycles are divided into several groups:

- **Simple fixed cycles** such as pecking and tapping as well as the milling operations slot milling, circular pocket milling and rectangular pocket milling.
- **SL (Subcontour List) cycles**, which allow machining of relatively complex contours composed of several overlapping subcontours.
- **Coordinate transformation cycles** which enable datum shift, rotation, mirror image, enlarging and reducing for various contours.
- **Special cycles** such as dwell time, program call and oriented spindle stop.

Programming a cycle

Defining a cycle

Pressing the CYCL DEF key opens the cycle directory. Select the desired cycle and program it in the dialog. The following example shows how to define any cycle:

CYCL DEF	Open the cycle directory.

CYCL DEF 1 PECKING

↓ / ↑	Select a cycle with the vertical arrow keys (in this example, cycle 17).
GOTO □ 1 7 ENT	Address the desired cycle directly with GOTO.

CYCL DEF 17 RIGID TAPPING

ENT	Confirm entry of the selected cycle.

⋮

⋮

The TNC then requests the data for the selected cycle:

SETUP CLEARANCE?	
+/– 2 ENT	Enter setup clearance, for example –2 mm.

TOTAL HOLE DEPTH?	
+/– 3 0 ENT	Enter total hole depth, for example –30 mm.

THREAD PITCH?	
0 . 7 5 ENT	Enter thread pitch, for example +0.75 mm.

Resulting NC block:	*17.0*	*RIGID TAPPING*
	17.1	*SET UP –2*
	17.2	*DEPTH –30*
	17.3	*PITCH +0.75*

Cycle call

The following cycles become effective immediately upon being defined in the part program:

- Coordinate transformation cycles
- Dwell time
- The SL cycle CONTOUR

All other cycles must be called separately. Further information on cycle calls is provided in the descriptions of the individual cycles.

If the cycle is to be programmed after the block in which it was called up, program the cycle call

- with CYCL CALL

CYCL CALL ▶

MISCELLANEOUS FUNCTION	
3 ENT	Cycle call with miscellaneous function M3.

- with the miscellaneous function M99.

If the cycle is to be run after every positioning block, it must be called with the miscellaneous function M89 (depending on the machine parameters).

M89 is cancelled through
- M99
- CYCL CALL
- a new cycle definition

Prerequisites:

The following data must be programmed before a cycle call:

- BLK FORM for graphic display
- Tool call
- Positioning block for starting position X, Y
- Positioning block for starting position Z (setup clearance)
- Direction of rotation of the spindle (miscellaneous functions M3/M4)
- Cycle definition (CYCL DEF).

Dimensions in the tool axis

The dimensions for tool axis movement are always referenced to the position of the tool at the time of the cycle call and interpreted by the control as incremental dimensions. It is not necessary to press the incremental key.

The algebraic signs for SETUP CLEARANCE, TOTAL HOLE DEPTH and JOG INCREMENT define the working direction. They must be entered identically (usually negative).

Customized macros

The machine tool builder can store additional cycles in the control memory. These cycles can be called up under cycle numbers 68 to 99. Information on these cycles is available from the machine builder.

The TNC assumes that at the beginning of the cycle the tool is positioned over the workpiece at the clearance height.

8.2 Simple Fixed Cycles

PECKING (Cycle 1)

Process:

- The tool drills at the entered feed rate to the first pecking depth.
- The tool is then retracted at rapid traverse (FMAX) to the starting position and advances again to the first pecking depth, minus the advanced stop distance *t* (see calculations).
- The tool advances with another infeed at the programmed feed rate.
- These steps are repeated until the programmed total hole depth is reached.
- After a dwell time at the bottom of the hole, the tool is retracted to the starting position at FMAX for chip breaking.

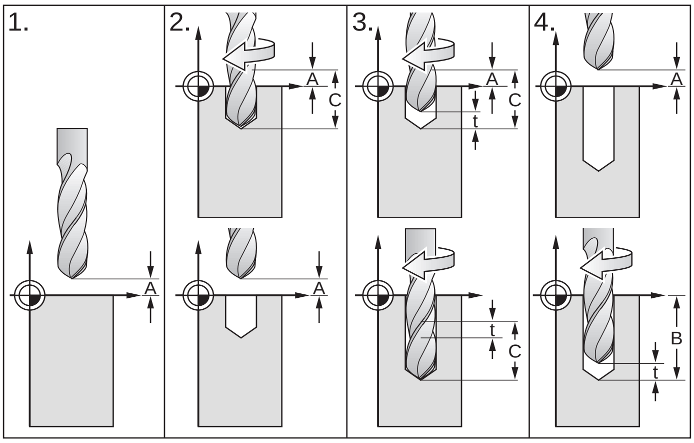


Fig. 8.1: PECKING cycle

Input data

- SETUP CLEARANCE Ⓐ:
  Distance between tool tip (at starting position) and workpiece surface
- TOTAL HOLE DEPTH Ⓑ:
  Distance between workpiece surface and bottom of hole (tip of drill taper)
- PECKING DEPTH Ⓒ:
  Infeed per cut.
  If the TOTAL HOLE DEPTH equals the PECKING DEPTH, the tool will drill to the programmed hole depth in one operation. The PECKING DEPTH does not have to be a multiple of the TOTAL HOLE DEPTH. If the PECKING DEPTH is greater than the TOTAL HOLE DEPTH, the tool only advances to the TOTAL HOLE DEPTH.
- DWELL TIME:
  Length of time the tool remains at the total hole depth for chip breaking.
- FEED RATE:
  Traversing speed of the tool when drilling

Calculations

The advanced stop distance is automatically calculated by the control:

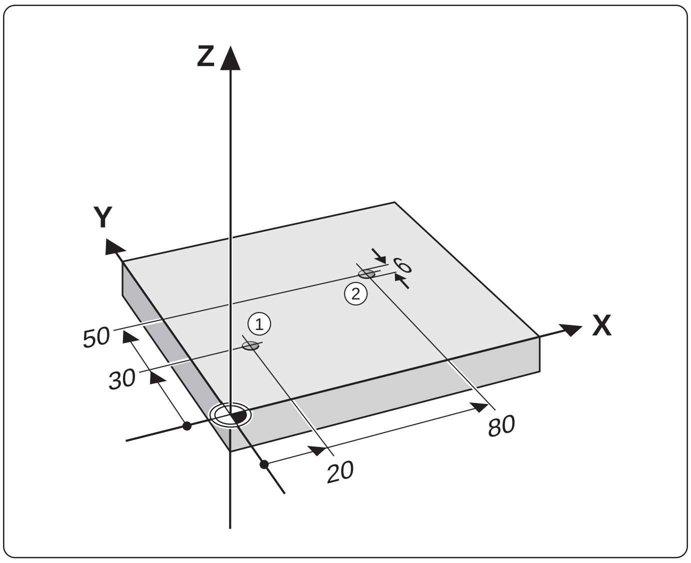
- Total hole depth up to 30 mm: t = 0.6 mm
- Total hole depth over 30 mm: t = Total hole depth / 50
  maximum advanced stop distance: 7 mm

Example: Pecking

Hole coordinates:

1	X	=	20 mm	Y	=	30 mm
2	X	=	80 mm	Y	=	50 mm

Hole diameter:	6	mm
Setup clearance:	2	mm
Total hole depth:	15	mm
Pecking depth:	10	mm
Dwell time:	1	s
Feed rate:	80	mm/min



PECKING cycle in a part program

```
0    BEGIN PGM 360086 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+3
4    TOOL CALL 1 Z S1000
5    CYCL DEF 1.0 PECKING
6    CYCL DEF 1.1 SET UP –2 ..................................Setup clearance
7    CYCL DEF 1.2 DEPTH –15 ................................Total hole depth
8    CYCL DEF 1.3 PECKG –10 ................................Pecking depth
9    CYCL DEF 1.4 DWELL 1 ...................................Dwell time
10   CYCL DEF 1.5 F 80 ...........................................Feed rate
11   L Z+100 R0 FMAX M6 ......................................Approach tool change position
12   L X+20 Y+30 FMAX M3 ....................................Pre-positioning for first hole, spindle on
13   L Z+2 FMAX M99 ..............................................Pre-positioning in Z, first hole, cycle call
14   L X+80 Y+50 FMAX M99 ..................................Approach second hole, cycle call
15   L Z+100 FMAX M2
16   END PGM 360086 MM
```

TAPPING with floating tap holder (cycle 2)

Process

- The thread is cut in one pass.
- When the tool reaches the total hole depth, the direction of spindle rotation is reversed. After the programmed dwell time the tool is retracted to the starting position.
- At the starting position, the direction of rotation is reversed once again.

Required tool

A floating tap holder is required for tapping. The floating tap holder compensates the tolerances for feed rate and spindle speed during the tapping process.

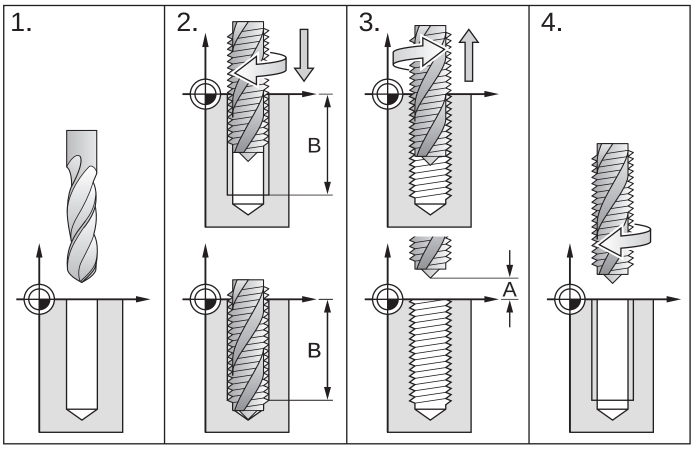


Fig. 8.2: TAPPING cycle

Input data

- SETUP CLEARANCE Ⓐ:
  Distance between tool tip (starting position) and workpiece surface.
  Standard value: 4x thread pitch.
- TOTAL HOLE DEPTH Ⓑ (thread length):
  Distance between workpiece surface and end of thread
- DWELL TIME:
  Enter a value between 0 and 0.5 seconds to prevent wedging of the tool when retracted. (Further information is available from the machine manufacturer.)
- FEED RATE F:
  Traversing speed of the tool during tapping.

The signs for setup clearance and total hole depth are the same and depend on the working direction.

Calculations

The feed rate is calculated as follows:

$F = S \times p$

F: Feed rate (mm/min)
S: Spindle speed (rpm)
p: Thread pitch (mm)

When a cycle is being run, the spindle speed override control is disabled. The feed rate override control is only active within a limited range (preset by the machine tool builder).

For tapping right-hand threads activate the spindle with M3; for left-hand threads use M4.

Example: Tapping with a floating tap holder

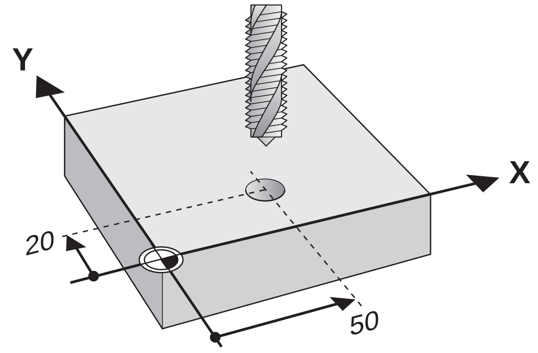
Cutting an M6 thread at 100 rpm

Coordinates of the hole:

X	=	50 mm	Y	=	20 mm
Pitch			p	=	1 mm

F = S x p ⇨ F = 100 · 1 = 100 mm/min

Setup clearance:	3	mm
Thread depth:	20	mm
Dwell time:	0.4	s
Feed rate:	100	mm/min



TAPPING cycle in a part program

```
0    BEGIN PGM 360088 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+3
4    TOOL CALL 1 Z S1000
5    CYCL DEF 2.0 TAPPING
6    CYCL DEF 2.1 SET UP –3 .................................. Setup clearance
7    CYCL DEF 2.2 DEPTH –20 ................................ Thread depth
8    CYCL DEF 2.3 DWELL 0.4 ................................ Dwell time
9    CYCL DEF 2.4 F 100 .......................................... Feed rate
10   L Z+100 R0 FMAX M6 ...................................... Approach tool change position
11   L X+50 Y+20 FMAX M3 .................................... Pre-positioning, spindle on clockwise
12   L Z+3 FMAX M99 .............................................. Pre-positioning in Z, cycle call
13   L Z+100 FMAX M2
14   END PGM 360088 MM
```

RIGID TAPPING (Cycle 17)

Process

The thread is cut without a floating tap holder in one or several passes.

Advantages over tapping with a floating tap holder:

- Higher machining speeds
- Repeated tapping of the same thread; repetitions are made possible by spindle orientation to the 0° position during cycle call (depending on machine parameters)
- Increased traverse range of the spindle axis

Machine and control must be specially prepared by the machine manufacturer to enable rigid tapping.

Input data

- SETUP CLEARANCE Ⓐ:
  Distance between tool tip (starting position) and workpiece surface.
- TAPPING DEPTH Ⓑ:
  Distance between workpiece surface (beginning of thread) and end of thread

The signs for setup clearance and thread pitch are the same and depend on the working direction.

- THREAD PITCH Ⓒ:
  The sign differentiates between right-hand and left-hand threads:
  + = Right-hand thread
  – = Left-hand thread

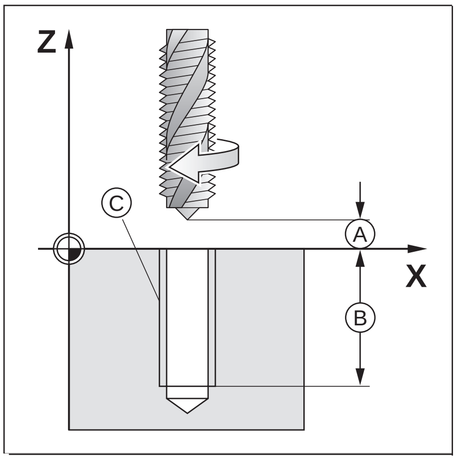


Fig. 8.3: Infeeds and distances in the ROUGH-OUT cycle

The control calculates the feed rate from the spindle speed. If the spindle speed override knob is turned during tapping, the control automatically adjusts the feed rate accordingly. The feed rate override is disabled.

SLOT MILLING (Cycle 3)

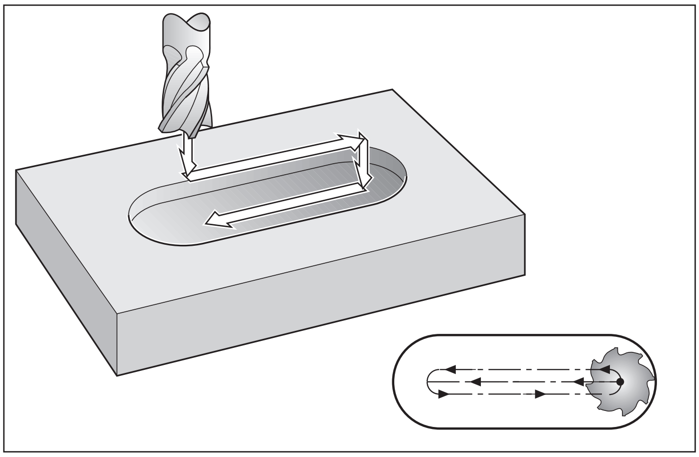


Fig. 8.4: SLOT MILLING cycle

Process

Roughing process:

- The tool penetrates the workpiece from the starting position and mills in the longitudinal direction of the slot.
- After downfeed at the end of the slot, milling is performed in the opposite direction.
  These steps are repeated until the programmed milling depth is reached.

Finishing process:

- The control advances the tool in a quarter circle at the bottom of the slot by the remaining finishing cut. The tool subsequently climb mills the contour (with M3).
- At the end of the cycle, the tool is retracted in rapid traverse to the setup clearance.
  If the number of infeeds was odd, the tool returns to the starting position at the level of the setup clearance.

Required tool

This cycle requires a center cut end mill (ISO 1641). The cutter diameter must not be larger than the width of the slot and not smaller than half the width of the slot. The slot must be parallel to an axis of the current coordinate system.

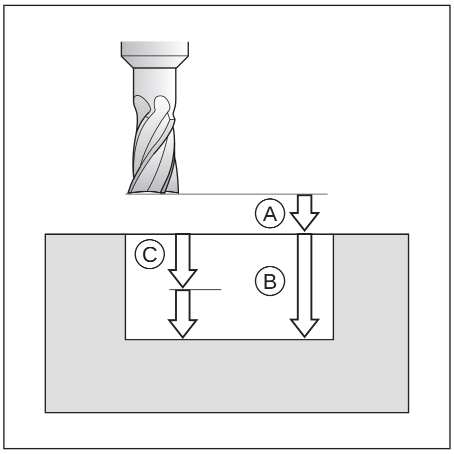


Fig. 8.5: Infeeds and distances for the SLOT MILLING cycle

Input data

- Setup clearance Ⓐ
- Milling depth Ⓑ: Depth of the slot
- Pecking depth Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration.
- FIRST SIDE LENGTH Ⓓ:
  Length of the slot. Specify the sign to determine the first milling direction.
- SECOND SIDE LENGTH Ⓔ:
  Width of the slot
- FEED RATE:
  Traversing speed of the tool in the working plane.

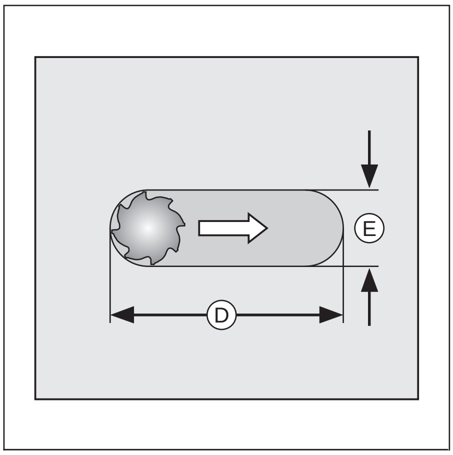


Fig. 8.6: Side lengths of the slot

Example: Slot milling

A horizontal slot 50 mm x 10 mm and a vertical slot 80 mm x 10 mm are to be milled.

The starting position takes into account the tool radius in the longitudinal direction of the slot.

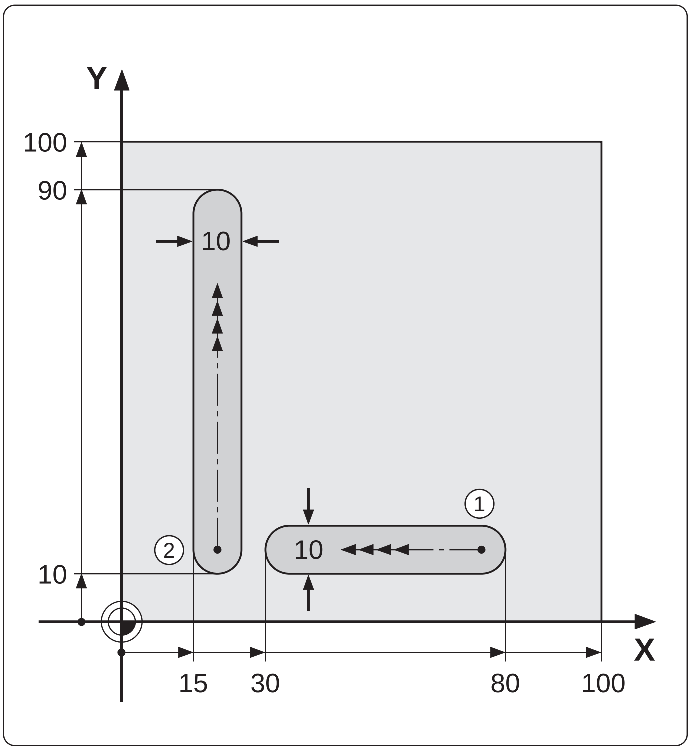
Starting position slot ①:
X = 76 mm Y = 15 mm

Starting position ②:
X = 20 mm Y = 14 mm

SLOT DEPTHS:	15 mm
Setup clearances:	2 mm
Milling depths:	15 mm
Pecking depths:	5 mm
Feed rate for pecking:	80 mm/min

	①	②
Slot length 1st milling direction	50 mm –	80 mm +

Slot widths:	10 mm
Feed rate:	120 mm/min



SLOT MILLING cycle in a part program

```
0    BEGIN PGM 360811 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+4
4    TOOL CALL 1 Z S1000
5    CYCL DEF 3.0 SLOT MILLING ........................... Slot parallel to X-axis
6    CYCL DEF 3.1 SET UP –2 .................................. Setup clearance
7    CYCL DEF 3.2 DEPTH –15 ................................ Milling depth
8    CYCL DEF 3.3 PECKG –5 F80 ........................... Pecking depth, feed rate for pecking
9    CYCL DEF 3.4 X–50 ........................................... Slot length and first milling direction (–)
10   CYCL DEF 3.5 Y+10 .......................................... Slot width
11   CYCL DEF 3.6 F120 ........................................... Feed rate
12   L Z+100 R0 FMAX M6
13   L X+76 Y+15 FMAX M3 .................................... Approach starting position, spindle on
14   L Z+2 F1000 M99 ............................................... Pre-positioning in Z, cycle call
15   CYCL DEF 3.0 SLOT MILLING ........................... Slot parallel to Y-axis
16   CYCL DEF 3.1 SET UP –2 .................................. Setup clearance
17   CYCL DEF 3.2 DEPTH –15 ................................ Milling depth
18   CYCL DEF 3.3 PECKG –5 F80 ........................... Pecking depth, feed rate for pecking
19   CYCL DEF 3.4 Y+80 .......................................... Slot length and first milling direction (+)
20   CYCL DEF 3.5 X+10 .......................................... Slot width
21   CYCL DEF 3.6 F110 ........................................... Feed rate
22   L X+20 Y+14 FMAX .......................................... Approach starting position
23   CYCL CALL ......................................................... Cycle call
24   L Z+100 FMAX M2
25   END PGM 360811 MM
```

POCKET MILLING (Cycle 4)

Process

The rectangular pocket milling cycle is a roughing cycle, in which

- the tool penetrates the workpiece at the starting position (pocket center)
- the tool subsequently follows the programmed path at the specified feed rate (see Fig. 8.9) .

