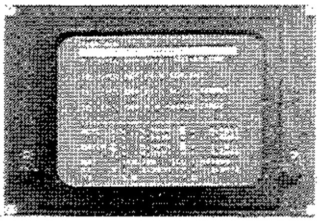
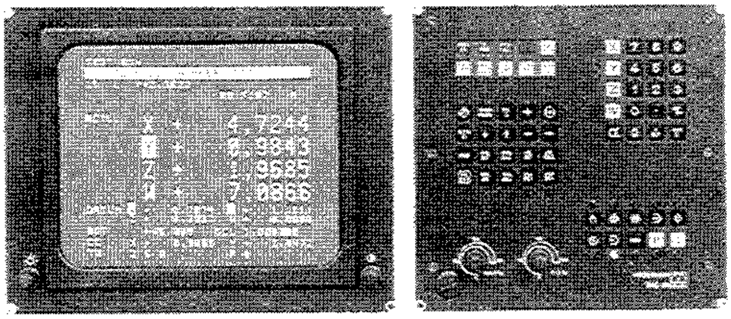
HEIDENHAIN

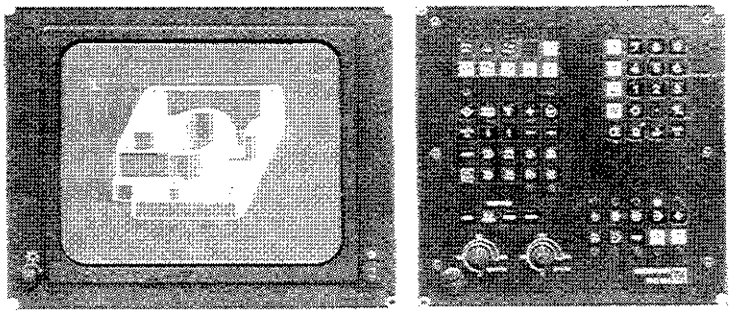
Operating Manual

HEIDENHAIN TNC 151 B/TNC 151 Q
HEIDENHAIN TNC 155 B/TNC 155 Q
Contouring Control





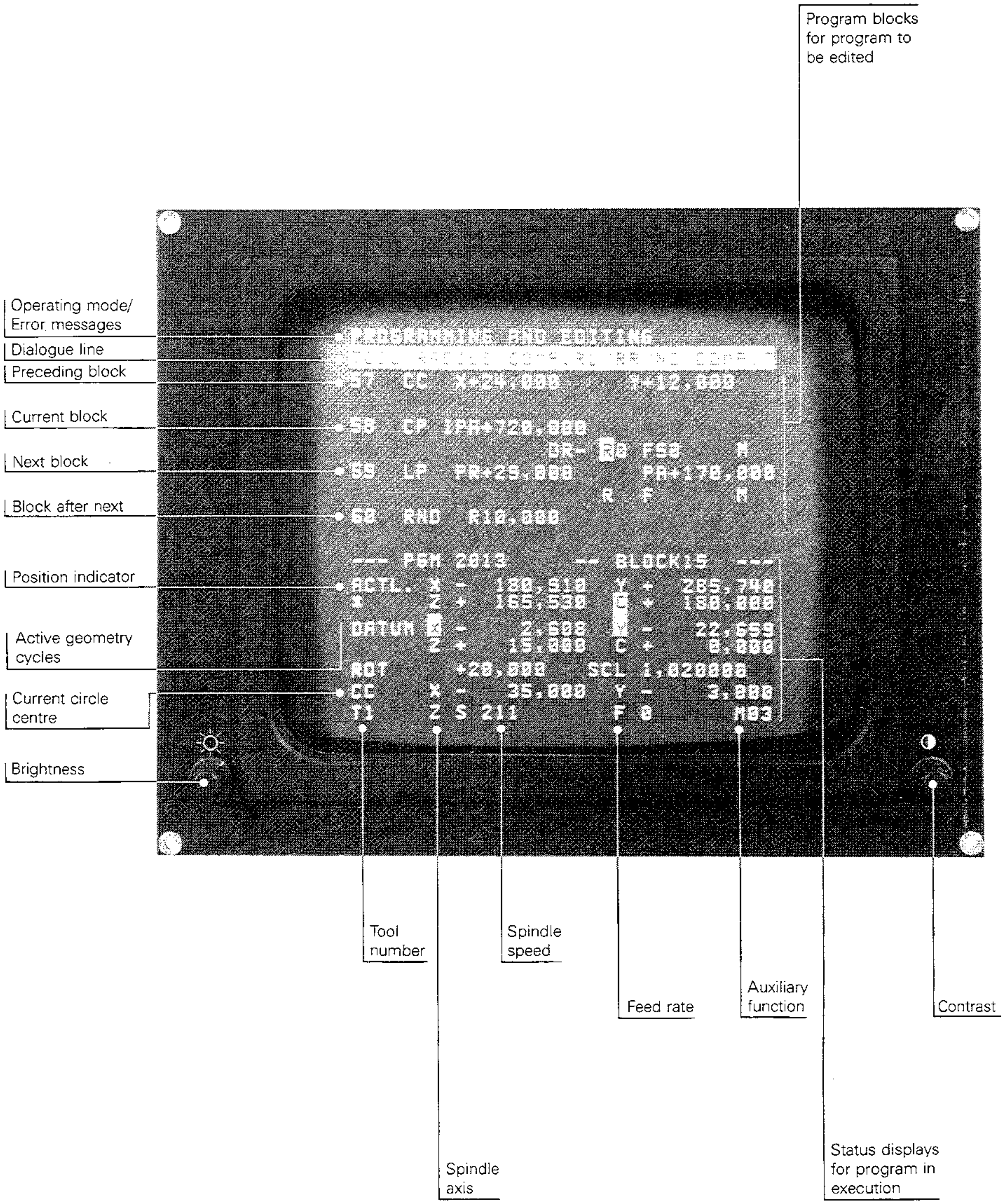
TNC 151



TNC 155

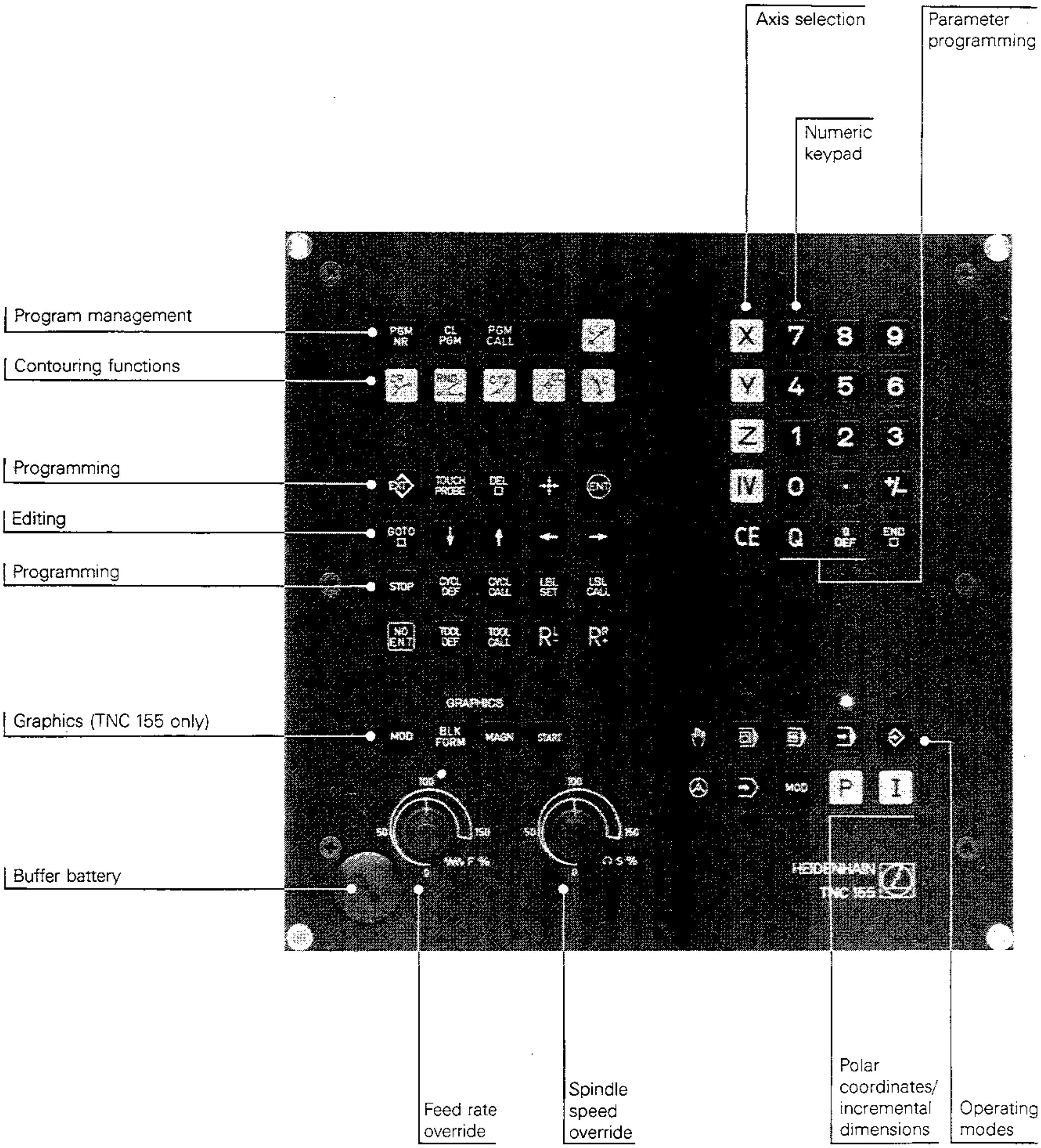
January '89

Screen display

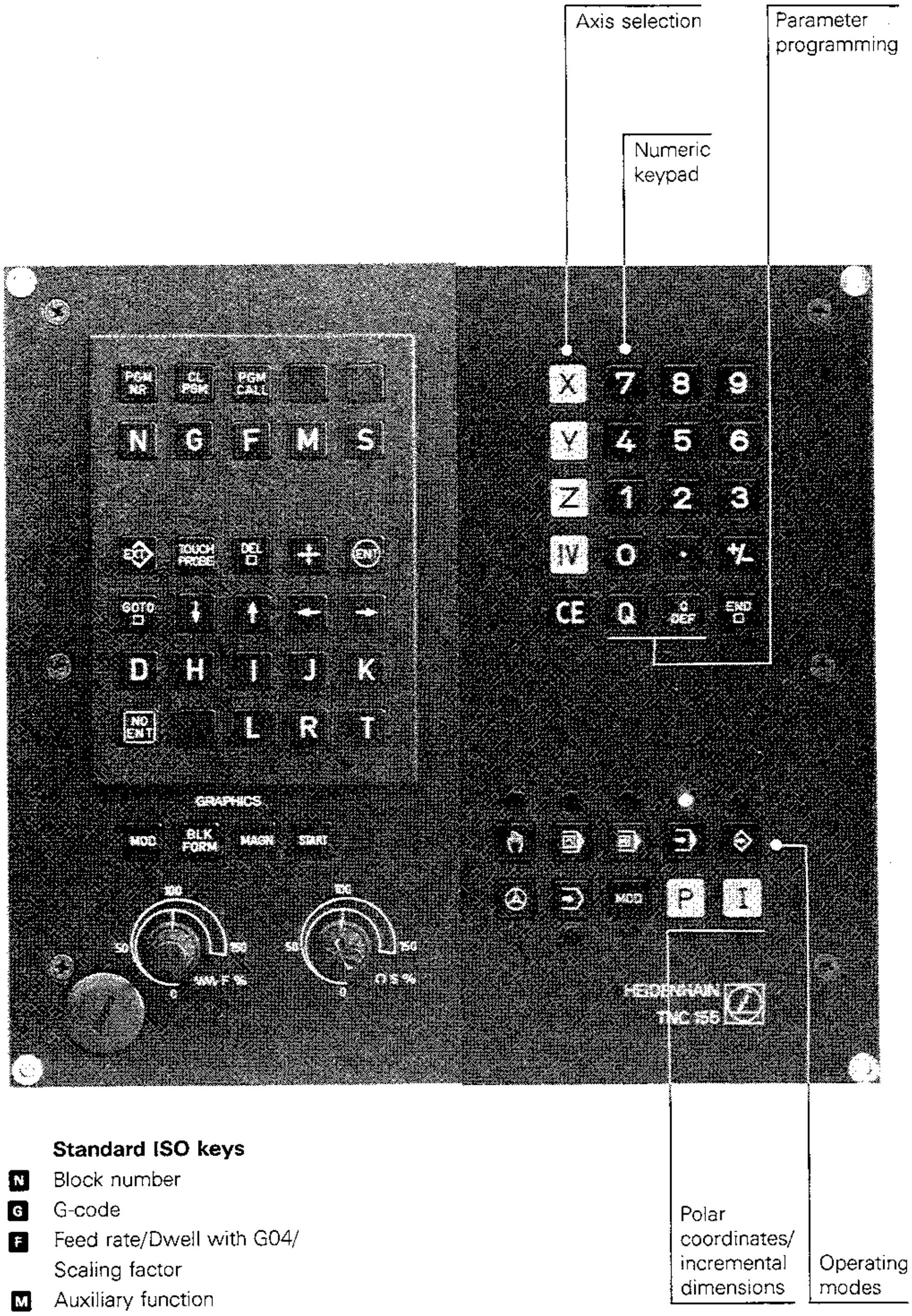


```
PROGRAMMING AND EDITING
TOOL RADIUS COMP. RL RR NO COMP.
57  CC  X+24,000      Y+12,000

58  CP  IPA+720,000
                DR- R0 F50     M
59  LP  PR+29,000       PA+170,000
                    R  F       M
60  RND  R10,000

--- PGM 2013      -- BLOCK15  ---
ACTL. X -   180,910  Y +  285,740
*     Z +   165,530  C +  180,000
DATUM X -     2,608  Y -   22,659
      Z +    15,000  C +    0,000
ROT      +20,000  SCL 1,020000
CC    X -    35,000  Y -    3,000
T1    Z S 211        F 0      M03
```

Control panel



Snap-on keyboard



Standard ISO keys

N Block number

G G-code

F Feed rate/Dwell with G04/
Scaling factor

M Auxiliary function

S Spindle speed, spindle orientation
with G36

D Parameter definition

H Polar coordinate angle/angle of rotation
in cycle G73

I X-coordinate of circle centre

J Y-coordinate of circle centre

K Z-coordinate of circle centre

L Set label number with G98/
Jump to label number/
Tool length with G99

R Polar coordinate radius/
Rounding-off radius with G25, G26, G27/
Chamfer with G24
Tool radius with G99/Circle radius
with G02, G03, G05

T Tool definition with G99/
Tool call

Keyboard

Program management

- PGM NR Program designation and call
- CL PGM Clear (erase) program
- CALL PGM Call program within another program

Workpiece contour entry

- L Line (linear interpolation)/Chamfer
- RND Rounding corner/Tangential contour approach and departure
- CT Circle tangentially adjoining previous contour (end position only)
- CC Circle centre/Pole
- C Circle (with centre and end position)
- CR Circle (with radius and end position)

Programming and editing

- EXT External data transfer
- TOUCH PROBE Touch-probe functions
- DEL Delete block
- ✛ Transfer/enter actual position
- ENT Enter data
- GOTO ↓ ↑ ← → Search and edit functions
- STOP Programmed stop, terminate
- CYCL DEF CYCL CALL Define and call canned cycles
- LBL SET LBL CALL Define and call program sections and subroutines
- NO ENT No data entry, skip dialogue prompts
- TOOL DEF TOOL CALL Define and call tool and tool compensation
- R L R R Radius compensation

Graphics (TNC 155 only)

- MOD Graphics modes
- BLK FORM Define workpiece blank, reset to blank
- MAGN Magnify
- START Start graphics

Entry values and axis selection

- X Y Z IV X Y Z IV axis address keys
- CE Clear (delete) previous entry
- END Terminate block entry

Parameter programming

- Q Set parameter
- Q DEF Define parameter

Operating modes

- Manual operation (TNC functions as position indicator)
- Positioning via manual data input (positioning block is run, but not saved)
- Program run – single block (program is executed block-by-block)
- Program run – full sequence (continuous program execution)
- Programming and editing (enter program manually or via data interface)
- Electronic handwheel
- Test run (check program without machine movement)
- MOD Supplementary operating modes (vacant blocks – mm/inch – position-display size – actual position/nominal position/distance to go/trailing error/distance to reference point – baud rate – axis software limits – user parameters – code number – NC/PLC software number – V.24 interface configuration
  For ISO programming: block number increment

Polar coordinates/Incremental dimensions

- P Enter position value in polar coordinates
- I Enter position value in incremental dimensions

This Operating Manual is valid for all currently available versions of the TNC 151/TNC 155:

Transducer inputs	TNC 151/TNC 155 version **without** separate PLC input/output board(s)	**TNC 151/TNC 155 version with PLC input/output** board(s)
Sinusoidal signals	TNC 151 B/TNC 155 B TNC 151 F/TNC 155 F*	TNC 151 Q/TNC 155 Q TNC 151 W/TNC 155 W*
Square-wave signals	TNC 151 BR/TNC 155 BR TNC 151 FR/TNC 155 FR*	TNC 151 QR/TNC 155 QR TNC 151 WR/TNC 155 WR*

* Does not include 3D positioning and "Transfer blockwise".

Because HEIDENHAIN is constantly striving to further develop its TNC control systems, details of a given control version may deviate from the version described in this Operating Manual.

Manufacturer's certificate
We hereby certify that the above device is radio-interference-suppressed in compliance with the provisions of the West German Official Register Decree No. 1046/1984.
The West German postal authorities have been notified of the deployment of this device and have been granted permission to inspect the series for compliance with said provisions.

Note:
If the device is incorporated by the user into an installation, the entire system must comply with the above-mentioned provisions.

Concept, copy and layout: Schwenkert, Kastenhuber und Partner GmbH, Munich

Contents

Control system in brief
TNC 151/TNC 155

Control type

The HEIDENHAIN TNC 151/TNC 155 is a 4-axis contouring control system. Axes X, Y and Z are linear axes; the fourth axis is provided for the attachment of an optional rotary table or use as an additional linear axis. The fourth axis can be connected or disconnected as required.

The four-axis contouring control permits:
- linear interpolation of any 3 axes,
- circular interpolation of two linear axes.

Complex contours can also be produced with the aid of parameter programming.
The TNC 151/TNC 155 can be equipped with an optional 5th axis for spindle orientation. This feature allows accurate positioning of the spindle, when using the TS 510/TS 511 infrared probing system, for example, or for certain tool change systems.

Program entry

Programs may be entered either
- in HEIDENHAIN plain-language interactive dialogue
  or
- in standard ISO 6983 format.

All interactive dialogue prompts, input values, machining programs and error messages are displayed on the control screen. The program memory can accommodate up to 32 programs with a total of 3,100 blocks. The machining program can be either keyed in or entered "electronically" via the data interface. In "Transfer blockwise" mode, machining programs can be transferred from an external storage medium and run simultaneously.
The TNC 151/TNC 155 allows you to enter or edit a program while another program is running.

External data storage

HEIDENHAIN provides the ME 101/ME 102 magnetic tape unit and the FE 401 floppy disk unit for external storage of programs. The magnetic tape units use mini-cassettes for data storage; the floppy disk unit uses 3 1/2" diskettes. Each unit is equipped with two interfaces, making it possible to connect a peripheral device, such as a printer, in addition to the TNC.

Control system in brief
TNC 151/TNC 155

Program test

In "Test run" mode, the TNC checks machining programs without moving the machine slides. Any errors in the program are displayed in the form of plain-language messages.
Graphic program simulation provides another option for testing the program. Machining procedures can be simulated on the three main axes with a constant tool axis using a cylindrical end mill.

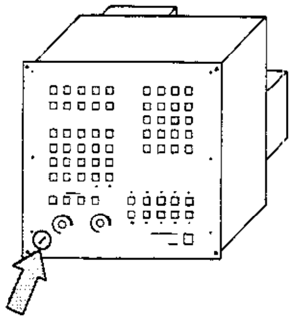
Upward compatibility

Programs created on the TNC 145 or TNC 150 can also be run on the TNC 151/TNC 155. The control system adapts the input later to the TNC 151/TNC 155.
Thus an existing TNC 145/TNC 150 program library can also be used for the TNC 151/TNC 155.

Changing buffer batteries

The buffer battery is the voltage source for the memory containing the machine parameters and for the control system program memory, in case the external voltage supply is switched off. It is located beneath the cover on the front panel of the control unit.
It is time to replace the three batteries when the message:
= EXCHANGE BUFFER BATTERY =
is displayed. (Memory contents will be maintained for at least one week after this message appears.)

Replace batteries with mains voltage connected. The TNC's memory units are then supplied with power from the mains supply. All data memory units (RAM) will be erased if the buffer battery is replaced while the mains supply is off and machine parameters will have to be re-entered.

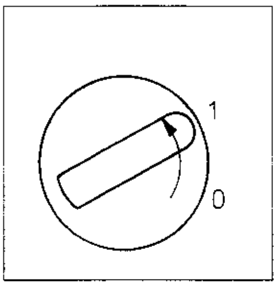


Battery type
Mignon cells, leak-proof
IEC designation: "LRG"
Recommended: VARTA type 4006

Switching on the control unit
Traversing reference points

The following symbols are used in diagrams in this Operating Manual:
○ ≙ Keys/buttons on external control panel
□ ≙ Keys/buttons on TNC control panel

Switch-on



Switch on power.

MEMORY TEST

The control checks the internal control electronics. Display is cleared automatically.

POWER INTERRUPTED □CE Clear error message.

RELAY EXT. DC VOLTAGE MISSING ○I Switch on control voltage.

PASS OVER Z-AXIS REFERENCE POINT
PASS OVER X-AXIS REFERENCE POINT
PASS OVER Y-AXIS REFERENCE POINT
PASS OVER 4TH AXIS REFERENCE POINT

○START Pass over the reference point of each axis.
Re-start each axis individually

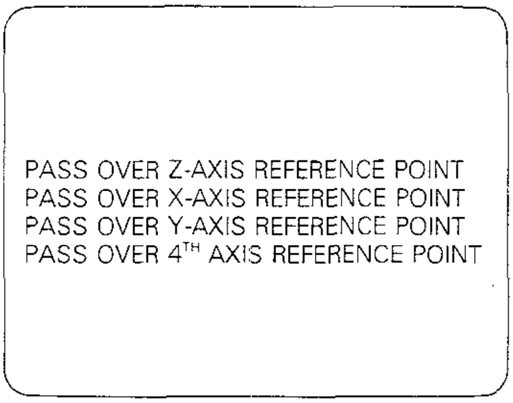
The axis sequence is determined by machine parameters set by the machine manufacturer.

In the case of linear measuring systems with interval-coded refe-ence marks, the traverse of each axis is reduced to max. 20 mm (0.78 in).

MANUAL OPERATION

Switching on the control unit
Traversing reference points

If the reference points cannot be overrun in the specified sequence due to the danger of collision, proceed as follows:



```
PASS OVER Z-AXIS REFERENCE POINT
PASS OVER X-AXIS REFERENCE POINT
PASS OVER Y-AXIS REFERENCE POINT
PASS OVER 4TH AXIS REFERENCE POINT
```

▶ MOD — Select supplementary mode.

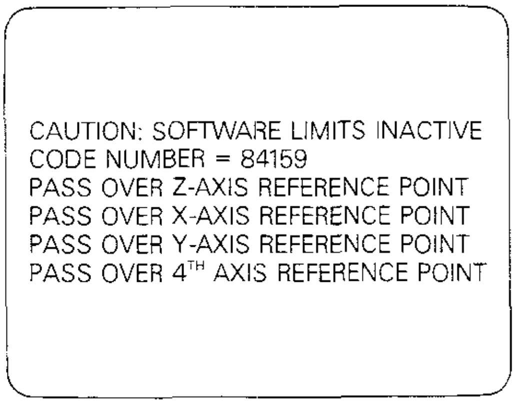
VACANT BLOCKS = 1654

▶ ↑ — Select MOD function "Code number".

CODE NUMBER =

▶ □ — Enter code number **84159**.

ENT — Press "ENTER".



```
CAUTION: SOFTWARE LIMITS INACTIVE
CODE NUMBER = 84159
PASS OVER Z-AXIS REFERENCE POINT
PASS OVER X-AXIS REFERENCE POINT
PASS OVER Y-AXIS REFERENCE POINT
PASS OVER 4TH AXIS REFERENCE POINT
```

▶ X — Pass over X-axis reference point.

Y — Pass over Y-axis reference point.

Z — Pass over Z-axis reference point.

IV — Pass over 4th axis reference point.

Reference points can be traversed manually, using the axis direction buttons, in any desired sequence, or via the external START button.

MANUAL OPERATION

Switching on the control unit
Traversing reference points

Machines with rotary encoders

If position measurement on your machine is performed by rotary encoders, and no additional cam is available for the reference pulse inhibitor, observe the following procedure after switching on the unit when traversing the reference points:

Procedure

If an axis, e.g. the X-axis, has reached the "Reference limit" cam, the message

PASS OVER X-AXIS REFERENCE POINT

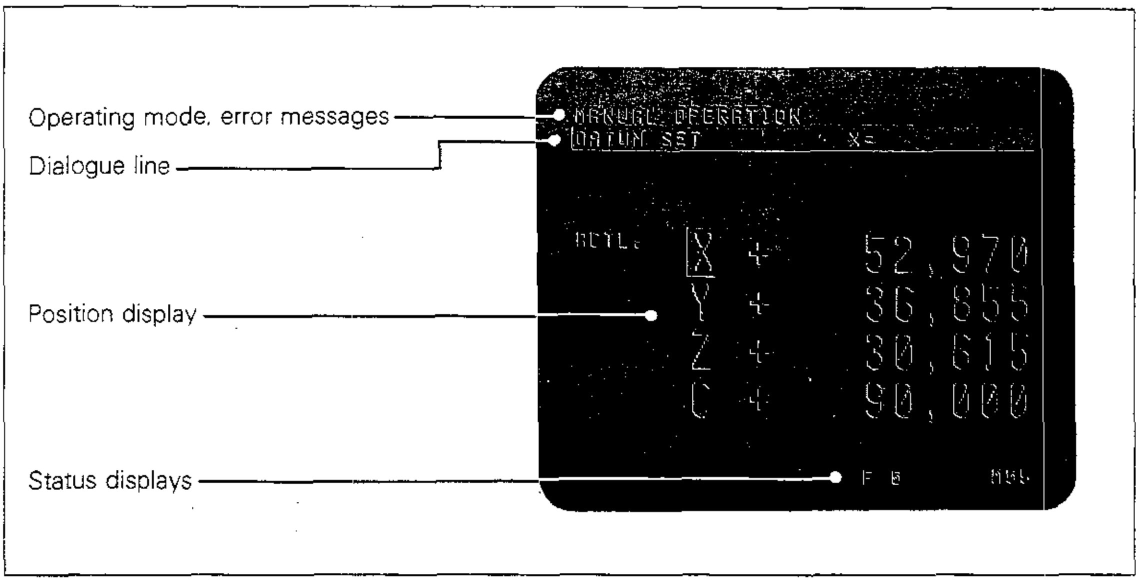
is highlighted on the screen.
The axis must be moved free of the cam before traversing the reference point. To do so, press the external START button.
If more than one axis has reached the cam, press the external START button START repeatedly. All axes must be cleared before the reference points can be traversed in the usual manner.

The procedure described above for traversing the reference points depends on a machine parameter. Your machine's manufacturer can tell you if this procedure is active.

Notes:

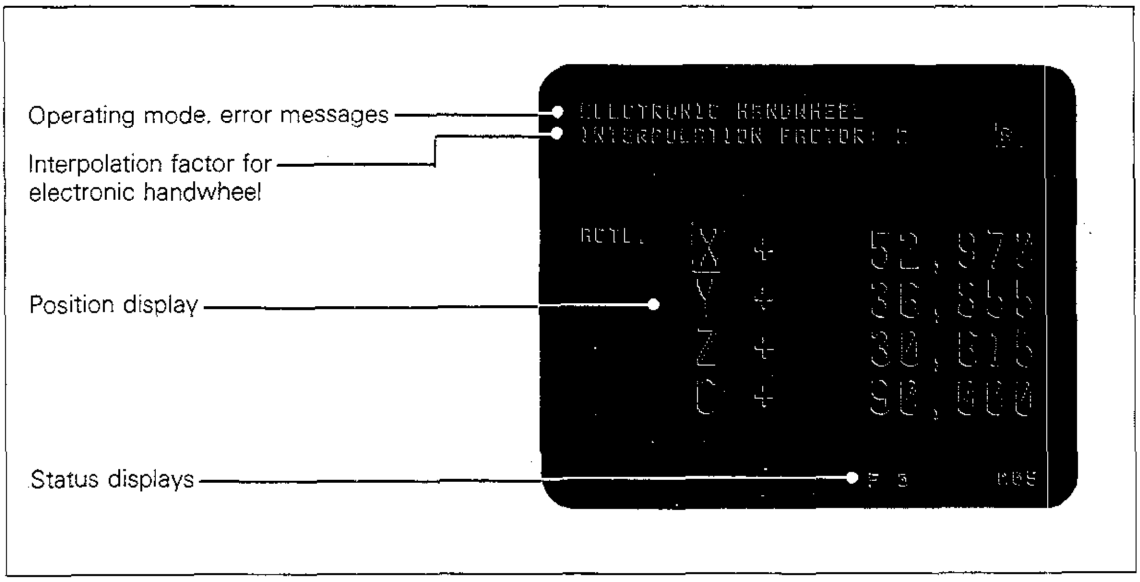
Operating modes and screen displays

Manual operation



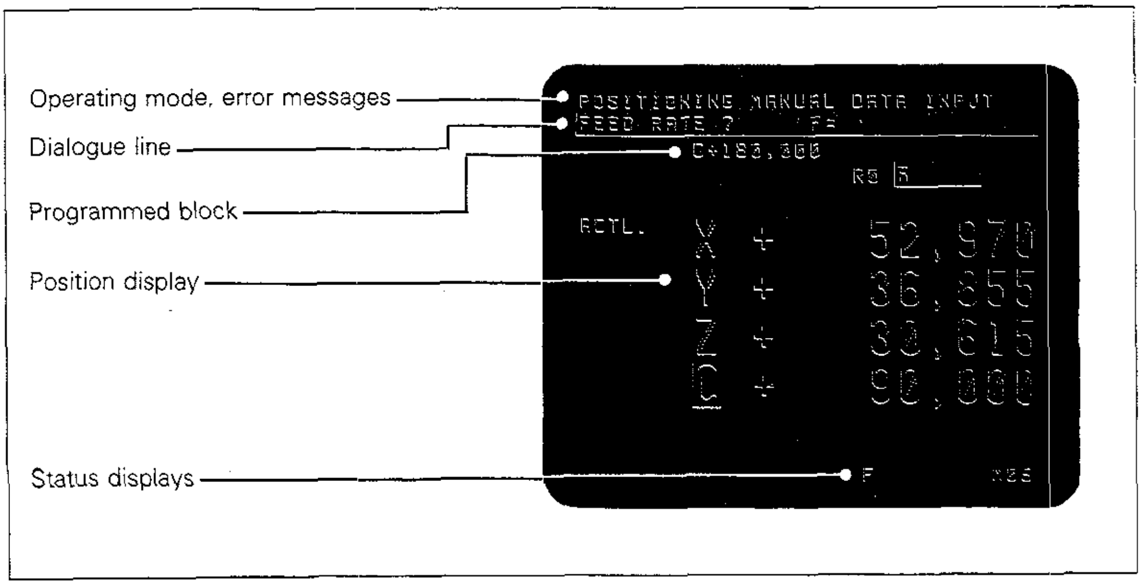
```
MANUAL OPERATION
DATUM SET              X=

ACTL.  X +      52,970
       Y +      36,855
       Z +      30,615
       C +      90,000

                    F 0       M05
```

Electronic handwheel


Operating mode, error messages
Interpolation factor for electronic handwheel
Position display
Status displays

```
ELECTRONIC HANDWHEEL
INTERPOLATION FACTOR: 3

ACTL.  X +      52,970
       Y +      36,855
       Z +      30,615
       C +      90,000

                    F 5       M05
```