The cutter begins milling in the positive axis direction of the longer side. With square pockets, the cutter begins in the positive Y-direction. At the end of the cycle, the tool returns to the starting position.

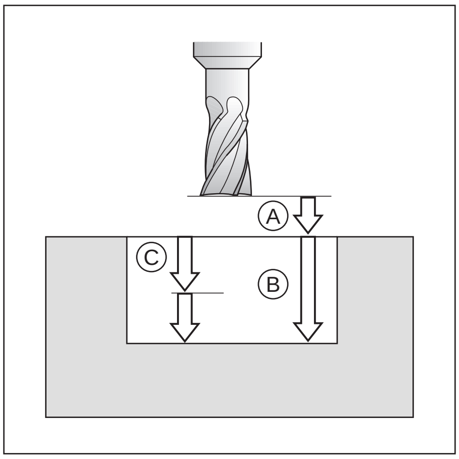


Fig. 8.7: Infeeds and distances for the POCKET MILLING cycle

Requirements / Limitations

This cycle requires a center-cut end mill (ISO 1641) or a separate pilot drilling operation at the pocket center. The pocket sides are parallel to the axes of the coordinate system.

Input data

- Setup clearance Ⓐ
- Milling depth Ⓑ
- Pecking depthⒸ
- FEED RATE FOR PECKING: Traversing speed of the tool during penetration.
- FIRST SIDE LENGTH Ⓓ: Length of the pocket, parallel to the first main axis of the working plane.
- SECOND SIDE LENGTH Ⓔ: Width of the pocket The signs of the side lengths are always positive
- FEED RATE: Traversing speed of the tool in the working plane.
- DIRECTION OF THE MILLING PATH:
  DR + : Climb milling with M3
  DR – : Up-cut milling with M3

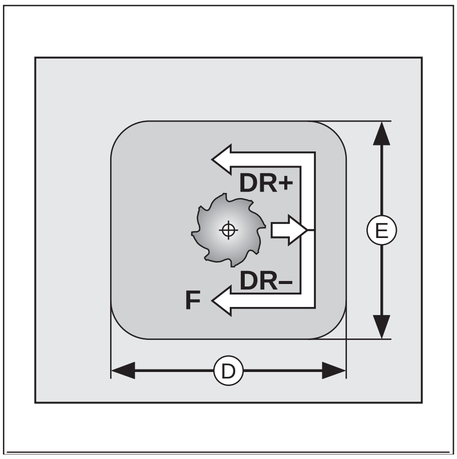


Fig. 8.8: Side lengths of the pocket

The radius of the pocket corners is determined by the cutter radius. The tool does not perform any circular movement in the pocket corners.

Calculations:

Stepover factor k:

k = K x R

K: Overlap factor (preset by the machine builder)
R: Cutter radius

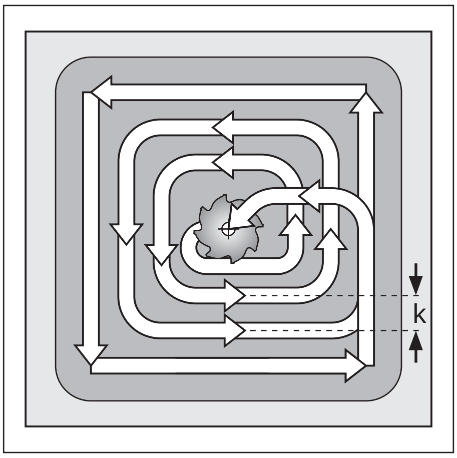


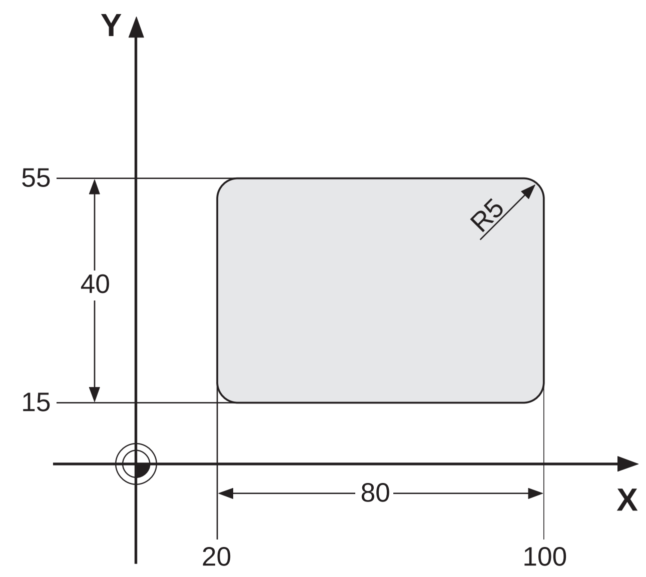
Fig. 8.9: Tool path for roughing out

Example: Rectangular pocket milling

Coordinates of the pocket center:

X = 60 mm   Y = 35 mm

Setup clearance:	2	mm
Milling depth:	10	mm
Pecking depth:	4	mm
Feed rate for pecking:	80	mm/min
First side length:	80	mm
Second side length:	40	mm
Milling feed rate:	100	mm/min
Direction of cutter path:		+



POCKET MILLING cycle in a part program

```
0    BEGIN PGM 360813 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+110 Y+100 Z+0 ..................... Note: BLK FORM has been changed!
3    TOOL DEF 1 L+0 R+5
4    TOOL CALL 1 Z S1000
5    CYCL DEF 4.0 POCKET MILLING
6    CYCL DEF 4.1 SET UP –2 ................................... Setup clearance
7    CYCL DEF 4.2 DEPTH –10 ................................ Milling depth
8    CYCL DEF 4.3 PECKG –4 F80 ............................ Pecking depth and feed rate for pecking
9    CYCL DEF 4.4 X+80 ........................................... First side length of pocket
10   CYCL DEF 4.5 Y+40 ........................................... Second side length of pocket
11   CYCL DEF 4.6 F100 DR+RADIUS 0 ................... Feed rate and direction of cutter path
12   L Z+100 R0 FMAX M6
13   L X+60 Y+35 FMAX M3 ..................................... Pre-positioning in X, Y (pocket center), spindle on
14   L Z+2 FMAX ........................................................ Pre-positioning in Z
15   CYCL CALL .......................................................... Cycle call
16   L Z+100 FMAX M2
17   END PGM 360813 MM
```

CIRCULAR POCKET MILLING (Cycle 5)

Process

- Circular pocket milling is a roughing cycle. The tool penetrates the workpiece from the starting position (pocket center).
- The cutter then follows a spiral path at the programmed feed rate (see illustration at right). The stepover factor is determined by the value of k (see Cycle 4, RECTANGULAR POCKET MILLING: calculations).
- The process is repeated until the programmed milling depth is reached.
- At the end of the cycle the tool returns to the starting position.

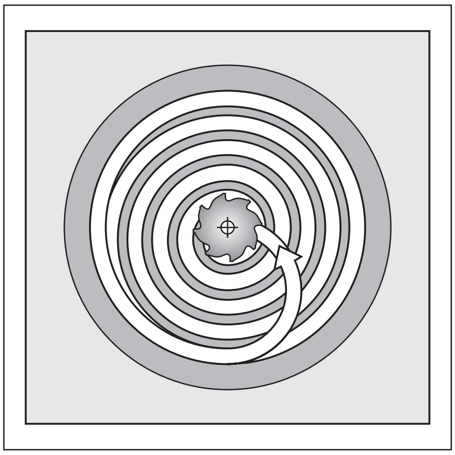


Fig. 8.10: Cutter path for roughing-out

Required tool

This cycle requires a center-cut end mill (ISO 1641) or a separate pilot drilling operation at the pocket center.

Input data

- SETUP CLEARANCE Ⓐ
- MILLING DEPTH Ⓑ: depth of the pocket
- PECKING DEPTH Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration
- CIRCLE RADIUS Ⓡ:
  Radius of the circular pocket
- FEED RATE:
  Traversing speed of the tool in the working plane
- DIRECTION OF THE MILLING PATH:
  DR + : Climb milling with M3
  DR – : Up-cut milling with M3

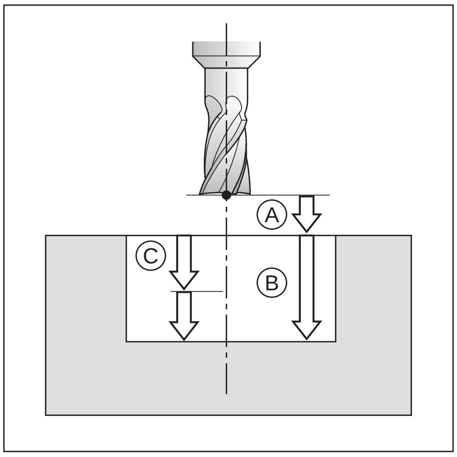


Fig. 8.11: Distances and infeeds with CIRCULAR POCKET MILLING

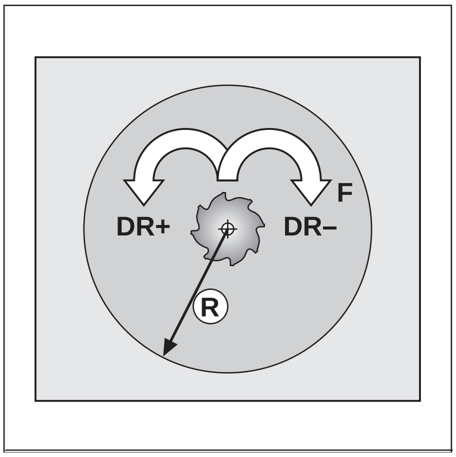


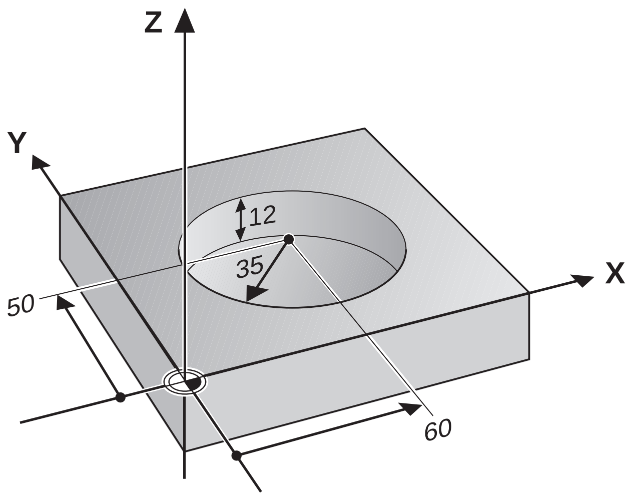
Fig. 8.12: Direction of the cutter path

Example: Milling a circular pocket

Coordinates of the pocket center:

X = 60 mm Y = 50 mm

Setup clearance:	2	mm
Milling depth:	12	mm
Pecking depth:	6	mm
Feed rate for pecking:	80	mm/min
Circle radius:	35	mm
Milling feed rate:	100	mm/min
Direction of the cutter path:	–	



CIRCULAR POCKET MILLING cycle in the part program

```
0     BEGIN PGM 360815 MM
1     BLK FORM 0.1 Z X+0 Y+0 Z–20
2     BLK FORM 0.2 X+100 Y+100 Z+0
3     TOOL DEF 1 L+0 R+10
4     TOOL CALL 1 Z S2000
5     CYCL DEF 5.0 CIRCULAR POCKET
6     CYCL DEF 5.1 SET UP –2 ................................... Setup clearance
7     CYCL DEF 5.2 DEPTH –12 ................................. Milling depth
8     CYCL DEF 5.3 PECKG –6 F80 ............................ Pecking depth and feed rate for pecking
9     CYCL DEF 5.4 RADIUS 35 ................................. Circle radius
10    CYCL DEF 5.5 F 100 DR– .................................. Milling feed rate and direction of cutter path
11    L Z+100 R0 FMAX M6
12    L X+60 Y+50 FMAX M3 .................................... Pre-positioning in X, Y, pocket center, spindle on
13    L Z+2 FMAX M99 .............................................. Starting position in Z, cycle call
14    L Z+100 FMAX M2
15    END PGM 360815 MM
```

8.3 SL Cycles

Subcontour list (SL) cycles are very powerful cycles that enable you to mill any plane contour. They are characterized by the following features:

- A contour can consist of superimposed subcontours. Pockets and islands compose the subcontours.
- The subcontours are programmed as subprograms.
- The control automatically superimposes the subcontours and calculates the points of intersection of the subcontours with each other.

Cycle 14 CONTOUR GEOMETRY contains the subcontour list and is a purely geometric cycle, containing no cutting data or infeed values.

Programming the parallel axes

Pockets and islands can also be machined in planes formed by parallel axes.

Prerequisite:

The plane has to be perpendicular to the tool axis in TOOL CALL.

Example:

Tool axis Z or W; possible planes X/Y, U/Y, X/V, U/V

The coordinates of the desired machining plane must be in the first coordinate block (positioning block or CC block) of the first subprogram named in cycle 14 CONTOUR GEOMETRY.

Example:

Tool axis Z, machining plane X/V

```
 .
 .
 .
TOOL DEF 1 L+0 R+3
TOOL CALL 1 Z S 1000
CYCL DEF 14.0 CONTOUR GEOM.
CYCL DEF 14.1 CONTOUR LABEL 1/2/3
 .
 .
 .
L....M2
LBL 1
CC X+20 V+10
 .
 .
 .
```

All other coordinates are then ignored.

The machining data are defined in the following cycles:

- PILOT DRILLING (cycle 15)
- ROUGH-OUT (cycle 6)
- CONTOUR MILLING (cycle 16)

Each subprogram defines whether RL or RR radius compensation applies. The sequence of points determines the direction of rotation in which the contour is to be machined. The control deduces from these data whether the specific subprogram describes a pocket or an island:

- For a pocket the tool path is inside the contour
- For an island the tool path is outside the contour

- The way the SL contour is machined is determined by MP7420.
- We recommend a graphical test run before you machine the part. This will show if all contours were correctly defined.
- All coordinate transformations are allowed in the subprograms for the subcontours.
- F and M words are ignored in the subprograms for the subcontours.

The following examples will at first use only the ROUGH-OUT cycle. Later, as the examples become more complex, the full range of possibilities of this group of cycles will be illustrated.

CONTOUR GEOMETRY (Cycle 14)

Application

Cycle 14 CONTOUR GEOMETRY contains the list of subcontours that make up the complete contour

Input data

Enter the LABEL numbers of the subprograms. A maximum of 12 subprograms can be listed.

Effect

Cycle 14 becomes effective as soon as it is defined.

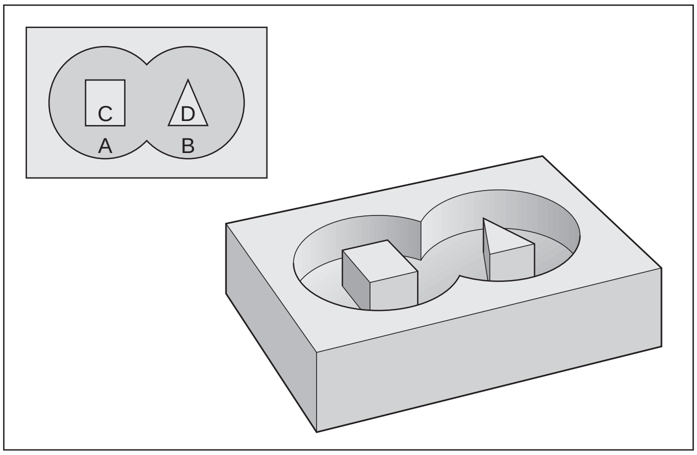


Fig. 8.13: Example of an SL contour: A, B = pockets; C, D = islands

ROUGH-OUT (Cycle 6)

Process

Cycle 6 specifies the cutting path and partitioning.

- The tool is positioned in the tool axis above the first infeed point, taking the finishing allowance into account.
- Then the tool penetrates into the workpiece at the programmed feed rate for pecking.

Milling the contour:

- The tool mills the first subcontour at the specified feed rate, taking the finishing allowance into account.
- When the tool returns to the infeed point, it is advanced to the next pecking depth.

This process is repeated until the programmed milling depth is reached.

- Further contours are milled in the same manner.

Roughing out pockets:

- After milling the contour the pocket is roughed out. The stepover is defined by the tool radius. Islands are jumped over.
- If necessary, pockets can be cleared out with several downfeeds.
- At the end of the cycle the tool returns to the setup clearance.

Required tool

This cycle requires a center cut end mill (ISO 1641) if the pocket is not separately pilot drilled or if the tool must repeatedly jump over contours.

Input data

- SETUP CLEARANCE Ⓐ
- MILLING DEPTH Ⓑ
- PECKING DEPTH Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration
- FINISHING ALLOWANCE Ⓓ:
  Allowance in the machining plane (positive number)
- ROUGH-OUT ANGLE α:
  Feed direction for roughing out. The rough-out angle is relative to the angle reference axis and can be set such that the resulting cuts are as long as possible with few cutting movements.
- FEED RATE:
  Traversing speed of the tool in the machining plane.

Machine parameters determine whether

- the contour is first milled and then surface machined, or vice-versa
- the contour is milled conventionally or by climb milling
- all pockets are first roughed out to the full milling depths and then contour milled, or vice-versa
- contour milling and roughing out are performed together for each pecking depth.

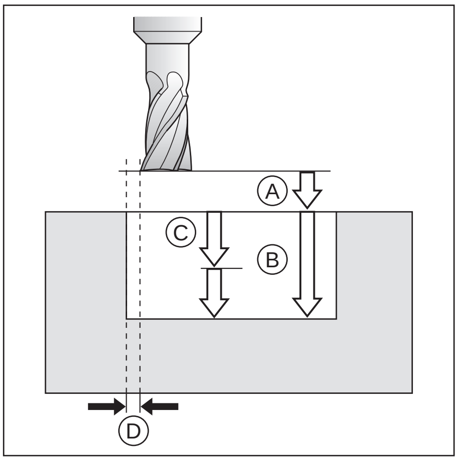


Fig. 8.14: Infeeds and distances with the ROUGH-OUT cycle

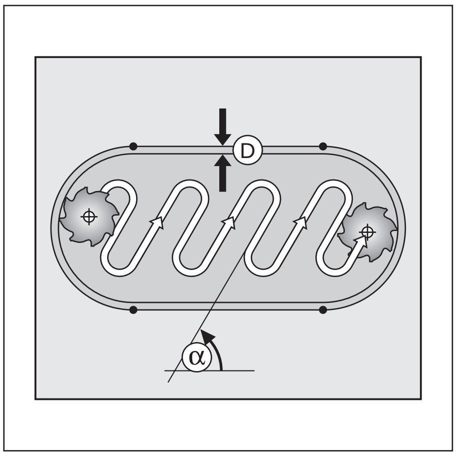


Fig. 8.15: Tool path for rough-out

Example: Roughing out a rectangular pocket

Rectangular pocket with rounded corners

Tool: center-cut end mill (ISO 1641),
radius 5 mm.

Coordinates of the island corners:

	X	Y
①	70 mm	60 mm
②	15 mm	60 mm
③	15 mm	20 mm
④	70 mm	20 mm

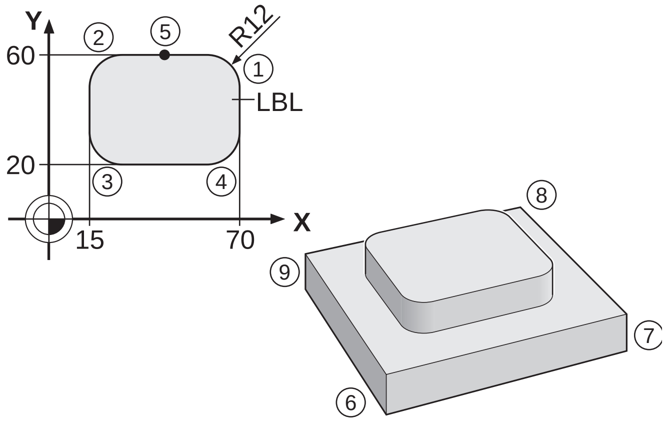
Coordinates of the auxiliary pocket:

	X	Y
⑥	–5 mm	–5 mm
⑦	105 mm	–5 mm
⑧	105 mm	105 mm
⑨	–5 mm	105 mm

Starting point for machining:

⑤ X = 40 mm Y = 60 mm

Setup clearance:	2	mm
Milling depth:	15	mm
Pecking depth:	8	mm
Feed rate for pecking:	100	mm/min
Finishing allowance:	0	
Rough-out angle:	0^0	
Feed rate for milling:	500	mm/min



Cycle in a part program

```
0    BEGIN PGM 360819 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+3
4    TOOL CALL 1 Z S1000
5    CYCL DEF 14.0 CONTOUR GEOM.
6    CYCL DEF 14.1 CONTOUR LABEL 2/1
7    CYCL DEF 6.0 ROUGH-OUT ............................. Cycle definition ROUGH-OUT
8    CYCL DEF 6.1 SET UP –2 DEPTH –15
9    CYCL DEF 6.2 PECKG –8 F100 ALLOW +0
10   CYCL DEF 6.3 ANGLE +0 F500
11   L Z+100 R0 FMAX M6
12   L X+40 Y+50 FMAX M3 .................................... Pre-positioning in X, Y, spindle on
13   L Z+2 FMAX M99 .............................................. Pre-positioning in Z, cycle call
14   L Z+100 FMAX M2
15   LBL 1
16   L X+40 Y+60 RR
17   L X+15
18   RND R12
19   L Y+20
20   RND R12
21   L X+70
22   RND R12
23   L Y+60
24   RND R12
25   L X+40
26   LBL 0
```

Subprogram 1:
Geometry of the island
(From radius compensation RR and counterclockwise machining, the control concludes that contour element 1 is an island)

```
27   LBL 2
28   L X–5 Y–5 RL
29   L X+105
30   L Y+105
31   L X–5
32   L Y–5
33   LBL 0
34   END PGM 360819 MM
```

Subprogram 2:
Geometry of the auxiliary pocket
External limitation of the machined surface
(From radius compensation RL and counter-clockwise machining, the control concludes that contour element 2 is a pocket)

SL Cycles: Overlapping contours

Pockets and islands can be overlapped to form a new contour. The area of a pocket can thus be enlarged by another pocket or reduced by an island.

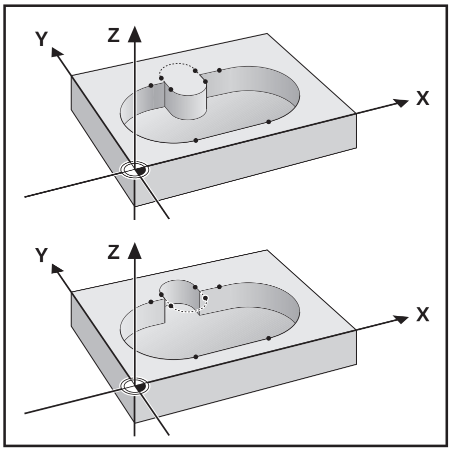


Fig. 8.16: Example for overlapping contours

Starting position

Machining begins at the starting position of the first pocket in cycle 14 CONTOUR GEOMETRY. The starting position should be located as far as possible from the overlapping contours.

Example: Overlapping pockets

Machining begins with the first contour label defined in block 6. The first pocket must begin outside the second pocket.

Inside machining with a center-cut end mill (ISO 1641), tool radius 3 mm.

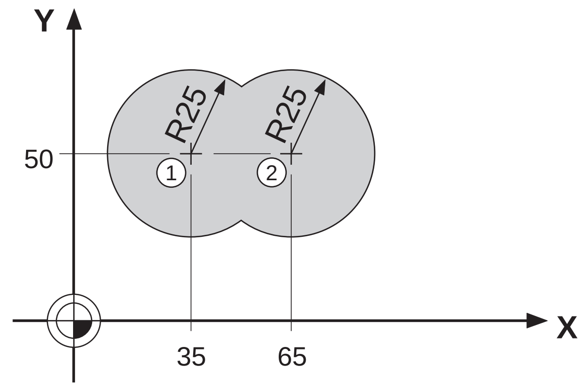
Coordinates of the circle centers:

①	X = 35 mm	Y = 50 mm
②	X = 65 mm	Y = 50 mm

Circle radii

R = 25 mm

Setup clearance:	2	mm
Milling depth:	10	mm
Pecking depth:	5	mm
Feed rate for pecking:	500	mm/min
Finishing allowance:	0	
Rough-out angle:	0	
Milling feed rate:	500	mm/min



Continued...

Cycle in a part program

```
0    BEGIN PGM 360821 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+3
4    TOOL CALL 1 Z S1000
5    CYCL DEF 14.0 CONTOUR GEOM.
6    CYCL DEF 14.1 CONTOUR LABEL 1/2 .............. "List" of contour subprograms
7    CYCL DEF 6.0 ROUGH-OUT ............................. Cycle definition ROUGH-OUT
8    CYCL DEF 6.1 SET UP –2 DEPTH –10
9    CYCL DEF 6.2 PECKG –5 F500 ALLOW +0
10   CYCL DEF 6.3 ANGLE +0 F500
11   L Z+100 R0 FMAX M6
12   L X+50 Y+50 FMAX M3 ..................................... Pre-positioning X, Y, spindle on
13   L Z+2 FMAX M99 ............................................... Setup clearance Z, cycle call
14   L Z+100 FMAX M2 ............................................ Retract, return to start of program
15   LBL 1                                        }
      .                                           }
      .                                           }
      .                                           }
19   LBL 0                                        }  Subprograms on pages 8-21 and
20   LBL 2                                        }  8-22 are inserted here
      .                                           }
      .                                           }
      .                                           }
24   LBL 0                                        }
25   END PGM 360821 MM
```

Subprograms: Overlapping pockets

The pocket elements A and B overlap.
The control automatically calculates the points of intersection S1 and S2, so these points do not have to be programmed.
The pockets are programmed as full circles.

```
15   LBL 1                }
16   L X+10 Y+50 RL       }   A   Left pocket
17   CC X+35 Y+50         }
18   C X+10 Y+50 DR+      }
19   LBL 0
20   LBL 2                }
21   L X+90 Y+50 RL       }
22   CC X+65 Y+50         }   B   Right pocket
23   C X+90 Y+50 DR+      }
24   LBL 0
25   END PGM 360821 MM
```

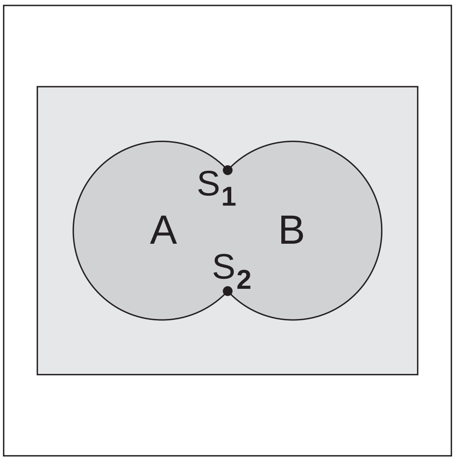


Fig. 8.17: Points of intersection S_1 and S_2 of pockets A and B

Depending on the control setup (machine parameters), machining starts either with the outline or the surface:

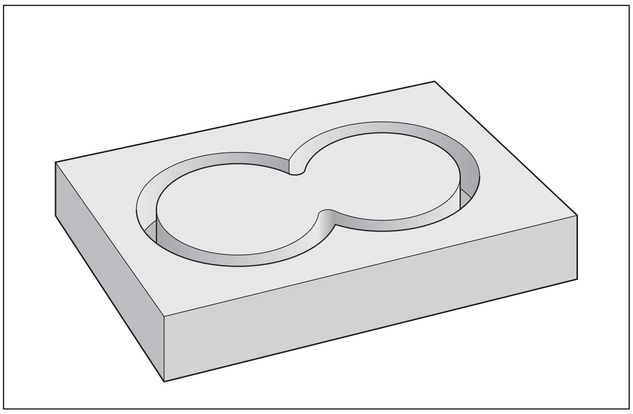


Fig. 8.18: Outline is machined first

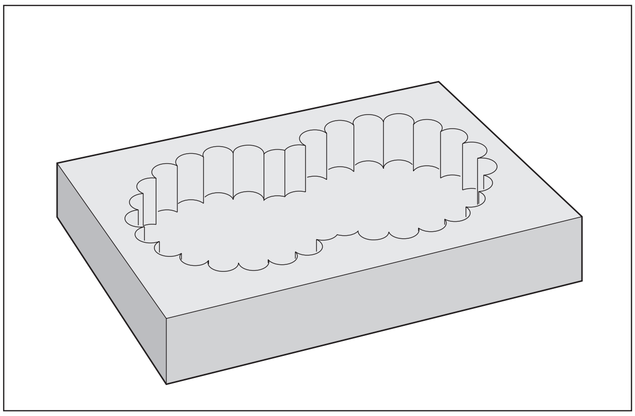


Fig. 8.19: Surface is machined first

Area of inclusion

Both areas (element A and element B) are to be machined — including the area of overlap.

- A and B must be pockets.
- The first pocket (in cycle 14) must start outside the second.

```
15  LBL 1
16  L X+10 Y+50 RL
17  CC X+35 Y+50
18  C X+10 Y+50 DR+
19  LBL 0

20  LBL 2
21  L X+90 Y+50 RL
22  CC X+65 Y+50
23  C X+90 Y+50 DR+
24  LBL 0
```

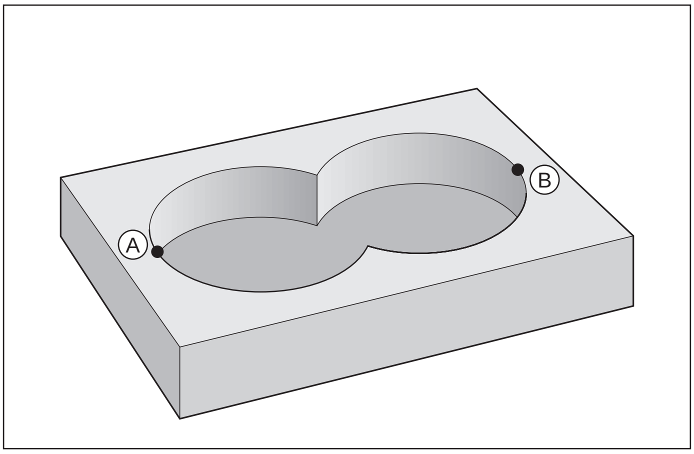


Fig. 8.20: Overlapping pockets: area of inclusion

Area of exclusion

Surface A is to be machined without the portion overlapped by B:

- A must be a pocket and B an island.
- A must start outside of B.

```
15  LBL 1
16  L X+10 Y+50 RL
17  CC X+35 Y+50
18  C X+10 Y+50 DR+
19  LBL 0

20  LBL 2
21  L X+90 Y+50 RR
22  CC X+65 Y+50
23  C X+90 Y+50 DR+
24  LBL 0
```

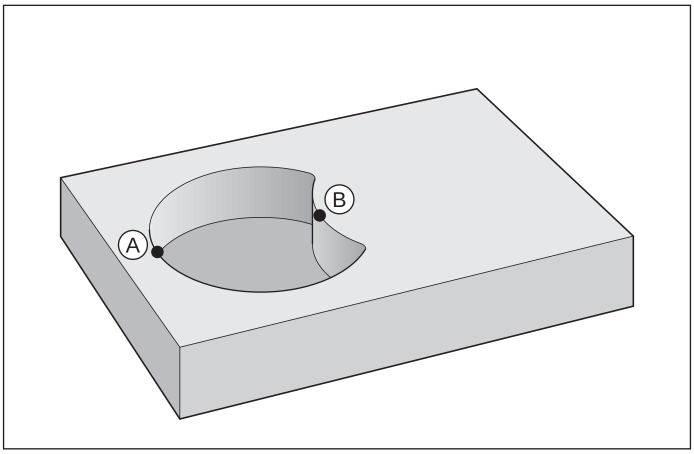


Fig. 8.21: Overlapping pockets: area of exclusion

Area of intersection

Only the area of intersection of A and B is to be machined.

- A and B must be pockets.
- A must start inside B.

```
15  LBL 1
16  L X+60 Y+50 RL
17  CC X+35 Y+50
18  C X+60 Y+50 DR+
19  LBL 0

20  LBL 2
21  L X+90 Y+50 RL
22  CC X+65 Y+50
23  C X+90 Y+50 DR+
24  LBL 0
```

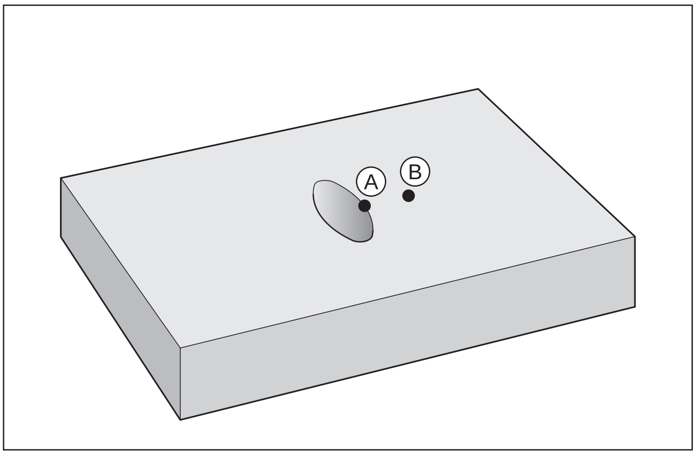


Fig. 8.22: Overlapping pockets: area of intersection

The subprograms are used in the main program on page 8-21.

Subprograms: Overlapping islands

An island always requires a pocket as an additional boundary (here, LBL 1). A pocket can also reduce several island surfaces. The starting point of this pocket must be within the first island. The starting points of the remaining intersecting island contours must lie outside the pocket.

```
0     BEGIN PGM 360823 MM
1     BLK FORM 0.1 Z X+0 Y+0 Z–20
2     BLK FORM 0.2 X+100 Y+100 Z+0
3     TOOL DEF 1 L+0 R+2.5
4     TOOL CALL 1 Z S1000
5     CYCL DEF 14.0 CONTOUR GEOM.
6     CYCL DEF 14.1 CONTOUR LABEL 2/3/1
7     CYCL DEF 6.0 ROUGH-OUT
8     CYCL DEF 6.1 SET UP –2 DEPTH –10
9     CYCL DEF 6.2 PECKG –5 F500 ALLOW +0
10    CYCL DEF 6.3 ANGLE +0 F500
11    L Z+100 R0 FMAX M6
12    L X+50 Y+50 FMAX M3
13    L Z+2 FMAX M99
14    L Z+100 FMAX M2
15    LBL 1
16    L X+5 Y+5 RL
17    L X+95
18    L Y+95
19    L X+5
20    L Y+5
21    LBL 0
22    LBL 2
       .
       .
       .
26    LBL 0
27    LBL 3
       .
       .
       .
31    LBL 0
32    END PGM 360823 MM
```

Area of inclusion

Elements A and B are to be left unmachined including the mutually overlapped surface:

- A and B must be islands.
- The first island must start outside the second island.

```
22    LBL 2
23    L X+10 Y+50 RR
24    CC X+35 Y+50
25    C X+10 Y+50 DR+
26    LBL 0
27    LBL 3
28    L X+90 Y+50 RR
29    CC X+65 Y+50
30    C X+90 Y+50 DR+
31    LBL 0
32    END PGM 360823 MM
```

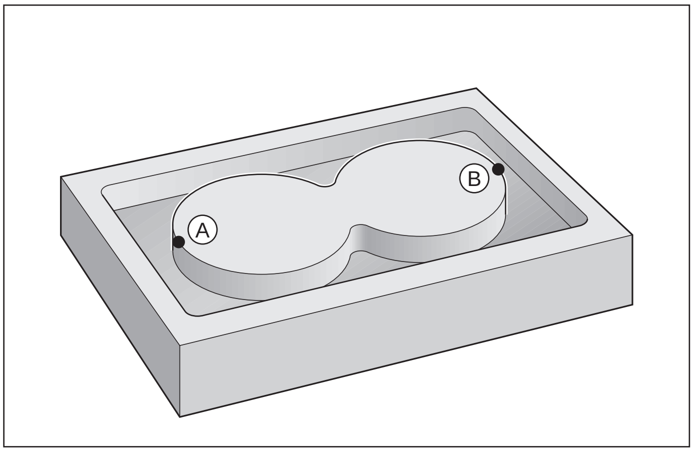


Fig. 8.23: Overlapping islands: area of inclusion

The supplements and subprograms are entered in the main program on page 8-23.

Area of exclusion

All of surface A is to be left unmachined except the portion overlapped by B:

- A must be an island and B a pocket.
- B must start inside A.

```
22   LBL 2
23   L X+10 Y+50 RR
24   CC X+35 Y+50
25   C X+10 Y+50 DR+
26   LBL 0
27   LBL 3
28   L X+40 Y+50 RL
29   CC X+65 Y+50
30   C X+40 Y+50 DR+
31   LBL 0
32   END PGM 360823 MM
```

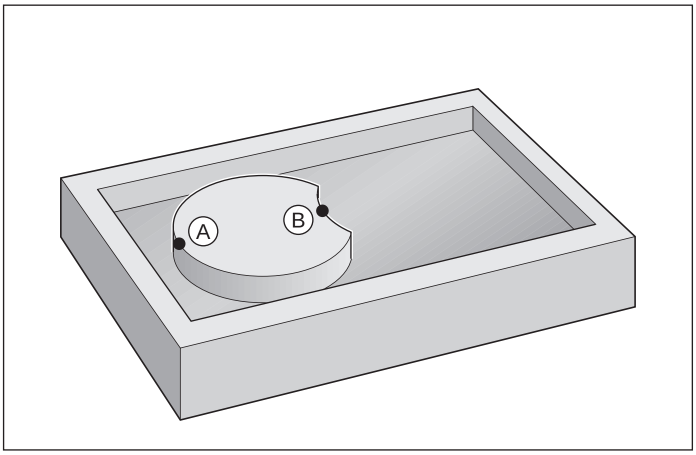


Fig. 8.24: Overlapping islands: area of exclusion

Area of intersection

Only the area of intersection of A and B is to remain unmachined.

- A and B must be islands.
- A must start inside B.

```
22   LBL 2
23   L X+60 Y+50 RR
24   CC X+35 Y+50
25   C X+60 Y+50 DR+
26   LBL 0
27   LBL 3
28   L X+90 Y+50 RR
29   CC X+65 Y+50
30   C X+90 Y+50 DR+
31   LBL 0
32   END PGM 360823 MM
```

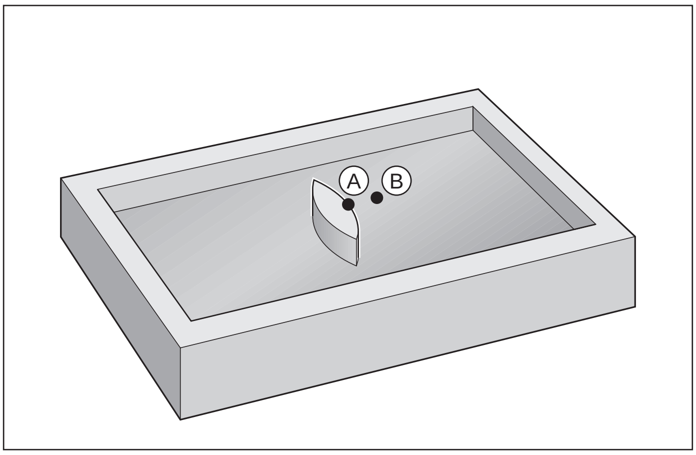


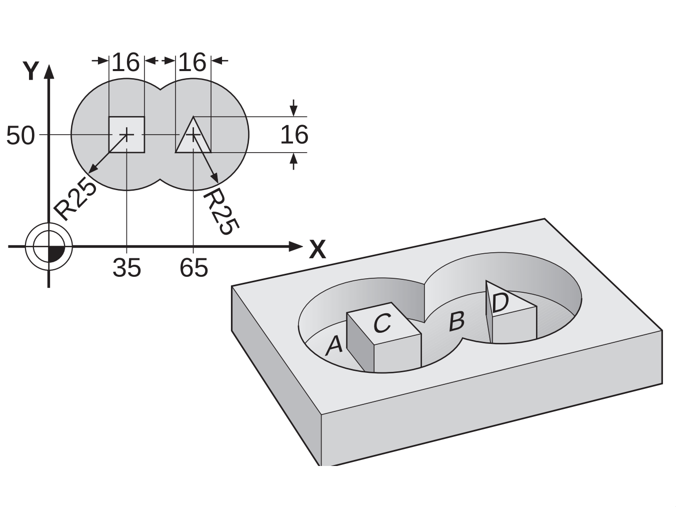
Fig. 8.25: Overlapping islands: area of intersection

Example: Overlapping pockets and islands

PGM 360825 is an expansion of PGM 360821 for the inside islands C and D.

Tool: Center-cut end mill (ISO 1641), radius 3 mm.

The SL contour is composed of the elements A and B (two overlapping pockets) as well as C and D (two islands within these pockets).