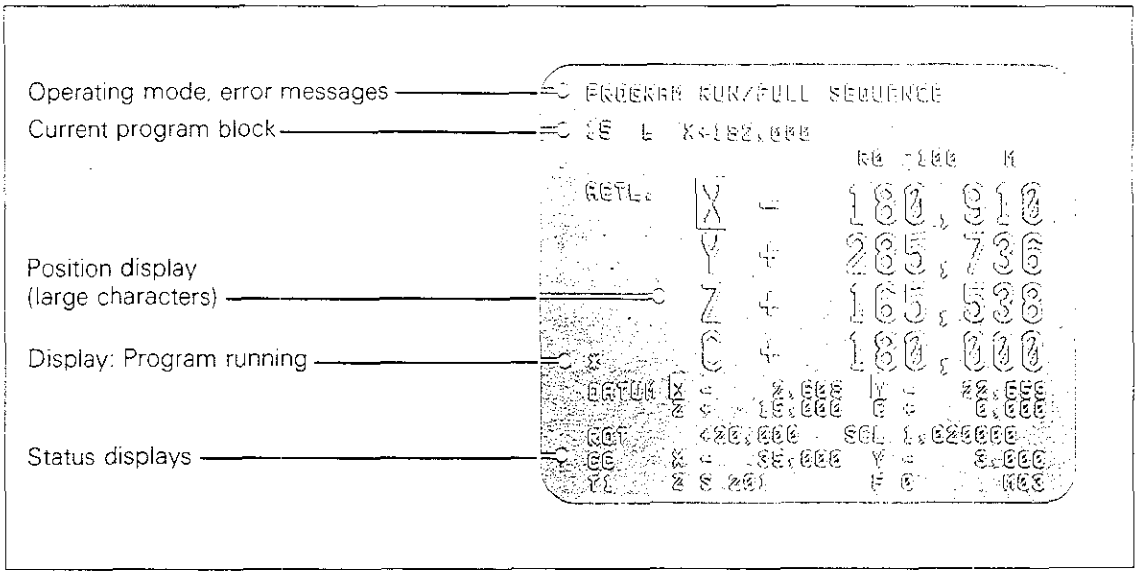
Positioning – manual data input



```
POSITIONING MANUAL DATA INPUT
FEED RATE ?     F=
      C+180,000
                    R0 R

ACTL.  X +      52,970
       Y +      36,855
       Z +      30,615
       C +      90,000

                    F         M05
```

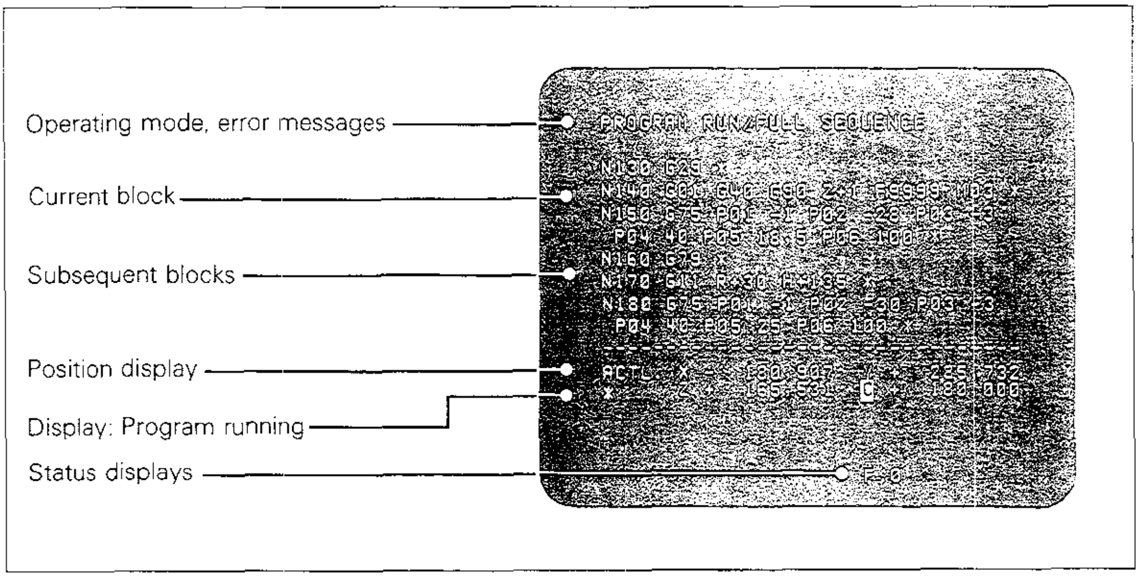
Operating modes and screen displays

Program run – full sequence (HEIDENHAIN dialogue)


Operating mode, error messages
Current program block
Position display (large characters)
Display: Program running
Status displays

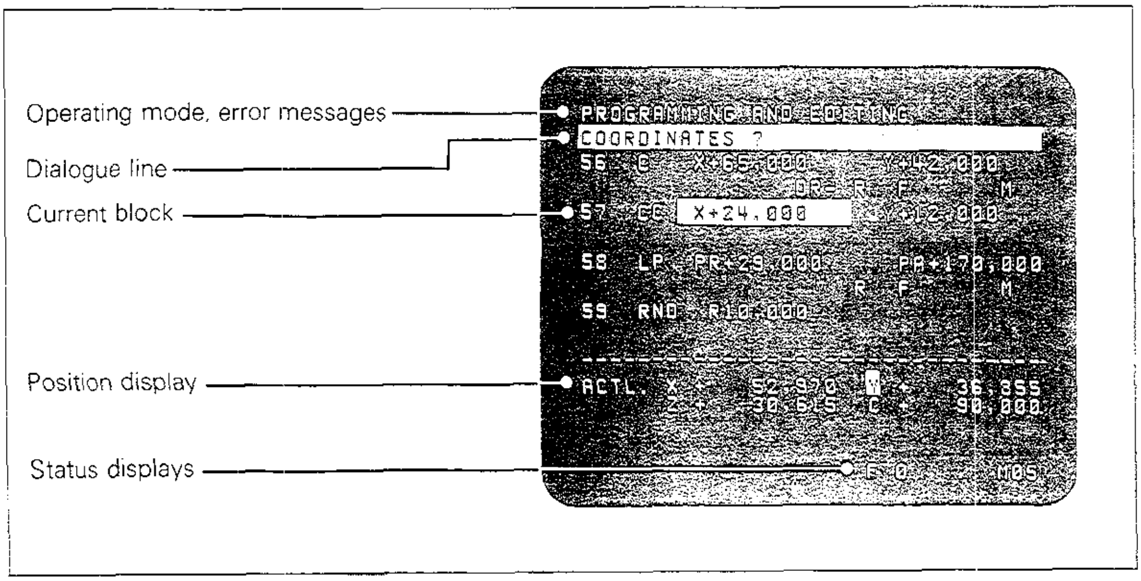
```
PROGRAM RUN/FULL SEQUENCE
15  L  X+182,000
                    R0  F100    M
ACTL.  X  -   180,910
       Y  +   285,736
       Z  +   165,538
*      C  +   180,000
DATUM X +    2,000  Y +   22,000
      Z +   15,000  C +    0,000
ROT     +20,000   SCL 1,020000
CC     X +   35,000  Y +    3,000
T1     Z S 201       F 0       M03
```

Program run – full sequence (ISO format)



```
PROGRAM RUN/FULL SEQUENCE
N130 G29 *
N140 G00 G40 G90 Z+1 F9999 M03 *
N150 G75 P01 -1 P02 -28 P03 -3
  P04 40 P05 18,5 P06 100 *
N160 G79 *
N170 G11 R+30 H+135 *
N180 G75 P01 -1 P02 -30 P03 -3
  P04 40 P05 25 P06 100 *
-------------------------------------
ACTL. X -  180,907  Y +  285,732
*     Z +  165,531  C +  180,000

                F 0
```

Programming and editing



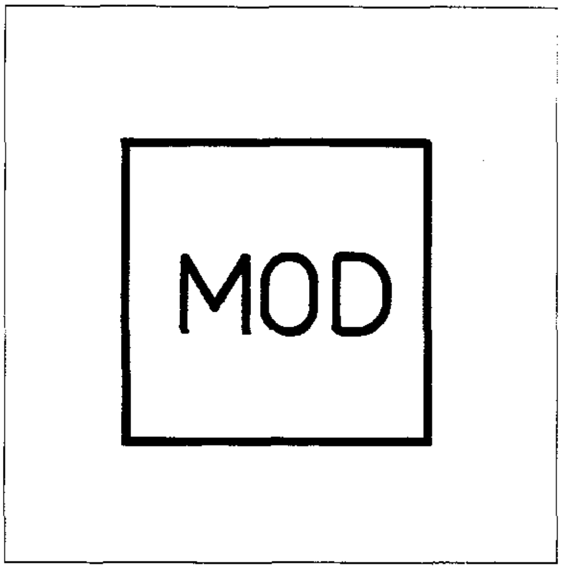
```
PROGRAMMING AND EDITING
COORDINATES ?
56  C   X+65,000       Y+42,000
               DR- R  F       M
57  CC  X+24,000       Y+12,000

58  LP  PR+29,000      PA+170,000
                   R  F       M
59  RND  R10,000

-------------------------------------
ACTL.  X +   52,970  Y +   36,855
       Z +   30,615  C +   90,000

                F 0          M05
```

Supplementary operating modes

Introduction

In addition to the main operating modes, the TNC 151/TNC 155 provides a number of **supplementary operating modes,** or MOD* functions. The supplementary modes are selected by pressing the MOD key. When this key is pressed, the first MOD function "Vacant blocks" is displayed on the dialogue line.
You can use the ↓ ↑ keys to page forward and backward through the MOD function menu. You can page forward with the MOD key.

Exit the supplementary mode function by pressing the DEL □ key.



* MOD is a shortened form of the word "mode".

Restrictions

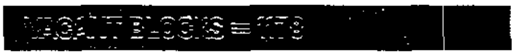
While a program is running in modes ≣→ or →, only the following supplementary operating modes can be selected:
- position display size (large or small characters)
- vacant blocks

The following supplementary operating modes can be selected while the message
= POWER INTERRUPTED =
is displayed on the screen:
- code number
- user parameters
- NC software number
- PLC software number

Vacant blocks

The MOD function "Vacant blocks" indicates the number of blocks still available in program memory.
When programming in ISO format, the number of available characters (bytes) is displayed.

Sample display:



VACANT BLOCKS = 1178

Supplementary operating modes
How to select and exit MOD functions

Select function

Operating mode ______________________

Initiate dialogue ______________________ MOD

VACANT BLOCKS = 1974

Select MOD functions via editing keys ▶ ↓ ↑

or
by paging (forward only) with the MOD key. ▶ MOD

Exit function

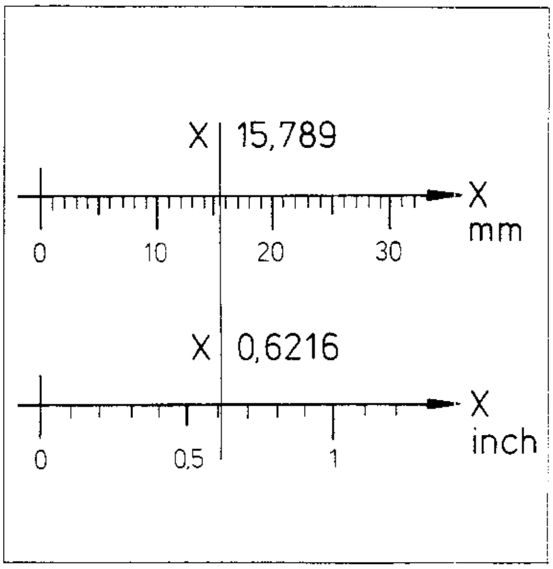
LIMIT X+ = X+ 350.000 ▶ DEL □ Exit supplementary mode.

Press the ENT key to transfer numerical entries to memory before exiting MOD functions.

Supplementary operating modes

Changeover mm/inch

You can use the MOD function "mm/inch" to determine whether the control system displays position data in millimetres or inches. Press the ENT key to change from inch to mm and vice versa. When this key is pressed, the control system switches to the alternate measuring unit.



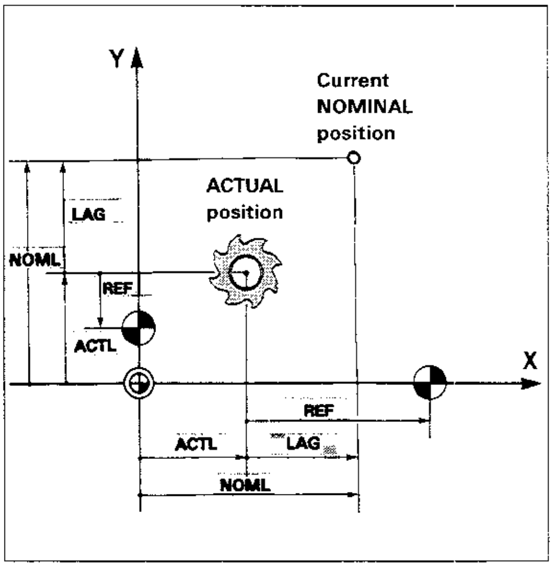
You can recognize whether the current display is in mm or inches by noting the number of decimal places following the decimal point:

X 15.789	mm display
X 0.6216	inch display

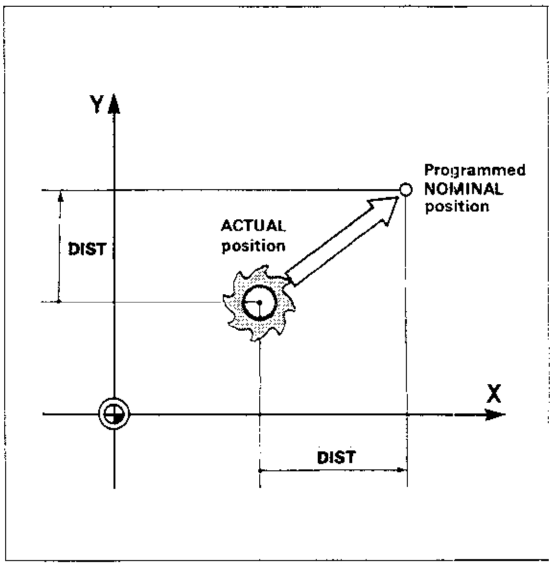
Position data display

The type of position data displayed on the screen can be selected via the MOD function "position display":

- current actual position display: **ACTL**
- distance from reference points: **REF**
- difference between current nominal and actual positions (trailing error or lag): **LAG**
- Current nominal position calculated by control system: **NOML**



- Display of distance to go to nominal position (difference between programmed nominal and current actual position): **DIST**



Beginning with software version 03, jammed axes are indicated on the screen by a decimal point behind the axis designation.

Supplementary operating modes

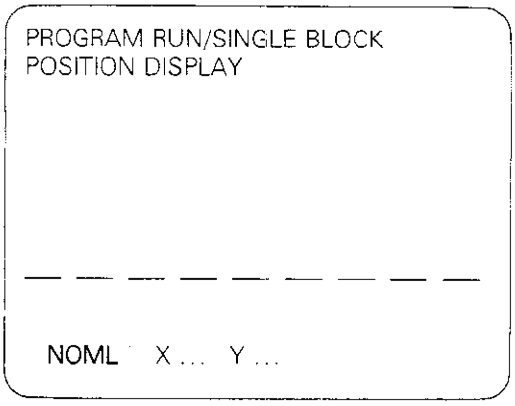
Changeover mm/inch

Select MOD function.	▶ MOD ↓	
CHANGE MM/INCH The control displays position data in mm, you want to change to inch:	▶ ENT	Changeover.

Change from inch to metric in the same way.

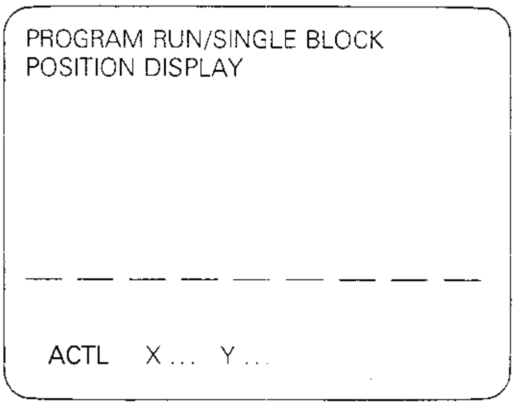
Position display

Select MOD function. ▶ MOD ↓


PROGRAM RUN/SINGLE BLOCK
POSITION DISPLAY

NOML X ... Y ...

To change the display back to "actual position":	▶ ENT	Changeover. (Press repeatedly until ACTL appears.)


PROGRAM RUN/SINGLE BLOCK
POSITION DISPLAY

ACTL X ... Y ...

To change the display back to "nominal position":	▶ ENT	Changeover. (Press repeatedly until NOML appears.)

Follow the same procedure to change the position data display to REF, LAG and DIST.

Supplementary operating modes

Position display large/small

You can change the height of the characters in the position display on the screen in ⊟→ "Program run/single block" or → "Program run/full sequence" (automatic) modes.
In the case of small-character display, the screen displays four program blocks (preceding, current, next and block after next); with large-character display, only the current block is shown.

If you are programming in ISO format, position data cannot be displayed in large characters because program blocks may be longer than two lines.

Block number increment

If you are programming in ISO format, you can determine the interval between block numbers via the MOD function "Block number increment". If the increment is 10, for example, blocks will automatically be numbered as follows:
N10
N20
N30
etc.
Block increments may lie within a range of 0 - 99.

Baud rate

The MOD function "Baud rate" is used to determine the data transmission speed for the interface (see "Baud rate entry").

V.24-interface

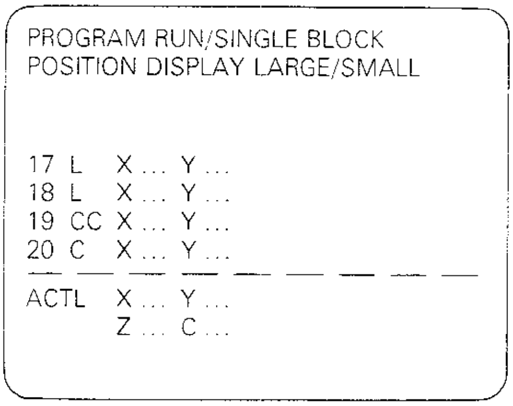
The interface can be switched to the following operating modes via the MOD function "V.24 . interface":
- magnetic tape operation (ME)
- floppy disk operation (FE)
- EXT-operation with other external devices.

(see "V.24 interface – definition").

Supplementary operating modes

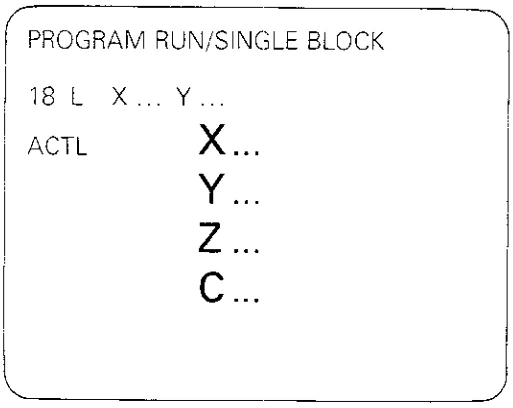
Position display large/small

Select MOD function "Position display large/small": ▶ MOD ↓



```
PROGRAM RUN/SINGLE BLOCK
POSITION DISPLAY LARGE/SMALL


17  L   X ...  Y ...
18  L   X ...  Y ...
19  CC  X ...  Y ...
20  C   X ...  Y ...
—  —  —  —  —  —  —  —  —  —
ACTL    X ...  Y ...
        Z ...  C ...
```

To switch position display to large: ▶ ENT



```
PROGRAM RUN/SINGLE BLOCK

18  L   X ...  Y ...

ACTL         X...
             Y...
             Z...
             C...
```

Follow the same procedure to switch from large to small display. ▶ ENT

Block number increment (ISO only)

Select MOD function "Block number increment": ▶ MOD ↓

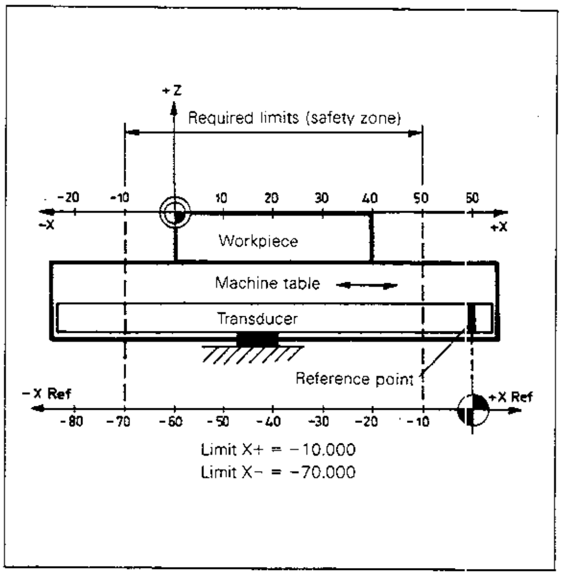
BLOCK NO. INCREMENT = ▶ □ Enter increment for block numbers.

ENT Transfer ent'y to memory.

Supplementary operating modes

Software limits

Using the MOD function "Limits", you can confine tool traverse to specified limits, to prevent collisions with certain workpieces, for example. Maximum traverse ranges are defined by software limit switches. Traverse range limits are determined on each axis consecutively in + and – directions, based on the reference point. For this reason, the position display must be switched to REF when defining the limiting positions.



Supplementary operating modes

Setting the software limits

Operating mode ______ ☞ or ⊛

☞ Switch position display to REF to set traverse range limits.

Select MOD function "Limits". ▶ MOD ↓

▼

LIMIT X+ = +30000.000

Use the external axis direction buttons or the electronic handwheel to move to the limiting position.

Program position indicated, e. g. −10.000 ▶ □ Enter X-value.

▼ ENT Transfer to memory.

▼

LIMIT X+ = −10.000

Select next MOD function "Limits": ▶ ↓

▼

LIMIT X− = −30000.000

Use the external axis direction buttons or the electronic handwheel to move to the limiting position.

Program position indicated, e. g. −70.000 ▶ □ Enter X-value.

▼ ENT Transfer to memory.

▼

LIMIT X− = −70.000

Follow the same procedure to limit remaining traverse ranges.

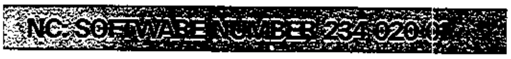
☞ If you decide not to limit the traverse ranges, enter the values +30000.000 or −30000.000 for the corresponding axes.

Supplementary operating modes

NC software number

This MOD function displays the software number of the TNC control system.

Sample display:

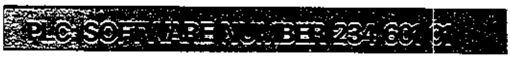


NC: SOFTWARE NUMBER 234 020 01

PLC software number

This MOD function displays the software number of the integrated PLC.

Sample display:



PLC: SOFTWARE NUMBER 234 601 01

User parameters

This MOD function provides the user with access to up to 16 machine parameters. User parameters are defined by the machine manufacturer, who can also provide you with further information.

Code number

With this MOD function, code numbers can be used to select a special procedure for "reference point approach" or to cancel the "edit/erase protection" for programs (see appropriate chapter).

Supplementary operating modes

User parameters

Select MOD function "User parameters".	MOD ↓	
USER PARAMETERS	ENT	Transfer MOD function.
USER PAR.1 = 0 *	↓ ↑	Select desired user parameter.
* The dialogue text is determined by the machine manufacturer. The above message is displayed if no dialogue text has been defined.	□	Change parameter if required.
	ENT	Transfer entry to memory.

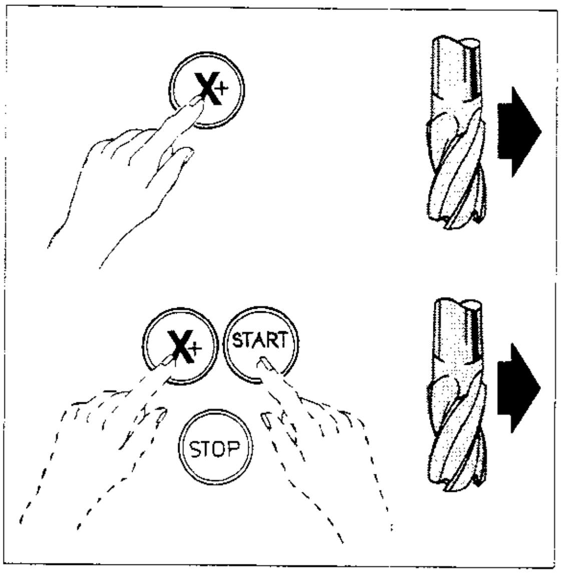
Exit user parameter function

To exit the MOD function "User parameters":	DEL	Exit MOD function.

Notes:

Manual operation
Operating mode: "Manual operation"

In "Manual operation" mode ✋, the machine axes can be moved via the external axis direction buttons (X) (Y) (Z) (IV).



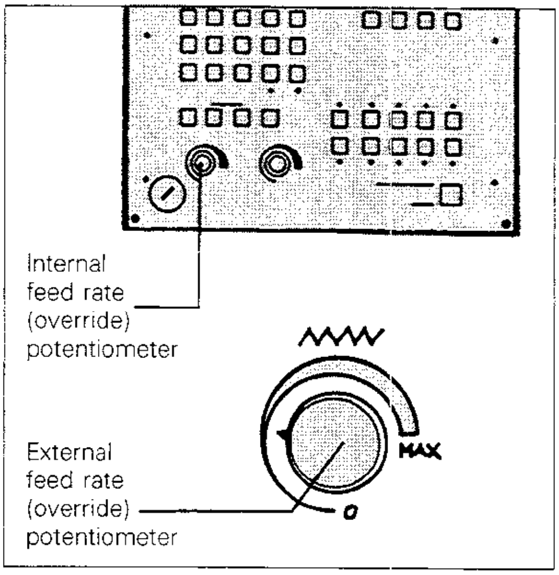
Jog mode

The machine axis is moved as long as the appropriate external axis direction button is pressed. The machine axis stops immediately when the axis direction button is released.
Multiple axes may be traversed simultaneously in jog mode.

Continuous operation

If an **axis direction button** and the **external start button** are pressed **at the same time,** the selected machine axis will continue to move after the button is released. Movement can be **stopped** by pressing the **external stop button.**

> In the ✋ operation mode, the X Y Z IV buttons are used to define the workpiece datum (see "Workpiece datum).


Internal feed rate (override) potentiometer
MAX
0
External feed rate (override) potentiometer

Feed rate

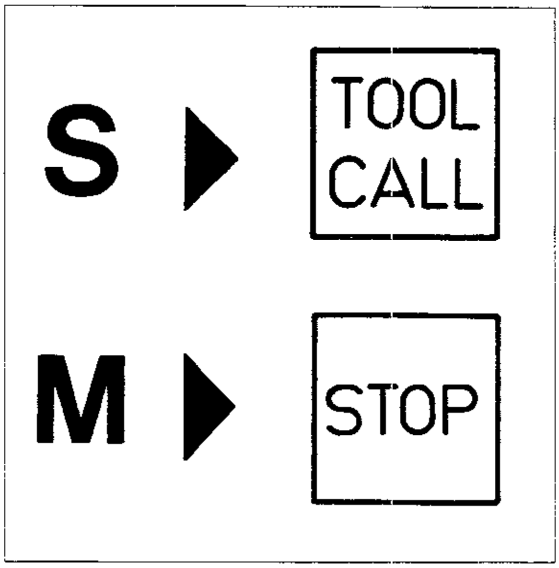
The traversing speed (feed rate) can be set
- via the control system's **internal feed rate override,** or
- via the machine's **external feed rate override** (depending on the specified machine parameters).

The specified feed rate is displayed on the screen.

Spindle speed

Spindle speed can be adjusted via the TOOL CALL key (see "TOOL CALL").
With analogue output, the programmed spindle speed can be altered via the spindle override function while the program is running.

> Your machine tool manufacturer or supplier can tell you whether your machine operates with coded or analogue output for spindle speeds.



Miscellaneous functions

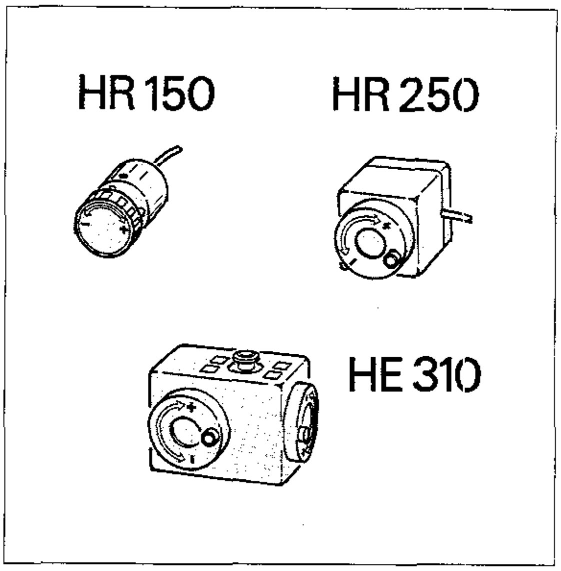
You can enter miscellaneous functions via the STOP key (see "Program stop").

Manual operation
Operating mode: "Electronic handwheel"

Versions

The control unit can be equipped with an electronic handwheel that can be used for machine set-up, for example.
The electronic handwheel is available in three versions:

- **HR 150:** 1 handwheel for integration in machine control panel,
- **HR 250:** 1 handwheel in portable unit,
- **HE 310:** 2 handwheels in portable unit with axis keys and supplementary emergency stop button.



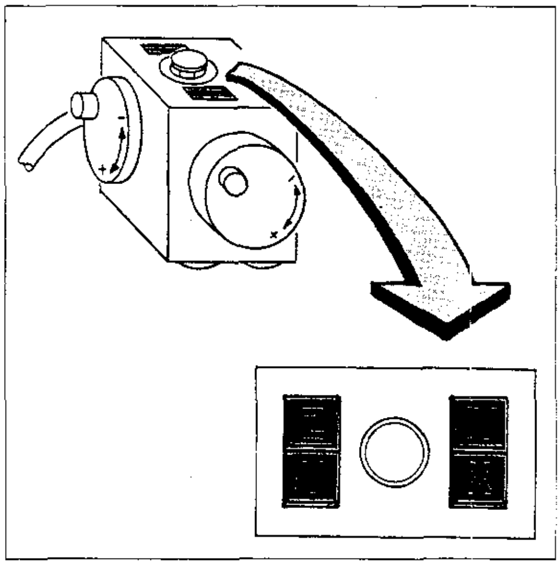
Interpolation factor

The interpolation factor determines the distance traversed per handwheel revolution (see chart at right).

Interpolation factor	Distance traversed per revolution in mm
1	10.0
2	5.0
3	2.5
4	1.25
5	0.625
6	0.313
7	0.156
8	0.078
9	0.039
10	0.020

Operation

Versions HR 150 and **HR 250:**
Use the X Y Z IV axis keys of the control unit to select machine axes for the handwheel.
Version HE 310 with two handwheels: the handwheel unit is equipped with additional X Y Z IV axis keys. They can be used to assign one handwheel to either the X- or IV-axis and the other handwheel to the Y- or Z-axis.
The axis being controlled by the electronic handwheel is highlighted on the screen.



In ⊛ mode, the machine axes can also be moved by means of the external axis direction buttons X Y Z IV.

Manual operation
Operation mode: "Electronic handwheel"

Operation HR 150/ HR 250

Operating mode and dialogue initiation

INTERPOLATION FACTOR: 3	▶ □	Specify required interpolation factor, e.g. 4.
	▼ ENT	Press ENTER key.

INTERPOLATION FACTOR: 4 **4**	▶ Y	Specify required axis, e.g. Y.

The tool can now be moved in positive or negative Y-direction by means of the electronic handwheel.

Operation HE 310

Operating mode and dialogue initiation

INTERPOLATION FACTOR: 4	▶ □	Specify required interpolation factor, e.g. 6.
	▼ ENT	Press ENTER key.

INTERPOLATION FACTOR: 6 **6**	▶ Z	Select first traversing axis on handwheel unit, e.g. Z.
	▼ X	Select second traversing axis on handwheel, e.g. X.

The tool can be moved in positive or negative Z-direction via the first handwheel, in positive or negative X-direction via the second handwheel.

Manual Operation
Step Positioning

Step Positioning

Beginning with software version 03:
The step positioning can be activated via the integrated PLC. This makes it possible to enter a supplementary step measure in the operation mode "electronic handwheel". When an axis direction key is pressed the corresponding axis moves by the distance that was entered.