Cycle in a part program

```
0    BEGIN PGM 360825 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+3

4    CYCL DEF 14.0 CONTOUR GEOM.
5    CYCL DEF 14.1 CONTOUR LABEL 1/2/3/4

6    CYCL DEF 6.0 ROUGH-OUT
7    CYCL DEF 6.1 SET UP –2 DEPTH –10
8    CYCL DEF 6.2 PECKG –5 F100 ALLOW +2
9    CYCL DEF 6.3 ANGLE +0 F100

10   TOOL CALL 1 Z S1000

11   L Z+2 R0 FMAX M3

12   CYCL CALL

13   L Z+100 R0 FMAX M2

14   LBL 1
15   L X+10 Y+50 RL
16   CC X+35 Y+50
17   C X+10 Y+50 DR+
18   LBL 0

19   LBL 2
20   L X+90 Y+50 RL
21   CC X+65 Y+50
22   C X+90 Y+50 DR+
23   LBL 0

24   LBL 3
25   L X+27 Y+42 RL
26   L Y+58
27   L X+43
28   L Y+42
29   L X+27
30   LBL 0

31   LBL 4
32   L X+57 Y+42 RR
33   L X+73
34   L X+65 Y+58
35   L X+57 Y+42
36   LBL 0

37   END PGM 360825 MM
```

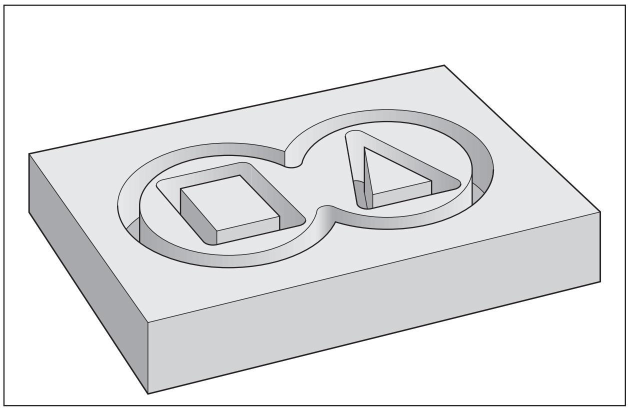


Fig. 8.26: Milling the outlines

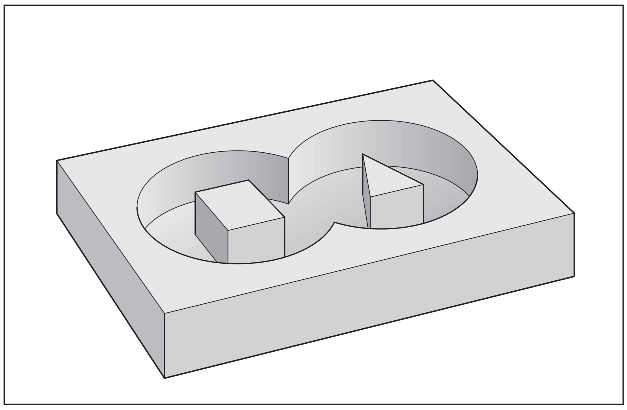


Fig. 8.27: Milling completed

PILOT DRILLING (Cycle 15)

Process

Pilot drilling of holes for cutter infeed at the starting points of the subcontours. With SL contours that consist of several overlapping surfaces, the cutter infeed point is the starting point of the first subcontour:

- The tool is positioned above the first infeed point.
- The subsequent drilling sequence is identical to that of cycle 1 PECKING.
- The tool is then positioned above the next infeed point, and the drilling process is repeated.

Input data

- SETUP CLEARANCE
- TOTAL HOLE DEPTH
- PECKING DEPTH
- DWELL TIME
- FEED RATE

} Identical to cycle 1 PECKING

- FINISHING ALLOWANCE
  Allowed material for the drilling operation (see Fig. 8.29).
  The sum of the tool radius and finishing allowance should be the same for pilot drilling and roughing out.

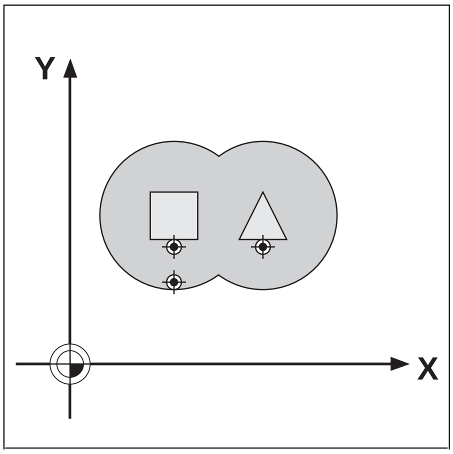


Fig. 8.28: Example of cutter infeed points for PECKING

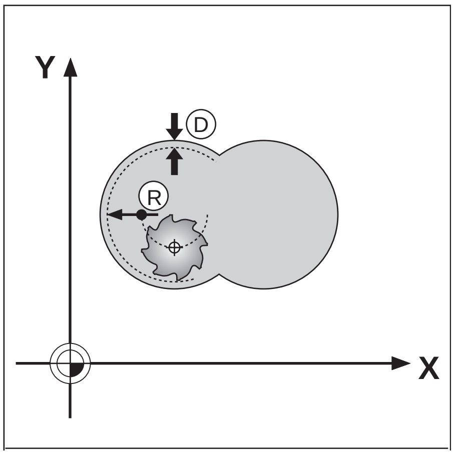


Fig. 8.29: Finishing allowance

CONTOUR MILLING (Cycle 16)

Cycle 16 CONTOUR MILLING is used to finish-mill the contour pocket. This cycle can also be used generally for milling contours.

Process

- The tool is positioned above the first starting point.
- The tool then penetrates at the programmed feed rate to the first pecking depth.
- On reaching the first pecking depth, the tool mills the first contour at the programmed feed rate and in the specified direction of rotation.
- At the infeed point, the tool is advanced to the next pecking depth.

This process is repeated until the programmed milling depth is reached. The remaining subcontours are milled in the same manner.

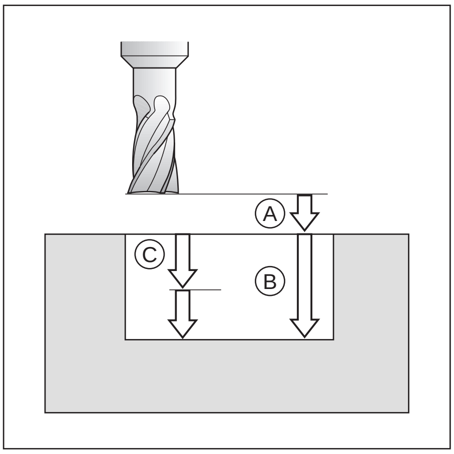


Fig. 8.30: Infeeds and distances for CONTOUR MILLING

Required tool

This cycle requires a center-cut end mill (ISO 1641).

Input data

- SETUP CLEARANCE Ⓐ
- MILLING DEPTH Ⓑ
- PECKING DEPTH Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration.
- DIRECTION OF ROTATION FOR CONTOUR MILLING:
  The following is valid for M3:
  DR+: Climb milling for pocket and island
  DR–: Up-cut milling for pocket and island
- FEED RATE:
  Traversing speed of the tool in the machining plane.

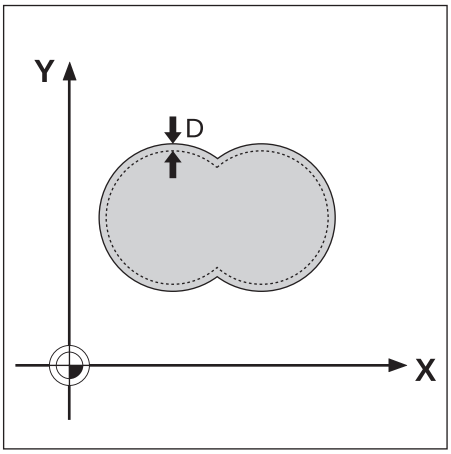


Fig. 8.31: Finishing allowance

The following scheme illustrates the application of the cycles Pilot Drilling, Rough-Out and Contour Milling in part programming:

1. **List of contour subprograms**

   CYCL DEF 14.0 CONTOUR GEOM.
   Cycle call not required.

2. **Drilling**

   Define and call the drilling
   CYCL DEF 15.0 PILOT DRILLING
   Pre-positioning
   Cycle call required!

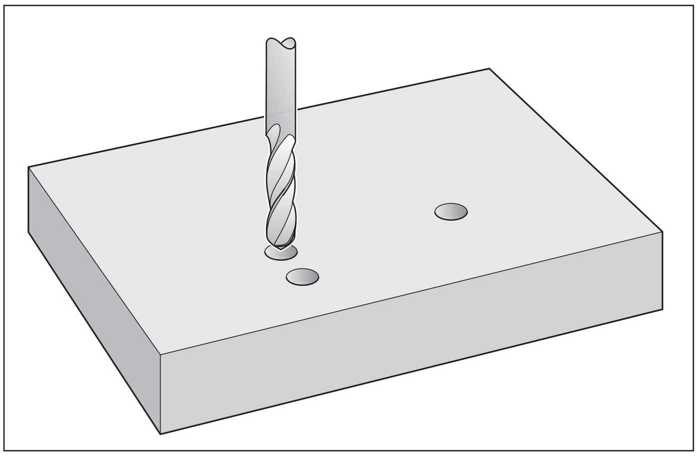


Fig. 8.32: PILOT DRILLING cycle

3. **Rough-out**

   Define and call tool for rough milling
   CYCL DEF 6.0 ROUGH-OUT
   Pre-positioning
   Cycle call required!

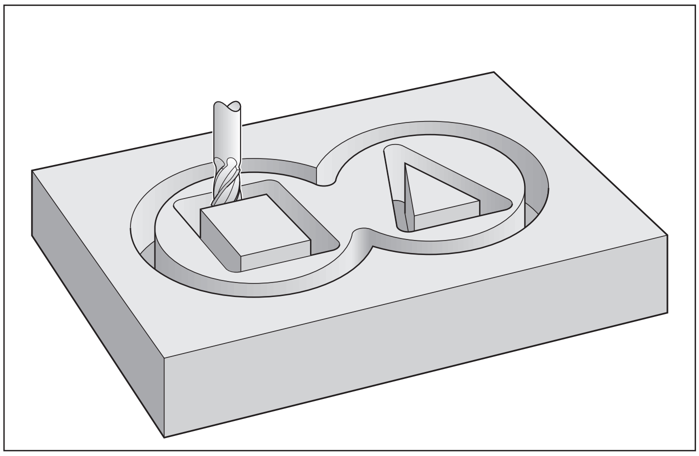


Fig. 8.33: ROUGH-OUT cycle

4. **Finishing**

   Define and call finish milling tool
   CYCL DEF 16.0 CONTOUR MILLING
   Pre-positioning
   Cycle call required!

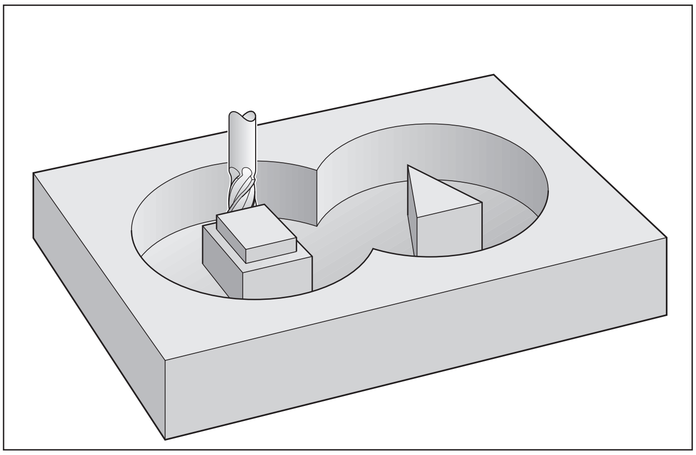


Fig. 8.34: CONTOUR MILLING cycle

5. **Contour subprograms**

   STOP M02
   Subprograms for the subcontours.

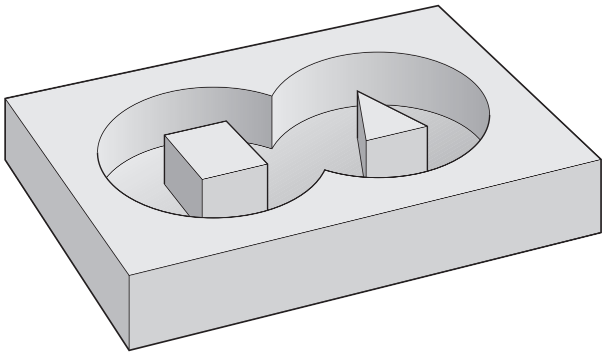
Example: Overlapping pockets with islands

Inside machining with pilot drilling, roughing out and finishing.

PGM 360830 is based on 360825:

The main program has been expanded by the cycle definition and cycle calls for pilot drilling and finishing.

The contour subprograms 1 to 4 are identical to those in PGM 360825 (see page 8-25) and are to be added after block 39.



```
0    BEGIN PGM 360830 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+2.2 .................................... Drill
4    TOOL DEF 2 L+0 R+3 ...................................... Rough mill
5    TOOL DEF 3 L+0 R+2.5 .................................... Finish mill
6    CYCL DEF 14.0 CONTOUR GEOM.
7    CYCL DEF 14.1 CONTOUR LABEL 1/2/3/4
8    CALL LBL 10

9    STOP M6
10   TOOL CALL 1 Z S 2000
11   CYCL DEF 15.0 PILOT DRILL                   }
12   CYCL DEF 15.1 SET UP –2 DEPTH –10           } Pilot drilling
13   CYCL DEF 15.2 PECKG –5 F500 ALLOW +2.8      }
14   L Z+2 R0 FMAX
15   CYCL CALL M3
16   CALL LBL 10

17   STOP M6
18   TOOL CALL 2 Z S 1750
19   CYCL DEF 6.0 ROUGH-OUT                      }
20   CYCL DEF 6.1 SET UP –2 DEPTH –10            }
21   CYCL DEF 6.2 PECKG –5 F100 ALLOW +2         } Rough out
22   CYCL DEF 6.3 ANGLE +0 F500                  }
23   L Z+2 R0 FMAX                               }
24   CYCL CALL M3
25   CALL LBL 10

26   STOP M6
27   TOOL CALL 3 Z S 2500
28   CYCL DEF 16.0 CONTOUR MILLING               }
29   CYCL DEF 16.1 SET UP –2 DEPTH –10           } Finishing
30   CYCL DEF 16.2 PECKG –5 F100 DR– F500        }
31   L Z+2 R0 FMAX
32   CYCL CALL M3
33   CALL LBL 10
34   L Z+20 R0 FMAX M2 ....................................... Retract and rapid return

35   LBL 10
36   TOOL CALL 0 Z ........................................... Tool change
37   L Z+100 R0 FMAX
38   L X–20 Y–20 R0 FMAX
39   LBL 0
```

From block 40: add the subprograms listed on page 8-25

```
63   END PGM 360830 MM
```

8.4 Cycles for Coordinate Transformations

Coordinate transformations enable a programmed contour to be changed in its position, orientation or size. A contour can be:

- shifted (cycle 7 DATUM SHIFT)
- mirrored (cycle 8 MIRROR IMAGE)
- rotated (cycle 10 ROTATION)
- made smaller or larger (cycle 11 SCALING)

The original contour must be identified as a subprogram or program section.

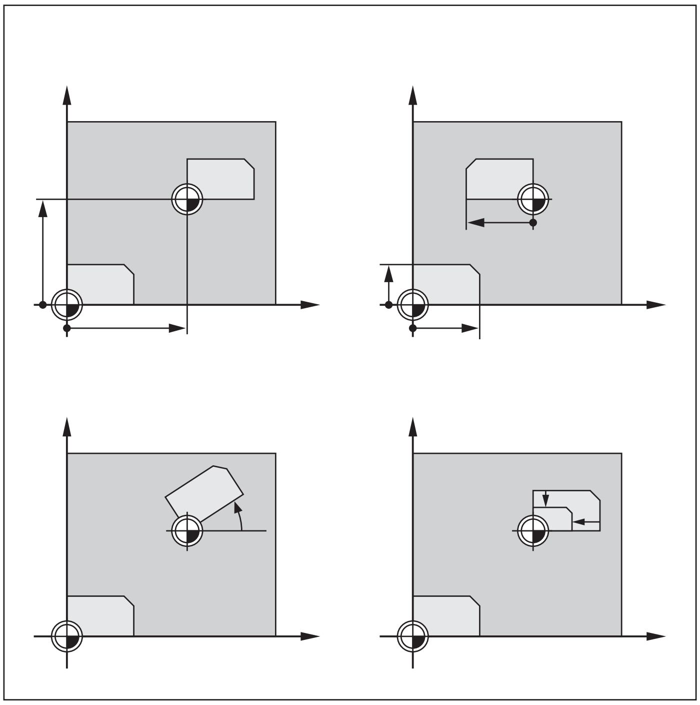


Fig. 8.35: Examples of coordinate transformations

Activation of coordinate transformation

Immediate activation: A coordinate transformation becomes effective as soon as it is defined (it does not have to be called). The transformation remains effective until it is changed or cancelled.

To cancel a coordinate transformation:

- Define cycle for basic behavior with new values, such as scaling factor 1.0
- Execute miscellaneous function M02, M30 or END PGM block (depending on machine parameters)
- Select a new program

DATUM SHIFT (Cycle 7)

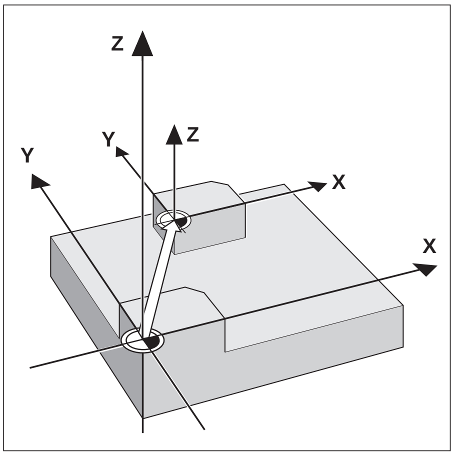


Fig. 8.36: Activation of the datum shift

Application

With the aid of a datum shift, machining operations can be repeated at various locations on the workpiece.

Activation

When the DATUM SHIFT cycle has been defined, all coordinate data are based on the new datum. Shifted axes are identified in the status display by the letter N.

Input data

Only the coordinates of the new datum need to be entered. Absolute values are based on the workpiece datum manually defined with datum setting. Incremental values are based on the last valid datum; this datum can itself be shifted.

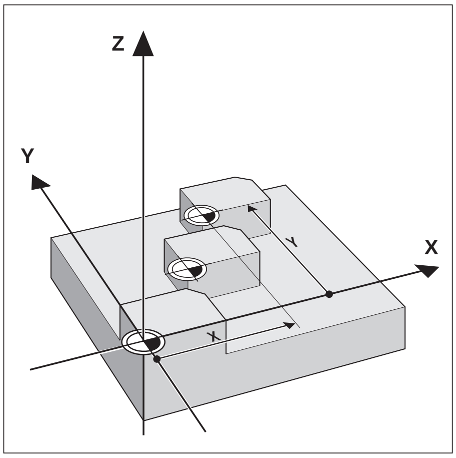


Fig. 8.37: Datum shift, absolute

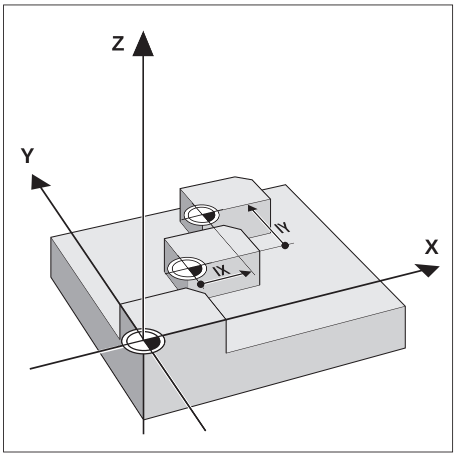


Fig. 8.38: Datum shift, incremental

Cancellation

To cancel a datum shift, enter the datum shift coordinates X = 0, Y = 0 and Z = 0.

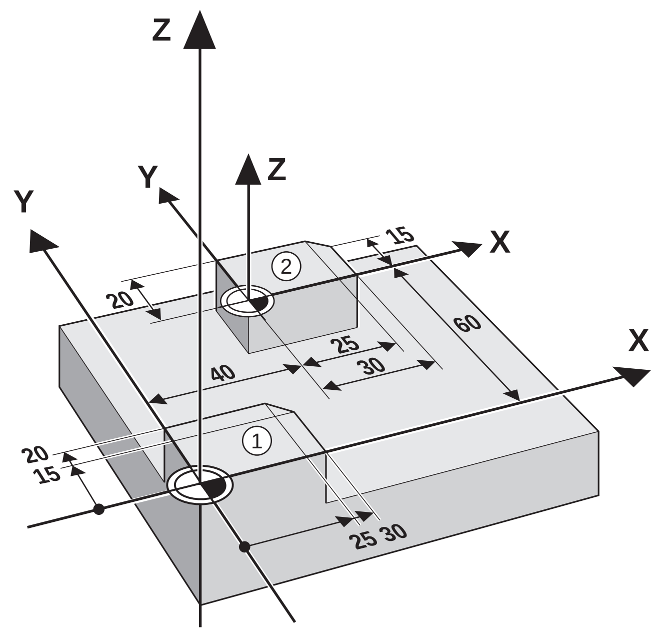
When combining transformations, program the datum shift first.

Example: Datum shift

A machining sequence in the form of a subprogram is to be executed twice:

a) once, referenced to the specified datum ① X+0/Y+0 and

b) a second time, referenced to the shifted datum ② X+40/Y+60.



Cycle in a part program

```
0    BEGIN PGM 360833 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+4
4    TOOL CALL 1 Z S1000
5    L Z+100 R0 FMAX
6    CALL LBL 1 ........................................................ Without a datum shift
7    CYCL DEF 7.0 DATUM SHIFT
8    CYCL DEF 7.1 X+40
9    CYCL DEF 7.2 Y+60
10   CALL LBL 1 ........................................................ With a datum shift
11   CYCL DEF 7.0 DATUM SHIFT ............................ Cancellation of datum shift
12   CYCL DEF 7.1 X+0
13   CYCL DEF 7.2 Y+0
14   L Z+100 R0 FMAX M2
15   LBL 1                                  }
16   L X–10 Y–10 R0 FMAX M3                 }
17   L Z+2 FMAX                             }
18   L Z–5 F200                             }
19   L X+0 Y+0 RL                           }
20   L Y+20                                 }
21   L X+25                                 } Subprogram for the geometry of the original contour
22   L X+30 Y+15                            }
23   L Y+0                                  }
24   L X+0                                  }
25   L X–10 Y–10 R0                         }
26   L Z+2 FMAX                             }
27   LBL 0
28   END PGM 360833 MM
```

The location of the subprogram (NC block) depends on the transformation cycle:

	LBL 1	**LBL 0**
Datum shift	Block 15	Block 27
Mirror image, rotation, scaling	Block 19	Block 31

MIRROR IMAGE (Cycle 8)

Application

This cycle makes it possible to machine the mirror image of a contour in the machining plane.

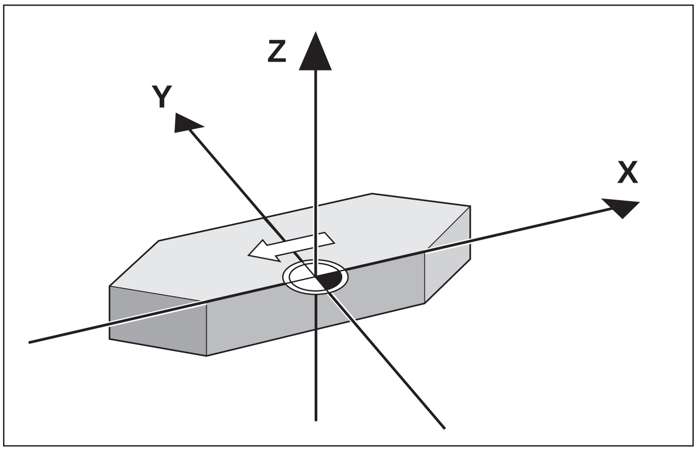


Fig. 8.39: MIRROR IMAGE cycle

Activation

The Mirror Image cycle becomes active as soon as it is defined:
Mirrored axes are identified in the status display by the letter S.

- If one axis is mirrored, the machining direction of the tool is reversed (this holds only for machining cycles).
- If two axes are mirrored, the machining direction remains the same.

The mirror image depends on the location of the datum:

- If the datum is located **on** the mirrored contour, the part "flips over."
- If the datum is located **outside** the mirrored contour, the part flips over and also moves to another location.

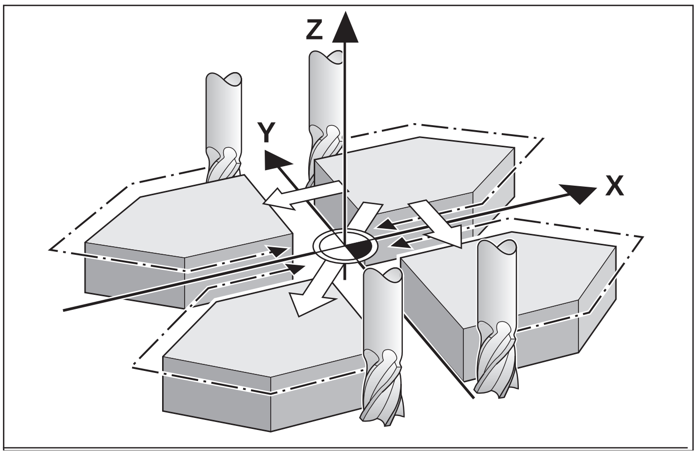


Fig. 8.40: Multiple mirroring and milling direction

Input data

Enter the axis that you wish to mirror. The tool axis cannot be mirrored.

Cancellation

To cancel a mirror image, answer the dialog query with NO ENT.

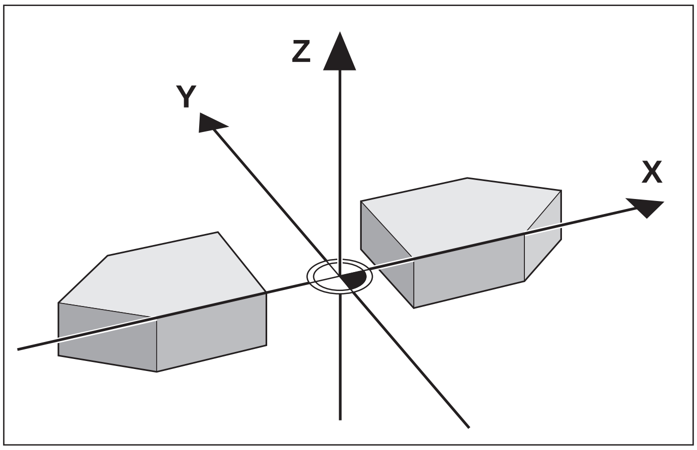
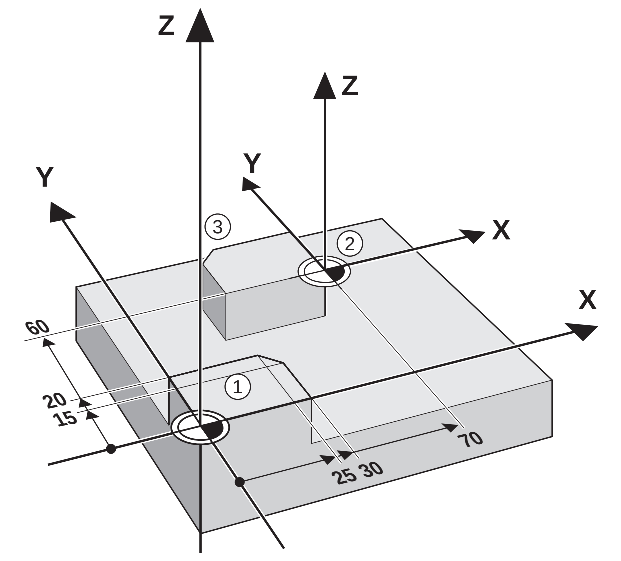


Fig. 8.41: Datum lies outside the mirrored contour

Example: Mirror image

A machining sequence (subprogram 1) is to be executed once – as originally programmed – referenced to the datum X+0/Y+0 ①and then again referenced to X+70/Y+60 ② mirrored ③ in X.



MIRROR IMAGE cycle in a part program

```
0     BEGIN PGM 360836 MM
1     BLK FORM 0.1 Z X+0 Y+0 Z–20
2     BLK FORM 0.2 X+100 Y+100 Z+0
3     TOOL DEF 1 L+0 R+4
4     TOOL CALL 1 Z S1000
5     L Z+100 R0 FMAX
6     CALL LBL 1 ........................................... Not mirrored ①; mirrored execution sequence:
7     CYCL DEF 7.0 DATUM ................................... 1. Datum shift ②
8     CYCL DEF 7.1 X+70
9     CYCL DEF 7.2 Y+60
10    CYCL DEF 8.0 MIRROR IMAGE ............................ 2. Mirror image ③
11    CYCL DEF 8.1 X
12    CALL LBL 1 ........................................... 3. Subprogram call
13    CYCL DEF 8.0 MIRROR IMAGE ............................ Cancel mirror image
14    CYCL DEF 8.1
15    CYCL DEF 7.0 DATUM ................................... Cancel datum shift
16    CYCL DEF 7.1 X+0
17    CYCL DEF 7.2 Y+0

18    L Z+100 R0 FMAX M2

19    LBL 1                          ┐
20    L X–10 Y–10 R0 FMAX M3         │
21    L Z+2 FMAX                     │
22    L Z–5 F200                     │
23    L X+0 Y+0 RL                   │
24    L Y+20                         │  This subprogram is identical to the subpro-
25    L X+25                         │  gram on page 8-32
26    L X+30 Y+15                    │
27    L Y+0                          │
28    L X+0                          │
29    L X–10 Y–10 R0                 │
30    L Z+2 FMAX                     ┘
31    LBL 0
32    END PGM 360836 MM
```

ROTATION (Cycle 10)

Application

Within a program the coordinate system can rotated about the active datum in the working plane.

Activation

A rotation becomes active as soon as the cycle is defined. This cycle is also effective in the POSITIONING WITH MANUAL INPUT mode.

Reference axis for the rotation angle:

• X/Y plane	X-axis
• Y/Z plane	Y-axis
• Z/X plane	Z-axis

The active rotation angle is indicated in the status display with ROT.

Input data

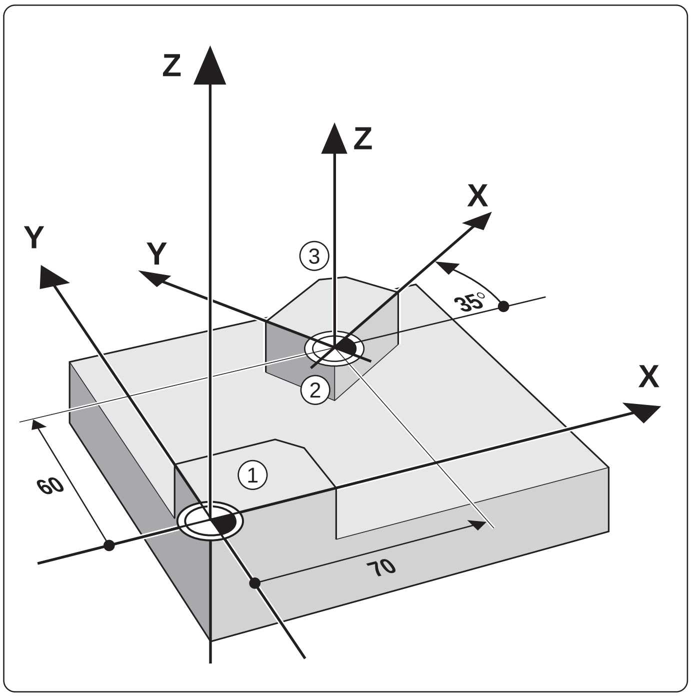
The angle of rotation is entered in degrees (°).
Entry range: –360° to +360° (absolute or incremental)

Cancellation

To cancel a rotation, enter a rotation angle of 0°.

Example: Rotation

A contour (subprogram 1) is to be executed once – as originally programmed – referenced to the datum X+0/Y+0 and then executed again referenced to X+70 Y+60 and rotated by 35°.



Continued...

Cycle in a part program

```
0    BEGIN PGM 360838 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+5
4    TOOL CALL 1 Z S1000
5    L Z+100 R0 FMAX
6    CALL LBL 1 ........................................................ Non-rotated execution ①
7    CYCL DEF 7.0 DATUM ...................................... Rotated execution. Sequence:
8    CYCL DEF 7.1 X+70
9    CYCL DEF 7.2 Y+60 .......................................... 1. Datum shift ②
10   CYCL DEF 10.0 ROTATION ............................... 2. Rotation ③
11   CYCL DEF 10.1 ROT +35
12   CALL LBL 1 ........................................................ 3. Subprogram call
13   CYCL DEF 10.0 ROTATION ............................... Cancel rotation
14   CYCL DEF 10.1 ROT 0
15   CYCL DEF 7.0 DATUM ...................................... Cancel datum shift
16   CYCL DEF 7.1 X+0
17   CYCL DEF 7.2 Y+0
18   L Z+100 R0 FMAX M2
19   LBL 1
       .
       .
       .
     LBL 0
     END PGM 360838 MM
```

The corresponding subprogram (see page 8-32) is programmed after M02.

SCALING FACTOR (Cycle 11)

Application

This cycle allows you to increase or reduce the size of contours within a program, such as for shrinkage or finishing allowances.

Activation

A scaling factor becomes effective as soon as the cycle is defined.
Scaling factors can be applied

- in the machining plane, or to all three coordinate axes at the same time (depending on MP7410)
- to the dimensions in cycles
- also in the parallel axes U, V, W

The scaling factor is indicated in the status display with SCL.

Input data

The cycle is defined by entering the scaling factor SCL. The TNC multiplies the coordinates and radii by the SCL factor (as described under "Activation" above).

To increase the size:	enter SCL greater than 1 (max. 99.999 999)
To reduce the size:	enter SCL less than 1 (down to 0.000 001)

Cancellation

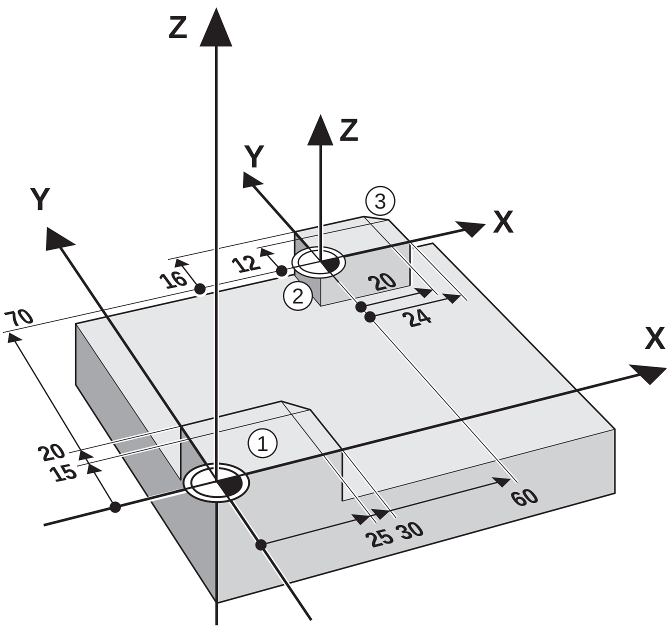
To cancel a scaling factor, enter a scaling factor of 1.

Prerequisite

Before entering a scaling factor it is advisable to set the datum to an edge or corner of the contour.

Example: Scaling factor

A contour (subprogram 1) is to be executed once – as originally programmed – at the manually set datum X+0/Y+0 and then executed again referenced to the position X+60/Y+70 and with a scaling factor of 0.8.



SCALING FACTOR cycle in a part program

```
0    BEGIN PGM 360839 MM
1    BLK FORM 0.1 Z X+0 Y+0 Z–20
2    BLK FORM 0.2 X+100 Y+100 Z+0
3    TOOL DEF 1 L+0 R+5
4    TOOL CALL 1 Z S1000
5    L Z+100 R0 FMAX
6    CALL LBL 1 ........................................................ Execution at original size ①
7    CYCL DEF 7.0 DATUM ....................................... Execution with scaling factor. Sequence:
8    CYCL DEF 7.1 X+60
9    CYCL DEF 7.2 Y+70 .......................................... 1. Datum shift ②
10   CYCL DEF 11.0 SCALING ................................... 2. Define scaling factor ③
11   CYCL DEF 11.1 SCL 0.8
12   CALL LBL 1 ........................................................ 3. Call subprogram (scaling factor active)
13   CYCL DEF 11.0 SCALING ................................... Cancel transformations
14   CYCL DEF 11.1 SCL 1
15   CYCL DEF 7.0 DATUM
16   CYCL DEF 7.1 X+0
17   CYCL DEF 7.2 Y+0
18   L Z+100 R0 FMAX M2
19   LBL 1
20   L X–10 Y–10 R0 FMAX M3
21   L Z+2 FMAX
22   L Z–5 F200
23   L X+0 Y+0 RL
24   L Y+20
25   L X+25
26   L X+30 Y+15
27   L Y+0
28   L X+0
29   L X–10 Y–10 R0
30   L Z+2 FMAX
31   LBL 0
32   END PGM 360839 MM
```

8.5 Other Cycles

DWELL TIME (Cycle 9)

Application

Within a running program, the execution of the next block is delayed by the programmed dwell time.

The dwell time cycle can be used, for example, for chip breaking.

Activation

This cycle becomes effective as soon as it is defined. Modal conditions (such as a spindle rotation) are not affected.

Input data

A dwell time is entered in seconds.
Entry range: 0 to 30 000 s (approx. 8.3 hours) in increments of 0.001 s.

PROGRAM CALL (Cycle 12)

Application and activation

Part programs such as special drilling cycles, curve milling or geometric modules, can be written as main programs and then called for use just like fixed cycles.

Input data

Enter the file name of the program to be called.

The program is called with

- CYCL CALL (separate block) or
- M99 (blockwise) or
- M89 (modally)

Example: Program call

A callable program (program 50) is to be called into a program with a cycle call.

Part program

```
 .
 .
 .
CYCL DEF 12.0 PGM CALL ....................................... Definition:
CYCL DEF 12.1 PGM 50 ............................................ “Program 50 is a cycle”
L X+20 Y+50 FMAX M99 .......................................... Call of program 50
 .
 .
 .
```

ORIENTED SPINDLE STOP (Cycle 13)

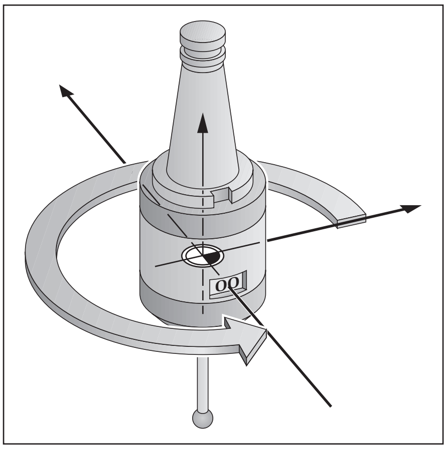


Fig. 8.42: Oriented spindle stop

Application

The control can address the machine tool spindle as a 5th axis and turn it to a certain angular position. Oriented spindle stops are required for:

- Tool changing systems with a defined tool change position
- Orientation of the transmitter/receiver window of the TS 511 Touch Probe System from HEIDENHAIN

Activation

The angle of orientation defined in the cycle is positioned to with M19. If M19 is executed without a cycle definition, the machine tool spindle will be oriented to the angle set in the machine parameters.

> Oriented spindle stops can also be programmed in machine parameters.

Prerequisite

The machine must be set up for this cycle.

Input data

Angle of orientation (based on the reference axis of the machining plane)

Input range: 0 to 360°
Input resolution: 0.1°

The digitizing option enables you to reduce a three-dimensional part into discrete digital information by scanning it with the TS 120 touch probe.

The following components are required for digitizing:
- TS 120 three-dimensional touch probe
- "Digitizing option" software module in the TNC
- External data storage, such as:
  HEIDENHAIN FE 401 floppy disk unit
  or
  PC (IBM-compatible) with HEIDENHAIN TNC.EXE data transfer software

The digitized surface data can be evaluated with the
- SUSA evaluation software for IBM-compatible PCs

.

The TNC and machine must have been prepared by the machine tool builder for the use of a 3D touch probe.

9.1 The Digitizing Process

The touch probe scans a 3D surface point-for-point in a selectable grid. The scanning speeds vary from 200 to 600 mm/min (about 8 to 24 ipm).

The TNC transmits the digitized positions as straight-line blocks in HEIDENHAIN format. The interface function PRINT (see page 7-15) determines where the blocks are stored:
- In the program memory of the TNC
- Externally via RS-232 interface

If very large amounts of data are generated, you will have to store them in a PC.

Generating programs with digitized data

The TNC automatically converts the digitized data into an NC part program. Such a program can be run without any additional processing provided that the cutter has the same radius as the probe stylus tip.

The HEIDENHAIN evaluation software SUSA calculates male/female transformations and tool paths for tool radii and tool shapes that differ from the shape of the probe stylus tip.

Overview: Digitizing cycles

The following digitizing cycles are available:

• RANGE	For defining the scanning range
• MEANDER	For digitizing line by line
• CONTOUR LINES	For digitizing contour lines

Transferring digitized data

The digitized data are stored externally in a file which you name in cycle 5: RANGE.

- The digitizing cycles operate in HEIDENHAIN conversational dialog.
- Digitizing cycles are programmed only for the axes X, Y and Z.
- Coordinate transformations or a basic rotation must not be active during digitizing.

9.2 Digitizing Range

The digitizing range is defined in cycle 5: RANGE. The model to be scanned must lie within this range. You also enter the name of the file for the digitized data as well as a clearance height for pre-positioning the touch probe.

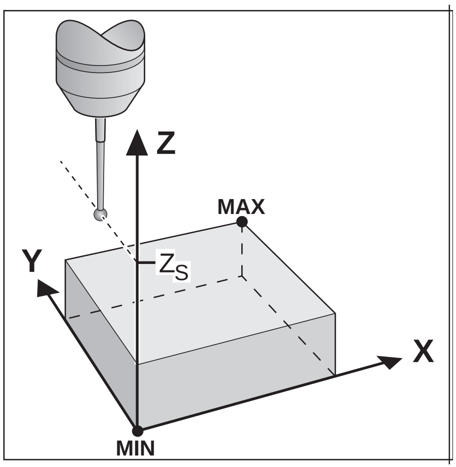


Fig. 9.1: Clearance height and digitizing range

Input data

- PGM NAME
  Name of the file in which the digitized data is to be stored
- MIN POINT RANGE
  Coordinates of the lowest point in the range to be digitized
- MAX POINT RANGE
  Coordinates of the highest point in the range to be digitized
- CLEARANCE HEIGHT
  Position in the probe axis at which the probe cannot collide with the model

Setting the scanning range

TOUCH PROBE ▶

TCH PROBE: 0 REF. PLANE

GOTO □ 5 ENT — Select digitizing cycle 5: RANGE.

TCH PROBE: 5 RANGE

ENT — Confirm selection.

PGM NAME?

Enter the name of the file in which the digitizing data are to be stored.

TCH PROBE AXIS?

e.g. Z — Enter the touch probe axis.

.
.
.

.
.
.

MIN. POINT RANGE ?	
e.g. 0 ENT 0 ENT 0 ENT	Enter in sequence the X, Y and Z coordinates of the range MIN point.

MAX. POINT RANGE?	
e.g. 1 0 ENT e.g. 1 0 ENT e.g. 2 0 ENT	Enter in sequence the X, Y and Z coordinates of the range MAX point.

CLEARANCE HEIGHT?	
e.g. 1 0 0 ENT	Enter the clearance height for the touch probe.

Resulting NC blocks:

```
  .
  .
  .
TCH  PROBE  5.0  RANGE
TCH  PROBE  5.1  PGM NAME: 5007
TCH  PROBE  5.2  Z   X+0 Y+0 Z+0
TCH  PROBE  5.3      X+10 Y+10 Z+20
TCH  PROBE  5.4  HEIGHT: + 100
  .
  .
  .
```

9.3 Line-By-Line Digitizing

The MEANDER cycle scans and digitizes a 3D contour in a back-and-forth ("meandering") series of parallel lines.