Manual Operation
Step measure entry

Step measure entry

Operation mode and Dialog initiation ——— ⊛

SUBDIVISION FACTOR: 3	▶ ↓	move bright field one line downward
FEED: 1.000	▶ □ ▼ ENT	enter desired feed e.g. 2 (maximum entry value: 10 mm) transfer entry to memory
FEED: 2.000	▶	X Y Z IV Press external axis key. The chosen axis moves by the distance entered.

Notes:

Coordinate system and dimensioning

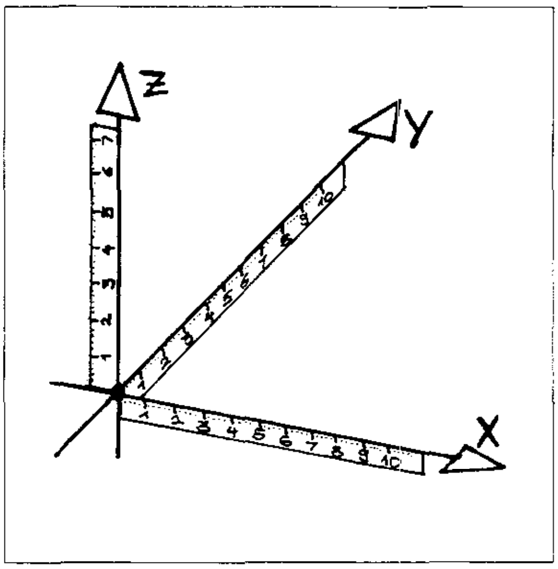
Introduction

An NC machine cannot process a workpiece automatically unless all the machining operations are completely defined by the NC program. The nominal positions of the tool, relative to the workpiece, must be defined in the NC program. A reference system, a system of coordinates, is required to define the nominal tool positions. The TNC allows you to use either rectangular or polar coordinates, depending on how the workpiece is dimensioned.

Rectangular or Cartesian * coordinate system

A rectangular coordinate system is formed by two axes in the plane and by three axes in space. These axes intersect at a single point and are perpendicular to one another. The point where the axes intersect is called the origin or zero point of the coordinate system. The axes are identified by the letters X, Y and Z.
Imaginary scales, the zero points of which coincide with the zero point of the coordinate system, are located on the axes. The arrow indicates the positive direction of the scales.

* Named for the French mathematician René Descartes, referred to in Latin as Renatus Cartesius (1596 - 1650)

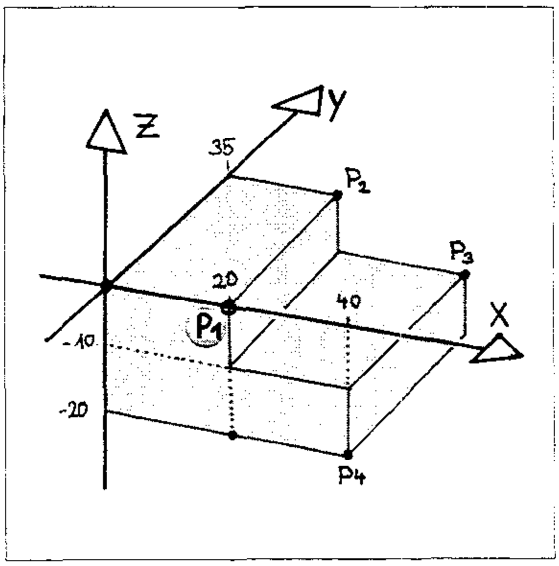


Example

Any point on a workpiece can be described with the aid of the Cartesian coordinate system by indicating the appropriate X, Y and Z coordinates:

P1 X = 20
Y = 0   abbreviated: P1 (20; 0; 0)
Z = 0

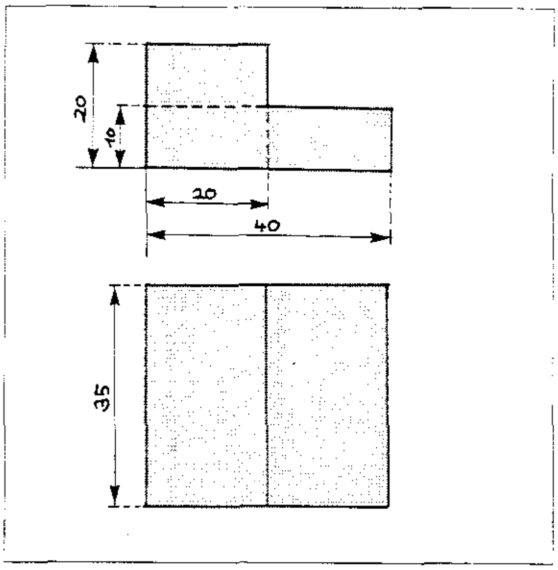
P2 (20; 35; 0)
P3 (40; 35; −10)
P4 (40; 0; −20)



Coordinate system and dimensioning

Coordinate data

The Cartesian coordinate system is particularly suitable if the production drawing is dimensioned "rectangularly".
In the case of workpiece with circular elements or angular dimensions, it is often more convenient to define positions in polar coordinates.

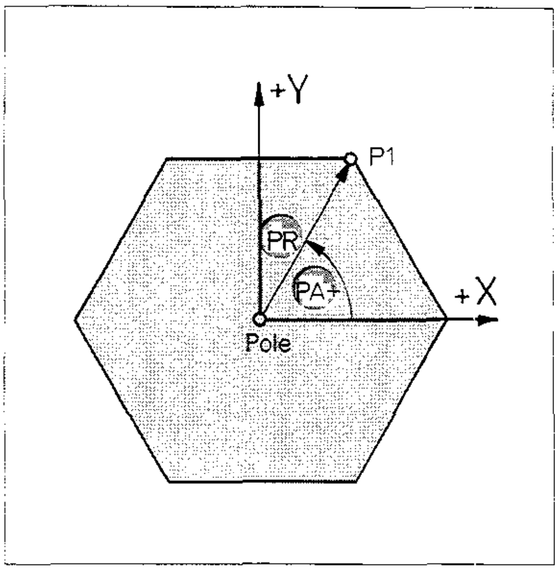


Polar coordinates

The polar coordinate system is used to define points in a plane. The point of reference is the pole (the zero point of the coordinate system) and one direction (reference axis for the specific angle).

Points are described as follows:
By indicating the polar coordinate radius **PR** (distance between pole and point P1) and by the angle **PA** formed between the reference direction (in the illustration, the + X-axis) and the connecting line pole-to-point P1.

A is the abbreviation for angle.



Input range

The polar coordinate angle PA is entered in degrees (°), (decimal notation).
Input range for linear interpolation:
absolute or incremental –360° to +360°

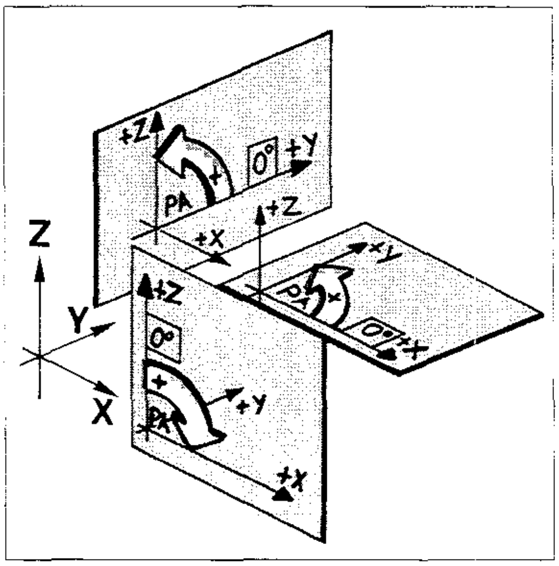
Input range for circular interpolation:
absolute or incremental –5400° to +5400°

PA positive: angle specified counterclockwise
PA negative: angle specified clockwise

Angular reference axis

The angular reference axis (0°-axis) is
the +X-axis in the X, Y plane,
the +Y-axis in the Y, Z plane,
the +Z-axis in the Z, X plane.

The prefix sign for the angle PA can be determined with the aid of the illustration at the right.

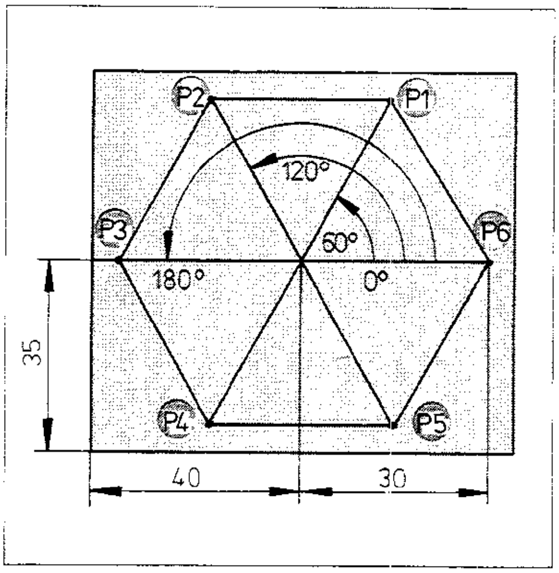


Coordinate system and dimensioning

Example

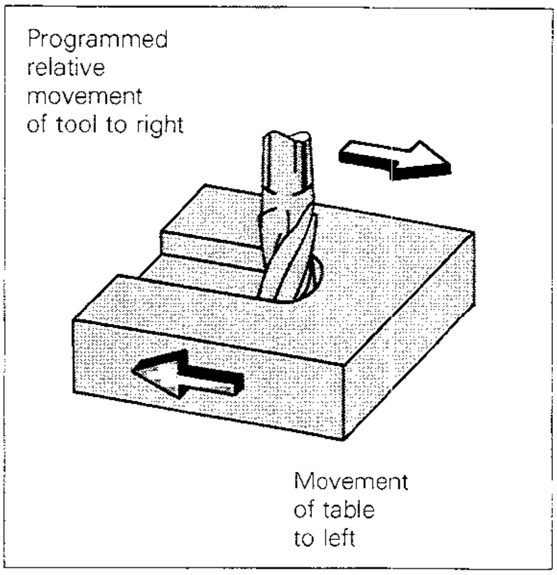
Point	Polar coord. radius PR	Polar coord. absolute	Angle PA incremental
P1	30	60°	60°
P2	30	120°	60°
P3	30	180°	60°
P4	30	240°	60°
P5	30	300°	60°
P6	30	360°	60°

The polar coordinate system is particularly suitable for describing points on a workpiece if the production drawing contains primarily angles, as shown in the example at right.



Relative tool movement

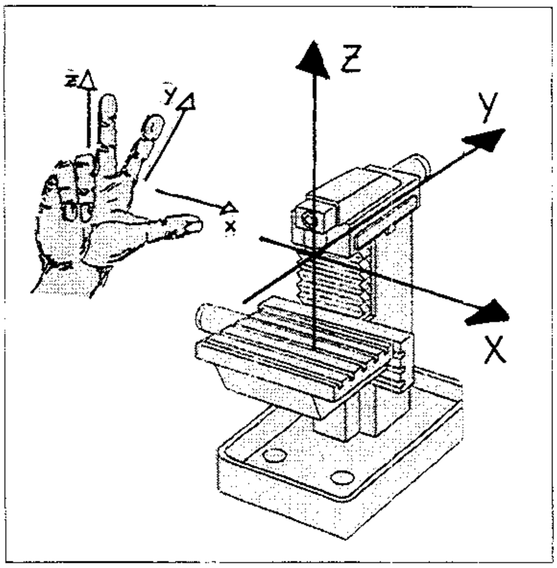
When machining a workpiece, it makes no difference whether the **tool** moves on a stationary workpiece, or whether the **workpiece** moves while the tool remains stationary.
Only the relative tool/workpiece movement is important when creating a program.
This means, for example:
If the worktable of the milling machine, together with the clamped workpiece, moves to the left, the movement of the too , relative to the workpiece, is to the right. If the table moves upward, the relative tool movement is down. The tool actually moves only when the headstock moves; thus machine movement always corresponds to the relative tool movement.


Programmed relative movement of tool to right
Movement of table to left

Correlation of machine slide movement and the coordinate system

Two factors must be determined before the control system can properly interpret workpiece coordinates in the machining program:

- which slide will move parallel to which coordinate axis (correlation of machine axis and coordinate axis)
- what relationship exists between the position of the machine slides and the coordinate data in the program.

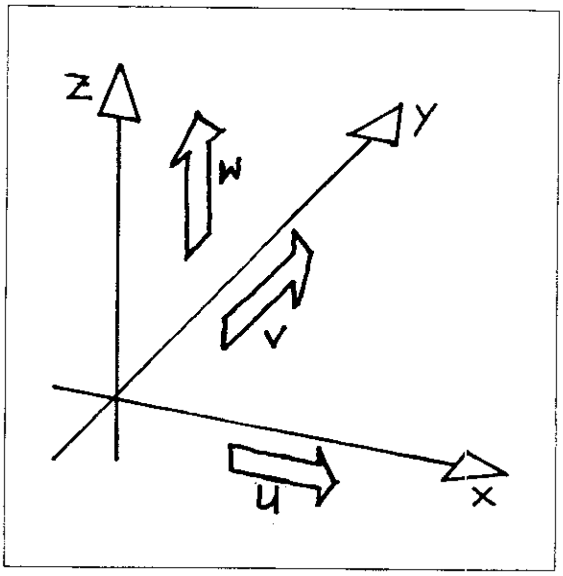


The three main axes

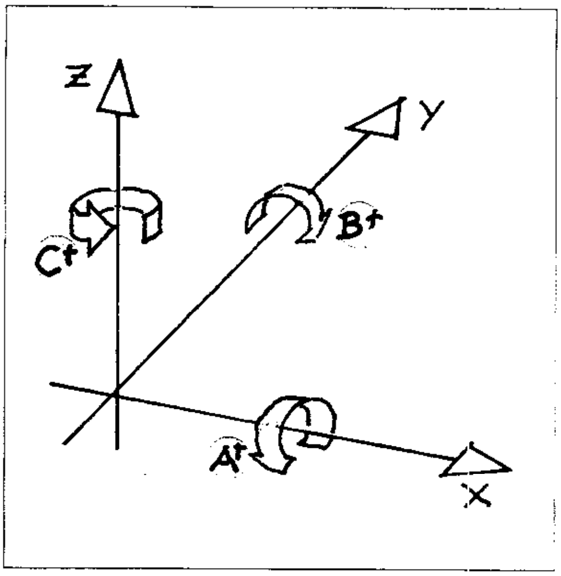
The allocation of the three workpiece coordinate axes to the machine axes has been defined by the ISO 841 standard for various machine tools. The direction of traverse can be easily noted by applying the "right-hand rule".

The fourth axis

If a fourth axis is used, the machine manufacturer will determine whether it controls a **rotary table** or an additional **linear axis** (e.g. a controlled quill) and how the axis is identified on the screen. An additional linear axis moving parallel to the X-, Y- or Z-axis is referred to as the U-, V- or W-axis.



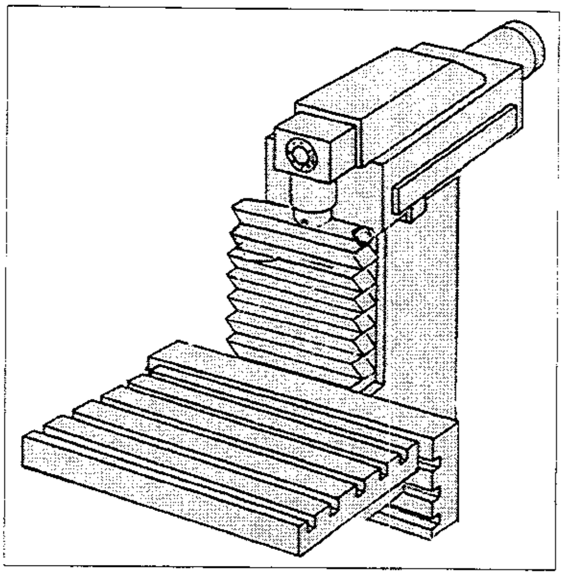
When programming the movement of a rotary table, the angle of table rotation on the A-, B- or C-axis is indicated in degrees (°), (decimal notation).
In this case, we refer to an A-, B- or C-axis movement, meaning a rotation about the X-, Y- or Z-axis.

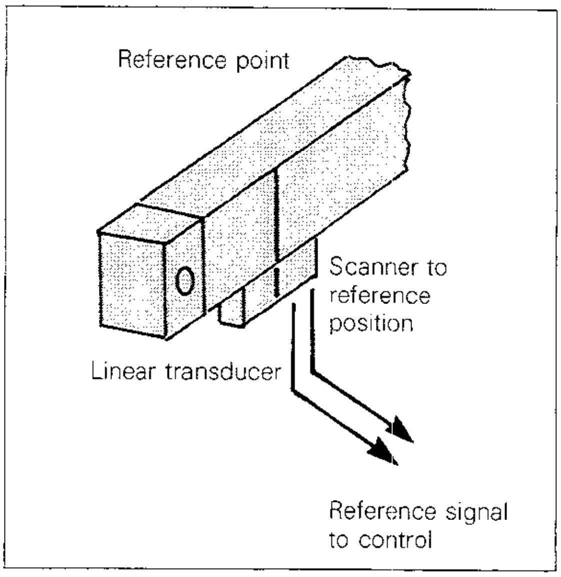


Coordinate system and dimensioning

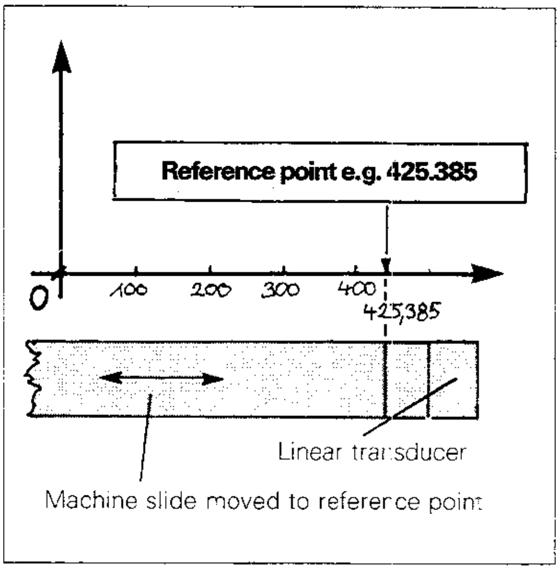
Allocation of the coordinate system

The position of the machine coordinate system is determined as follows:
The machine slide is moved over a defined position, the reference position (also called the reference point). When this point is traversed, the transducer issues an electrical signal, the reference signal, to the control system. Once this signal is received, the control system assigns a given coordinate value to the reference point. The procedure is repeated for all machine slides in order to define the position of the machine's coordinate system.





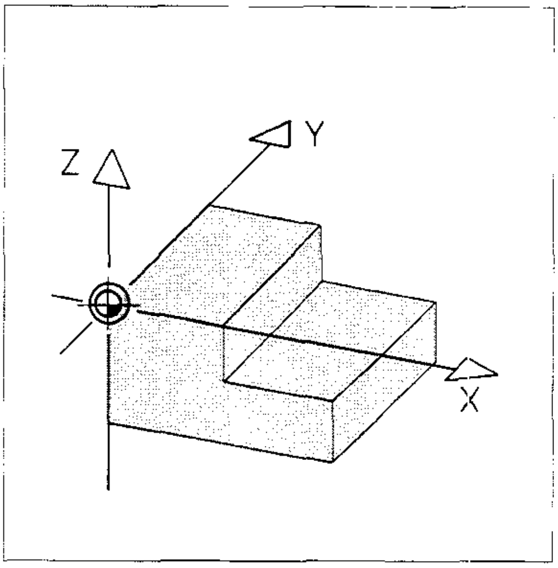
The reference points must be traversed after every interruption of the power supply, which causes the correlation between the coordinate system and the machine slide position to be lost. All operating options are disabled until the reference points are traversed.
Once the reference points have been traversed, the control system "recognizes" the previous workpiece datum (see next chapter) and the software limit switches again.


Reference point e.g. 425.385
0
100
200
300
400
425,385
Linear transducer
Machine slide moved to reference point

Coordinate system and dimensioning
Setting the workpiece datum

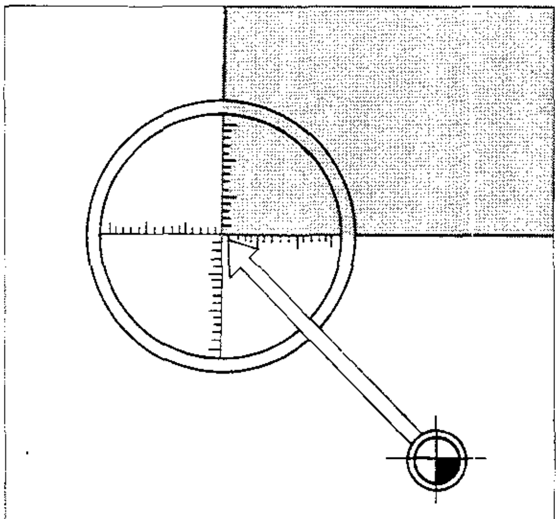
Setting the workpiece datum

To avoid unnecessary calculating effort, the workpiece datum is located at that point on the workpiece on which workpiece dimensions are based. For reasons of safety, the workpiece datum is almost always located at the "highest" point of the workpiece on the tool axis.



Setting the workpiece datum in the machining plane with an optical contour scanner

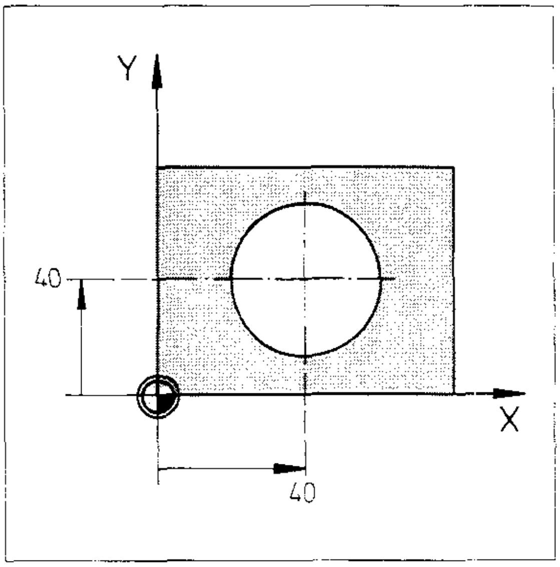
Approach the desired workpiece datum and reset the indicator for both axes of the machining plane to zero.



Symbol for workpiece datum

With a centring device

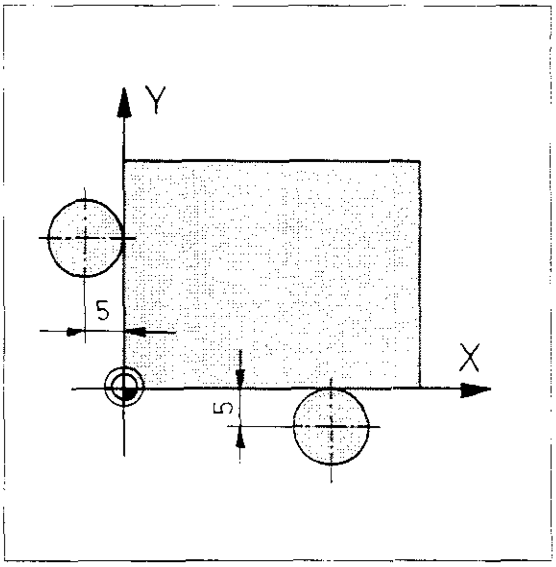
Move to a known position, e. g. to the centre of a hole, with the aid of the centring device. Then enter the coordinates of the hole centre into the control system (in this case X = 40 mm, Y = 40 mm). This defines the location of the workpiece datum.



Coordinate system and dimensioning
Setting the workpiece datum

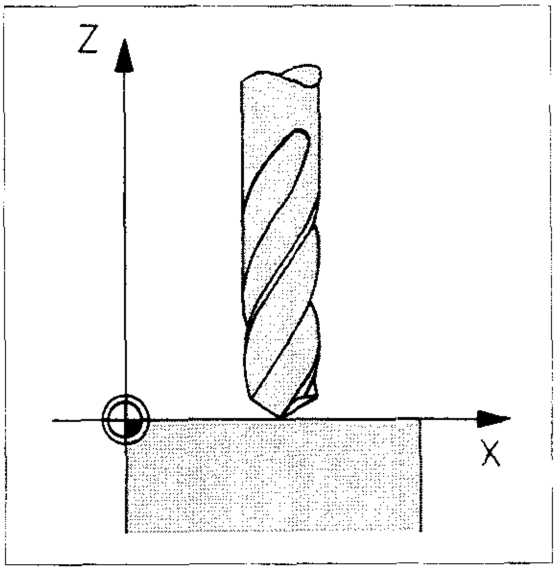
With edge finder or tool

Move the tool to the workpiece reference surfaces. When the tool contacts the surface, set the actual value display for the corresponding axis to the value of the tool radius, with a negative prefix sign (in this case e. g. X = –5 mm, Y = –5 mm).



Setting the workpiece datum on the tool axis by tool contact with the workpiece surface

Move the zeroing tool to the workpiece surface. When the tool contacts the surface, set the actual value display for the tool axis to zero.
If contact with the workpiece surface is not desired, place a thin piece of sheet metal of known thickness (approx. 0.1 mm) between the tool tip and the workpiece. Enter the thickness of the sheet metal (e. g. Z = 0.1 mm) instead of zero.

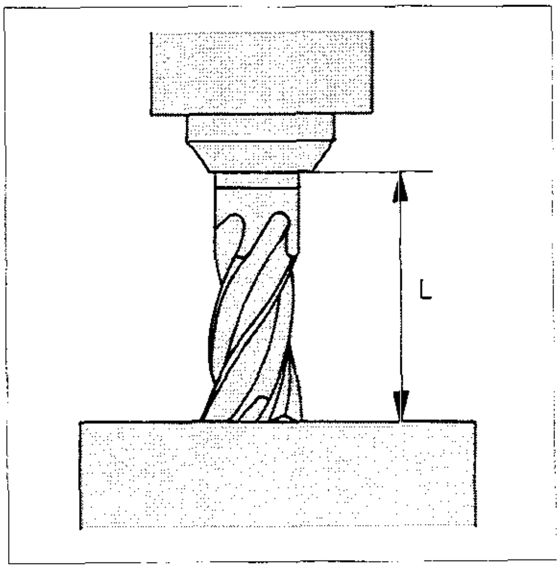


With preset tools

If preset tools are used, i. e. if tool lengths are known in advance, probe the workpiece surface with any of the tools. To assign the value "0" to the surface, specify the length L of the tool, with a positive prefix sign, as actual value of the tool axis. If the workpiece surface has a value other than zero, enter the following actual value:

(actual value Z) = (tool length L) + (position of surface)

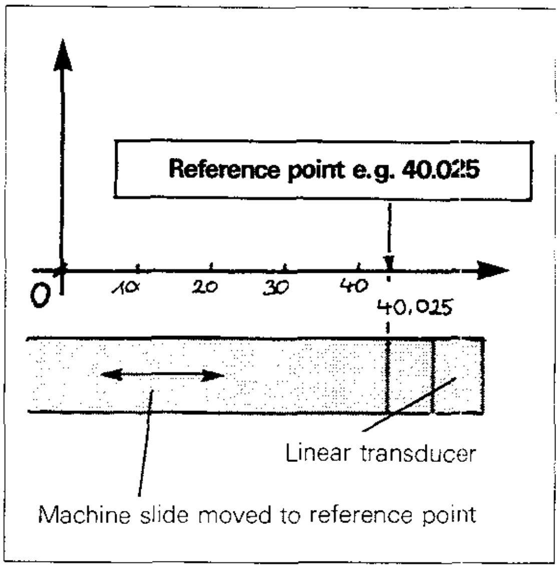
Example:
Tool length L = 100 mm
Position of workpiece surface = + 50 mm
Actual value Z = 100 mm + 50 mm = 150 mm



Coordinate system and dimensioning
Setting the workpiece datum

REF-values

In setting the workpiece datum, defined numerical values, called "REF-values", are assigned to the reference points. These values are automatically saved by the control system. This makes it possible to find the previously defined workpiece datum after an interruption of power, by simply traversing the reference points.


Reference point e.g. 40.025
0
10
20
30
40
40,025
Linear transducer
Machine slide moved to reference point

Coordinate system and dimensioning
Setting the workpiece datum

Setting the workpiece datum

Operating mode ✋

The workpiece datum cannot be set unless the actual position is displayed.
Select this display via the MOD function if required.

Dialogue initiation X

DATUM SET X =	▶ □	Specify value for X-axis.
	▼ ENT	Press ENTER.

Dialogue initiation Y

DATUM SET Y =	▶ □	Specify value for Y-axis.
	▼ ENT	Press ENTER.

Dialogue initiation Z

DATUM SET Z =	▶ □	Specify value for Z-axis.
	▼ ENT	Press ENTER.

Dialogue initiation IV

DATUM SET C =	▶ □	Specify value for 4th axis.
	▼ ENT	Press ENTER.

Depending on the specified machine parameters, the 4th axis is identified and displayed as A, B, C or U, V or W.

If the dialogue for setting the datum was initiated inadvertently and you do not wish to set the datum, proceed as follows:

- if programming in HEIDENHAIN format, press NO ENT .
- if programming in standard ISO format, press DEL □

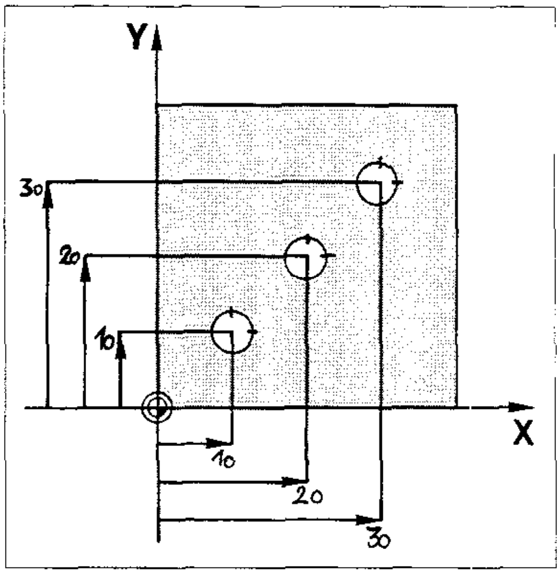
Coordinate system and dimensioning
Absolute/incremental dimensions

Dimensioning

Dimensions in workpiece drawings are indicated either in absolute or in incremental (chain) dimensions.

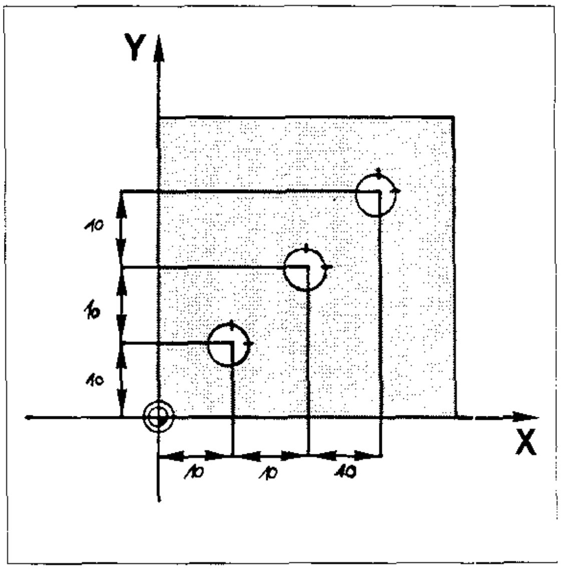
Absolute dimensions

Absolute dimensions in the machining program are based on a fixed, absolute point, e.g. the zero point of the coordinate system (corresponds to the workpiece datum).



Incremental dimensions

Incremental dimensions in the machining program are based on the previous programmed nominal position of the tool.

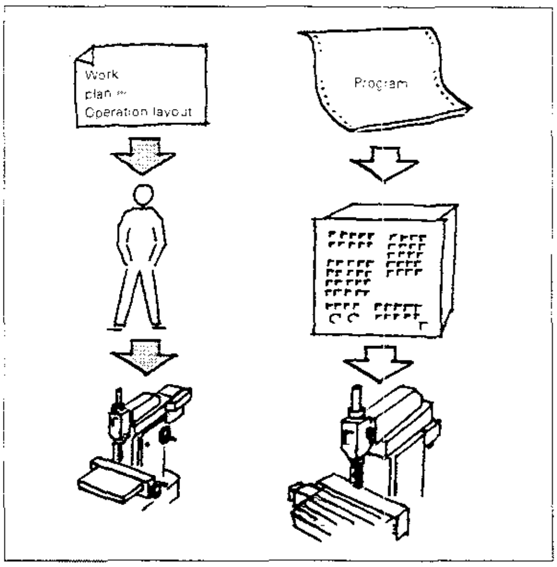


Programming
Introduction

Introduction

As in the case of conventional, manually operated machine tools, a work plan, called an "operation layout" is required for operations with a CNC machine tool. The operating sequence is the same in both cases.