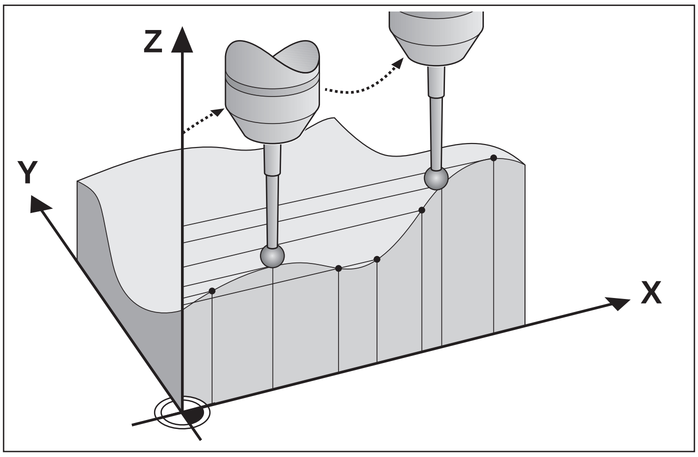


Fig. 9.2: Scanning a line on the 3D surface

Starting position

- Coordinates from the RANGE cycle:
  X and Y coordinates of the MIN point
  Z coordinate = CLEARANCE HEIGHT
- Automatically move to the starting position:
  first Z, then X and Y

Contour approach

The touch probe moves in the negative Z direction towards the model. Upon contact, the TNC stores the position.

Input data

- LINE DIRECTION
  Coordinate axis in whose positive direction the touch probe moves (beginning with the first contour point).
- LIMIT IN NORMAL LINES DIRECTION
  Distance the probe is retracted from the model after each deflection of the stylus during scanning
- LINE SPACING
  The offset by which the probe moves at the ends of the lines before scanning the next line
- MAX. PROBE POINT INTERVAL
  Maximum spacing between consecutive digitized positions

The LINE SPACING and MAX. PROBE POINT INTERVAL cannot exceed 5 mm.

The touch probe moves in the positive direction of the axis entered under LINE DIRECTION. When the probe reaches the MAX coordinate on this axis, it moves by the line spacing (L.SPAC) in the positive direction of the other axis in the working plane (i.e. in the column direction). It then moves back in the negative line direction, and at the other end moves again by the programmed line spacing.

This process is repeated until the entire range has been scanned.

While the probe is moving, the coordinates of the center of the probe tip are stored at intervals equal to or less than the programmed probe point interval.

When the entire range has been scanned, the touch probe returns to the CLEARANCE HEIGHT.

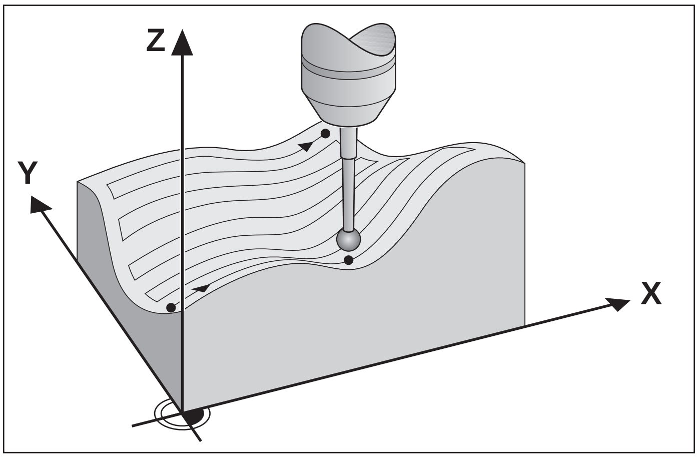


Fig. 9.3: Digitizing with the MEANDER cycle

Setting the digitizing parameters:

TOUCH PROBE ▶

TCH PROBE: 0 REF PLANE

GOTO □ 6 ENT — Select the digitizing cycle 6: MEANDER.

TCH PROBE: 6 MEANDER

ENT — Confirm your selection.

LINE DIRECTION ?

e.g. X ENT — Enter the line direction, for example X.

LIMIT IN NORMAL LINES DIRECTION ?

e.g. 0 . 5 ENT — Enter the distance by which the probe is to retract from the surface, for example 0.5 mm.

LINE SPACING ?

e.g. 0 . 2 ENT — Enter the desired line spacing, for example 0.2 mm.

.
.
.

.
.
.

MAX. PROBE POINT INTERVAL ?	
e.g. 0 . 8 ENT	Enter the maximum probe point interval, for example 0.8 mm.

Resulting NC blocks:

```
TCH   PROBE   6.0    MEANDER
TCH   PROBE   6.1    DIRECTN: X
TCH   PROBE   6.2    TRAVEL: 0.5 L.SPAC: 0.2
                     P.P. INT: 0.8
```

☞ Before cycle 6: MEANDER the program must have a range defined in digitizing cycle 5: RANGE.

9.4 Contour Line Digitizing

The CONTOUR LINES cycle scans a 3D contour by circling around the model in a series of upwardly successive levels.

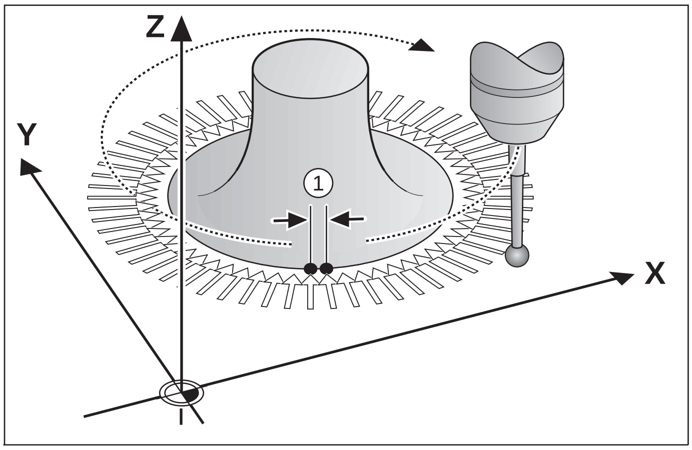


Fig. 9.4: Scanning one level of a 3D surface

Starting position

- Z coordinate of the MIN point from the RANGE cycle if the line spacing was entered as a positive value, or Z coordinate of the MAX point if the line spacing was entered as a negative value.
- Define the X and Y coordinates in the CONTOUR LINES cycle
- Automatically approach the starting point:
  first in Z to the CLEARANCE HEIGHT (from RANGE cycle), then in X and Y

Contour approach

The probe moves towards the surface in the programmed direction. When it makes contact, the TNC stores the position coordinates.

Input data

- TIME LIMIT
  The time within which the probe must orbit the model and reach the first probe point. If the time limit is exceeded, the control aborts the digitizing cycle. The input value 0 means there is no time limit.
- STARTING POINT
  Coordinates of the starting point in the plane perpendicular to the probe axis.
- AXIS AND DIRECTION OF APPROACH
  Coordinate axis and direction in which the probe approaches the model.
- STARTING PROBE AXIS AND DIRECTION
  Coordinate axis and direction in which the probe begins scanning the model.
- LIMIT IN NORMAL LINES DIRECTION
  Distance by which the probe is retracted from the model after a stylus deflection.
- LINE SPACING
  Offset by which the probe moves to start a new contour line. The algebraic sign determines the direction.
- MAX. PROBE POINT INTERVAL
  Maximum distance between digitized positions.

- The LINE SPACING and MAX. PROBE POINT INTERVAL cannot exceed 5 mm.
- After digitizing, the TNC moves the 3D touch probe back to the programmed STARTING POINT.

Limits of the scanning range

- In the touch probe axis:
  The defined range must be lower than the highest point of the 3D model by at least the radius of the probe tip.
- In the plane perpendicular to the touch probe:
  The defined range must be larger than the 3D model by at least the radius of the probe.

The probe starts scanning in the direction that was entered as the STARTING PROBE AXIS AND DIRECTION. The scanned positions are digitized at intervals equal to or less than the MAX. PROBE POINT interval.

When the probe has orbited the model and returned to the first probe point, it then moves in Z direction by the programmed LINE SPACING:

- Positive LINE SPACING:
  offset in positive Z direction
- Negative LINE SPACING:
  offset in negative Z direction

The probe must return to the coordinates of the first digitized position to within one-quarter of the programmed point spacing. The process is repeated until the entire range is scanned.

When the entire range has been scanned, the probe returns to the CLEARANCE HEIGHT.

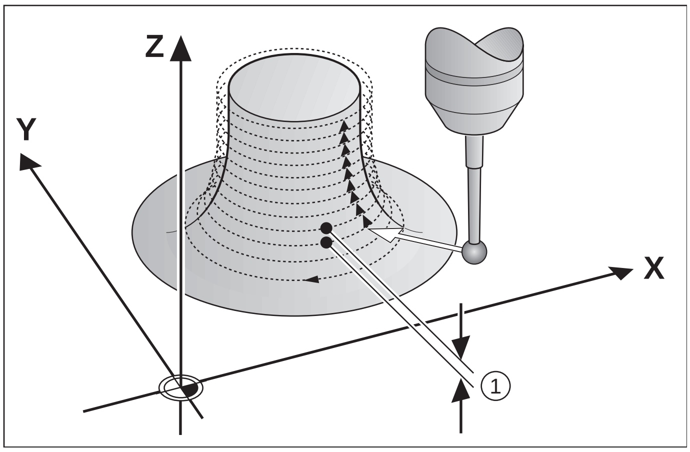


Fig. 9.5: Digitizing with the CONTOUR LINES cycle

Setting the digitizing parameters

TOUCH PROBE ▶

TCH PROBE: 0 REF. PLANE	
GOTO □ 7 ENT	Select digitizing cycle 7: CONTOUR LINES.

TCH PROBE: 7 CONTOUR LINES	
ENT	Confirm your selection.

TIME LIMIT ?	
e.g. 2 0 0 ENT	Enter the time limit, for example 200 seconds.

STARTING POINT ?	
X e.g. 5 0 Y e.g. 0 ENT	Enter the coordinates of the starting point (for example, X = 50 mm and Y = 0), and confirm your entry.

:

.
.
.

AXIS AND DIRECTION OF APPROACH ?

z.B. Y +/– ENT	Enter the approach direction, for example Y–.

STARTING PROBE AXIS AND DIRECTN?

e.g. X +/– ENT	Enter the starting direction, for example X+.

LIMIT IN NORMAL LINES DIRECTION ?

e.g. 0 . 5 ENT	Enter the distance the probe is retracted, for example 0.5 mm.

LINE SPACING AND DIRECTION ?

e.g. +/– ENT	Enter the line spacing, here 1 mm. The algebraic sign determines the direction in which the probe moves to start the next contour line.

MAX. PROBE POINT INTERVAL ?

e.g. 0 . 2 ENT	Enter the maximum probe point interval, for example 0.2 mm.

Resulting NC blocks:

```
TCH   PROBE   7.0    CONTOUR LINES
TCH   PROBE   7.1    TIME: 200 X+50 Y+0
TCH   PROBE   7.2    ORDER Y–/X+
TCH   PROBE   7.3    TRAVEL: 0.5 L.SPAC: –1 P.P. INT: 0.2
```

☞ Before Cycle 7: CONTOUR LINES, the program must have a range defined in digitizing Cycle 5: RANGE.

The TNC generates an NC part program from the digitized data. The program name is entered in the scanning cycle RANGE.

During machine execution, the tool radius determines the shape of the machined contour.

When tool radius equals the effective probe tip radius

The program can be run without any changes. The model that has been scanned is reproduced.

When the tool radius does not equal the effective probe tip radius

In this case the machined part will be either smaller or larger than the model. The HEIDENHAIN evaluation software SUSA can reproduce the original shape of workpiece models that were scanned with the meander scanning process.

9.5 Using Digitized Data in a Part Program

Program example with digitized data from the CONTOUR LINES cycle

```
0   BEGIN PGM DATA MM ..................................... Program name DATA.H is entered in the RANGE cycle
1   L Z+40 FMAX ........................................... Starting point in Z
2   L X+0 Y–25 FMAX ....................................... Starting point in X, Y
3   L X+2.005 Y–12.561 .................................... 1st digitized position
4   L X+2.025 Y–12.375 .................................... 2nd digitized position
.
.
.
    L X+2.005 Y–12.560 .................................... Contour line completed: probe has returned to first digitized
.                                                           position
.
.
    L Z+0.5 X+0 Y–10.423 .................................. 1st digitized position at the height of the new line
.
.
.
    L X+0 Y–12.560 ........................................ Last digitized position
.
.
.
    L X+0 Y–25 FMAX ....................................... Return to starting point in X, Y
    L Z+40 FMAX ........................................... Return to clearance height
    END PGM DATA MM ....................................... Program end
```

Note:

- The feed rate of the touch probe system for approaching the starting point and departing the end point is set in machine parameters for the touch probe.
- The program length is limited only by the capacity of the external storage device. After block 65535 the numbering begins again with 0.
- The probe scans the contour up to the next contour line.
- The TNC automatically marks the program beginning and end for data transfer.

Executing a part program from digitized data

Before the digitized data program can be transferred blockwise (see page 3-6) and executed, the TNC must receive the following information from another program:

- Tool radius and length
- Feed rate of tool
- Radius compensation
- Spindle axis and rpm
- Miscellaneous function for spindle

The program must contain the following five lines:

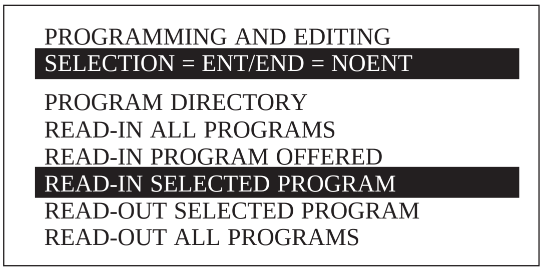
```
0   BEGIN PGM 444 MM .......................................... Any program number
1   TOOL DEF 1 L+30 R+4 ....................................... Tool
2   TOOL CALL 1 Z S1000 ....................................... Tool axis and spindle speed
3   L R0 F500 M3 .............................................. No radius compensation
4   L R F M xy ................................................ M xy: M function defined by the machine builder, through
                                                                      which the tool, feed rate and direction of spindle
                                                                      rotation remain effective even when a new program
                                                                      (the digitized data program) is selected.
5   END PGM 444 MM
```

At the end of the digitized data program generated by the CONTOUR LINES cycle, the tool is returned to the programmed starting point.

The TNC features an RS-232-C data interface for transferring data to and from other devices. It can be used in the PROGRAMMING AND EDITING operating mode and in a program run mode.

Possible applications:

- Blockwise transfer (DNC mode)
- Downloading program files into the TNC
- Transferring program files from the TNC to external storage devices
- Printing files



```
PROGRAMMING AND EDITING
SELECTION = ENT/END = NOENT

PROGRAM DIRECTORY
READ-IN ALL PROGRAMS
READ-IN PROGRAM OFFERED
READ-IN SELECTED PROGRAM
READ-OUT SELECTED PROGRAM
READ-OUT ALL PROGRAMS
```

Fig. 10.1: Menu for external data transfer

10.1 Menu for External Data Transfer

To select external data transfer:

EXT	Menu for external data transfer appears on the screen.

Use the arrow keys to select the individual menu options.

Function	**Menu option**
Display program numbers of the programs on the storage medium	PROGRAM DIRECTORY
Transfer all programs from the storage medium into the TNC	READ-IN ALL PROGRAMS
Display programs for transfer into the TNC	READ-IN PROGRAM OFFERED
Transfer selected program into the TNC	READ-IN SELECTED PROGRAM
Transfer selected program to an external device	READ-OUT SELECTED PROGRAM
Transfer all programs which are in TNC memory to an external device	READ-OUT ALL PROGRAMS

Aborting data transfer

To abort a data transfer process, press END.

If you are transferring data between two TNCs, the receiving control must be started first.

Blockwise transfer

In the operating modes PROGRAM RUN/FULL SEQUENCE and SINGLE BLOCK, it is possible to transfer programs which exceed the memory capacity of the TNC by means of blockwise transfer with simultaneous execution (see page 3-6).

10.2 Pin Layout and Connecting Cable for the Data Interface

RS-232-C/V.24 Interface

HEIDENHAIN devices

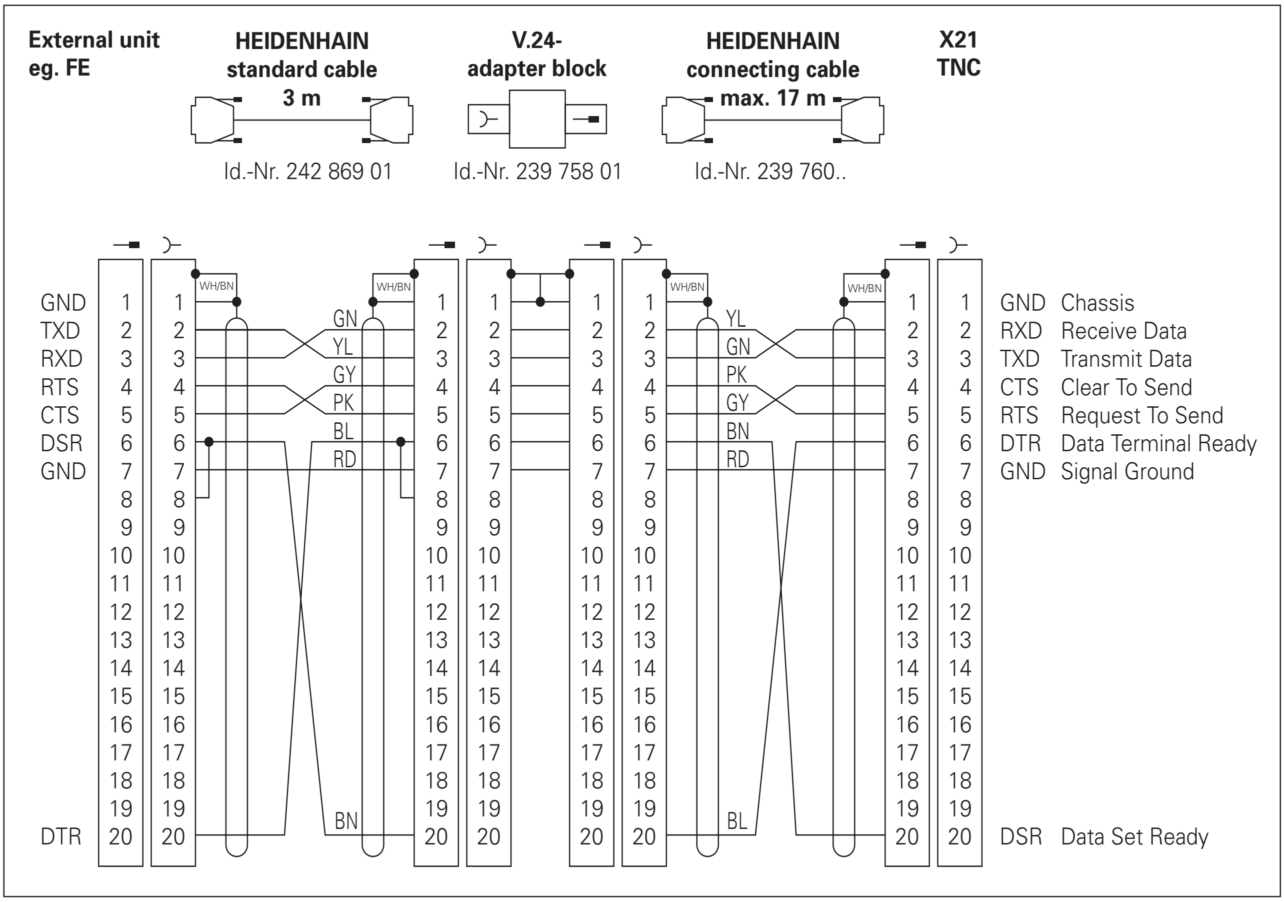


Fig. 10.2: Pin layout of the RS-232-C/V.24 interface for HEIDENHAIN devices

The connecting pin layout on the TNC logic unit (X25) is different from that on the adapter block.

Non-HEIDENHAIN devices

The connector pin layout on a non-HEIDENHAIN device may differ considerably from that on a HEIDENHAIN device. The pin layout will depend on the unit and the type of data transfer.

10.3 Preparing the Devices for Data Transfer

HEIDENHAIN Devices

HEIDENHAIN devices (FE floppy disk unit and ME magnetic tape unit) are designed for use with the TNC. They can be used for data transfer without further adjustments.

Example: FE 401 Floppy Disk Unit

- Connect the power cable to the FE
- Connect the FE and the TNC with data transfer cable
- Switch on the FE
- Insert a diskette into the upper drive
- Format the diskette if necessary
- Set the interface (see page 11-3)
- Transfer the data

> The baud rate can be selected on the FE 401 floppy disk unit.

Non-HEIDENHAIN devices

The TNC and non-HEIDENHAIN devices must be adapted to each other.

Adapting a non-HEIDENHAIN TNC

- PC: Adapt the software
- Printer: Adjust the DIP switches

Adapting the TNC for a non-HEIDENHAIN device

- Set user parameter 5020

The MOD functions provide additional displays and input possibilities. The MOD functions available depend on the selected operating mode.