While, in the case of the conventional machine, the individual steps are performed by the operator, the electronic control system of the CNC machine calculates the tool path, coordinates the feed motions of the machine slides and monitors spindle speed. The control system receives the data required to carry out these tasks from a **program** which has been entered in advance.



Program

The program is nothing more than a set of instructions, like the operation layout, compiled in a language that the control system can understand.

Programming

Programming is therefore the creation and entry of an operation layout in a language that can be understood by the control system.

Programming language

In the machining programm, each **NC program block** corresponds to one step in the operation layout. A block is made up of **individual commands.**

Examples

Programmed command	Meaning
Y –50.000	Move Y-axis slide to position –50.000.
F250	Move machine slide at feed rate of 250 mm/min.
TOOL CALL 1	Call compensation values for tool No. 1.

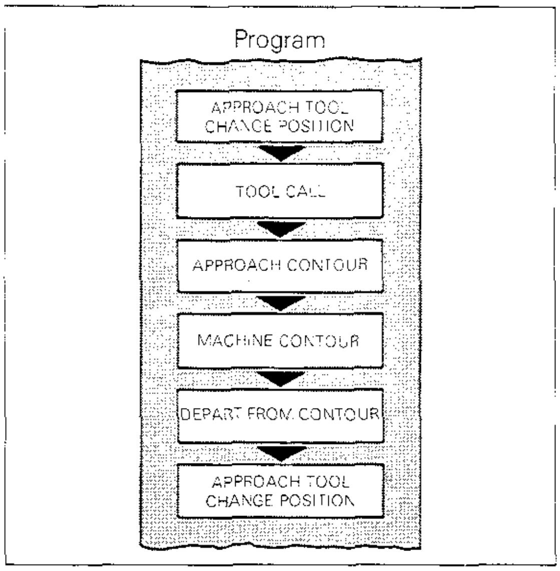
Programming
Program

Program structure

A program for producing a workpiece can be divided into the following **sections:**

- Approach tool change position.
- Insert tool.
- Approach workpiece contour.
- Machine workpiece contour.
- Depart from workpiece contour.
- Approach tool change position.

Each program section is composed of individual program blocks.



Block number

The control system automatically assigns a block number to each block. The **block number** identifies the program block within the machining program.

The block number is maintained when a block is erased; the subsequent block takes the place (and the number) of the erased one.

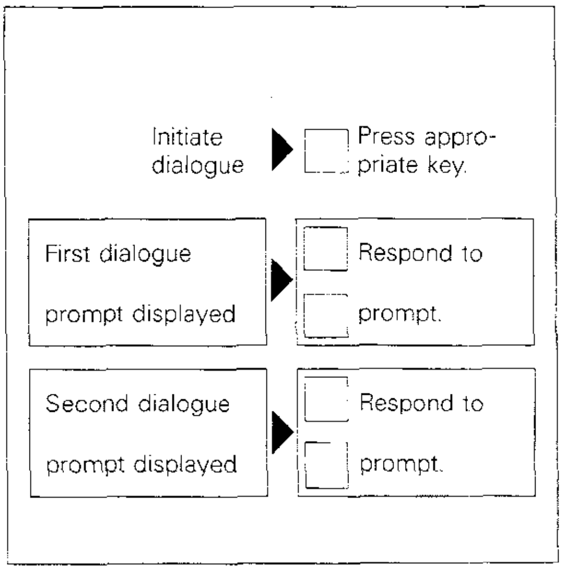
```
 7  L    Z-20.000
                       R0 F9999   M03
 8  L    X-12.000      Y+60.000
                       R0 F9999   M
 9  L    X+20.000      Y+60.000
                       RR F40     M
10  RND R+5.000
                        F20
11  L    X+50.000      Y+20.000
                       RR F40     M
12  CC   X-10.000      Y+80.000
13  C    X+70.000      Y+51.715
             DR+       RR F40     M
14  CC   X+150.000     Y+80.000
15  C    X+90.000      Y+20.000
             DR+       RR F40     M
16  L    X+120.000     Y+20.000
                       RR F40     M
```

Dialogue prompting

Programming is dialogue-prompted, meaning that the control system asks for the required information in plain language during program entry. The appropriate dialogue sequence for each program block is initiated via the dialogue-initiation key, e. g. TOOL DEF (the control system prompts the operator for the tool number, then for the tool length etc.).

Errors made while entering a program are also displayed in plain language. Incorrect entries can be corrected immediately, during program entry.

Programs are entered in "PROGRAMMING AND EDITING" mode ⇳.



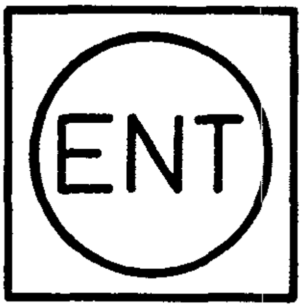
Programming
Responding to dialogue prompts

Responding to dialogue prompts

Every dialogue prompt requires a response. The response is displayed in the highlighted field on the screen. Following the response to the dialogue prompt, the entry is transferred to the program by pressing the ENT key.
The control system then issues the next dialogue prompt.

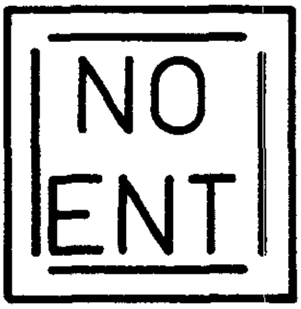
"ENT" is an abbreviation for "ENTER"

Do not press ENT when programming an axis without a numerical value (e. g. for mirror-imaging).



Skipping dialogue prompts

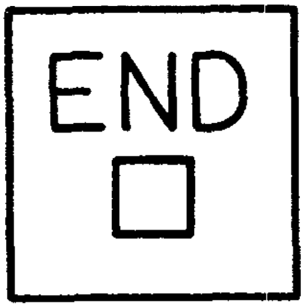
Certain entries do not change from block to block, e. g. feed rate or spindle speed. The corresponding dialogue prompt does not require a response in this case and may be "skipped" by pressing NO ENT. Entries already displayed in the highlighted field will be deleted and the next prompt will appear on screen. The values programmed previously at the corresponding address will be valid when the program is run.



Terminating a block prematurely

By pressing the END □ key, the programming of positioning blocks, tool calls or the cycles "datum shift" and "mirror image" can be terminated prematurely. Following the last prompt, the END □ key can be used much in the same way as the ENT key to transfer data, or immediately following the next prompt, in the same way as NO ENT.
The values programmed previously at the corresponding address will be valid when the program is run.

□ is the symbol for a program block.



Programming
Entering numerical values

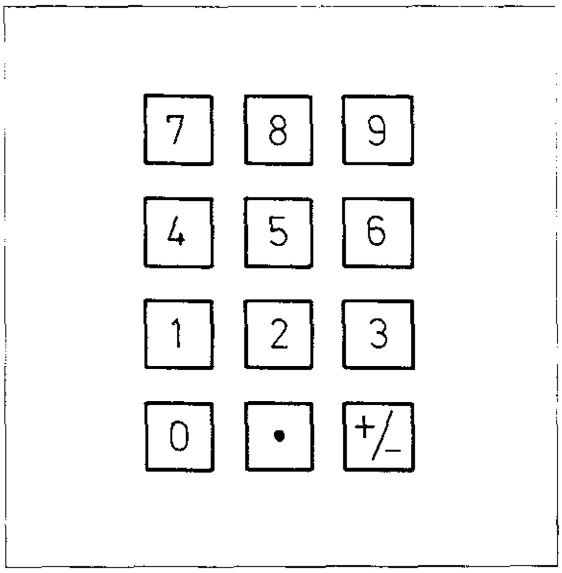
Entering numerical values

Numerical values are entered from the numeric keypad, which also features decimal point and prefix sign keys. Leading zeros in front of the decimal point may be omitted (the decimal point may be shown on the screen as a decimal comma).

Prefix signs may be entered before, during and after numerical entries.

Incorrectly entered numbers can be deleted by pressing the CE key before transferring them, and then re-entered correctly.

A zero is displayed in the highlighted field when the CE key is pressed.
Press the NO ENT key if you do not wish to enter data.



Notes:

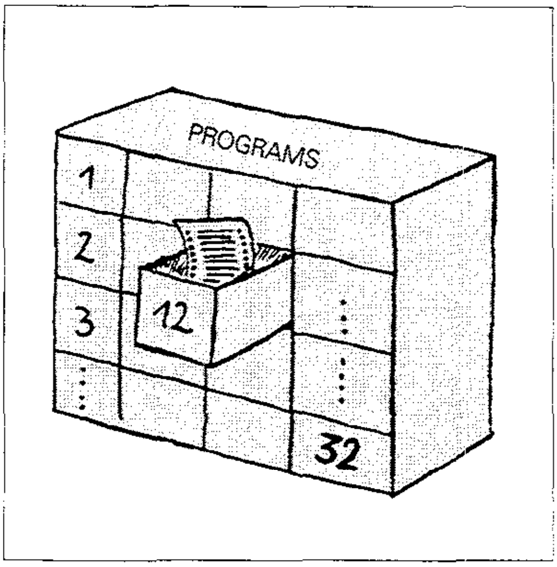
Program management

Entering a new program

The control system can save and store up to 32 programs with a total of 3.100 program blocks. A machining program can contain up to 999 blocks.
To distinguish the various programs, each machining program must be identified by a **program numer.**

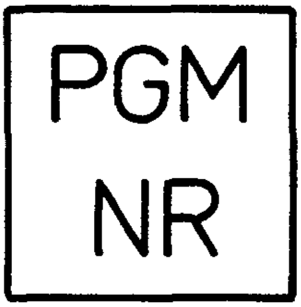
Erase/edit protection

Programs can be protected from direct access (e.g. erasure or editing).



Directory

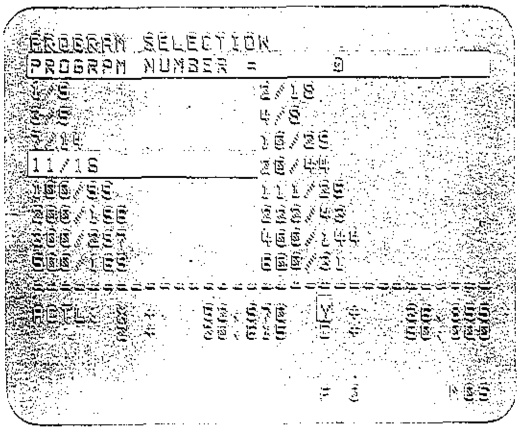
The dialogue for entering or calling up a program number is initiated by pressing PGM NR.
A table, or **directory**, showing the programs stored in the TNC's memory is displayed on the screen.
The length of the program is indicated following the program number. In HEIDENHAIN plain-language format, this display shows the number of program blocks; in ISO format, the number of characters (bytes) is displayed.
The directory can be exited with NC ENT or DEL □.



Calling an existing program

Programs that have already been entered are called via the program number. There are two ways of doing this:

- The programs stored in the control system are listed on the screen, by their program numbers. The most recently entered or called number is highlighted. The highlighted field, also called a cursor, can be moved around in the directory to the desired program number by means of the editing keys ↓ ↑ ← →. The program is called by pressing ENT.
- A program can also be called by typing the program number and pressing the ENT key.



```
PROGRAM SELECTION
PROGRAM NUMBER =        0
1/6                     2/16
3/6                     4/6
5/24                    10/26
11/16                   20/44
100/55                  111/26
200/156                 222/40
300/257                 400/144
500/105                 600/31
```

Program management

Entering a new program number

Operating mode ________ ⇨

Dialogue initiation ________ PGM NR

PROGRAM SELECTION

PROGRAM NUMBER =

▶ □ Enter program number (max. 8 digits).

▼ ENT Press ENT.

▼

MM = ENT / INCH = NO ENT

▶ ENT for **dimensions in mm**

or

▶ NO ENT for **dimensions in inches**

Sample display

```
0 BEGIN PGM 12345678    MM

1 END PGM 12345678      MM
```

The program number is 12345678; dimensions are in millimetres.
When programming, the program is inserted between the BEGIN-block and the END-block.

Selecting an existing program number

Operating mode ________ ⇨ or → or ⇉ or →

Dialogue initiation ________ PGM NR

PROGRAM SELECTION

PROGRAM NUMBER =

Select the program number using the highlighted cursor.

▶ ↓ ↑ ← → Place cursor over desired number.

▼ ENT Press ENT.

Or enter the program number.

▶ □ Enter number.

▼ ENT Press ENT.

Sample display

```
0 BEGIN PGM 8324        MM

1 L...
```

The beginning of the selected program is displayed on screen.

Program management
Edit-protected programs

Erase/edit protection

After a program is compiled, it can be protected against erasure and editing. Erase/edit-protected programs are identified at the beginning and end by a "P".

A protected program cannot be erased or changed unless the erase/edit protection is removed. This is done by selecting the program and entering the code number 86 357.

Program management
Edit-protected programs

Entering erase/edit protection

Operating mode ______ ◇

Dialogue initiation ______ PGM NR

PROGRAM SELECTION

PROGRAM NUMBER = ▶ □ Enter number of program to be protected.

▼

ENT Press ENT.

▼

0 BEGIN PGM 22 MM ▶ → Press key until dialogue prompt PGM protection is displayed.

▼

PGM PROTECTION

0 BEGIN PGM 22 MM ■ ▶ ENT

▼

→ Erase/edit protection is programmed.

Sample display

0 BEGIN PGM 22 MM P

"P" appears at end of line to identify erase/edit protection.

Notes:

Program management
Edit-protected programs

Cancelling erase/edit protection

Operating mode ______ ⇨

Dialogue initiation ______ PGM NR

PROGRAM SELECTION		
PROGRAM NUMBER =	▶ □	Enter number of program from which protection is to be cancelled.
	ENT	Press ENT.

▼

0 BEGIN PGM 22 MM **P**	▶ MOD	Select supplementary mode.

▼

VACANT BLOCKS 2951	▶ ↑	Select MOD function "Code number".

▼

CODE NUMBER =	▶ □	Enter code number **86 357**.
	ENT	Erase/edit protection is cancelled.

Sample display

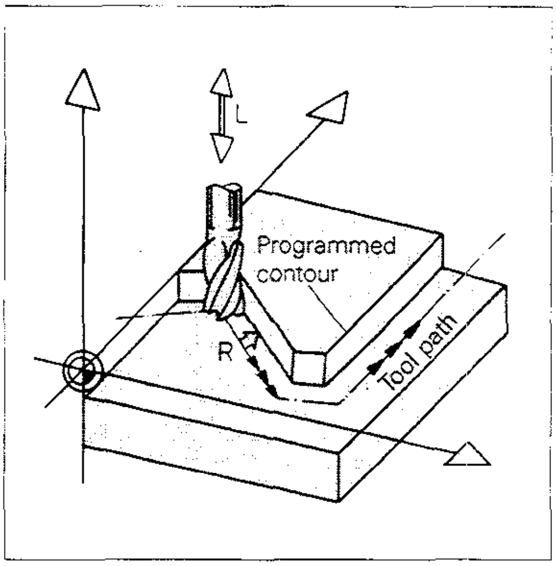
0 BEGIN PGM 22 MM

The "P" identifying the erase/edit protection disappears from display.

Programming the workpiece contour

Tool definition TOOL DEF

To enable the control system to calculate the tool path from the programmed workpiece co^ntour, tool lenght and radius must be specified. These data are programmed in the TOOL DEFINITION feature.



Tool number

The compensation (or offset) values always refer to a certain tool, which is identified by a number. The possible entry values for the tool number depend on how the machine is equipped:

with automatic tool changer: 1 – 99
(see "central tool memory")

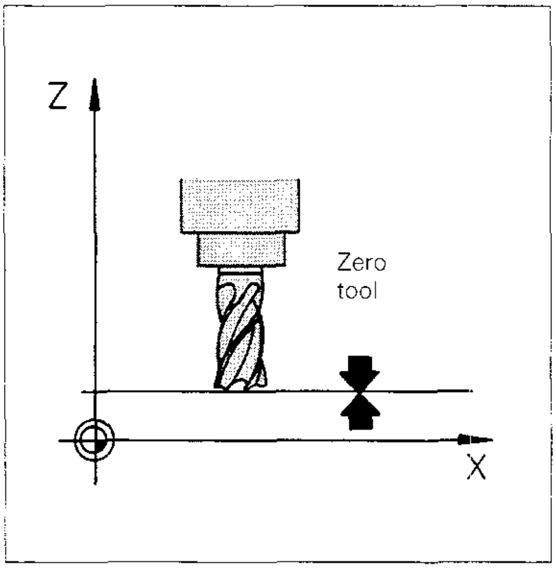
without automatic tool changer: 1 – 254.

Tool length

The **offset value** for tool length can be determined on the machine or on a tool presetter.

If the length compensation value is determined on the machine, the workpiece datum ⊕ should be defined first. The tool used to set the datum has a compensation value of "0" and is called the **zero tool.**

The **differences in length** of the remaining clamped tools, relative to the zero tool, are programmed as **tool length compensations.**

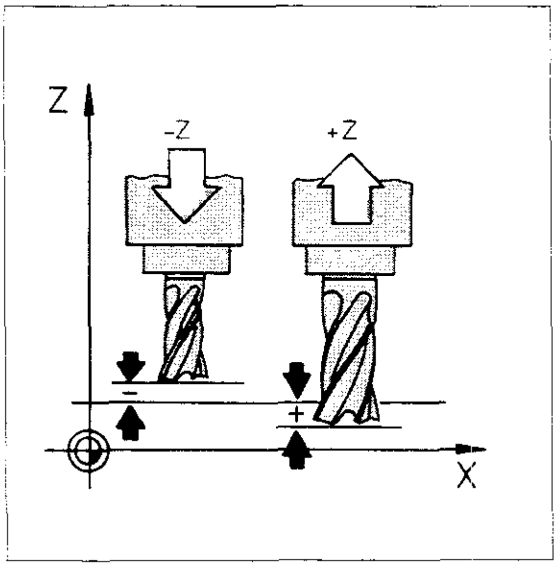


Prefix signs

If a tool is shorter than the zero tool, the difference is entered as a negative tool length compensation.

If a tool is longer than the zero tool, the difference is entered as a positive tool length compensation.

If a **tool presetter** is used, all tool lengths are already known. The compensation values are entered from a list, together with the correct prefixes.

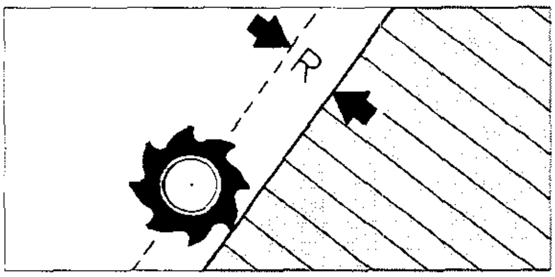


If the tool length is determined on the machine, the difference in length can be entered and transferred to memory by pressing ⌖ .

Programming tool compensation

Tool radius

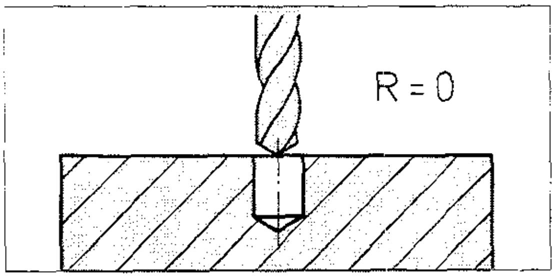
The tool radius is always entered as a positive value, except in the case of radius compensation for playback programming.



When using drilling and boring tools, the value "0" can be entered for the tool radius.

Possible input range: ± 30000.000.

A tool radius must be programmed if a machining program is to be checked with the aid of the TNC 155 graphics option.

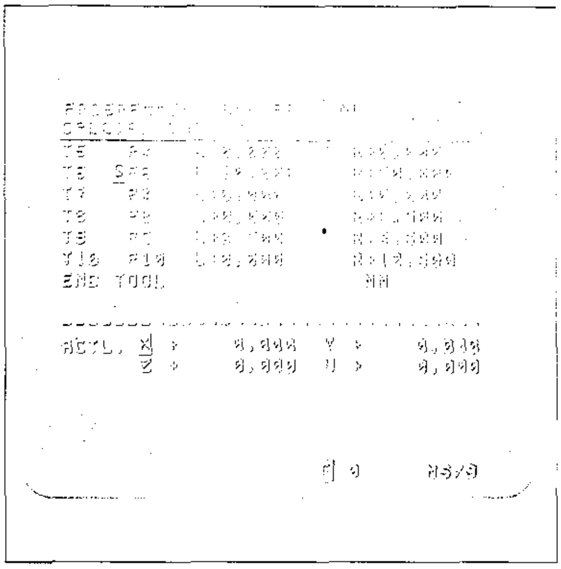


Programming tool compensation
Central tool memory

Central tool memory

In the TNC 151 B/Q and TNC 155 B/Q control systems, a central tool memory can be activated via machine parameters.

The central tool memory is selected via program number "0" and modified, printed out and loaded in ⇨ "Programming and editing" mode. Data for up to 99 tools can be stored. The tool number, length, radius and location of each tool is entered.



```
[illegible]
T5    P4    [illegible]        [illegible]
T6    S[illegible]  [illegible]        [illegible]
T7    P3    [illegible]        [illegible]
T8    P[illegible]  [illegible]        [illegible]
T9    P[illegible]  [illegible]        [illegible]
T10   P10   [illegible]        [illegible]
END TOOL                       MM

ACTL. X +      0,000   Y +      0,000
      Z +      0,000   [illegible] +      0,000

                       [illegible]       [illegible]
```

Tool changer with random-select feature

When using a tool changer with random select (variable tool location coding), the control system handles tool location management. The random select feature works like this: While one tool is being used for machining, the control system preselects the next tool to be used and exchanges the two tools at the programmed tool change. The control system records which tool number is stored at which location. The preselected tool is programmed via TOOL DEF. (Length and radius can only be entered in program 0).
Tools which due to their size require three locations may be defined as "special tools". A special tool is always deposited at the same defined location. A special tool is programmed by placing the cursor on the dialogue prompt
SPECIAL TOOL?
and pressing ENT.
For special tools, the previous and subsequent place numbers should be cancelled for safety reasons by setting the cursor and pressing the NO ENT key.
Beginning with software version 03:
A cancelled place number is replaced by an asterisk. "S" for special tool and "P" for place number is only displayed if this function was selected via machine parameter (entry value 3 in machine parameter G1).

When employing special tools, P0 (spindle) or another place in the magazine must be free.

Transfer blockwise

In "Transfer blockwise" mode, compensation values can be called up from the central tool memory.

Programming tool compensation
Tool definition

Entering a tool compensation

Operating mode ______________________

Dialogue initiation ______________________ TOOL DEF

TOOL NUMBER ?	▶ □	Enter tool number.
	ENT	Press ENT.

The tool number "0" should not be programmed under TOOL DEF. This number is allocated internally (see "TOOL CALL 0"). Tool length and radius can also be entered in playback mode (see "Tool compensation for playback").

TOOL LENGTH L ?		
If tool length is known:	▶ □	Enter compensation value or difference in length from zero tool.
	+/−	with correct prefix.
	ENT	Press ENT.
If tool length is determined on machine:	▶ □	Transfer difference in length from zero tool.
	ENT	Press ENT.

TOOL RADIUS R ?	▶ +	Enter tool radius.
	ENT	Press ENT.

Sample display

15 TOOL DEF 28 L + 15.780
R + 20.000

Tool No. 28 has the compensation value 15.780 for length and 20.000 for the radius.

Programming tool compensation
Tool call

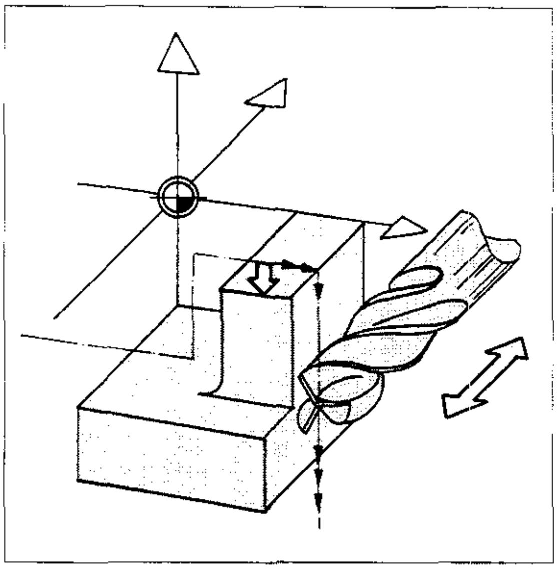
TOOL CALL

TOOL CALL is used to access a new tool and the corresponding compensation values for length and radius.

In addition to the **tool number,** the control system must also know the spindle axis, in order to perform length compensation on the correct axis, or radius compensation in the proper plane.

The **spindle speed** is entered immediately following the spindle axis. If the specified speed is outside the range permitted for the machine, the error message = WRONG RPM = is displayed.

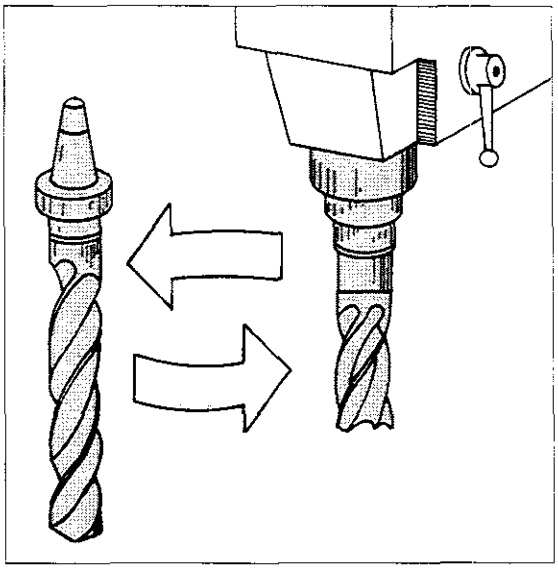
A TOOL-CALL block ends the linear and radius compensation.
Beginning with software version 3:
If during a TOOL-CALL block only the spindle speed is changed, then the TNC continues to execute the linear and radius compensation.



Tool change

Tool change occurs at a predefined **tool change position.** Thus the control system must move the tool to the **uncompensated nominal values** for the tool change positions. To do this, the compensation data for the tool currently in use must be cancelled.

This is done via the **TOOL CALL 0** function:
The moves to the desired uncompensated nominal position are programmed in the next block.
The tool change position can also be approached with M91, M92 (see "Auxiliary functions M) or via PLC positioning.
Contact your machine manufacturer or supplier for information.

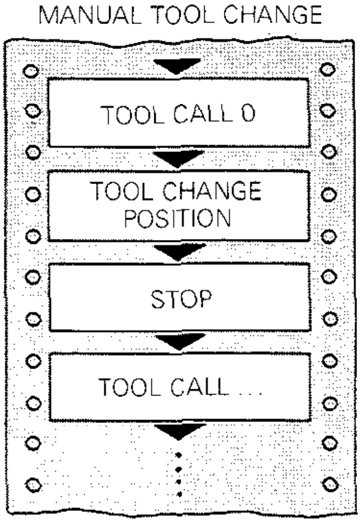


Program structure

Because the program run must be interrupted for a **manual tool change,** a program STOP command must be entered before TOOL CALL. The program run is then interrupted until the external start button is pressed.

The program STOP can be omitted only if a tool call is programmed merely for the purpose of changing slewing speed.

No programm STOP is required for an **automatic tool change.** The program run is resumed when the tool change is completed.



Programming tool compensation
Tool call/Program STOP

Entering a tool call command

Operating mode ____________ ⇨

Dialogue initiation ____________ TOOL CALL

TOOL NUMBER ?	▶ □	Enter tool number.
	▼ ENT	Press ENT.

SPINDLE AXIS PARALLEL X/Y/Z ?	▶ Z	Enter spindle axis, e.g. Z.
		Spindle axis is X/Y/Z or IV axis if IV axis is designated as U/V or W.

SPINDLE SPEED S IN RPM ?	▶ □	Enter spindle speed (see chart next page).
	▼ ENT	Press ENT.

Sample display

```
TOOL CALL 5     Z

          S 125.000
```

Tool No. 5 is called. The spindle axis operates in the direction of the Z-axis. Spindle speed is 125 rpm.

Entering a programmed STOP

Operating mode ____________ ⇨

Dialogue initiation ____________ STOP

AUXILIARY FUNCTION M ?

If auxiliary function is desired:	▶ □	Enter auxiliary function.
	▼ ENT	Press ENT.
No auxiliary function desired:	▶ NO ENT	Press NO ENT.

Sample display

```
18 STOP

            M
```

Program run interrupted in block 18.

No auxiliary function.

Tool call
Spindle speeds

Programmable spindle speeds (for coded output)

S in rpm	S in rpm	S in rpm	S in rpm	S in rpm
0	1	10	100	1000
0.112	1.12	11.2	112	1120
0.125	1.25	12.5	125	1250
0.14	1.4	14	140	1400
0.16	1.6	16	160	1600
0.18	1.8	18	180	1800
0.2	2	20	200	2000
0.224	2.24	22.4	224	2240
0.25	2.5	25	250	2500
0.28	2.8	28	280	2800
0.315	3.15	31.5	315	3150
0.355	3.55	35.5	355	3550
0.4	4	40	400	4000
0.45	4.5	45	450	4500
0.5	5	50	500	5000
0.56	5.6	56	560	5600
0.63	6.3	63	630	6300
0.71	7.1	71	710	7100
0.8	8	80	800	8000
0.9	9	90	900	9000

For coded output, spindle speeds must be within the range of standard values. If required, the control system will round off to the next higher standard value.

Programmable spindle speeds (for analogue output)

Programmed spindle speeds need not correspond to the values indicated in the table. Any desired spindle speed can be programmed, provided that it is not below the minimum speed and does not exceed the maximum speed of 30.000 rpm.
With the "Spindle override" potentiometer, the programmed speed can be increased or decreased by the set %-value.

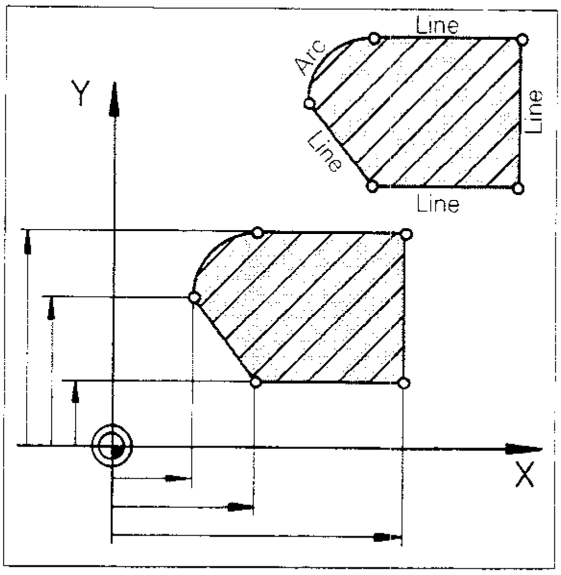
Contact your machine tool manufacturer or supplier to determine whether your machine operates with coded or analogue spindle speed output.

Beginning with software version 03:
With analog output of the spindle slewing speed the maximal spindle slewing speed is 99.999.999 rpm.

Programming workpiece contours
Contour

The workpiece contour

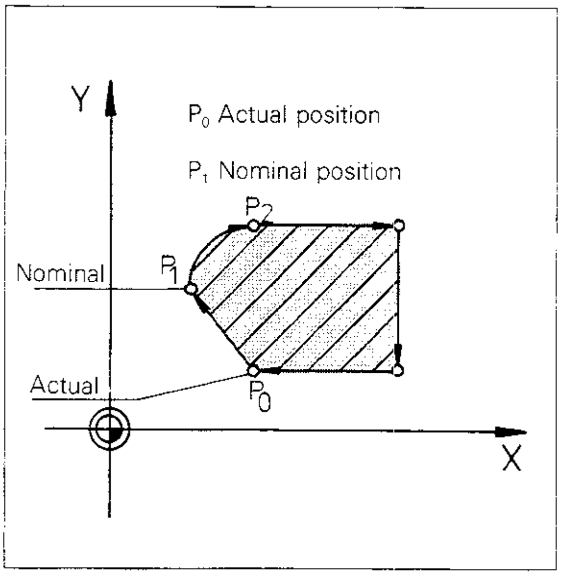
Workpiece contours programmed with the TNC 151/TNC 155 consist of the contour elements **straight lines** and **arcs.**



Generating a workpiece contour

To generate a contour, the control system has to know the type and location of the individual contour elements. Because the next machining step is defined in each program block, it is sufficient to

- enter the **coordinates** of the next target position and
- specify **what** type of **path** (straight line, arc or spiral) the tool will follow to the target point.



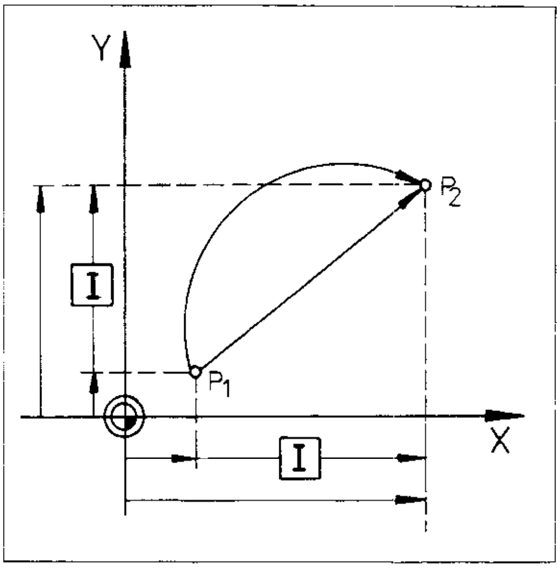
Programming coordinates

The **path** to a given target point must be specified before the coordinates of the point can be programmed.

The path is programmed with one of the **contouring keys** (see next page). These keys also initiate the input dialogue at the same time.

Incremental/ absolute dimensions

To enter the coordinates of a point in **incremental dimensions,** first press the I key. The red signal lamp indicates that the entry is being I accepted as an incremental dimension.
The I key acts as a "latch" switch; pressing it again switches back to **absolute dimensions** and the red signal lamp goes out.



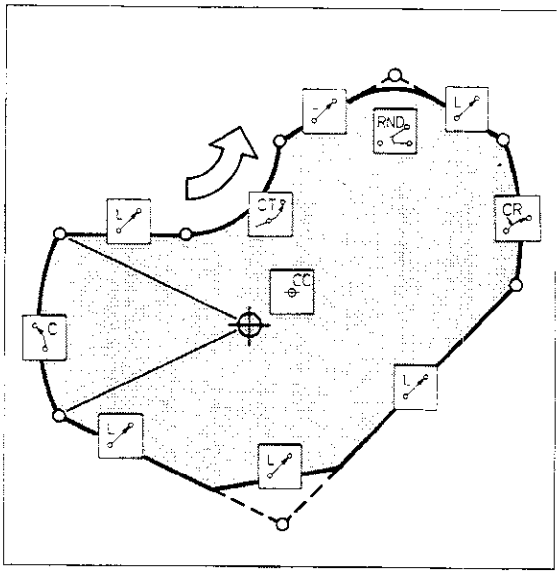
Programming workpiece contours
Contouring keys/Cartesian coordinates

Contouring keys

Linear interpolation L ("Line"):
The tool moves along a straight path. The end position of the straight line is programmed.

Circular interpolation C ("Circle"):
The tool moves along a circular path, or arc. The end position of the arc is programmed.

Circle centre CC (also pole for programming polar coordinates):
Used for programming the circle centre for circular interpolation or the pole for entering polar coordinates.



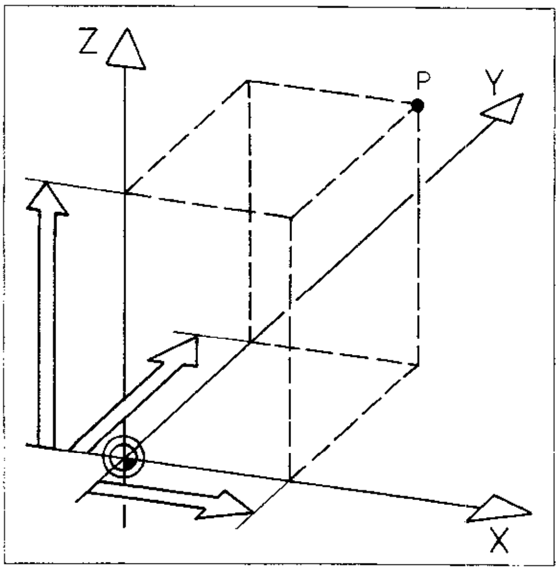
Rounding corners RND ("Rounding"):
The tool inserts an arc with tangential transitions between two contours. The radius of the arc and the contour elements of the corner to be rounded must be programmed.

Tangential arc CT ("Circle tangential"):
The tool inserts an arc with a tangential transition onto the preceding contour element. Only the end position of the arc need be programmed.

Circular interpolation CR ("Circle radius"):
The tool moves along a circular path. The circle radius and the end position of the arc are programmed.

Cartesian coordinates

A maximum of three axes can be programmed with linear interpolation and two axes with circular interpolation, using the corresponding numerical values. If the fourth axes is used for a rotary table axis (A-, B- or C-axis) the control system bases the entered value on "°" (degrees).



Programming workpiece contours
Cartesian coordinates

Entering Cartesian coordinates

Dialogue prompt:

COORDINATES ?	▶ X	Select axis, e.g. X.
	▼ I	Incremental – absolute?
	▼ □	Type numerical value.
	Y	Enter next coordinate, e.g. Y, and third coordinate, if required (max. 3 axes).
When **all coordinates** have been entered:	▶ ENT	Press ENT.

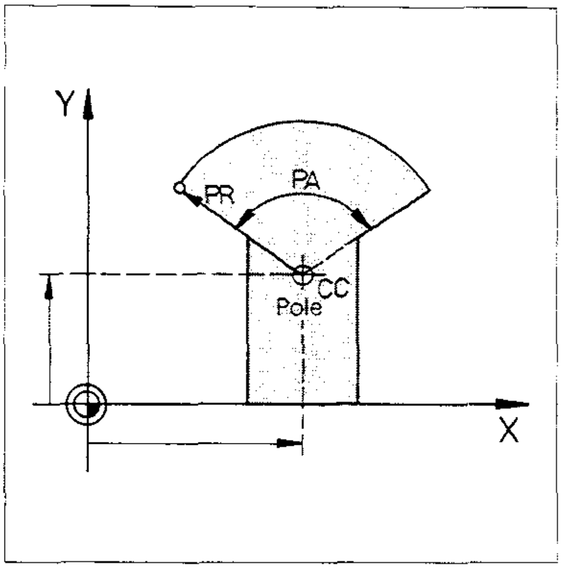
Programming workpiece contours
Polar coordinates/Pole

Pole CC

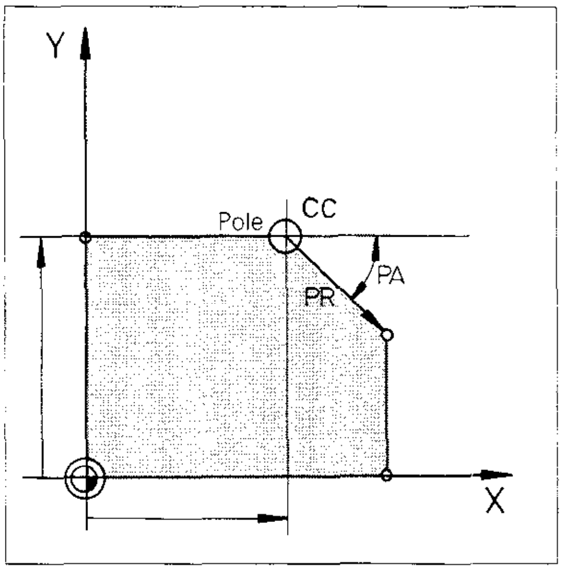
In the polar coordinate system, the reference point for the polar coordinates is the pole.
The pole must be defined **before entering the polar coordinates.**

There are three ways to define the pole:

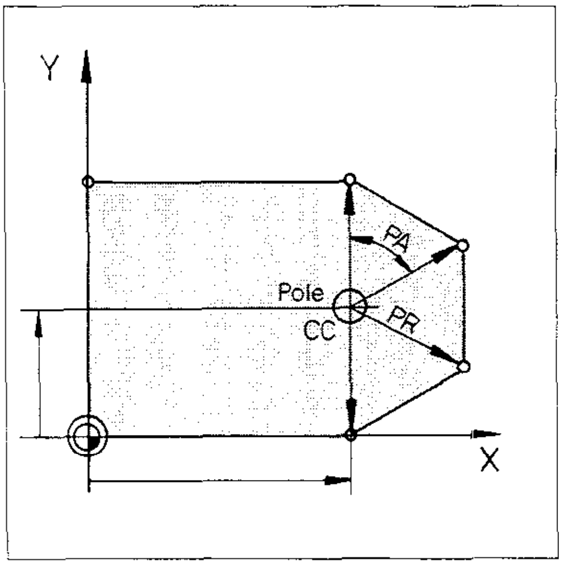
- The pole is redefined by Cartesian coordinates. A CC block is programmed with the coordinates of the machining plane.



- The last nominal position is used as the pole. A blank CC block is programmed. The most recently programmed coordinates of the program are then used to define the pole.



- The pole has the coordinates programmed in the last CC block. The CC block may be omitted.



The pole can be programmed only in Cartesian coordinates.
CC in absolute dimensions: The pole is based on the workpiece datum.
CC in incremental dimensions: The pole is based on the previous nominal position of the tool.

Programming workpiece contours
Polar coordinates/Pole

Entering the pole

Operating mode ______________________

Dialogue initiation ______________________ CC

COORDINATES ?	X	Select first axis, e.g. X.
	I	Incremental – absolute?
	□	Type numerical value.
If only one coordinate of the previous nominal position changes, the remaining coordinate need not be entered.	Y	Select second axis, e.g. Y.
	I	Incremental – absolute?
	□	Type numerical value.
	ENT	Press ENT.

To use the last nominal position as the pole, press NO ENT or END □. Both machining plane coordinates in the last positioning block must be defined.

Sample display 1

```
27 CC X + 10.000   IY + 45.000
```

The pole has the absolute X-coordinate 10.000 and the incremental Y-coordinate 45.000.

Sample display 2

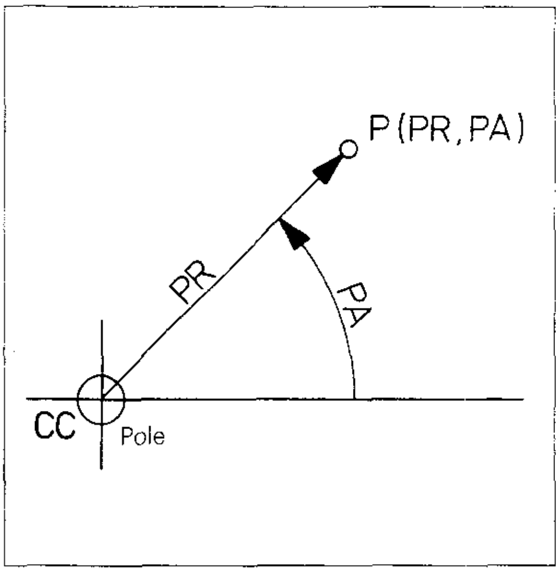
```
92 L   X + 20.500    Y + 33.000

                     R   F         M

93 CC
```

The pole in block 93 has the coordinates X 20.500 and Y 33.000.

Programming workpiece contours
Polar coordinates

Polar coordinates

If desired, points can also be defined by polar coordinates (polar coordinate radius PR, polar coordinate angle PA).
Polar coordinates are always based on a given **pole CC.**



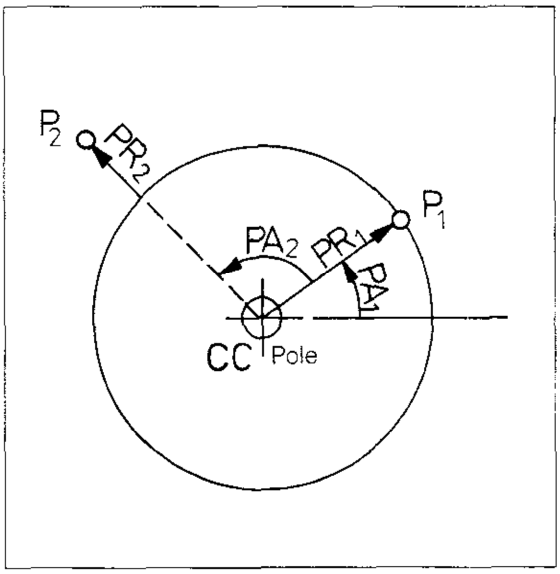
Incremental input

In the case of incremental data entry, the polar coordinate radius increases by the programmed value. An incremental polar coordinate angle PA is based on the side of the last angle entered.

Example: The polar coordinates of point P1 are PR1 (absolute) and PA1 (absolute).

The polar coordinates of point P2 are PR2 (incremental) and PA2 (incremental). Only the **change in radius** for PR2 and the **change in angle** for PA2 are entered as numerical values.

Thus point P2 has the absolute values
PR = (PR1 + PR2) and PA = (PA1 + PA2).



Programming workpiece contours
Polar coordinates

Initiating the dialog

The P key must be pressed **before** the L , C , CT contour keys are pressed.

Entering polar coordinates

Dialogue prompt:

POLAR COORDINATE RADIUS PR ?	▶ I	Incremental – absolute?
	▼ □	Enter polar coordinate radius PR to target point.
	▼ ENT	Press ENT.

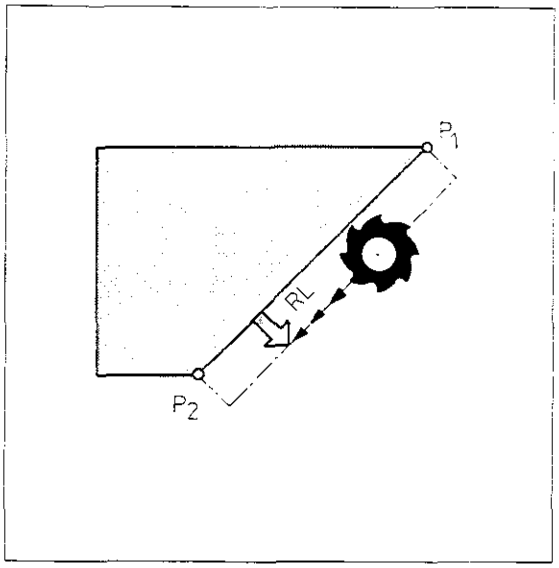
▼

POLAR COORDINATE ANGLE PA ?	▶ I	Incremental – absolute?
	▼ □	Enter angle PA to reference axis.
	▼ ENT	Press ENT.

Programming workpiece contours
Radius compensation – Tool path compensation

Tool radius compensation

For automatic compensation of tool length and radius – as entered in the TOOL DEF blocks – the control system has to know whether the tool will be located to the left or right of the contour, or directly on the contour, based on the direction of feed.

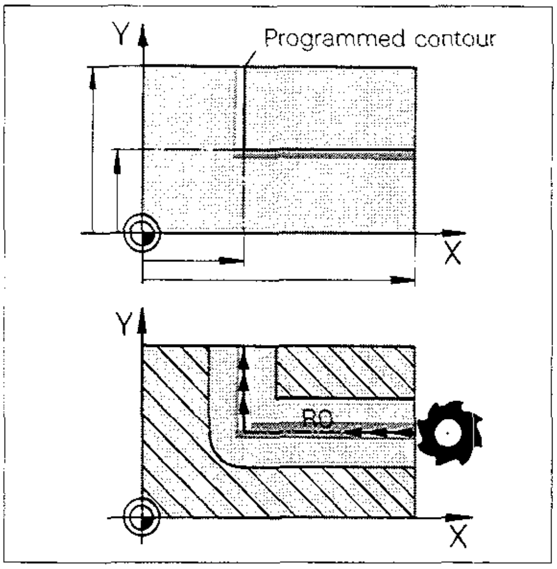


Tool path compensation

If the tool moves with path compensation, i.e. if the cutter axis moves with the programmed tool radius taken into account, it follows a path running parallel to and at a distance from the contour equal to the tool radius (equidistant).

Programming tool radius compensation

The radius compensation is programmed via the two-way switches RL and RR. The red signal lamp beneath each key indicates how the tool radius is offset by the control system.



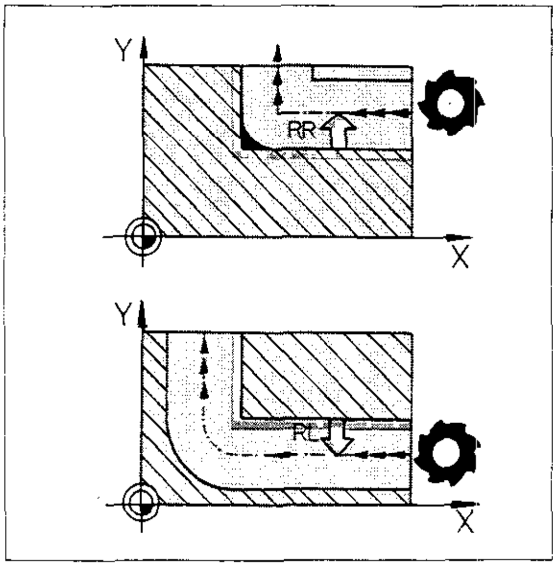
R0

If the tool is to move along the programmed contour, no radius compensation should be active in the positioning block.

The red signal lamps under the RR and RL keys must be off. The R0 entry is made with the ENT key.

RR

If the tool is to move to the **right** of the programmed **contour,** offset at a distance equal to the radius, press the RR key. The red signal lamp shows that the RR function is active. The RR entry is then made with the ENT key.



RL

If the tool is to move to the **left** of the programmed **contour,** offset at a distance equal to the radius, press the RL key. The red signal lamp shows that the RL function is active. The RL entry is then made with the ENT key.

Programming workpiece contours
Radius compensation

Programming the radius compensation

Dialog prompt:

TOOL RADIUS COMP.: RL/RR/NO COMP. ?

The tool should travel on the **left** of the programmed contour.	▶ RL	Select RL.
	▼ ENT	transfer to memory.
The tool should travel on the **right** of the programmed contour.	▶ RR	Select RR.
	▼ ENT	transfer to memory.
The tool should travel **on** the programmed contour.	▶ ENT	transfer to memory.
Make sur the red signal lamps below RR and RL are off.		
The radius compensation should be taken over from the previous block.	▶ NO ENT	Confirm **R**.

R0 and R have different meanings:

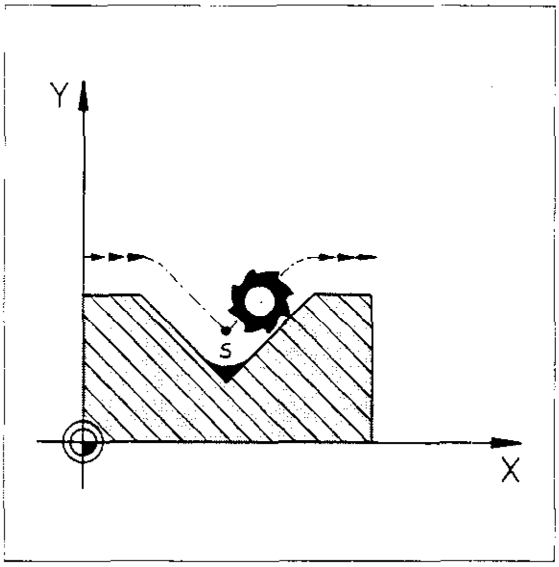
R0 The tool travels on the programmed contour.

R The radius compensation is taken over from the previous block.

Programming workpiece contours
Tool path compensation

Path compensation on internal corners

After radius compensation is activated, the control system automatically computes on **internal corners** the **intersection S** of the contour-parallel (equidistant) path of the cutter.
This prevents back-cutting on the contour on internal corners, and resulting damage to the workpiece.

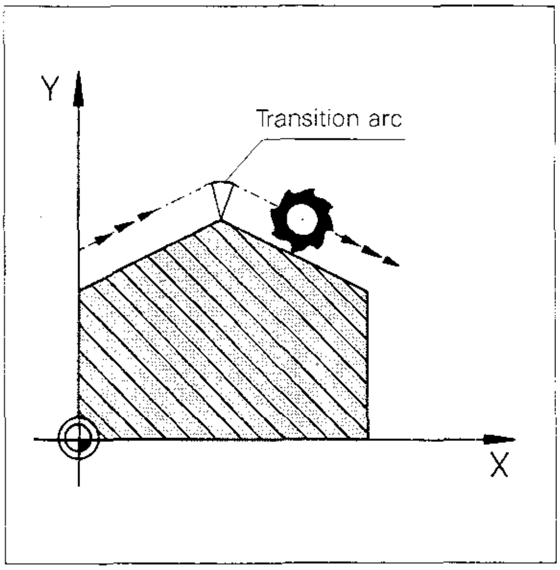


Path compensation on external corners

If radius compensation has been programmed, the control system inserts a **transition arc** (blend) on external corners, which allows the tool to "roll" around the corner point.

In most cases, this guides the tool around the corner at a constant tool path feed rate. If the programmed feed rate is too high for the transition arc, the tool path feed rate is reduced (resulting in a more precise contour). The limit value is permanently programmed in the control system.

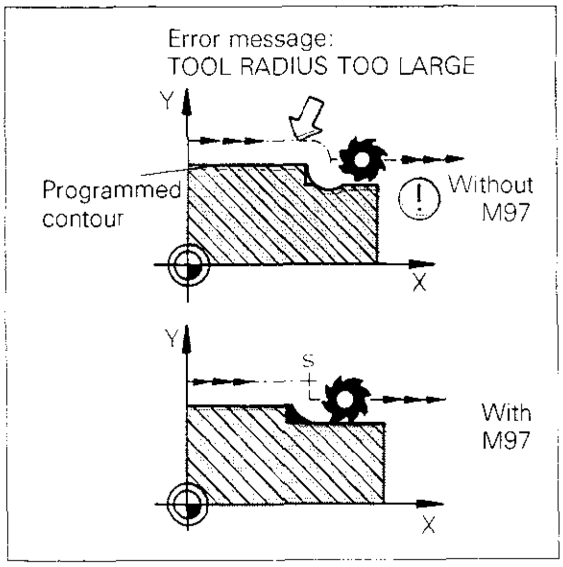
The automatic feed rate reduction can be disabled, f required, by programming the auxiliary function M90 (see "Feed rate").



Contour intersection compensation M97

If the tool radius is larger than the **contour shoulder,** the transition arc on external corners can cause damage to the contour. In this case the error message
= TOOL RADIUS TOO LARGE =
is displayed and the corresponding positioning block is not executed. Programming the auxiliary function **M97** prevents the insertion of a transition arc. The control system computes an additional **contour intersection S** and guides the tool over this point without damaging the contour.

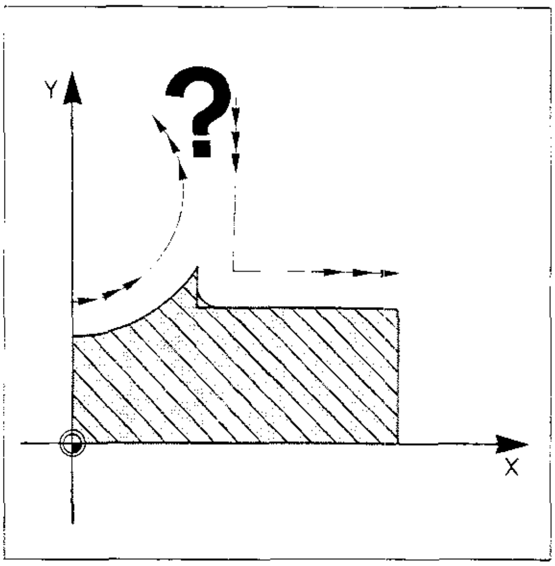
The contour intersection compensation M97 is a non-modal command. It must be programmed in the same block as the external corner point.



Programming workpiece contours
Tool path compensation

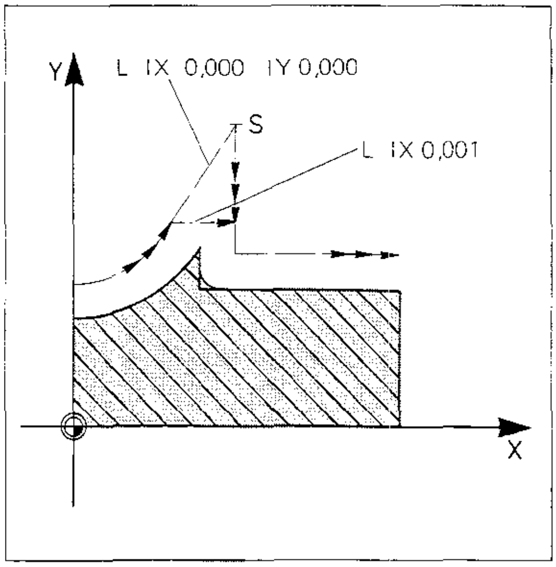
Special cases with M97

Under some circumstances e.g. the intersection of a circle with a straight line, the control system is unable to find a contour intersection with tool path compensation using M97. The error message
= TOOL RADIUS TOO LARGE =
is displayed when the program is run.



Remedy

An auxiliary positioning block is inserted in the program that extends the end point of the arc by the length "0". The control system then performs a linear interpolation, resulting in the calculation of the intersection S.



Example

```
 16 CC Circle centre
 17  C Arc end point
```

```
 18  L IX 0.000  IY 0.000
                 R F            M97
```

```
 19  L Straight line
```

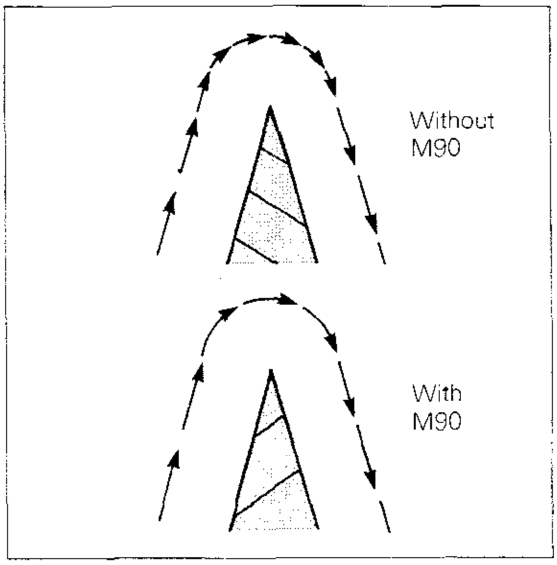
A straight element with length "0" was programmed in block 18
or:

```
 18  L IX 0.001
                 R F            M97
```

A straight element of length 0.001 was programmed in block 18.

Constant feed rate on corners M90

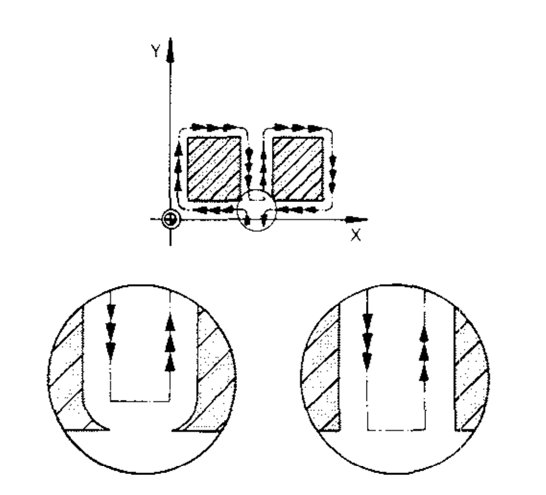
For external corners the TNC normally reduces the feed rate, for internal corners the tool stops.
The reduction of the feed rate on corners can be cancelled with the auxiliary function M90, which, however, can cause a slight distortion of the contour. Increased accelleration can also occur, i.e. the maximum acceleration defined in the machine parameters can be exceeded.
This auxiliary function depends on the stored machine parameters (operation with trailing error). Contact your machine manufacturer to determine whether your control system operates in this way.



Terminating tool path compensation

Tool path compensation can be ended with
- TOOL CALL
- STOP
- M98

The auxiliary function M98 in the positioning block for the last point on the contour causes the respective contour element to be completely machined. If an additional contour has been programmed, as shown in the example at the right, M98 will cause the tool to approach the first point on the contour with radius compensation and this contour will also be completely machined (see "Departure command").



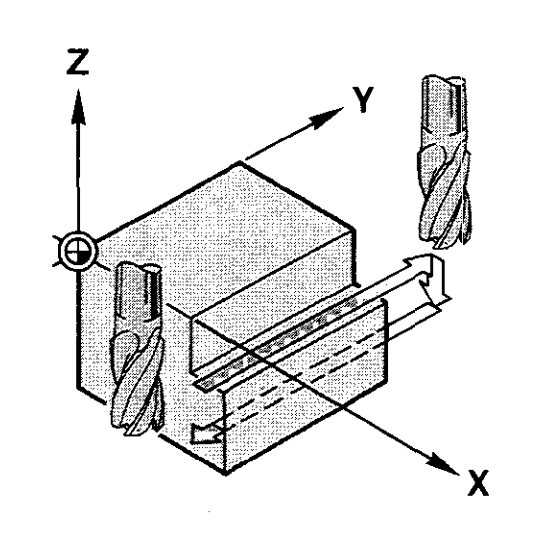
Without M98 With M98

Line milling with M98

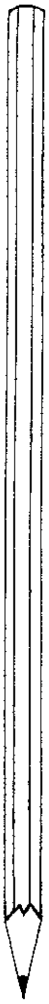
Another potential application for M98: line milling with downfeed on Z.

Example

```
L   X+20.000 Y-10.000
                        RR F15999       M
L   Z-10.000
                        R  F            M
L   Y+110.000
                        R  F20          M98
L   Z-20.000
                        R  F15999       M
L   Y+110.000
                        RL F20          M
L   Y-10.000
                        R  F            M98
```



Notes:



Programming workpiece contours
Feed rate F/Auxiliary function M

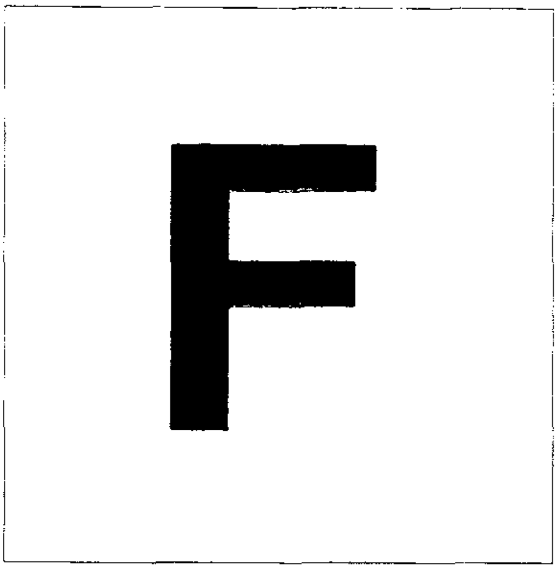
Feed rate

The **feed rate F,** i.e. the traversing speed of the tool along its path, is programmed in mm/min or 0.1 inch/min. If a rotary table is used (A-, B- or C-axis) the value is entered in degrees (°) per minute.

The **feed rate override,** located on the front panel of the control unit, can be used to vary the feed rate within a range of 0 to 150%.

The **maximum input values** (rapid traverse) for the feed rate are:
- 15.999 mm/min or
- 6.299/10 inch/min.

The maximum feed rate of the individual machine axes is determined by the machine manufacturer via the machine parameters.



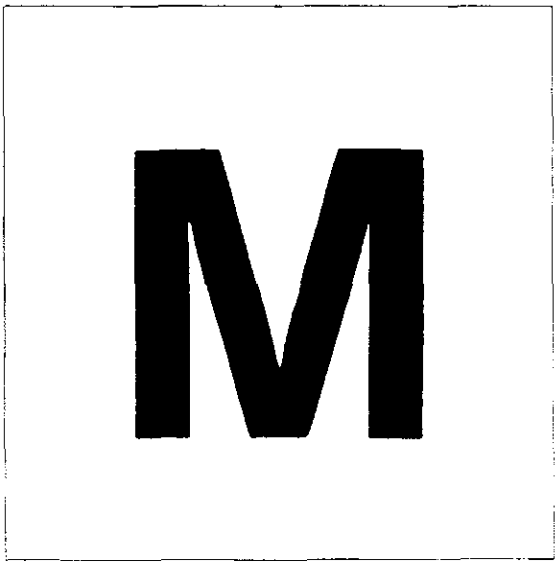
The current feed rate is indicated in the status display at the lower right of the screen. If this display is highlighted and the axes no longer move, the feed rate has not been enabled at the control unit interface. If this happens, contact your machine tool manufacturer or supplier.