Functions available in the operating modes PROGRAMMING AND EDITING and TEST RUN:

- Display NC software number
- Display PLC software number
- Enter code number
- Set the data interface
- Machine-specific user parameters
- Selection of axes for L block generation

Functions available in all other modes:

- Display NC software number
- Display PLC software number
- Select position display
- Select unit of measurement (mm/inch)
- Select programming language
- Set traverse limits
- Selection of axes for L block generation

11.1 Selecting, Changing and Exiting the MOD Functions

To select the MOD functions:

MOD	Select the MOD functions.

To change the MOD functions:

Select the desired MOD function with the arrow keys.

↑ / ↓ ENT Repeatedly	Page through the MOD functions until you find the desired function.

To exit the MOD functions:

END □	Close the MOD functions.

11.2 NC and PLC Software Numbers

The software numbers of the NC and PLC are displayed in the dialog field when the corresponding MOD function is selected.

11.3 Entering the Code Number

The TNC asks for a code number before allowing access to certain functions:

Function	Code number
Cancel erase/edit protection (status P)	86357
Select user parameters	123
Timers for: Control ON Program run Spindle ON	857282

Code numbers are entered in the dialog field after the corresponding MOD function is selected.

11.4 Setting the External Data Interfaces

Two functions are available for setting the external data interface:

- BAUD RATE
- RS-232 INTERFACE

Use the vertical arrow keys to select the functions.

BAUD RATE

The baud rate is the speed of data transfer in bits per second.

Permissible baud rates (enter with the numerical keys):
110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 baud

The ME 101 has a baud rate of 2400.

RS-232-C Interface

The proper setting depends on the connected device.

Use the ENT key to select the baud rate.

External device	RS-232-C interface =
HEIDENHAIN FE 401 and FE 401B floppy disk units	FE
HEIDENHAIN ME 101 magnetic tape unit (no longer in production)	ME
Non-HEIDENHAIN units such as printers, tape punchers, and PCs without TNC.EXE	EXT
No transfer of data	– empty –

11.5 Machine-Specific User Parameters

The machine tool builder can assign functions to up to 16 USER PARAMETERS. For more detailed information, refer to the operating manual for the machine tool.

11.6 Selecting Position Display Types

The positions indicated in Fig. 11.1 are:

- Starting position Ⓐ
- Target position of the tool Ⓩ
- Workpiece datum Ⓦ
- Scale datum Ⓜ

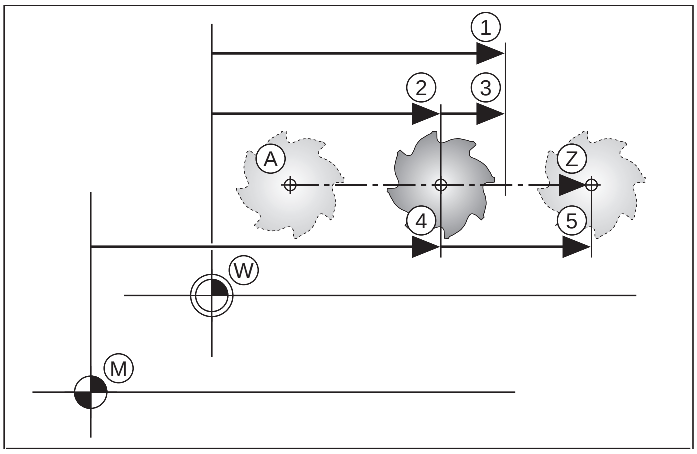


Fig. 11.1: Characteristic positions on the workpiece and scale

The TNC position display can show the following coordinates:

- Nominal position (the value presently commanded by the TNC) ① .... NOML.
- Actual position (the position at which the tool is presently located) ② .... ACTL.
- Servo lag (difference between the nominal and actual positions) ③ .... LAG
- Reference position (the actual position as referenced to the scale datum) ④ .... REF
- Distance remaining to the programmed position (difference between actual and target position) ⑤ .... DIST.

Select the desired information with the ENT key. It is then displayed directly in the status field.

11.7 Selecting the Unit of Measurement

This MOD function determines whether coordinates are displayed in millimeters or inches.

- Metric system: e.g. X = 15.789 (mm)
  MOD function CHANGE MM/INCH
  The value is displayed with 3 places after the decimal point
- Inch system: e.g. X = 0.6216 (inch)
  MOD function CHANGE MM/INCH
  The value is displayed with 4 places after the decimal point

11.8 Selecting the Programming Language

The MOD function PROGRAM INPUT lets you choose between programming in HEIDENHAIN plain language format and ISO format:

- To program in HEIDENHAIN format:
  Set the PROGRAM INPUT function to HEIDENHAIN
- To program in ISO format:
  Set the PROGRAM INPUT function to: ISO

11.9 Axes for L Block from Actual Position Capture

With the MOD function AXIS SELECTION you can determine which axis coordinates will be stored in the L block generated through actual position capture. Press the orange axes keys to select the desired axes. You can select up to three axes.

> The machine and TNC must be prepared for this feature by the machine tool builder.

11.10 Setting the Axis Traverse Limits

The AXIS LIMIT mod function allows you to set limits to axis traverse within the machine's maximum working envelope.

Possible application:
to protect an indexing fixture from tool collision.

The maximum traverse range is defined by software limit switches. This range can be additionaly limited through the AXIS LIMIT mod function. With this function you can enter the maximum traverse positions for the positive and negative directions. These values are referenced to the scale datum.

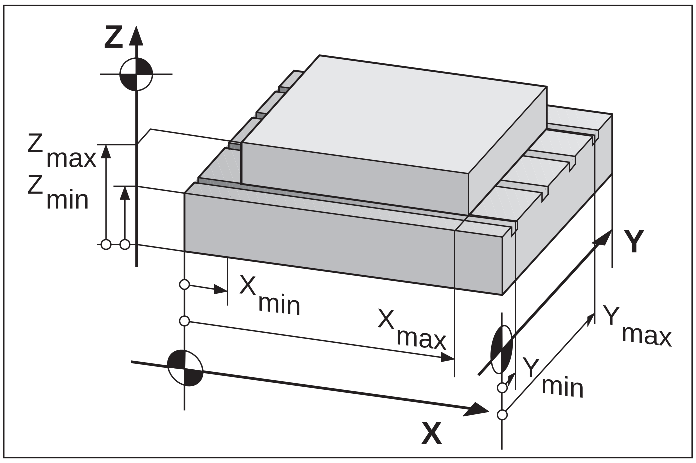


Fig. 11.2: Traverse limits on the workpiece

Working without additional traverse limits

To allow certain coordinate axes to use their full range of traverse, enter the maximum traverse of the TNC (+/– 30 000 mm) as the AXIS LIMIT.

To find and enter the maximum traverse:

Select POSITION DISPLAY REF.

Move the spindle to the desired positive and negative end positions of the X, Y and Z axes.

Write down the values, noting the algebraic sign.

MOD	Select the MOD functions.

Enter the values that you wrote down as LIMITS in the corresponding axes.

END □	Exit the MOD functions.

- The tool radius is not automatically compensated in the axis traverse limits values.
- Traverse range limits and software limit switches become active as soon as the reference marks are crossed over.
- In every axis the TNC checks whether the negative limit is smaller than the positive one.
- The reference positions can also be captured directly with the Actual Position Capture function (see page 4-19).

12.1 General User Parameters

General user parameters are machine parameters which affect the behavior of the TNC. These parameters set such things as:

- Dialog language
- Interface behavior
- Traversing speeds
- Machining sequence
- Effect of the overrides

Selecting the general user parameters

General user parameters are selected with code number 123 in the MOD functions.

The MOD functions also include machine-specific user parameters.

Parameters for external data transfer

These parameters define control characters for blockwise transfer.

Input values: Number between 0 and 32 382
(ASCII character with 16-bit coding)

Note:
The character defined here for end of program is also valid for the setting of the standard interface.

MP 5010

Function	M P	Bit
• End of program	5010.0	0 to 7
• Beginning of program	5010.0	8 to 15
• Data input	5010.1	0 to 15
• Data output	5010.2	0 to 15
• Beginning of command block	5010.3	0 to 7
• End of command block	5010.3	8 to 15
• Positive acknowledgment	5010.4	0 to 7
• Negative acknowledgment	5010.4	8 to 15
• End of data transfer	5010.5	0 to 15

Integrating the TNC interfaces to external devices: Data format and transmission stop

Input value: number between 0 and 255
Sum of the individual values from the "value" column.

MP 5020

Function	Selections	Values
• Number of data bits	7 data bits (ASCII code, 8th bit = parity)	+0
	8 data bits (ASCII code, 9th bit = parity)	+1
• Block Check Character (BCC)	BCC can be any character	+0
	BCC control character not allowed	+2
• Transmission stop with RTS	Active	+4
	Inactive	+0
• Transmission stop with DC3	Active	+8
	Inactive	+0
• Character parity	Even	+0
	Odd	+16
• Character parity	Not desired	+0
	Desired	+32
• Number of stop bits	1 1/2 stop bits	+0
	2 stop bits	+64
	1 stop bit	+128
	1 stop bit	+192

Example

To adapt the TNC interface to an external non-HEIDENHAIN device, use the following setting:
8 data bits, BCC any character, transmission stop with DC3, even character parity, character parity desired, 2 stop bits.

Input value: 1+0+8+0+32+64 = 105, so enter 105 for MP 5020.

Interface type

MP 5030

Function	Selections	Value
• Interface type	Standard	0
	Interface for blockwise transfer	1

Parameters for 3D Touch Probes

Signal transmission type

MP 6010

Function	Value
• Cable transmission	0
• Infrared transmission	1

Traversing behavior of touch probe

Parameter	Function	Value
MP 6120	Probing feed rate in mm/min	80 to 30 000
MP 6130	Maximum measuring range to first scanning point in mm	0 to 30 000
MP 6140	Safety clearance over probing point during automatic probing, in mm	0 to 30 000
MP 6150	Rapid traverse for probe cycle in mm/min	80 to 30 000

Parameters for TNC Displays and the Editor

Programming station

MP 7210

Function	Value
• TNC with machine	0
• TNC as programming station with active PLC	1
• TNC as programming station with inactive PLC	2

Block number increment with ISO programming

MP 7220

Function	Value
• Block number increment	0 to 255

Dialog language

MP 7230

Function	Value
• National dialog language	0
• Dialog language English (standard)	1

Edit-protect OEM cycles

For protection against editing of programs whose program number is the same as an OEM cycle number.

MP 7240

Function	Value
• Edit-protect OEM cycles	0
• No edit protection of OEM cycles	1

Defining a tool table (program 0)

Input: numerical value

Parameter	Function	Value
• ***MP 7260***	Total number of tools in the table	0 to 99
• ***MP 7261***	Number of tools with pocket numbers	0 to 99
• ***MP 7264***	Number of reserved pockets next to special tools	0 to 3

Settings for MANUAL OPERATION mode

Entry values 0 to 3:
Sum of the individual values from the “value” column.

MP 7270

Function	Selections	Value
• Display feed rate in manual mode	Display feed rate	+1
	Do not display feed rate	+0
• Spindle speed S and M functions still active after STOP	S and M still active	+0
	S and M no longer active	+2

Decimal character

MP 7280

Function	Value
• Decimal point	1
• Decimal comma	0

Display step for coordinate axes

MP 7290

Function	Value
• Display step 0.001 mm	0
• Display step 0.005 mm	1

Q parameters and status display

MP 7300

Function	Selections	Value
• Q parameters and status display	Do not erase	+0
	Erase with M02, M30 and END PGM	+1
• Last programmed tool after power interruption	Do not activate	+0
	Activate	+4

Graphics display

Entry range: 0 to 3 (sum of the individual values)

MP 7310

Function	Selections	Value
• View in 3 planes according to ISO 6433	Projection method 1	+0
	Projection method 2	+1
• Rotate coordinate system by 90° in the working plane	Rotate	+2
	Do not rotate	+0

Parameters for machining and program run

Effect of cycle 11 SCALING

MP 7410

Function	Value
• SCALING effective in 3 axes	0
• SCALING effective in the working plane	1

MP 7411 Tool compensation data in the TOUCH PROBE block

Function	Value
• Overwrite current tool data with the calibrated data of the touch probe	0
• Retain current tool data	1

Behavior of machining cycles

This general user parameter affects pocket milling.

Entry value: 0 to 15 (sum of the individual values in the "value" column)

MP 7420

Function	Cases	Value
• Milling direction for a channel around the contour	Clockwise for pockets, counterclockwise for islands	+1
	Counterclockwise for pockets, clockwise for islands	+0
• Sequence of roughing out and channel milling	First mill contour channel, then rough out	+0
	First rough out, then mill contour channel	+2
• Merge contours	Merge compensated contours	+0
	Merge uncompensated contours	+4
• Milling in depth	At each pecking depth, mill channel and rough out before going to next depth	+8
	Mill contour channel to full pocket depth, then rough out to full pocket depth	+0

Overlapping with pocket milling

Overlap factor with pocket milling:
product of MP7430 and the tool radius

MP 7430

Function	Value
• Overlap factor for pockets	0.1 to 1.414

Effect of M functions

The M functions M6 and M89 are influenced by MP 7440:

Entry range: 0 to 7
(Sum of the individual values in the "value" column)

MP 7440

Function	Cases	Value
• Programmable stop with M06	Program stop with M06	+0
	No program stop	+1
• Modal cycle call with M89	Modal cycle call with M89	+2
	M89 vacant M function	+0
• Axes are stopped when M function carried out	Axis stop with M functions	+4
	No axis stop	+0

Safety limit for machining corners at constant path speed

Corners whose inside angle is less than the entered value are no longer machined at constant path speed with M90.

MP 7460

Function	Value
• Maintain constant path speed at inside corners for angles of (degrees)	0 to 179.999

Coordinate display for rotary axis

MP 7470

Function	Value
• Angle display up to ± 359.999°	0
• Angle display up to ± 30 000°	1

Parameters for override behavior and electronic handwheel

Override

Entry range: 0 to 7 (sum of the individual values in the “value” column)

MP 7620

Function	Cases	Value
• Feed rate override when rapid traverse key pressed in program run mode	Override effective Override not effective	+1 +0
• Increments for overrides	1% increments 2% increments	+0 +8
• Feed rate override when rapid traverse key and machine axis direction button pressed	Override effective Override not effective	+4 +0

Setting the TNC for handwheel operation

Entry range: 0 to 5

MP 7640

Function	Value
• No handwheel	0
• HR 330 with additional keys – the keys for traverse direction and rapid traverse are evaluated by the NC	1
• HR 130 without additional keys	2
• HR 330 with additional keys – the keys for traverse direction and rapid traverse are evaluated by the PLC	3
• HR 332 with 12 additional keys	4
• Multi-axis handwheel with additional keys	5

12.2 Miscellaneous Functions (M Functions)

Miscellaneous functions with predetermined effect

M	Function	Effective at start of block	Effective at end of block
M00	Stop program run / Spindle stop / Coolant off		•
M02	Stop program run / Spindle stop / Coolant off. Clear the status display (depending on machine parameter) / Return to block 1		•
M03	Spindle on clockwise	•	
M04	Spindle on counterclockwise	•	
M05	Spindle stop		•
M06	Tool change / Stop program run (depending on machine parameter) / Spindle stop		•
M08	Coolant on	•	
M09	Coolant off		•
M13	Spindle on clockwise / Coolant on	•	
M14	Spindle on counterclockwise / Coolant on	•	
M30	Same function as M02		•
M89	Vacant miscellaneous function	•	
	or Cycle call, modally effective (depending on machine parameter)		•
M90	Smoothing corners	•	
M91	**Within the positioning block:** Coordinates are referenced to the machine datum	•	
M92	**Within the positioning block:** Coordinates are referenced to a position defined by the machine tool builder (such as a tool change position)	•	
M93	**Within the positioning block:** Coordinates are referenced to the current tool position. Effective in blocks with R0, R+ R–	•	
M94	Limit display of rotary axis to value under 360°	•	
M95	Reserved		•
M96	Reserved		•
M97	Machine small contour steps		•
M98	Completely machine open contours		•
M99	Blockwise cycle call		•

Vacant miscellaneous functions

Vacant M functions are defined by the machine tool builder.
They are described in the operating manual of your machine tool.

Effect of vacant miscellaneous functions

M	Function	Effective at	
		start of block	end of block
M01			•
M07		•	
M10			•
M11		•	
M12			•
M16		•	
M17		•	
M18		•	
M19			•
M20		•	
M21		•	
M22		•	
M23		•	
M24		•	
M25		•	
M26		•	
M27		•	
M28		•	
M29		•	
M30		•	
M31		•	
M32			•
M33			•
M34			•
M35			•
M36		•	
M37		•	
M38		•	
M39		•	
M40		•	
M41		•	
M42		•	
M43		•	
M44		•	
M45		•	
M46		•	
M47		•	
M48		•	
M49		•	

M	Function	Effective at	
		start of block	end of block
M50		•	
M51		•	
M52			•
M53			•
M54			•
M55		•	
M56		•	
M57		•	
M58		•	
M59		•	
M60			•
M61		•	
M62		•	
M63			•
M64			•
M65			•
M66			•
M67			•
M68			•
M69			•
M70			•
M71		•	
M72		•	
M73		•	
M74		•	
M75		•	
M76		•	
M77		•	
M78		•	
M79		•	
M80		•	
M81		•	
M82		•	
M83		•	
M84		•	
M85		•	
M86		•	
M87		•	
M88		•	

12.3 Preassigned Q Parameters

The Q parameters Q100 to Q113 are assigned values by the TNC. Such values include:

- Values from the PLC
- Tool and spindle data
- Data on operating status, etc.

Values from the PLC: Q100 to Q107

The TNC uses the parameters Q100 to Q107 to transfer values from the PLC to an NC program.

Tool radius: Q108

The radius of the current tool is assigned to Q108.

Tool axis: Q109

The value of parameter Q109 depends on the current tool axis.
.

Tool axis	Parameter value
No tool axis defined	Q109 = –1
Z axis	Q109 = 2
Y axis	Q109 = 1
X axis	Q109 = 0

Spindle status: Q110

The value of Q110 depends on the M function last programmed for the spindle.

M function	Parameter value
No spindle status defined	Q110 = –1
M03: Spindle on clockwise	Q110 = 0
M04: Spindle on counterclockwise	Q110 = 1
M05 after M03	Q110 = 2
M05 after M04	Q110 = 3

Coolant on/off: Q111

M function	Parameter value
M08: Coolant on	Q111 = 1
M09: Coolant off	Q111 = 0

Overlap factor: Q112

The overlap factor for pocket milling (MP 7430) is assigned to Q112.

Unit of measurement: Q113

The value of Q113 specifies whether the highest-level NC program (for nesting with PGM CALL) is programmed in millimeters or inches.
After NC start, Q113 is set as follows:

Unit of measurement (main program)	Parameter value
Millimeters	Q113 = 0
Inches	Q113 = 1

Current tool length: Q114

The current value of the tool length is assigned to Q114.

Coordinates from probing during program run

Parameters Q115 to Q118 are assigned the coordinates of the spindle position upon probing during a programmed measurement with the 3D touch probe.

Coordinate axis	Parameter
X axis	Q115
Y axis	Q116
Z axis	Q117
IV axis	Q118

Current tool radius compensation

The current tool radius compensation is assigned to parameter Q123 as follows:

Current tool compensation	Parameter value
R0	Q123 = 0
RL	Q123 = 1
RR	Q123 = 2
R+	Q123 = 3
R–	Q123 = 4

12.4 Diagrams for Machining

Spindle speed S

The spindle speed S can be calculated from the tool radius R and the cutting speed v as follows:

$$S = \frac{V}{2 \cdot R \cdot \pi}$$

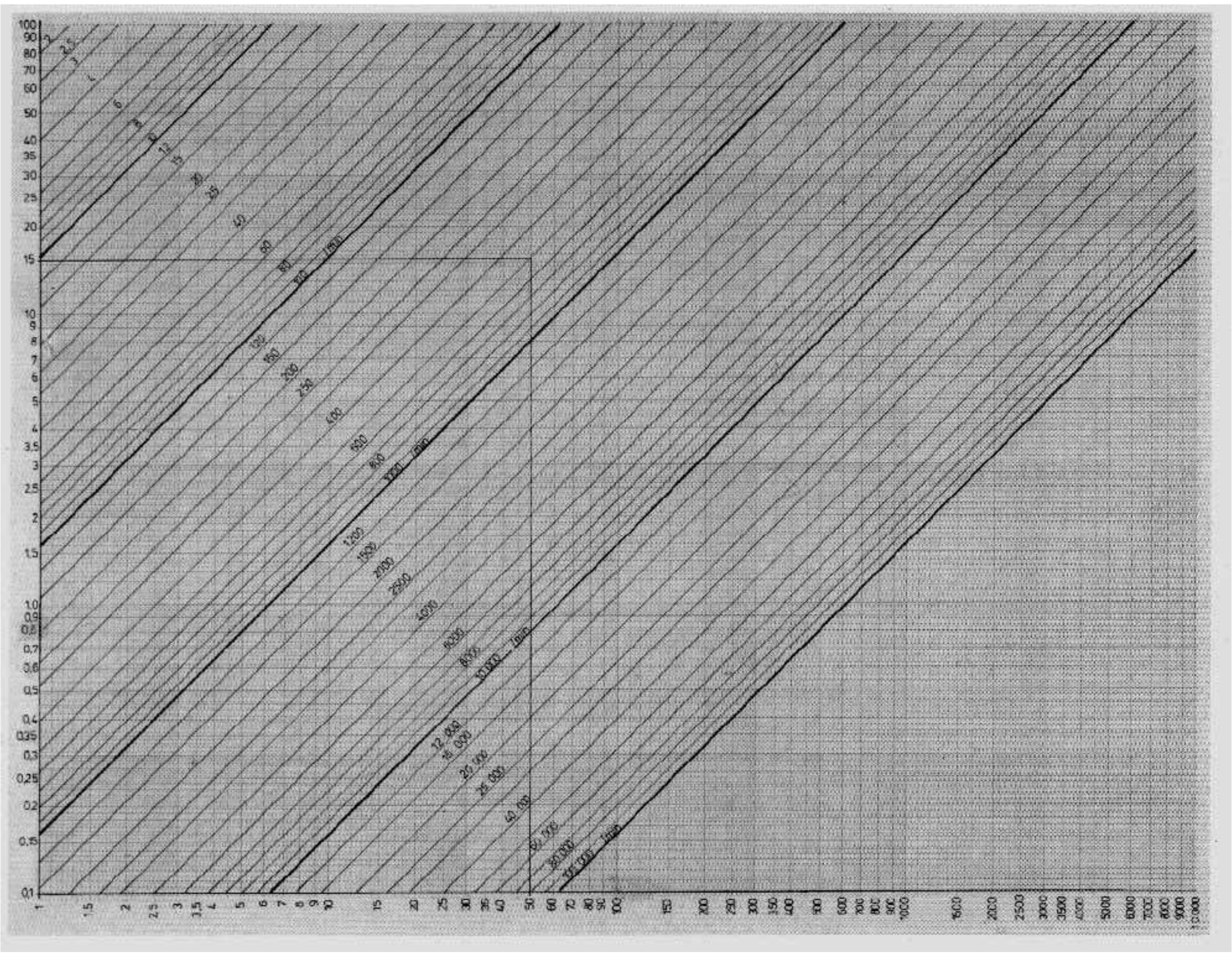
Units:

S in rpm
V in mm/min
R in mm

You can read the spindle speed directly from the diagram.