Auxiliary function

You can program auxiliary (miscellaneous) functions that control special machine functions (e.g. "Spindle ON") and influence tool contouring characteristics. The auxiliary functions consist of the **address M** and a **code number.**

When programming these functions, it should be noted that certain M functions are active at the beginning of a block (e.g. M03: "Spindle ON – clockwise), while others (e.g. M05: "Spindle STOP") are active at the end of a block.

All available M functions are listed on the following pages.



Programming workpiece contours
Entering the feed rate
Entering an auxiliary function

Entering the feed rate

Dialogue prompt:

FEED RATE ? F=	▶ □	Type numerical value.
	▼ ENT	Press ENT.

Entering an auxiliary function

Dialogue prompt:

AUXILIARY FUNCTION M ?	▶ □	Type code.
	▼ ENT	Press ENT.

Auxiliary functions M

M functions affecting program run

M	Function	Active at block	
		beginning	end
M00	Stop program run Spindle STOP Coolant OFF		●
M02	Stop program run Spindle STOP Coolant OFF Return to block 1		●
M03	Spindle ON: clockwise	●	
M04	Spindle ON: counterclockwise	●	
M05	Spindle STOP		●
M06	Tool change Stop program run (if req'd., depends on specified machine parameters) Spindle STOP		●
M08	Coolant ON	●	
M09	Coolant OFF		●
M13	Spindle ON: clockwise Coolant ON	●	
M14	Spindle ON: counterclockwise Coolant ON	●	
M30	same as M02		●
M89	Variable auxiliary function	●	
	or		
M89	Cycle call, modal (depends on specified machine parameters)		●
M90	Constant tool path feed rate at corners (see "Tool path feed rate")	●	
M91	**within positioning block:** Reference point substituted for workpiece datum	●	
M92	**within positioning block:** Specified workpiece datum replaced by position defined in machine parameters by machine manufacturer, e.g. tool change position	●	
M93	M-function assignment reserved by HEIDENHAIN	●	
M94	Reduction of displayed value for rotary table axis to below 360°	●	
M95	Modified approach characteristics (see "Approach statement M95")		●
M96	Modified approach characteristics (see "Approach statement M96")		●
M97	Contour intersection compensation on external corners		●
M98	End of contour compensation		●
M99	Cycle call		●

Auxiliary functions M

Variable auxiliary functions

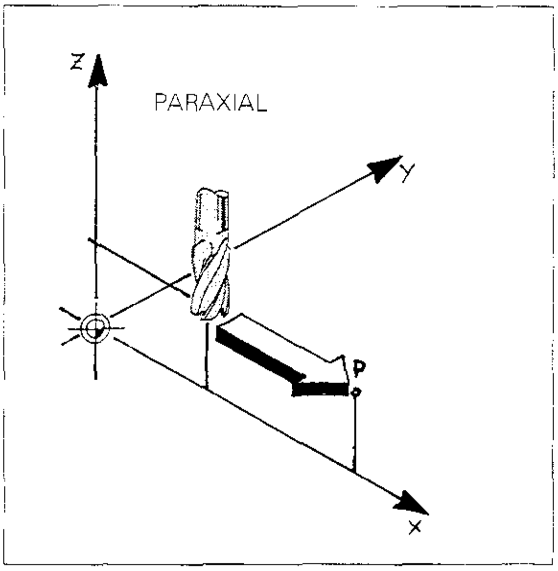
Variable auxiliary functions are defined by the machine manufacturer and explained in the machine Operating Manual.

M	Function	Active at	
		Block beginning	Block end
M01			●
M07		●	
M10			●
M11		●	
M12			●
M15		●	
M16		●	
M17		●	
M18		●	
M19			●
M20		●	
M21		●	
M22		●	
M23		●	
M24		●	
M25		●	
M26		●	
M27		●	
M28		●	
M29		●	
M31		●	
M32			●
M33			●
M34			●
M35			●
M36		●	
M37		●	
M38		●	
M39		●	
M40		●	
M41		●	
M42		●	
M43		●	
M44		●	
M45		●	
M46		●	
M47		●	
M48		●	
M49		●	
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		●	

Programming workpiece contours
Straight lines

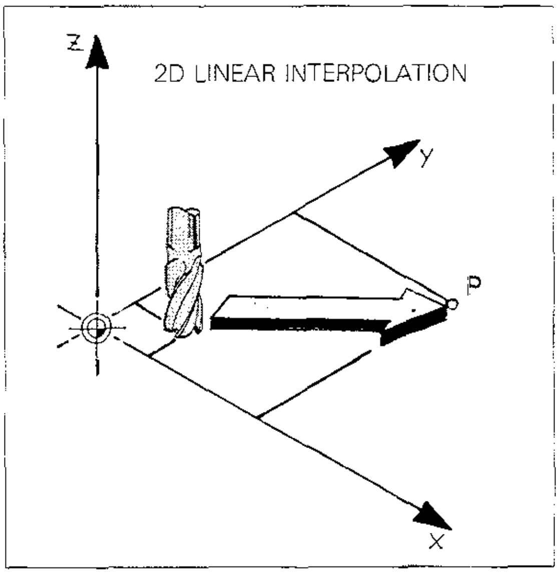
Paraxial movement

If the tool moves, relative to the workpiece, along a straight path, parallel to a **machine axis,** the movement is referred to as **paraxial** positioning or machining.



2D linear interpolation

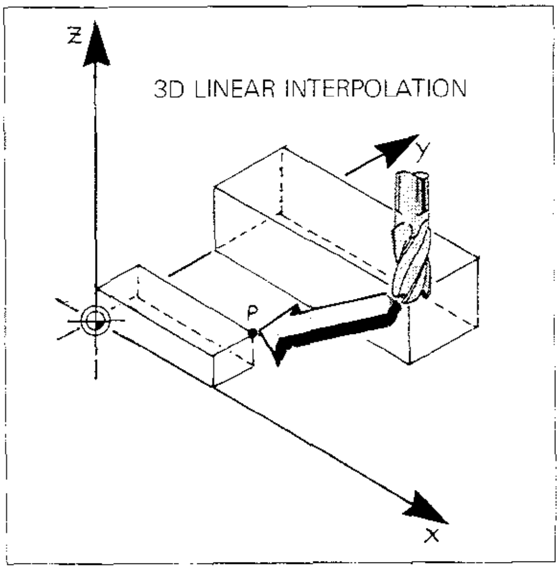
If the tool moves along a straight path in one of the **main planes** (XY, YZ, ZX), the movement is referred to as **2D linear interpolation.**



3D linear interpolation

If the tool moves relative to the workpiece along a straight path, with simultaneous movement of all **three machine axes,** the movement is referred to as **3D linear interpolation.**

Simultaneous movement of three machine axes along a straight path is not available on control system models TNC 151 F/TNC 155 F/ TNC 151 W/TNC 155 W.



Programming workpiece contours
Linear interpolation with a 4th axis

4th axis = linear axis

In the case of linear interpolation using the 4th axis as a linear axis, the axis, together with the corresponding coordinate data, must be programmed in each NC block. This also applies even in cases where the coordinate does not change from one block to the other. If no 4th axis is specified, the control system will traverse the main axes of the machining plane.

Example: linear interpolation with X and V, tool axis Z.

```
= CORRECT =

11  L  X+0.000      V+0.000

               RR F100          M

12  L  X+100.000    V+0.000

               R  F             M

13  L  X+150.000    V+70.000

               R  F             M
```

```
= INCORRECT =

11  L  X+0.000      V+0.000

               RR F100          M

12  L  X+100.000

               R  F             M

13  L  X+150.000    V+70.000

               R  F             M
```

4th axis = angular axis

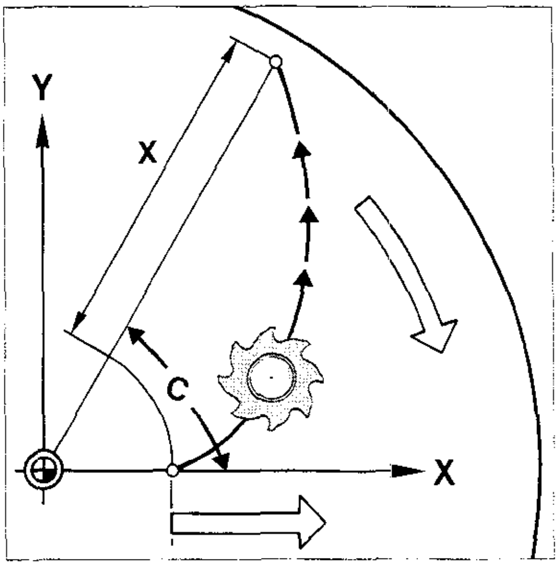
In the case of linear interpolation using one linear and one angular axis, the TNC interprets the programmed feed rate as the tool path feed rate. In this case, the feed rate is based on the relative speed between the workpiece and the tool. Thus the control system computes a feed rate value for the linear axis F (L) and a feed rate value for the angular axis F (W), for each point on the path:

$$F\,(L) = \frac{F \cdot \Delta L}{\sqrt{(\Delta L)^2 + (\Delta W)^2}}$$

$$F\,(W) = \frac{F \cdot \Delta W}{\sqrt{(\Delta L)^2 + (\Delta W)^2}}$$

Key:

F	= programmed feed rate
F (L)	= linear component of feed rate (machine slide)
F (W)	= angular component (rotary table)
Δ L	= distance traversed by linear axis
Δ W	= distance traversed by angular axis

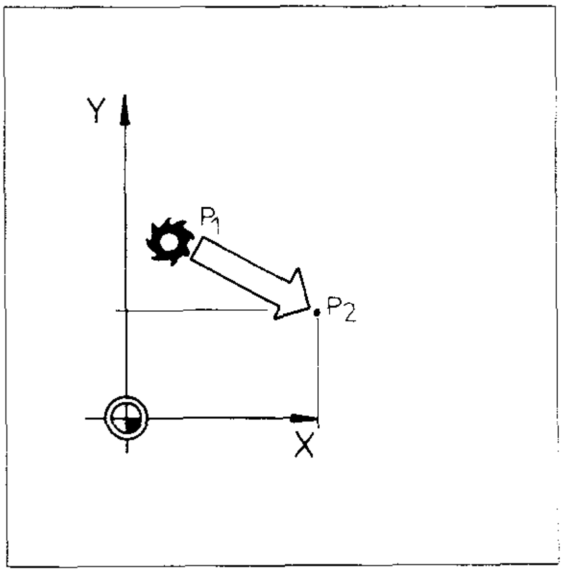


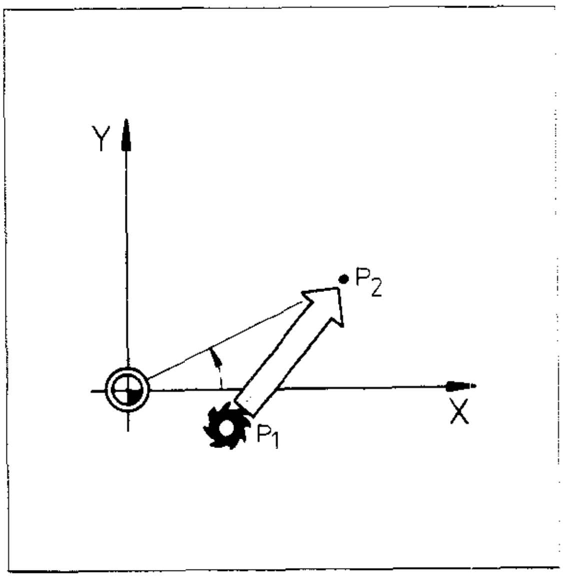
Programming workpiece contours
Straight lines

Straight line L

To move the tool along a straight path from the starting position P1 to the target position P2: Program the target position P2 (nominal position) of the straight line.

The nominal position P2 can be programmed in either Cartesian or polar coordinates.





Programming workpiece contours
Linear interpolation/Cartesian coordinates

Make sure the red signal lamp below the "P" key is off. Press "P" key to switch off if necessary.

Entering data in Cartesian coordinates

Operating mode ____________

Dialogue initiation ____________

COORDINATES ?	X	Select axis, e.g. X.
	I	Incremental – absolute?
	□	Type numerical value.
	Y	Enter next coordinate, e.g. Y and third coordinate if required (max. 3 axes).
When **all coordinates** of the target position have been entered:	ENT	Press ENT.

TOOL RADIUS COMP.: RL/RR/NO COMP. ?	R-L R+R	Enter radius compensation if required.
	ENT	Press ENT.

FEED RATE ? F =	□	Enter feed rate if required.
	ENT	Press ENT.

AUXILIARY FUNCTION M ?	□	Enter auxiliary function if required.
	ENT	Press ENT.

After coordinates have been entered, and if the remaining data are unchanged, positioning blocks can be shortened by pressing the END □ key.

Sample display

```
28 L X+20.000 IY+49.800
              RL F100      M13
```

The tool moves to position X 20.000 (absolute) and Y 49.800 (incremental), with a radius offset to the left of the programmed contour, at a feed rate of 100 mm/min.
Coolant flow starts at the beginning and the spindle rotates clockwise.

Notes:

Programming workpiece contours
Linear interpolation/Polar coordinates

Make sure the red signal lamp below the "P" key is on. Press "P" key to switch on if necessary.

Entering data in polar coordinates

Operating mode ______

Dialogue initiation ______ P

POLAR COORDINATES RADIUS PR ?

I — Incremental – absolute?

□ — Enter polar coordinate radius PR for end position of straight line.

ENT — Press ENT.

POLAR COORDINATE ANGLE PA ?

I — Incremental – absolute?

□ — Enter polar coordinate angle PA for end position of straight line.

ENT — Press ENT.

TOOL RADIUS COMP.: RL/RR/NO CAMP. ?

RL RR — Define radius compensation if required.

ENT — Press ENT.

FEED RATE ? F =

□ — Enter feed rate if required.

ENT — Press ENT.

AUXILIARY FUNCTION M ?

□ — Enter auxiliary function if required.

ENT — Press ENT.

After coordinates have been entered, and if the remaining data are unchanged, positioning blocks can be shortened by pressing the END □ key.

Sample display

```
39 LP PR+35.000 PA+45.000

            R  F              M
```

The tool moves along a straight path to a position 35.000 from the previously defined pole CC; the polar angle is 45° (absolute).
Radius compensation and feed rate are determined by the most recently programmed values.
No auxiliary function.

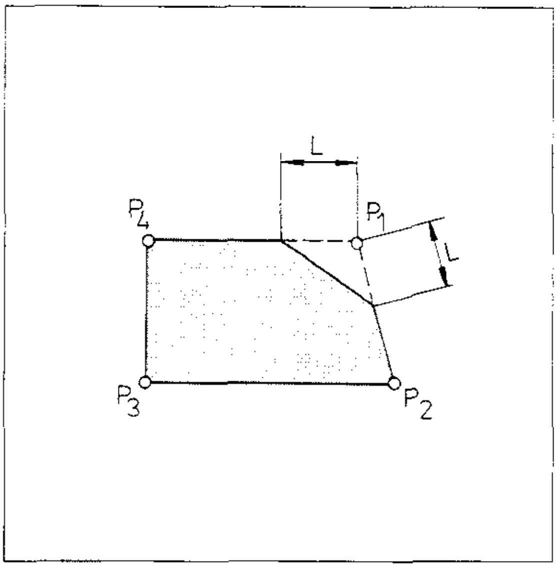
Programming workpiece contours
Chamfers

Chamfers

Contour corners produced by the intersection of two straight lines can be provided with chamfers. The angle between the two lines is variable.

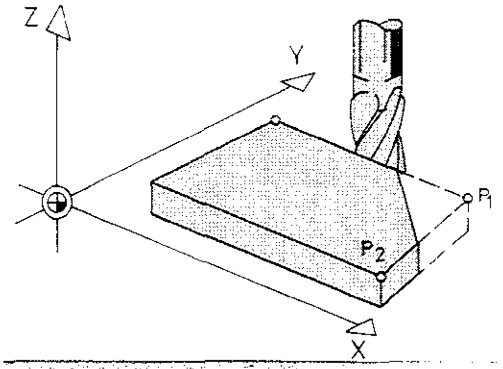
Entry

The chamfer is programmed with the key by specifying the chamfer length L.



Program

Chamfers can be inserted only in a main plane (XY, YZ, ZX), i.e. the positioning block preceding and following the "chamfer" block must contain the two coordinates of the machining plane. If the machining plane is not clearly defined (e.g. positioning block with X ... Y ... Z ...), the error message
= PLANE INCORRECTLY DEFINED =
is displayed.



```
26 straight line to P1 (X, Y)
27 L 10.000
28 straight line to P2 (X, Y)
```

Programming workpiece contours
Chamfers

Entry

Operating mode ______ ⎆

Dialogue initiation ______ L

COORDINATES ?	▶ □	Enter chamfer length L.
	▼ ENT	Press ENT.

Sample display

```
88 L 7.500
```

A chamfer with side length L = 7.500 is inserted between the contour elements programmed in the preceding and subsequent blocks.

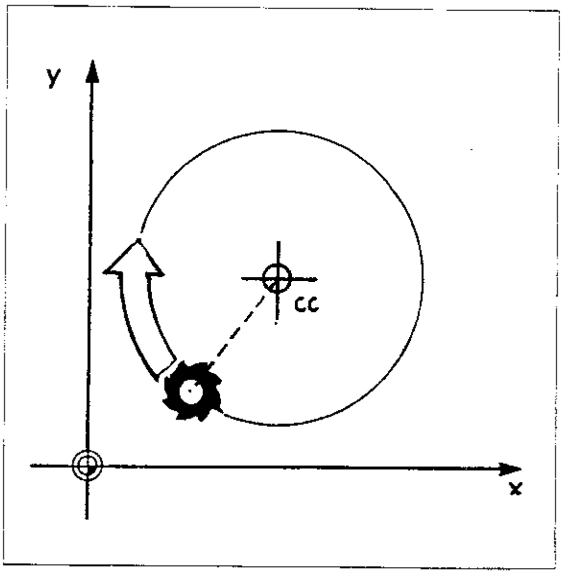
Programming workpiece contours
Circular interpolation/Circular path C

Circular interpolation

The control system controls two axes simultaneously in such a way that the tool, relative to the workpiece, follows the path of a circle or an arc.

With the TNC 151/TNC 155, an arc can be programmed in four ways:

- via the circle centre and end position using the CC and C keys,
- via the circle radius and end position using the CR key,
- for arcs with tangential transitions at both ends, via the circle radius only, using the RND key,
- for arcs joined tangentially to the preceding contour via the end position only, using the CT key.



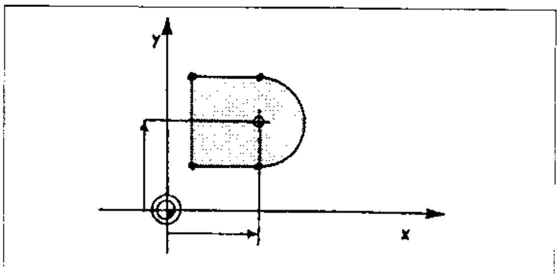
Circle centre CC

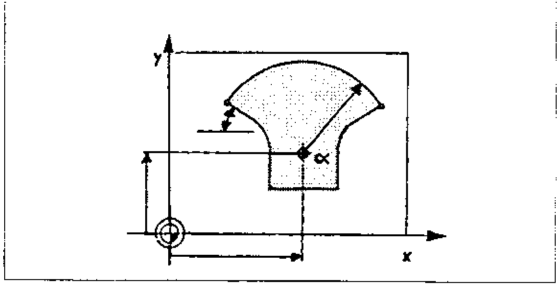
The circle centre CC must be programmed before the circular interpolation if the latter is programmed with the C key.
Two programming options are available:

- The circle centre CC is redefined by Cartesian coordinates.
- The coordinates programmed in the previous CC block are applied to the circle centre.

The input dialogue for the circle centre is initiated with the CC key (see "Pole").

CC in absolute dimensions: the circle centre is based on the workpiece datum.
CC in incremental dimensions: the circle centre is based on the previous nominal position of the tool.





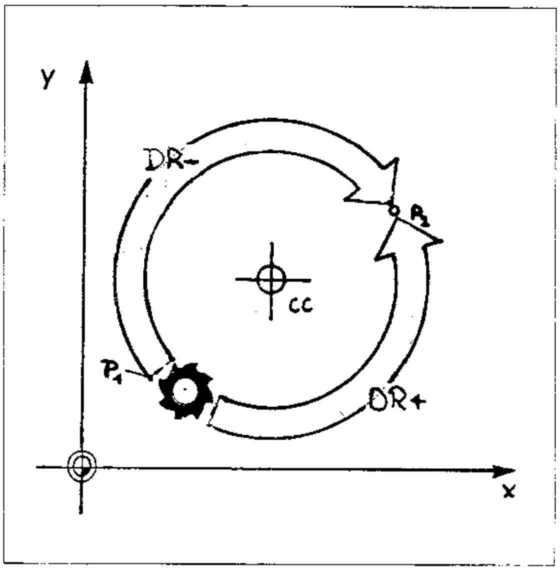
Circular path C

To move the tool from the actual position P1 along a circular path to the target position P2: program only P2.
The position P2 can be specified in either Cartesian or polar coordinates.

Direction of rotation

The **direction of rotation DR** for the circular path must be defined. The direction of rotation can be either positive DR+ (counterclockwise) or negative DR– (clockwise).

A compensated contour cannot be started with a circular path. Error message:
= PATH OFFSET INCORRECTLY STARTED =



Programming workpiece contours
Direction of rotation

Entry

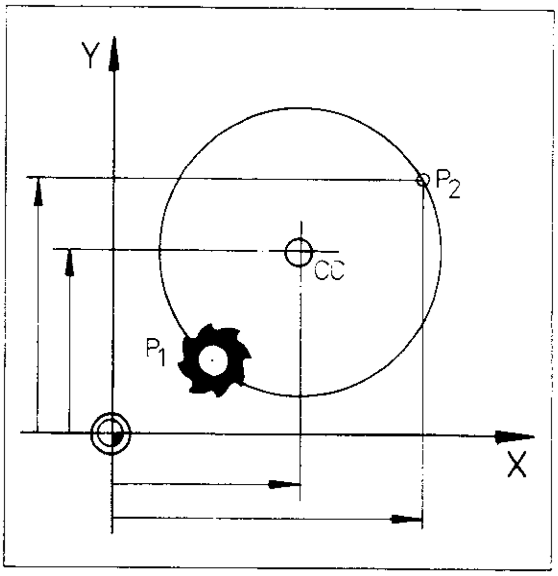
Dialogue prompt:

ROTATION CLOCKWISE: DR– ?	▶ +/–	
For clockwise rotation:	▼ ENT	Specify direction of rotation (–).
		Press ENT.
For counterclockwise rotation:	▶ +/–	Specify direction of rotation (+). (Press prefix key twice.)
	▼ +/–	
	▼ ENT	Press ENT.

Programming workpiece contours
Circular path C/Cartesian coordinates

Programming a circular path in Cartesian coordinates

Wher programming in Cartesian coordinates, make sure that starting position and target position (new nominal position) are located on the same circular path, i.e. that they are the same distance from the circle centre CC.
Otherwise, the error message
= CIRCLE END POS. INCORRECT =
will be displayed.



Programming workpiece contours
Circular path C/Cartesian coordinates

Make sure the red signal lamp beneath the "P" key is off. Press "P" to switch off if necessary.

Input in Cartesian coordinates

Operating mode ______ ⇨

Dialogue initiation ______ C

COORDINATES ?	X	Select axis, e.g. X.
	I	Incremental – absolute?
	□	Type numerical value.
	Y	Enter next coordinate, e.g. Y.
After **all coordinates of the circle end point** have been entered:	ENT	Press ENT.

ROTATION CLOCKWISE: DR– ?	+/–	Specify direction of rotation.
	ENT	Press ENT.

TOOL RADIUS COMP.: RL/RR/NO COMP. ?	RL RR	Specify radius compensation if required.
	ENT	Press ENT.

FEED RATE ? F=	□	Specify feed rate if required.
	ENT	Press ENT.

AUXILIARY FUNCTION M ?	□	Specify auxiliary function if required.
	ENT	Press ENT.

Sample display

```
87 C X+30.000 Y+48.000
          DR+ RR F        M
```

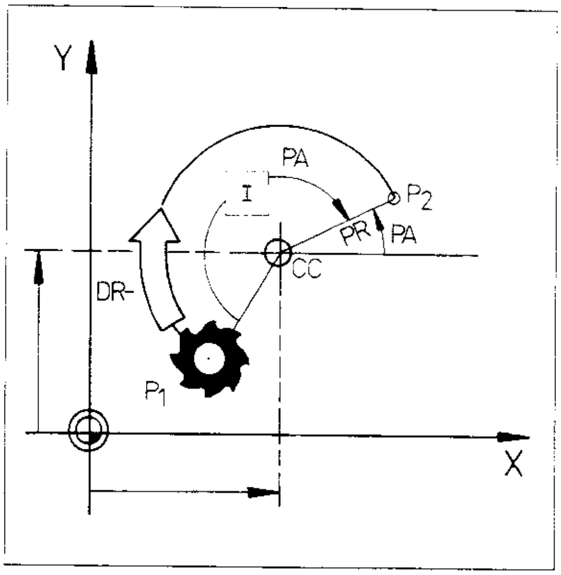
The tool moves along a circular path, in a positive direction of rotation (counterclockwise), with radius offset to the right of the contour, to position X 30.000 and Y 48.000.
The feed rate is defined by the most recently programmed value. No auxiliary function.

Programming workpiece contours
Circular path C/Polar coordinates

Programming a circular path in polar coordinates

If the target position on the arc is programmed in polar coordinates, only the polar angle PA (absolute or incremental) is required to define the end positicn. The radius is already defined by the positicn of the tool and programmed circle centre CC.

When programming a circular path in polar coordinates, the angle PA may be entered either as a positive or negative value. The angle PA indicates the end position of the arc. The direction of traverse DR can also be programmed as a positive or negative value. If the angle PA is specified in incremental dimensions, the prefixes of the angle and the direction of rotation should be identical. Based on the example at the right, both IPA and DR are negative.



If the tool is located at the pole or circle centre before circular interpolation begins, the error message
= ANGLE REFERENCE MISSING =
is disp ayed.

Programming workpiece contours
Circular path C/Polar coordinates

Make sure the red signal lamp beneath the "P" key is on. Press "P" to switch on if necessary.

Input in polar coordinates

Operating mode ________ ⇹

Dialogue initiation ________ P

POLAR COORDINATE ANGLE PA ?	▶ I	Incremental – absolute?
	▼ □	Specify polar angle PA for circle target position.
	▼ ENT	Press ENT.

▼

ROTATION CLOCKWISE: DR– ?	▶ +/–	Specify direction of rotation.
	▼ ENT	Press ENT.

▼

TOOL RADIUS COMP.: RL/RR/NO COMP. ?	▶ RL RR	Define radius compensation if required.
	▼ ENT	Press ENT.

▼

FEED RATE ? F=	▶ □	Specify feed rate if required.
	▼ ENT	Press ENT.

▼

AUXILIARY FUNCTION M ?	▶ □	Enter auxiliary function if required.
	▼ ENT	Press ENT.

Sample display

```
17 CP PA+60.000
      DR– RL F            M
```

The tool moves along a circular path in negative direction (clockwise), with radius offset, to the left of the programmed contour; the polar angle PA relative to the reference axis is 60°. The feed rate is defined by the most recently programmed value. No auxiliary function.

Programming workpiece contours
Circular path CR

Circular path CR

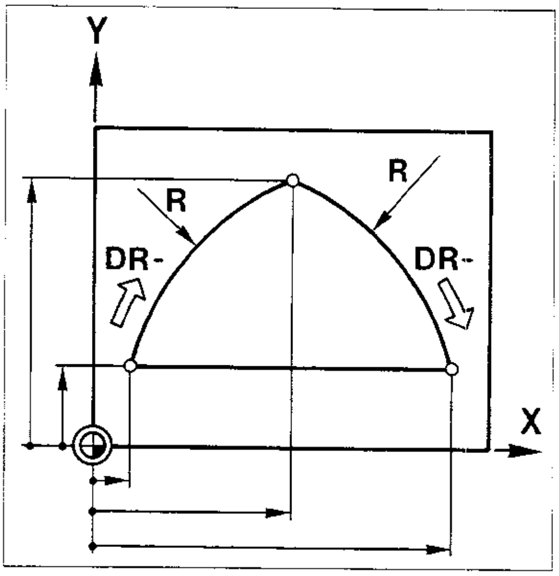
If the centre point of a circular path is not known, but the radius is specified, the circular path can be defined with the CR key via

- end position
- radius and
- direction of rotation

End position

The end position can be programmed in Cartesian coordinates only.

The distance between the starting point and end position of the path should not exceed 2 x R!

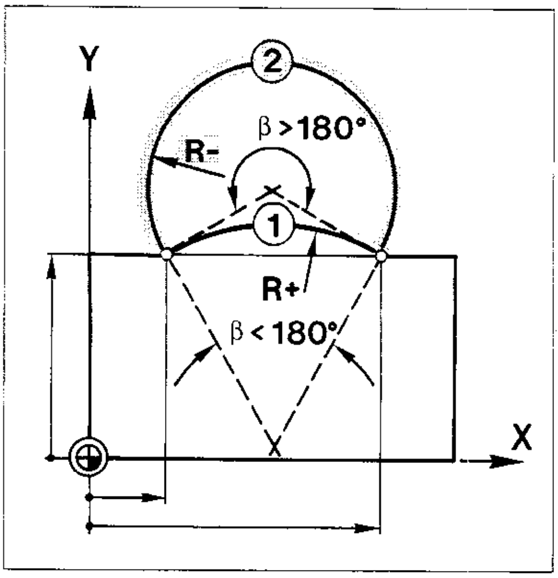


Radius

Two geometrical solutions are available for the circular path described above (see illustration). These solutions depend on the size of the central angle β:
the smaller **arc 1** has a central angle β < **180°**,
the larger **arc 2** has a central angle β > **180°.**

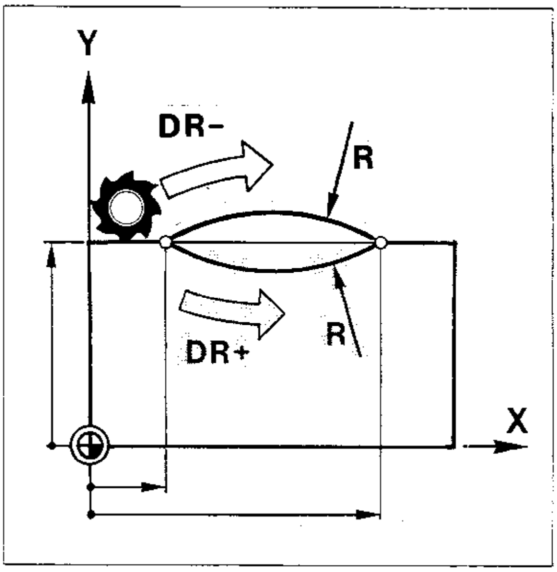
To program the **smaller arc** (β < 180°), enter a **positive radius** (the prefix + can be omitted).

To program the **larger arc** (β > 180°), enter a **negative radius.**



Direction of rotation

The direction of rotation DR indicates whether the circular path is concave or convex.
In the illustration at the right, DR– produces a convex contour element, DR+ a concave contour element.



Programming workpiece contours
Circular path CR

Input

Operating mode ______ ⇼

Dialogue initiation ______ CR

COORDINATES ?	▶ X	Select axis, e.g. X.
	▼ I	Incremental – absolute?
	▼ □	Type numerical value.
	▼ Y	Enter next coordinate, e.g. Y.
	▼ I	Incremental – absolute?
	▼ □	Type numerical value.
	▼ ENT	Press ENT.

▼

CIRCLE RADIUS ?	▶ □	Specify circle radius.
	▼ ENT	Press ENT.

▼

ROTATION CLOCKWISE: DR– ?	▶ +/–	Specify direction of rotation.
	▼ ENT	Press ENT.

▼

TOOL RADIUS COMP.: RL/RR/NO COMP. ?	▶ RL RR	Specify radius compensation if required.
	▼ ENT	Press ENT.

▼

FEED RATE ? F =	▶ □	Specify feed rate if required.
	▼ ENT	Press ENT.

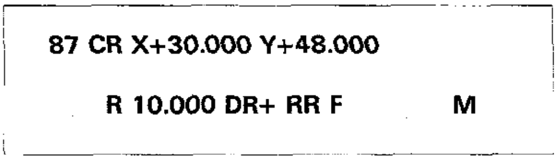
▼

AUXILIARY FUNCTION M ?	▶ □	Specify auxiliary function if required.
	▼ ENT	Press ENT.

Notes:

Programming workpiece contours
Circular path CR

Sample display



```
87 CR X+30.000 Y+48.000

    R 10.000 DR+ RR F          M
```

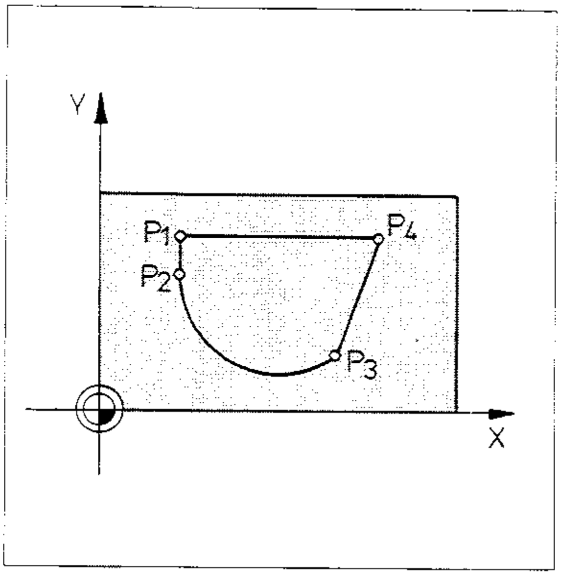
The tool moves along a circular path with a radius of 10.000, in a positive direction of rotation (counterclockwise), with radius offset to the right of the programmed contour, to position X 30.000 and Y 48.000.
The feed rate is defined by the most recently programmed value. No auxiliary function.

Radii up to 99 m can be machined if the radius is defined by Q-parameter programming (no entry via keyboard).

Programming workpiece contours
Tangential arc

Arc with tangential connection

Programming a circular path is simplified considerably if the arc is connected tangentially to the contour. Only the **end position of the arc** need be entered to define the arc.



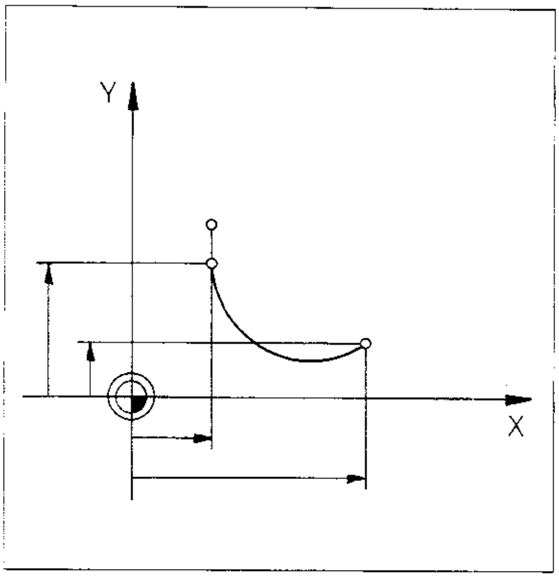
Requirements

The contour section to which the circular path is to be connected tangentially should be entered immediately before programming the tangential arc. If the contour section is missing, the following error message will be displayed:
= CIRCLE END POS. INCORRECT =

Both coordinates of the machining plane must be programmed in the positioning block preceding the tangential arc and in the positioning block for the tangential arc, otherwise, the error message:
= ANGLE REFERENCE MISSING =
will be generated.

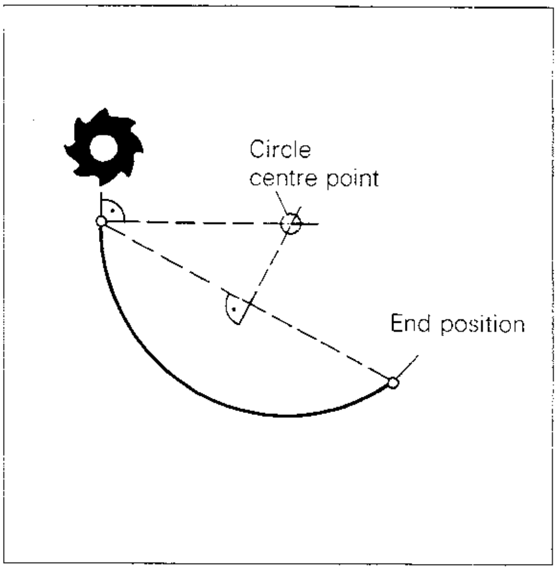
Input

The end position of the circular path can be programmed either in **Cartesian coordinates** or in **polar coordinates.**
Initiate the dialogue by pressing the CT key
or P CT.



Geometry

In the case of tangential transition to the contour, an **exact arc** is defined by the end position of the circular path.
Because the arc has a definite radius, a definite direction of rotation and a definite centre point, it is not necessary to program these data.



Programming workpiece contours
Tangential arc/Cartesian coordinates

Make sure the red signal lamp beneath the "P" key is off. Press "P" to switch off if necessary.

Input

Operating mode ______________ ⇨

Dialogue initiation ______________ CT

COORDINATES ?	X	Select axis, e.g. X.
	I	Incremental – absolute?
	□	Specify numerical value.
	Y	Enter next coordinate, e.g. Y.
	I	Incremental – absolute?
	□	Specify numerical value.
	ENT	Press ENT.

TOOL RADIUS COMP.: RL/RR/NO COMP. ?	RL RR	Specify radius compensation if required.
	ENT	Press ENT.

FEED RATE ? F =	□	Specify feed rate if required.
	ENT	Press ENT.

AUXILIARY FUNCTION M ?	□	Specify auxiliary function if required.
	ENT	Press ENT.

A full circle cannot be programmed.

Sample display

```
20 CT X+15.800 Y+35.000

         R F          M
```

An arc is connected tangentially to the last programmed contour section. The coordinates of the end position of the arc are X 15.800 and Y 35.000.

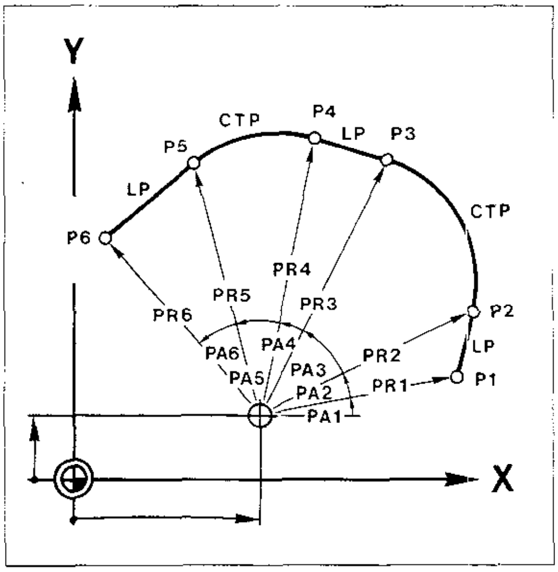
Programming workpiece contours

Tangential arc/Polar coordinates

Input in polar coordinates

Indicating the target position in polar coordinates simplifies the programming of cams, for example.

Make sure the pole CC is defined before programming in polar coordinates.



Programming workpiece contours
Tangential arc/Polar coordinates

Make sure the red signal lamp beneath the "P" key is on. Press "P" to switch on if necessary.

Input

Operating mode ______

Dialogue initiation ______ P CTP

POLAR COORDINATE RADIUS PR ?

I — Incremental – absolute?

Enter polar radius PR for the arc end position.

ENT — Press ENT.

POLAR COORDINATE ANGLE PA ?

I — Incremental – absolute?

Enter polar angle PA for the arc end position.

ENT — Press ENT.

TOOL RADIUS COMP.: RL/RR/NO COMP. ?

RL RR — Specify radius compensation if required.

ENT — Press ENT.

FEED RATE ? F=

Specify feed rate if requi-ed.

ENT — Press ENT.

AUXILIARY FUNCTION M ?

Specify auxiliary function if required.

ENT — Press ENT.

A full circle cannot be programmed.

Sample display

```
30 CTP PR+35.000 PA+90.000
                R F      M
```

An arc is connected tangentially to the last programmed contour section. The end position of the circular path is 35.000 from the last defined pole CC; the polar coordinate angle is 90° (absolute).
Too. radius compensation and feed rate are defined by the previously programmed values. No aux liary function.

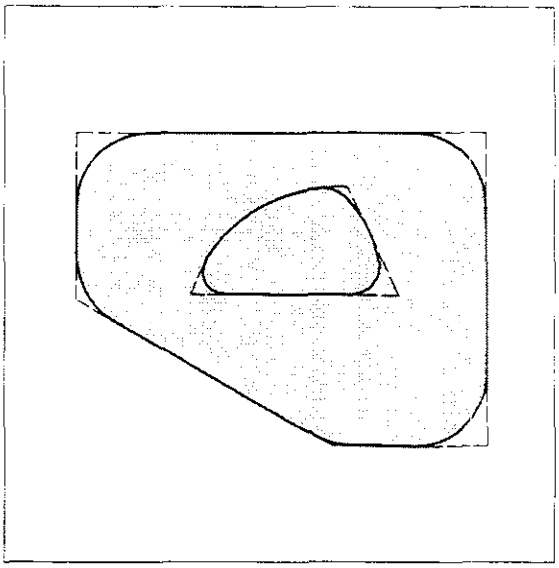
Programming workpiece contours
Rounding corners

Rounding corners RND

Contour corners can be rounded by inserting circular arcs. The arc blends tangentially into the preceding and subsequent contour segments.

A rounding radius can be inserted at any corner created by the intersection of the following contour elements:

- straight line – straight line
- straight line – arc or arc – straight line
- arc – arc



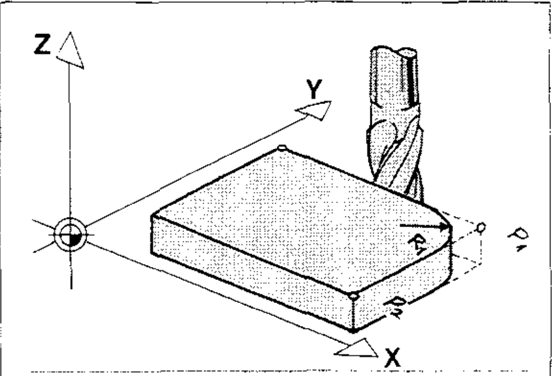
Programming tip

The rounding radius can only be inserted in a **main plane.** For this reason, the **machining plane** must be the same in the positioning blocks preceding and following the RND block. Otherwise, the following error message will be generated when the program is run:
= PLANE INCORRECTLY DEFINED =

Programming

The rounding radius is programmed immediately following contour point P1, where the corner is located.
The rounding radius and, if required, a reduced feed rate F for milling the rounded corner is entered.

The feed rate for rounding corners is effective only in the block in which it is programmed. The previously programmed feed rate is effective again after the RND block.

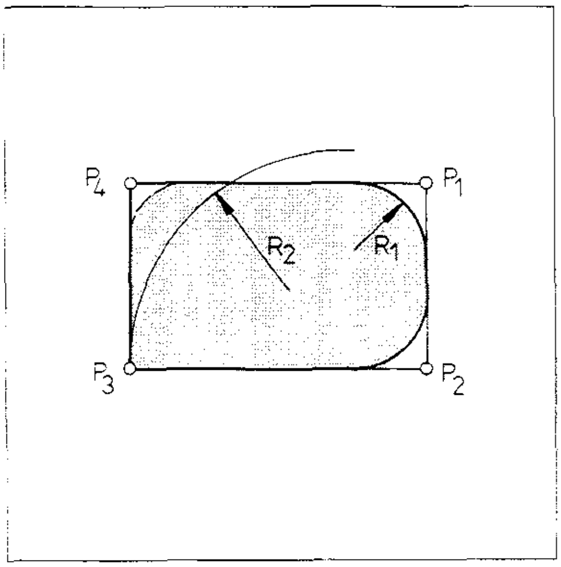


15 straight line to P1 (X, Y)

16 RND R 15.000 F80

17 straight line to P2 (X, Y)

The rounding radius should not be too large; it must "fit" between the contour elements.
If the radius selected is too large, the error message
= ROUNDING RADIUS TOO LARGE =
will be displayed.



Programming workpiece contours
Rounding corners

☞ Contour elements that are located in the same machining plane must be programmed before and after an RND block and must include both coordinates of the machining plane.

Input

Operating mode ______ ⇾

Dialogue initiation ______ RND

ROUNDING RADIUS R ?	▶□	Specify radius of corner arc.
	▼ ENT	Press ENT.

FEED RATE ? F =	▶□	Specify feed rate if required.
	▼ ENT	Press ENT.

Sample display

```
78 RND R 5.000

              F 20
```

A corner arc with radius R = 5.000 is inserted between the block programmed previously and the one programmed subsequently. The feed rate for milling the rounded corner is 20 mm/min.

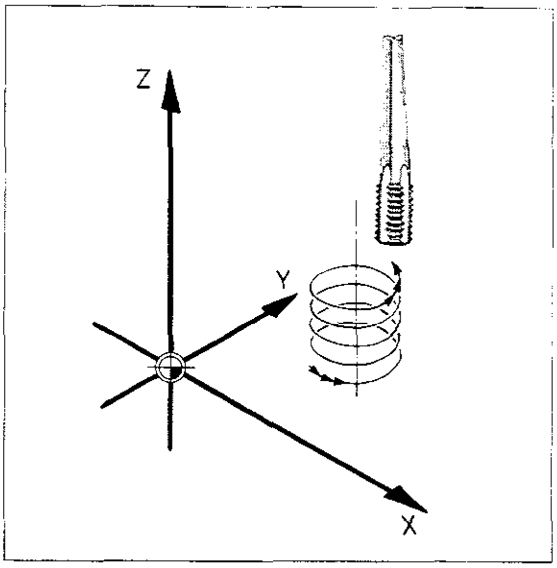
Programming workpiece contours
Helical interpolation

Helix

In the case of circular interpolation, two axes are traversed simultaneously in such a way that a circle is described in a main plane (XY, YZ, ZX). If a linear movement of the third axis (tool axis) is superimposed on this circular interpolation, the tool will follow a helical (spiral) path.
Tool axis is X, Y, Z or IV axis if the IV axis is designated as U, V or W.

Helical interpolation can be used to produce large-diameter internal and external threads or lubricating grooves.

Helical interpolation is not available on control system models TNC 151 F/TNC 155 F/TNC 151 W/TNC 155 W.

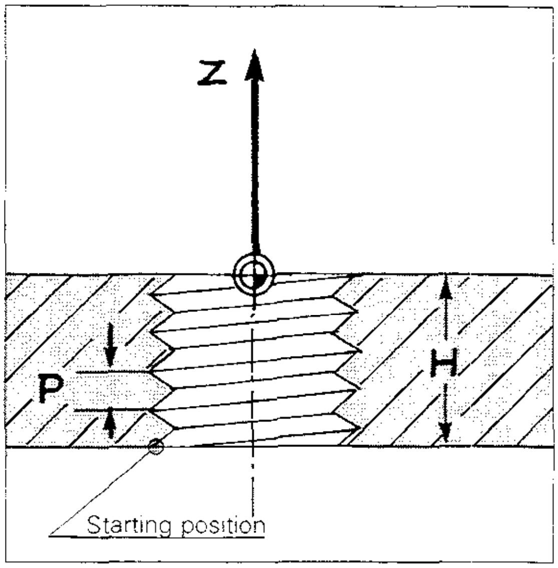


Input data

The helix can be programmed only in polar coordinates.
As in the case of circular interpolation, the **circle centre CC** must be defined **in advance.**
The total angle of rotation of the tool is indicated as the **polar angle PA in degrees:**
PA = number of rotations x 360°
Enter PA in incremental dimensions if the angle of rotation is greater than 360°.
Total height/depth is entered in response to the prompt **COORDINATES?.** The value depends on the desired pitch.

H = P x A	H = total height/depth
	P = pitch
	A = number of turns

The total height/depth can also be entered in either absolute or incremental dimensions.



Radius compensation

The value entered for radius compensation depends on:
- direction of rotation (CW/CCW)
- type of thread (internal/external)
- machining direction (pos./neg. axis direction)

Negative axis direction (−Z or −Y)			
Thread	Rotation direction	Radius compens.	
		intern.	extern.
Left-hand thr.	DR+	RL	RR
Right-hand thr.	DR−	RR	RL

Positive axis direction (+Z or +Y)			
Thread	Rotation direction	Radius compens.	
		intern.	extern.
Left-hand thr.	DR−	RR	RL
Right-hand thr.	DR+	RL	RR

Programming workpiece contours
Helical interpolation

Make sure the red signal lamp beneath the "P" key is on. Press "P" to switch on if necessary.

Input

Operating mode ______ ⇨

Dialogue initiation ______ P C

POLAR COORDINATE ANGLE PA ?	▶ I	Incremental – absolute?
	▼ □	Specify total angle of rotation in degrees.

COORDINATES ?	▶ Z	Select infeed axis.
	▼ I	Incremental – absolute?
	▼ □	Specify height or depth.
	▼ ENT	Press ENT.

ROTATION CLOCKWISE: DR– ?	▶ +/–	Specify direction of rotation.
	▼ ENT	Press ENT.

TOOL RADIUS COMP.: RL/RR/NO COMP. ?	▶ RL RR	Specify radius compensation.
	▼ ENT	Press ENT.

FEED RATE ? F=	▶ □	Specify feed rate if required.
	▼ ENT	Press ENT.

AUXILIARY FUNCTION M ?	▶ □	Specify auxiliary function if required.
	▼ ENT	Press ENT.

Sample display

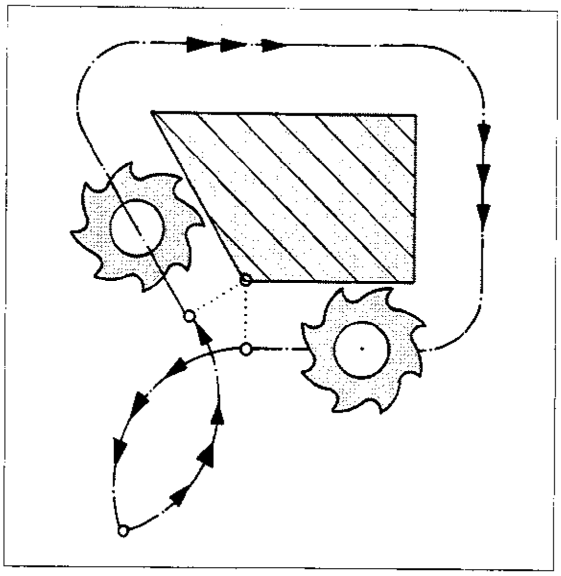
```
230 CP IPA+720.000 IZ+6.000
       DR+ RL F100          M
```

The tool completes two full revolutions, moving counterclockwise along a helical path. Total height is 6 mm; resulting pitch is 3 mm. The tool travels with radius offset to the left of the contour, producing an internal thread.

Contour approach and departure on an arc

Approach and departure on an arc

Approaching and departing the contour along an arc-shaped path offers the advantage of a "smooth" tangential approach and departure.
A smooth approach is programmed with the RND key.



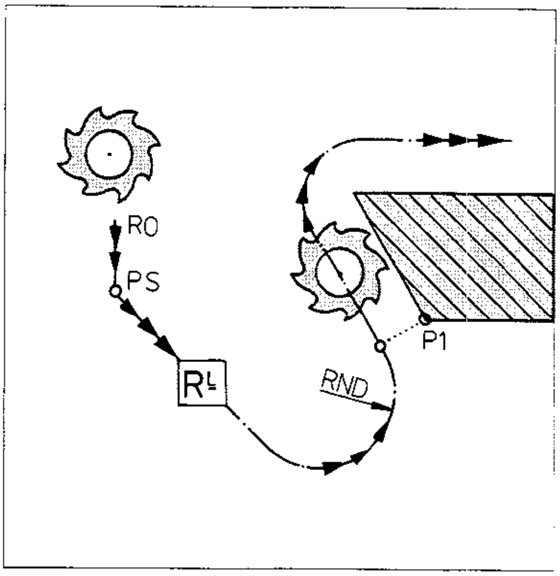
Approach

The tool moves to the starting position PS and then on to the location of the contour to be machined.

The positioning block for traverse to point PS should not contain a radius compensation (i.e. R0).

The positioning block for traverse to the first contour position P1 must contain a radius compensation (RR or RL).

Based on the data in the RND block, which follows the positioning block to contour position P1, the control system recognizes that a **tangential** approach to the contour is required.



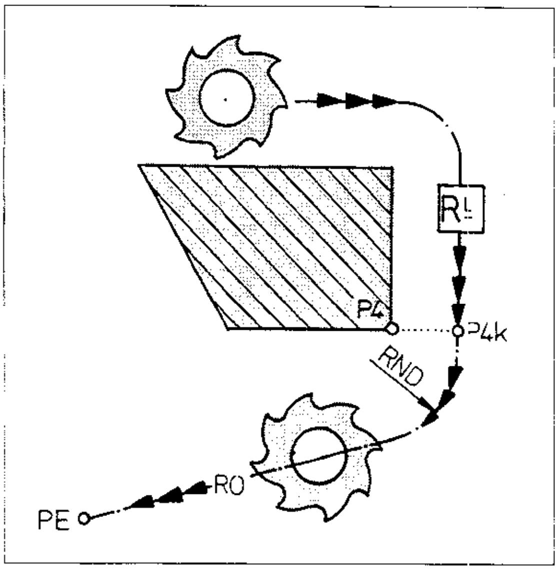
Departure

The tool reaches the last contour position P4 and then moves on to the end position PE.

The positioning block for traverse to P4 must contain a radius compensation (RR or RL).

The positioning block for traverse to point PE should not contain a radius compensation (i.e. R0).

Based on the data in the RND block, which follows the positioning block to the final contour position P4, the control system recognizes that a **tangential** departure from the contour is required.

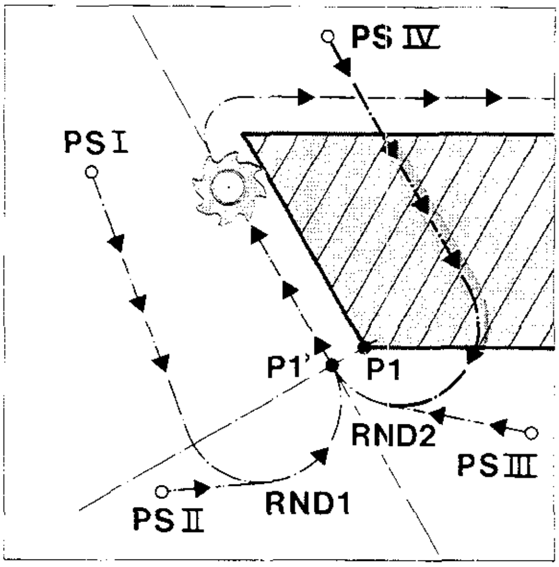


Contour approach and departure on an arc

Starting position

The starting position PS must be located in quadrant I, II or III.
The quadrants are formed by the starting direction (tangential direction in the case of an arc) in P1' and the corresponding perpendicular, which also intersects P1'. The workpiece will be damaged if the starting point is located in quadrant IV.

P1 = first point on contour
P1' = first compensated point on contour
PS = starting position (with R0)
RND1 = rounding radius for I, II
RND2 = rounding radius for III, IV



The feed rate in the RND block is effective blockwise. After the RND block the previously programmed feed rate is active again.

Programming an approach

```
20 L X+100.000 Y+50.000
     R0 F 15999              M
```

Positioning block to starting position PS with **R0.**

```
21 L X+65.000  Y+40.000
     RR F 50                 M13
```

Positioning block to first contour position P1 with radius compensation **RR.**

```
22 RND R 10.000
       F
```

Circular path radius for tangential approach.

```
23 L X+65.000  Y+100.000
     R  F                    M
```

Positioning block to next contour position P2.

If no feed rate for tangential approach is programmed in the RND block, then the feed rate of the next positioning block is effective in the RND block.

Programming a departure

```
30 L X+50.000  Y+65.000
     RR F 50                 M
```

Positioning block to last contour position P with radius compensation **RR.**

```
31 RND R 15.000
       F
```

Circular path radius for tangential departure.

```
32 L X+100.000 Y+85.000
     R0 F 15999              M00
```

Positioning block or end position PE with **R0.**

Caution when entering F 15999! Danger of collision!

A positioning block containing the two coordinates of the machining plane must be programmed before and after an RND block.

Contour approach and departure in a straight line
Introduction

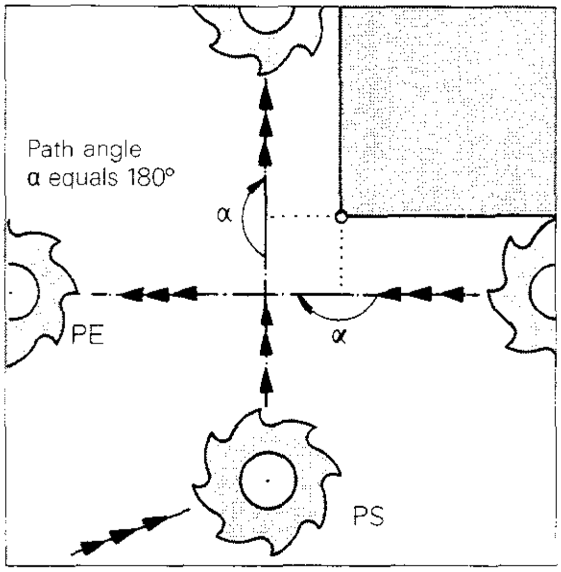
Approach and departure in a straight line

The tool is to approach the starting position PS and then proceed to the contour. After machining, the tool is to depart from the contour and move to end position PE.

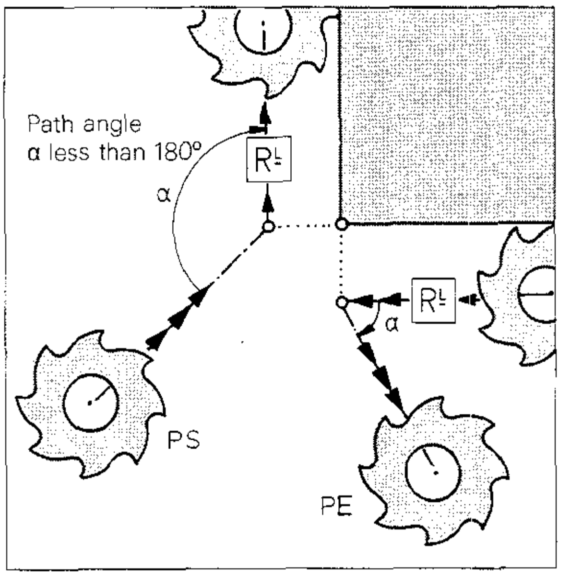
Path angle α

Approach and departure characteristics depend on the path angle α. The path angle is the angle formed by the first or last contour element and the straight-line approach or departure path. In general, three variations are possible:

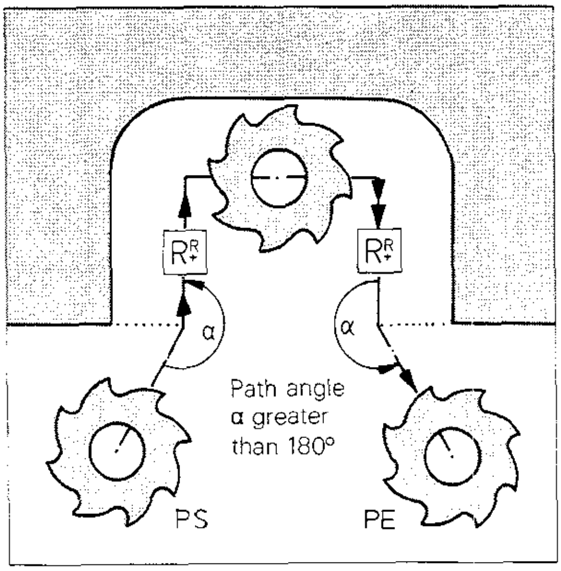
- Path angle α equals 180°



- Path angle α less than 180°



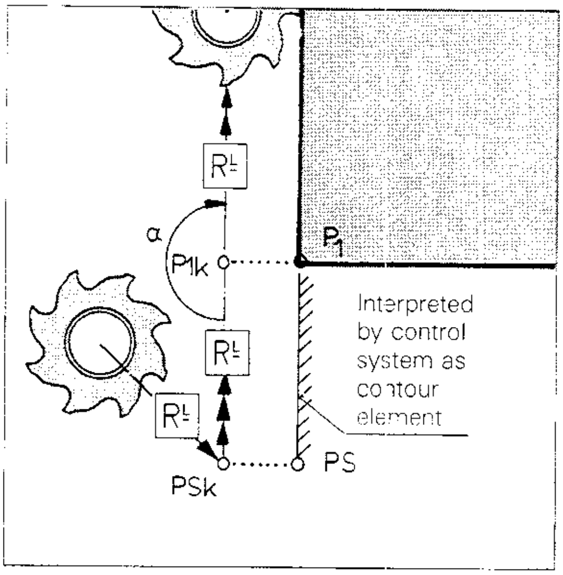
- Path angle α greater than 180°



Contour approach and departure in a straight line
Path angle α equals 180°

Path angle α equals 180°

If **path angle** α is equal to 180°, the starting and end positions are located on straight line extensions tangential to the first and last contour directions. The starting and end positions must be programmed **with radius compensation** (RL or RR).

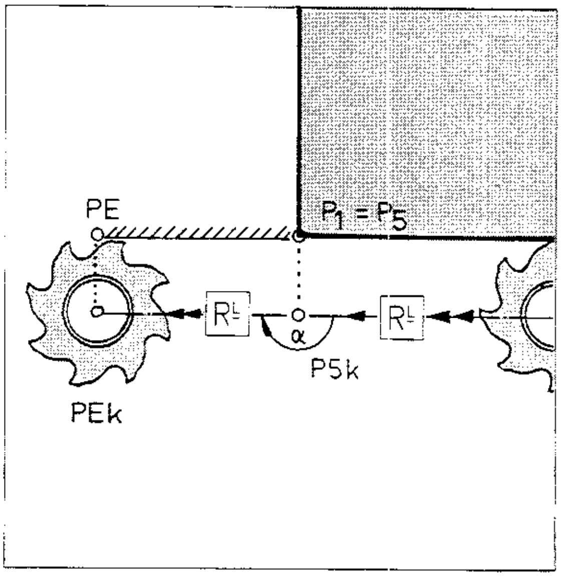


Approach

The control system moves the tool in a straight line to the compensated position PSk of the imaginary contour position PS and then follows the compensated path to position P1k.

Departure

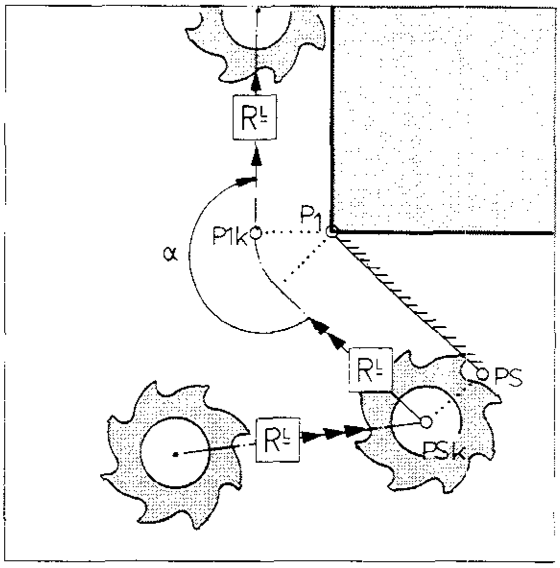
The control system moves the tool from compensated position P5k of contour position P5 to position PEk, following the compensated path.



Contour approach and departure in a straight line
Path angle α greater than 180°

Path angle α greater than 180°

If α is greater than 180°, the starting and end positions must be programmed **with radius compensation** (RI or RR). The first and last contour positions are assumed to form an external corner. The control system executes a path compensation on external corners and inserts a transition arc (blend).