Example:

Tool radius	R = 15 mm
Cutting speed	V = 50 000 mm/min
Spindle speed	S ≈ 500 rpm
	(calculated S = 530 rpm)



Feed rate F

The feed rate F of the tool is calculated from the number of tool teeth n, the permissible depth of cut per tooth d, and the spindle speed S:

F = n · d · S

Units:

F	in	mm/min
d	in	mm
S	in	rpm

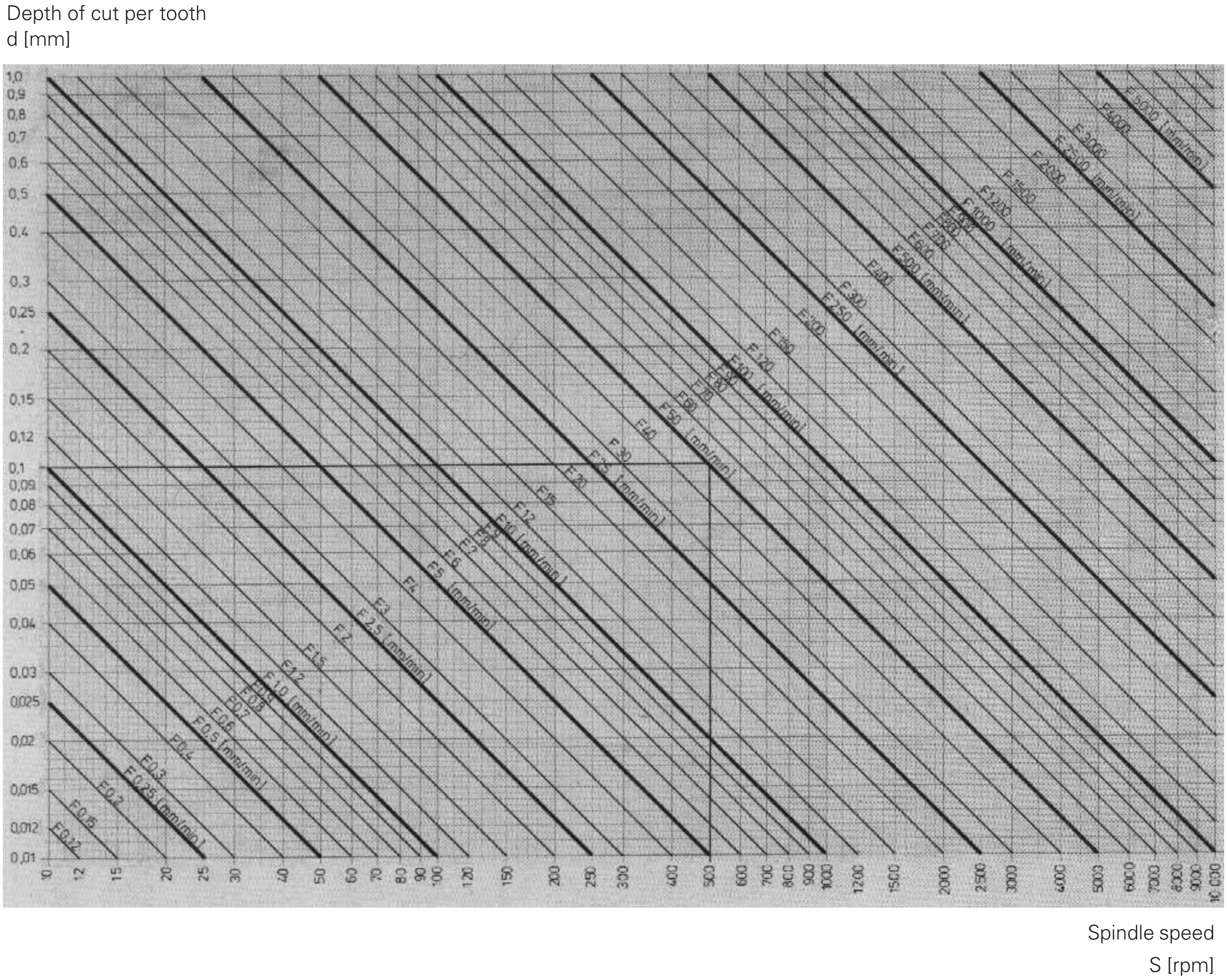
The feed rate read from the diagram must be multiplied by the number of tool teeth.

Example:

Depth of cut per tooth	d	=	0.1 mm
Spindle speed	S	=	500 rpm
Feed rate from diagram	F	=	50 mm/min
Number of tool teeth	n	=	6
Feed rate to enter	F	=	300 mm/min

The diagram provides approximate values and assumes the following:

- Downfeed in the tool axis = 0.5 · R and the tool is cutting through solid metal, or
- Lateral metal-to-air ratio = 0.25 · R and downfeed in the tool axis = R



Feed rate F for tapping

The feed rate for tapping F is calculated from the thread pitch p and the spindle speed S:

$F = p \cdot S$

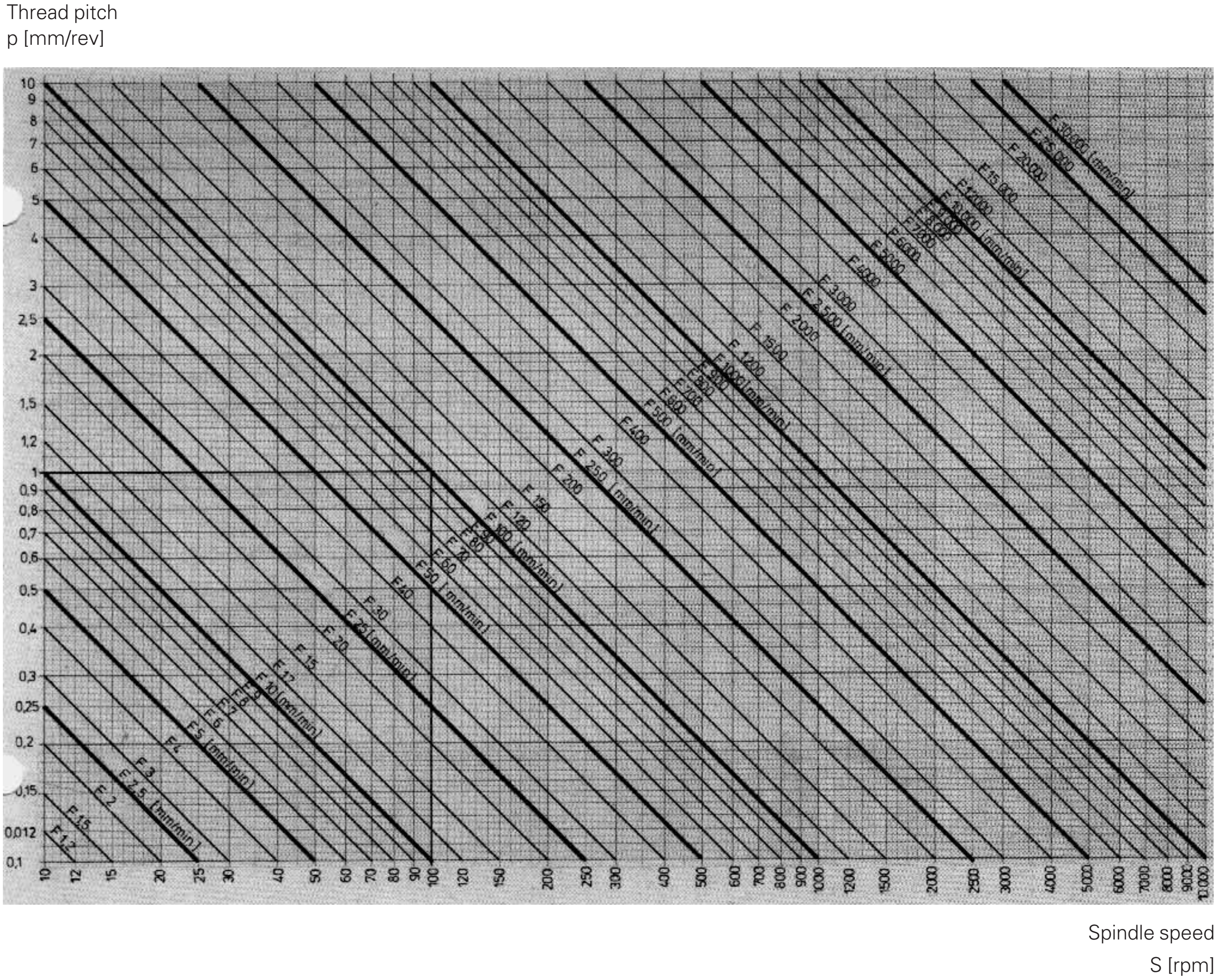
Units:

F	in	mm/min
p	in	mm/1
S	in	rpm

The feed rate for tapping can be read directly from the diagram below.

Example:

Thread pitch	p	=	1 mm/rev
Spindle speed	S	=	100 rpm
Feed rate for tapping	F	=	100 mm/min



12.5 Features, Specifications and Accessories

TNC 360

Description

Contouring control for up to 4 axes, with oriented spindle stop.

Components

Logic unit, keyboard, monochrome flat luminescent screen or CRT.

Data interface

RS-232-C / V.24

Simultaneous axis control for contour elements

• Straight lines up to	3 axes
• Circles in	2 axes
• Helices	3 axes

Background programming

For editing one part program while the TNC is running another.

Test run

Internally and with test run graphics.

Program types

- HEIDENHAIN plain Language format and ISO programs
- Tool table

Program memory

- Battery-buffered for up to 32 programs
- Capacity: approximately 4000 program blocks

Tool definitions

- Up to 254 tools in one program or up to 99 tools in the tool table (program 0).

Programmable Functions

Contour elements

Straight line, chamfer, circle center, circle radius, tangentially connecting arc, corner rounding.

Program jumps

Subprogram, program section repetition, main program as subprogram.

Fixed cycles

Peck drilling, tapping (also with synchronized spindle), rectangular and circular pocket milling, slot milling, milling pockets and islands from a list of subcontour elements.

Coordinate transformations

Datum shift, mirroring, rotation, scaling factor.

3D Touch Probe System

Probing functions for measuring and datum setting, digitizing 3D surfaces (optional).

Mathematical functions

Basic operations +, –, x and %, trigonometric functions
sin, cos, tan and arctan.
Square roots ($\sqrt{a}$) and root sum of squares ($\sqrt{a^2 + b^2}$).
Logical comparisons greater than, smaller than, equal to, not equal to.

TNC Specifications

Block execution time	1500 blocks/min (40 ms per block)
Control loop cycle time	6 ms
Data transfer rate	Max. 38400 baud
Ambient temperature	0°C to 45°C (operation) –30°C to 70°C (storage)
Traverse	Max. ± 30 m (1181 inches)
Traversing speed	Max. 30 m/min (1181 ipm)
Spindle speed	Max. 99 999 rpm
Input resolution	As fine as 1 µm (0.0001 in.) or 0.001°

Accessories

FE 401 Floppy Disk Unit

Description	Portable bench-top unit
Applications	All TNC contouring controls, TNC 131, TNC 135
Data interfaces	Two RS-232-C interface ports
Data transfer rate	• TNC : 2400 to 38400 baud • PRT : 110 to 9600 baud
Diskette drive s	Two drives, one for copying, capacity 795 kilobytes (approx. 25 000 blocks), up to 256 files
Diskette type	3.5", DS DD, 135 TPI

Triggering 3D Touch Probes

Description	Touch probe system with ruby tip and stylus with rated break point, standard shank for spindle insertion
Models	TS 120: Cable transmission, integrated interface TS 511: Infrared transmission, separate transmitting and receiving units
Spindle insertion	TS 120: manual TS 511: automatic
Probing reproducibility	Better than 1 µm (0.000 04 in.)
Probing speed	Max. 3 m/min (118 ipm)

Electronic Handwheels

HR 130	• Integrable unit
HR 330	• Portable version with cable transmission, equipped with axis address keys, rapid traverse key, safety switch, emergency stop button.

12.6 TNC Error Messages

The TNC automatically generates error messages when it detects such things as

- Incorrect data input
- Logical errors in the program
- Contour elements that are impossible to machine
- Incorrect use of the touch probe system

An error message containing a program block number was caused by an error in that block or in the preceding block. To clear a TNC error message, first correct the problem and then press the CE key.

Some of the more frequent TNC error messages are explained in the following list.

TNC error messages during programming

ENTRY VALUE INCORRECT

- Enter a correct LBL number.
- Observe the input limits.

EXT. IN-/OUTPUT NOT READY

The external device is not correctly connected.

FURTHER PROGRAM ENTRY IMPOSSIBLE

Erase some old files to make room for new ones.

LABEL NUMBER ALLOCATED

Label numbers can only be assigned once.

JUMP TO LABEL 0 NOT PERMITTED

Do not program CALL LBL 0.

TNC error messages during test run and program run

ANGLE REFERENCE MISSING

- Define the arc and its end points unambiguously.
- If you enter polar coordinates, define the polar coordinate angle correctly.

ARITHMETICAL ERROR

You have attempted to calculate with illegal values.

- Define values within the range limits.
- Choose probe positions for the 3D touch probe that are farther separated.
- Calculations must be mathematically possible.

AXIS DOUBLE PROGRAMMED

Each axis can only have one value for position coordinates.

BLK FORM DEFINITION INCORRECT

- Program the MIN and MAX points according to the instructions.
- Choose a ratio of sides less than 84:1.
- When programming with PGM CALL, copy the BLK FORM into the main program.

CHAMFER NOT PERMITTED

- A chamfer block must be inserted between two straight line blocks.

CIRCLE END POS. INCORRECT

- Enter complete information for tangential arcs.
- Enter end points that lie on the circular path.

CYCL INCOMPLETE

- Define the cycle with all data in the proper sequence.
- Do not call coordinate transformation cycles.
- Define a cycle before calling it.
- Enter a pecking depth other than 0.

EXCESSIVE SUBPROGRAMMING

- Conclude subprograms with LBL0.
- Program CALL LBL for subprograms without REP.
- Program CALL LBL for program section repeats to include the repetitions (REP).
- Subprograms cannot call themselves.
- Subprograms cannot be nested in more than 8 levels.
- Main programs cannot be nested as subprograms in more than 4 levels.

FEED RATE IS MISSING

- Enter the feed rate for the positioning block.
- Enter FMAX in each block.

GROSS POSITIONING ERROR

The TNC monitors positions and movements. If the actual position deviates too greatly from the nominal position, this blinking error message is displayed. To correct the error, press and hold the END key for several seconds (warm start).

KEY NON-FUNCTIONAL

This message always appears when you press a key that is not needed for the current dialog.

LABEL NUMBER NOT ALLOCATED

You can only call labels numbers that have been assigned.

PATH OFFSET WRONGLY ENDED

Do not cancel tool radius compensation in a block with a circular path.

PATH OFFSET WRONGLY STARTED

- Use the same radius compensation before and after a RND and CHF block.
- Do not begin tool radius compensation in a block with a circular path.

PGM SECTION CANNOT BE SHOWN

- Enter a smaller tool radius.
- Movements in a rotary axis cannot be graphically simulated.
- Enter a tool axis for simulation that is the same as the axis in the BLK FORM.

PLANE WRONGLY DEFINED

- Do not change the tool axis while a basic rotation is active.
- Define the main axes for circular arcs correctly.
- Define both main axes for CC.

PROBE SYSTEM NOT READY

- Orient transmitting/receiving window of TS 511 to face receiving unit.
- Check whether the touch probe is ready for operation.

PROGRAM-START UNDEFINED

- Begin the program only with a TOOL DEF block.
- Do not resume an interrupted program at a block with a tangential arc or pole transfer.

RADIUS COMPENSATION UNDEFINED

Enter radius compensation in the first subprogram to cycle 14: CONTOUR GEOM.

ROUNDING OFF NOT DEFINED

Enter tangentially connecting arcs and rounding arcs correctly.

ROUNDING RADIUS TOO LARGE

Rounding arcs must fit between contour elements.

SELECTED BLOCK NOT ADDRESSED

Before a test run or program run you must go to the beginning of the program by entering GOTO 0.

STYLUS ALREADY IN CONTACT

Before probing, pre-position the stylus so that it is not touching the workpiece surface.

TOOL RADIUS TOO LARGE

Enter a tool radius that
- lies within the given limits, and
- permits the contour elements to be calculated and machined.

TOUCH POINT INACCESSIBLE

Pre-position the 3D touch probe to a point nearer the surface.

WRONG AXIS PROGRAMMED

- Do not attempt to program locked axes.
- Program a rectangular pocket or slot in the working plane.
- Do not mirror rotary axes.
- Chamfer length must be positive.

WRONG RPM

Program a spindle speed within the permissible range.

WRONG SIGN PROGRAMMED

Enter the correct sign for the cycle parameter.

TNC error messages with digitizing

AXIS DOUBLE PROGRAMMED

Program two different axes for the coordinates of the starting point (CONTOUR LINES cycle).

EXCHANGE TOUCH PROBE BATTERY

Exchange the battery in the touch probe head (TS 511). This message is displayed when the probe reaches the end of a line.

FAULTY RANGE DATA

- Enter MIN coordinates that are smaller than their MAX coordinates.
- Define the RANGE within the limits set by software limit switches.
- Define the RANGE for the MEANDER and CONTOUR LINES cycles.

MIRRORING NOT PERMITTED

Reset all coordinate transformations before digitizing.

PLANE WRONGLY DEFINED

Define the starting position coordinates (CONTOUR LINES cycle) in axes different from the stylus axis.

PROBE SYSTEM NOT READY

- Orient transmitting/receiving window of TS 511 to face receiving unit.
- Check that the touch probe is ready for operation.
- The touch probe cannot be retracted (collision with workpiece).

RANGE EXCEEDED

Enter a RANGE that includes the entire 3D surface to be scanned.

ROTATION NOT PERMITTED

Reset all coordinate transformations before digitizing.

SCALING FACTOR NOT ALLOWED

Reset all coordinate transformations before digitizing.

START POSITION INCORRECT

Program the starting point coordinates for the CONTOUR LINES cycle so that they lie within the RANGE.

STYLUS ALREADY IN CONTACT

Pre-position the touch probe so that the stylus cannot be deflected before it reaches the RANGE.

TIME LIMIT EXCEEDED

Enter a TIME LIMIT that is appropriate to the 3D surface to be scanned (CONTOUR LINES cycle).

TOUCH POINT INACCESSIBLE

- The stylus must not be deflected before it reaches the RANGE.
- The stylus must be deflected somewhere within the RANGE.

WRONG AXIS PROGRAMMED

Enter calibrated touch probe axis in the RANGE cycle.

Miscellaneous Functions (M Functions)

Miscellaneous functions with predetermined effect

M	Function	Effective at	
		start of block	end of block
M00	Stop program run / Spindle stop / Coolant off		•
M02	Stop program run / Spindle stop / Coolant off. Clear the status display (depending on machine parameter) / Return to block 1		•
M03	Spindle on clockwise	•	
M04	Spindle on counterclockwise	•	
M05	Spindle stop		•
M06	Tool change / Stop program run (depending on machine parameter) / Spindle stop		•
M08	Coolant on	•	
M09	Coolant off		•
M13	Spindle on clockwise / Coolant on	•	
M14	Spindle on counterclockwise / Coolant on	•	
M30	Same function as M02		•
M89	Vacant miscellaneous function	•	
	or		
	Cycle call, modally effective (depending on machine paraemeter)		•
M90	Smoothing corners	•	
M91	**Within the positioning block:** Coordinates are referenced to the machine datum	•	
M92	**Within the positioning block:** Coordinates are referenced to a position defined by the machine tool builder (such as a tool change position)	•	
M93	**Within the positioning block:** Coordinates are referenced to the current tool position. Effective in blocks with R0, R+, R–	•	
M94	Limit display of rotary axis to value under 360°	•	
M95	Reserved		•
M96	Reserved		•
M97	Machine small contour steps		•
M98	Completely machine open contours		•
M99	Blockwise cycle call		•