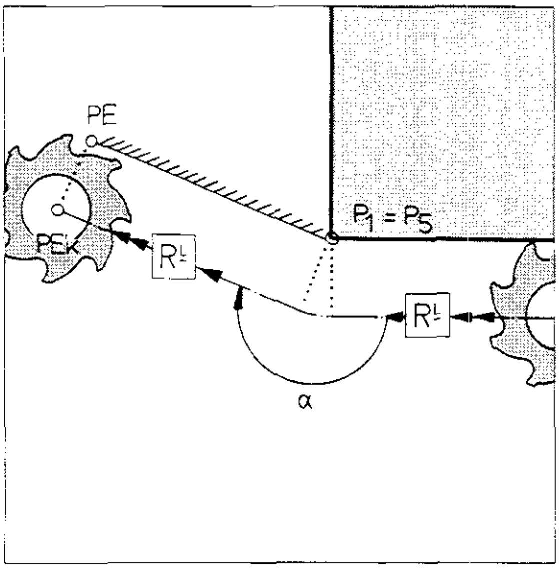


Approach

The control system considers starting position PS to be the first contour position. The tool moves to PSk and then to position P1k, following the compensated path.

Departure

The control system considers the end position PE to be the final contour position. The tool moves along the compensated path to end position PEk.



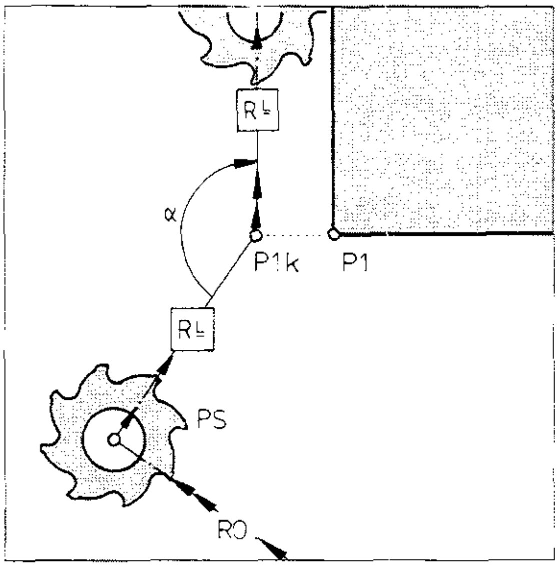
Contour approach and departure in a straight line
Path angle α less than 180°

Path angle α less than 180°

If α is less than 180°, the starting and end positions must be programmed **without radius compensation,** i.e. with R0. PS and PE are approached without path compensation.

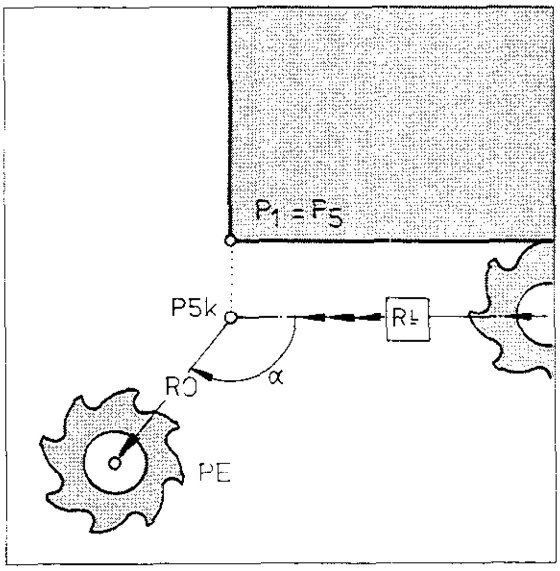
Approach

The control system moves the tool in a straight line to the compensated position P1k of contour position P1.



Departure

The control system moves the tool in a straight line from compensated position P5k of contour position P5 to uncompensated position PE.



Contour approach and departure in a straight line

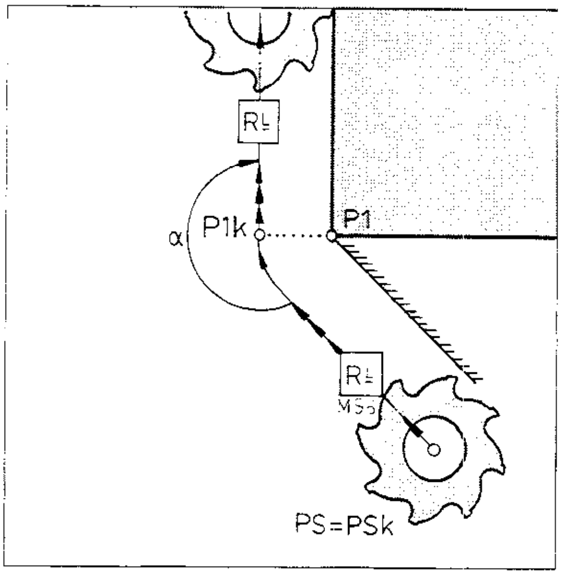
Approach command M96 for external corners
Departure command M98

Approach command M96 for external corners

The contour will be damaged if position PS was programmed without radius compensation and path angle α is greater than 180°.
With auxiliary function M96, the starting position PS is interpreted as a compensated contour position PSk. The tool moves along the compensated path to position P1k.
The auxiliary function M96 is programmed if the **approach angle α is greater than 180°.** M96 is programmed in the positioning block for P1.

M96 is always in effect if no path compensation is active at the beginning of the program.

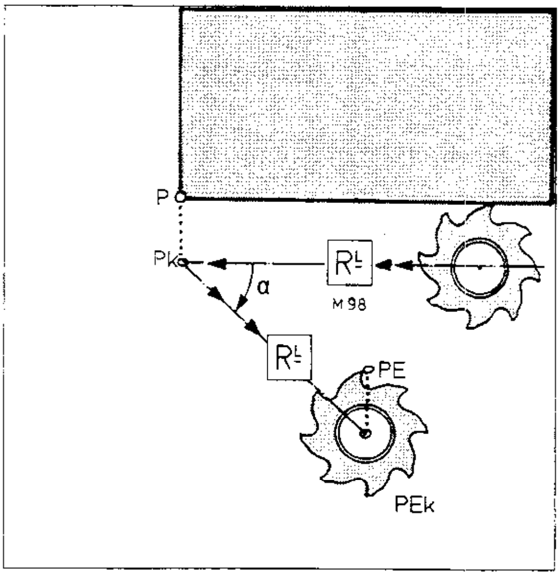
Incomplete machining of the contour will result if M96 is programmed and path angle α is less than 180°.



Departure command M98

If the end position was programmed with radius compensation and the **departure angle α is less than 180°,** the contour will be incompletely machined.

If auxiliary function M98 is programmed in the positioning block to P, the tool moves directly to point Pk and then to compensated point PEk. The direction PE – PEk equals the last executed radius compensation, in this case P – Pk.

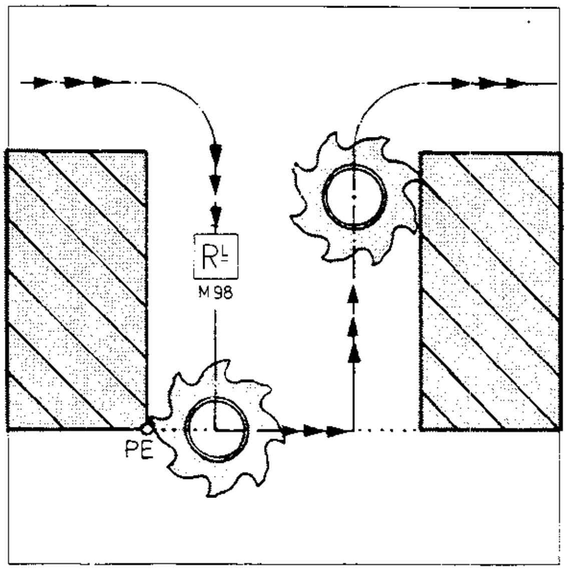


Terminating path compensation M98

If additional positions or contour points were programmed after PE, the required radius compensation direction depends on the direction of the next contour section.

M98 programmed in the positioning block to the final point on the contour causes the relevant contour element to be completed and, as shown in the example at the right, traverse with the last programmed radius compensation to the first point on the next contour.

The auxiliary function M98 is effective only in the block in which it is programmed. In the subsequent positioning block, M98 prevents the insertion of transition arcs on external corners and the calculation of path intersections on internal corners.



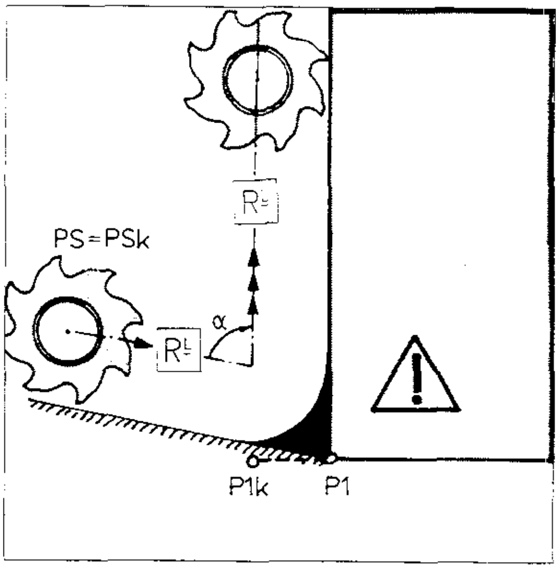
Contour approach and departure in a straight line

Tool at starting position

Approach command M95 for internal corners

Problems with approach angles α less than 180°

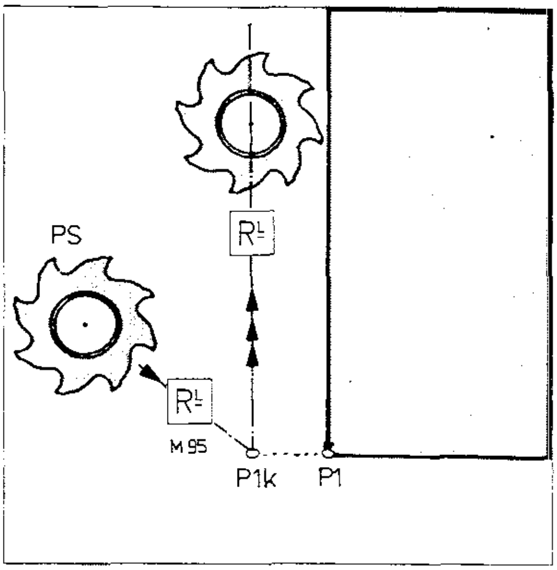
At the beginning of the program, the tool **happens to be at the actual position PS** and is to move to the nominal position P1 with radius compensation.
In this case, the control system interprets the random position PS as the compensated tool position PSk of an imaginary point on the contour and point P1k cannot be approached due to the path compensation.



Approach command M95

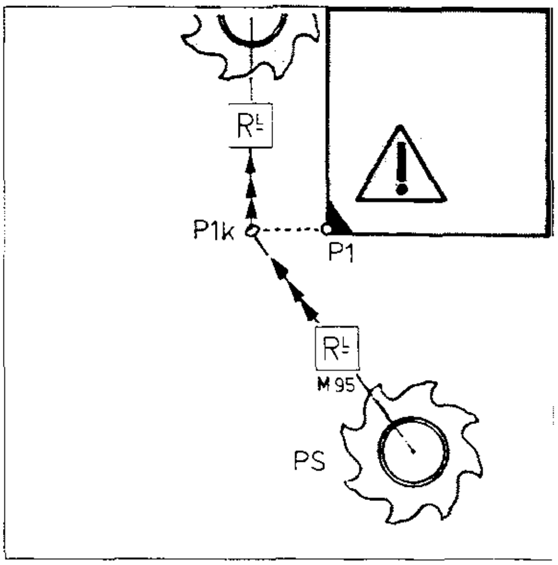
The auxiliary function M95 cancels the path compensation for the first positioning block. The tool moves without path compensation from position PS to the compensated contour point P1k.

Auxiliary function M95 is programmed if the approach angle α is less than 180°. M95 is programmed in the positioning block for P1.

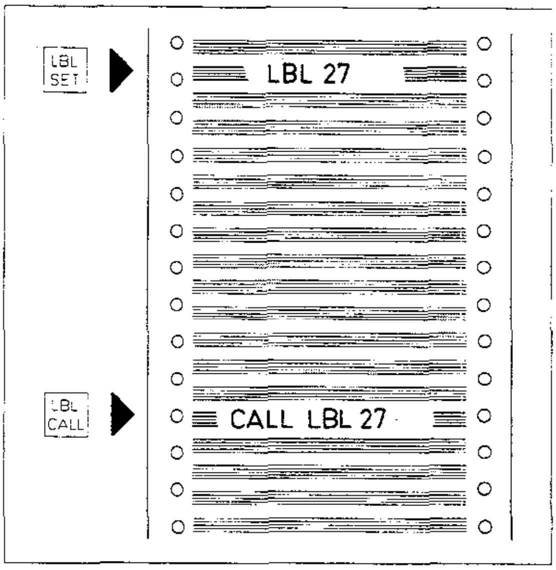


M95 is active only at the beginning of the machining program.
Use the function M98 (see "Terminating path compensation") to cancel path compensation within a machining program blockwise.

If M95 is programmed when the approach angle α is greater than 180°, the contour will be damaged.



Subroutines and program part repetition
Program markers (labels)



Labels

When programming, labels (program markers) with a specified number can be set to mark the start of a given program part, such as a subroutine.
You can then jump to these program markers while a program is running (e.g. to execute the subroutine in question).

Setting a label LBL SET

A label is set by pressing the LBL SET key.

Label number

You may choose label numbers from 0 to 254. The **label number 0** always marks the **end of a subroutine** (see "Subroutine"), and is therefore a return jump marker!

If you enter a label number that has already been set somewhere else in the program, the following error message will appear:
= LABEL NUMBER ALREADY ALLOCATED =

Calling up a label LBL CALL

Dialogue is initiated by pressing the LBL CALL key.
Using LBL CALL,

- **subroutines** can be called up, and
- **program part repetitions** can be programmed.

Label number

You may call up label numbers from 1 to 254.

If you enter the number 0, the following error message will appear:
= JUMP TO LABEL 0 NOT PERMITTED =

Program part repetition REP

For **program part repetition,** respond to the question "REPEAT REP" by entering the desired number of repetitions.
For **calling up a subroutine,** respond to the question REP by pressing the NO ENT key.

Subroutines and program part repetition
Labels

Setting a label

Operating mode ⟐

Dialogue initiation LBL SET

LABEL NUMBER ?	▶ □	Specify label number.
	ENT	Press ENT.

Sample display

```
118 LBL 27
```

Label number 27 has been set in block 18.

Calling up a label

Operating mode ⟐

Dialogue initiation LBL CALL

LABEL NUMBER ?	▶ □	Specify label number to be called up.
	ENT	Press ENT.

▼

REPEAT REP ?

If you want to enter a **program part repetition:**	▶ □	Specify the number of repetitions.
	ENT	Press ENT.
If you want to enter a **subroutine call:**	▶ NO ENT	Press NO ENT.

Sample display 1

```
29 CALL LBL 5 REP 2/2
```

A program part will be repeated twice. The number after the slash indicates the number of repetitions still to be executed in the program run. It decreases by 1 after each repetition.

Sample display 2

```
218 CALL LBL 27 REP
```

The subprogram labelled 27 is called up (machining is continued at block 118, see above).

Subroutines and program part repetition

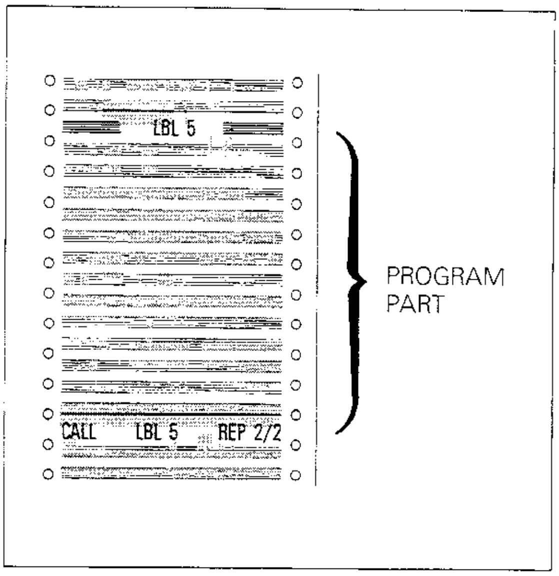
Program part repetition

Program part repetition

Program parts that have already been executed can be repeated upon completion of the program. This is referred to as a program loop or **program part repetition.**

The **beginning** of the program part which is to be repeated is marked with a **label number.** The end of the program part consists of a label number call **LBL CALL** and the programmed **number of program part repetitions REP.**

A program part can be repeated up to 65,534 times.



Program run

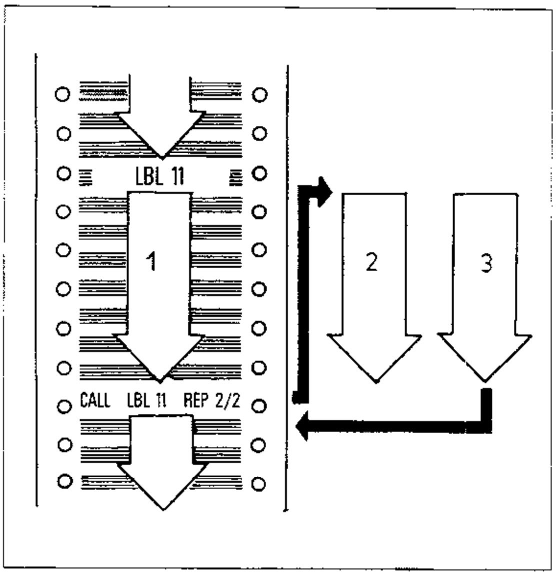
The control system executes the main program (together with the appropriate program part) up to the label number call.

Then it jumps to the specified program marker and the program part is repeated.

On the display screen, the number of remaining repetitions is reduced by 1: REP 2/1.
After another jump, the program part is repeated a second time.

Once all programmed repetitions have been executed (display: REP 2/0), machining with the main program is resumed.

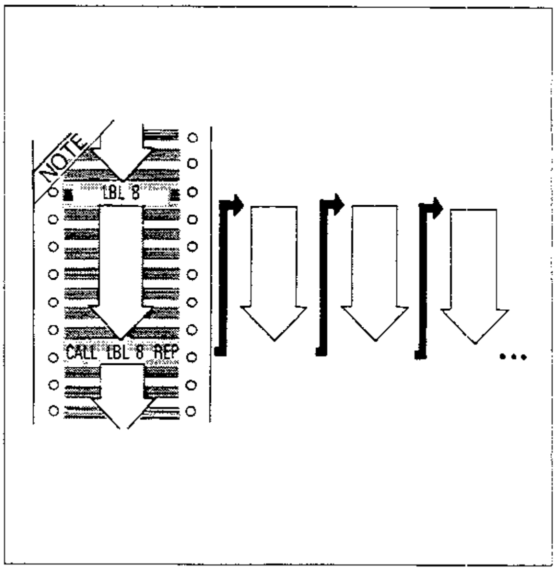
Altogether, the program part is always executed once more than the number of programmed repetitions.



Programming errors

If **no entry** is made (if you press the NO ENT key) in response to **REP** (number of repetitions), you will create a loop: the **label number call** will be **repeated 8 times.**

During the program run and in a test run, the following error message will appear on the display screen after the 8th repetition:
= EXCESSIVE SUBPROGRAMMING =



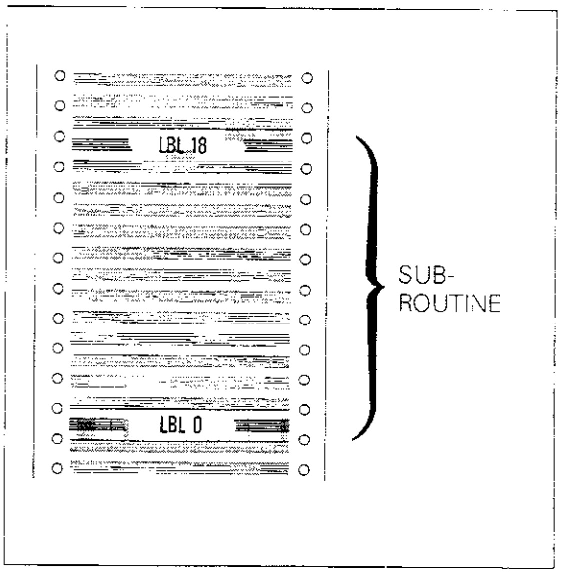
Subroutines and program part repetition
Subroutines

Subroutines

If a program part is required again at other points in the machining program, it can be marked as a subroutine.
The **beginning** of the subroutine is marked with any desired **label number.** The **end** of the subprogram is always designated by **label number 0.**

If the end of the subroutine is not marked by LBL 0, calling up the subroutine can result in excessive subroutine nesting, (see error message: EXCESSIVE SUBPROGRAMMING).

The subroutine is called up with LBL CALL, and can be called up at any location in the program, but not within the same subroutine.
After execution of the subroutine, a return jump is made to the jump location in the main program.



Program run

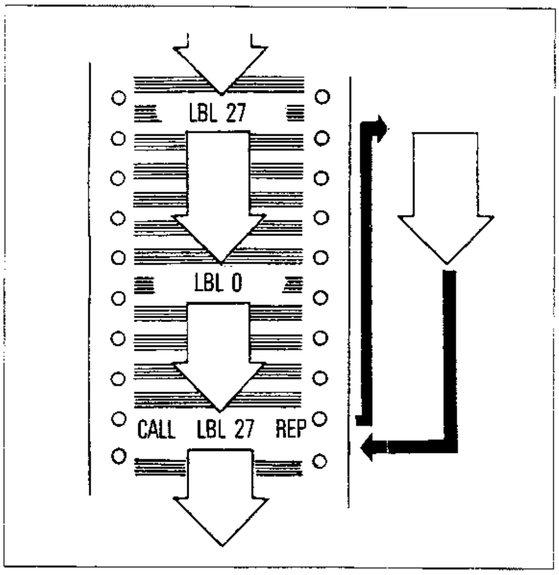
The control system executes the main program until a subroutine is called up (CALL LBL 27 REP).

Then it jumps to the program marker which has been called up.

The subroutine is executed up to label number 0 (end of subroutine).

Then a jump is made back into the main program.

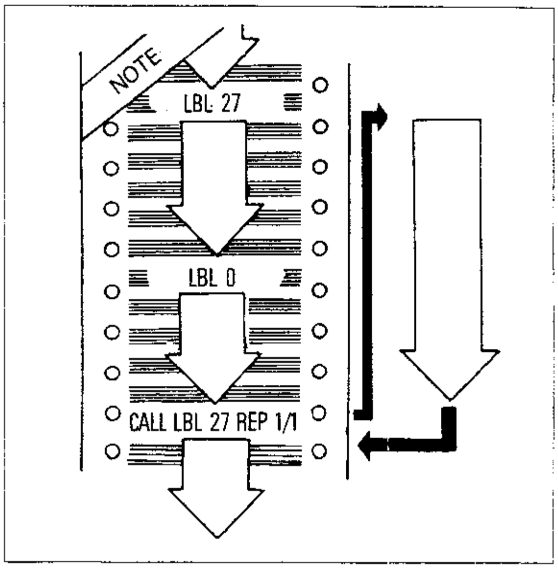
The main program continues at the block following the subroutine call.



If the subroutine is incorporated into the main program, as in the example above, it is run once during program execution without having to call it up.

A subroutine can only be executed once using a subroutine call! When calling up a program with LBL CALL, you must respond to the dialogue prompt REPEAT REP? by pressing the NO ENT key.

If a repetition, e.g. REP 1/1, is programmed, the program section between the called-up label number and the command LBL CALL is carried out as a program part repetition. The program marker LBL 0 is not taken into account.

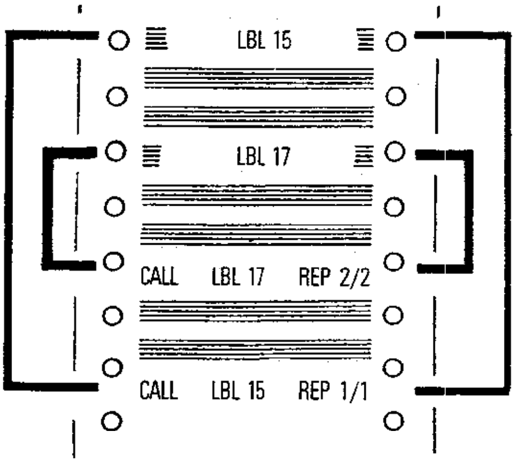


Subroutines and program part repetition
Nesting

Nesting

An additional subroutine or an additional program part repetition can be called up within a subroutine or a program part repetition.
This procedure is referred to as nesting.

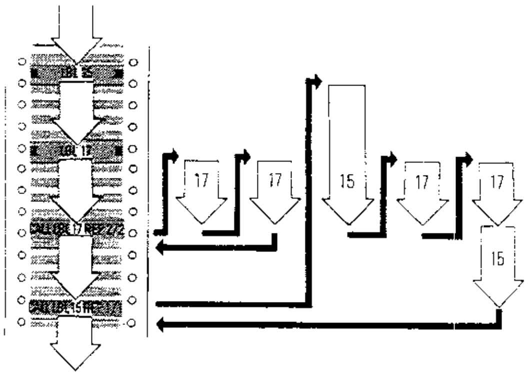
Program parts and **subroutines** can be nested up to 8 times, that is, the **nesting level** is 8.
If the nesting level is exceeded, the following error message will appear:
= EXCESSIVE SUBPROGRAMMING =



Program run with repetition

The main program is executed up to the jump to LBL 17. The program part is repeated twice.

Then the control system continues executing the main program up to the jump to LBL 15. The program part is repeated once up to CALL LBL 17 REP 2/2; the nested program part is again run twice. Then, the previously programmed repetition is continued after CALL LBL 17.

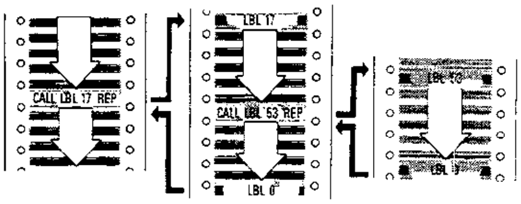


Program run with subroutines

The main program is executed until the jump command CALL LBL 17.

Then, the subroutine is executed from LBL 17 to the next subroutine call CALL LBL 53, etc. The subroutine nested at the lowest level is executed without interruption.

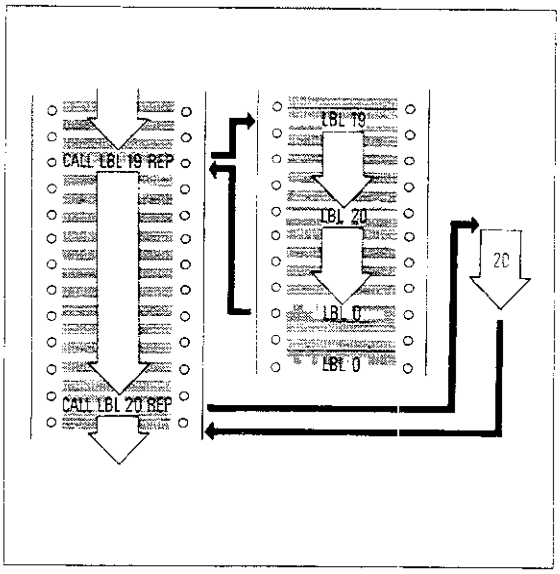
Before the end (LBL 0) of the final subroutine, a jump is made back to each preceding subroutine, until subsequently, the main program is reached.



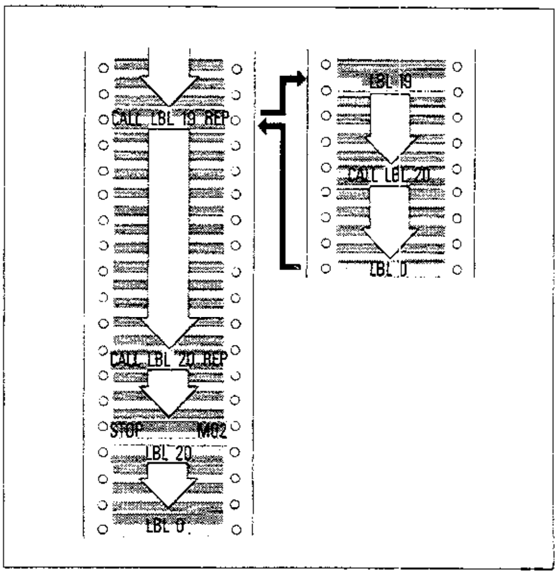
Subroutines and program part repetition
Nesting

Subroutine within a subroutine

A subroutine cannot be written into an existing subroutine. Therefore, each subroutine in the example shown is only executed up to the first label number 0.



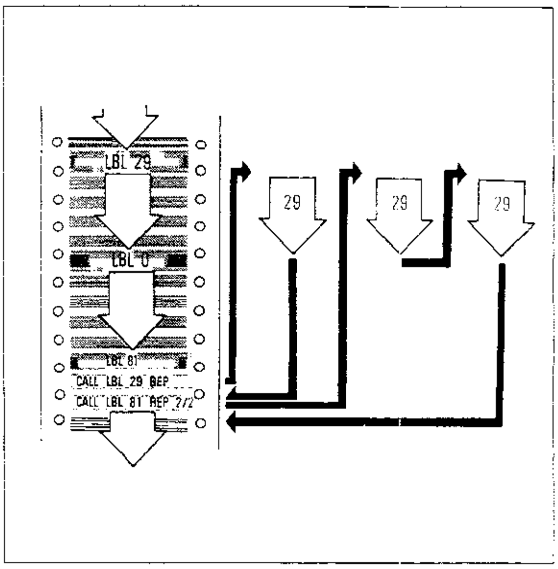
In this case, subroutine 20 should be programmed at the end of the machining program. It is separated from the main program by STOP M02.
Subroutine 20 is called up with CALL LBL 20 in subroutine 19.



Repetition of subroutines

It is possible to repeat subroutines with the aid of nesting.

A subroutine is called up within a program part repetition. The subroutine call is the only block in the program part repetition.
It is important to note that in a program run, the subroutine will be executed once more than the number of programmed repetitions.



Program Jump

Jump to another main program

The program management feature of the control system enables you to jump from one program to another.
Doing so enables:

- the creation of certain machining cycles (see "Cycle program call") in conjunction with parameter programming,

or

- the saving of tool files.

Jumps are programmed with the PGM CALL key.

If a program number is entered under which no program has been saved (e.g. CALL PGM 13), when you use the jump command to select the main program, the following error message will appear:
= PGM 13 UNAVAILABLE =

For program calls, no more than **four nesting levels** are permitted; that is, the nesting level is 4.

Program run example

The control system executes program 1 until the program call CALL PGM 28.

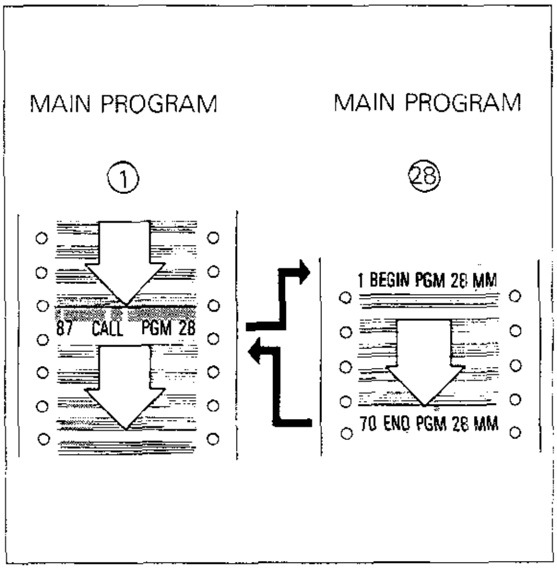
A jump is then made to program 28.

Program 28 is executed from beginning to end.

A jump is then made back to program 1.

Execution of program 1 is continued from the block following the program call.

Jumps back into the original program cannot be programmed into a program which is called up (causes excessive subroutine nesting).



Program Jump

Input

Operating mode ______________ ⇔

Dialogue initiation ______________ PGM CALL

PROGRAM NUMBER ?	▶ □	Enter the number of the program to be called up.
	▼ ENT	Press ENT.

Sample display

87 CALL PGM 28

In block 87, program 28 is called up and executed.

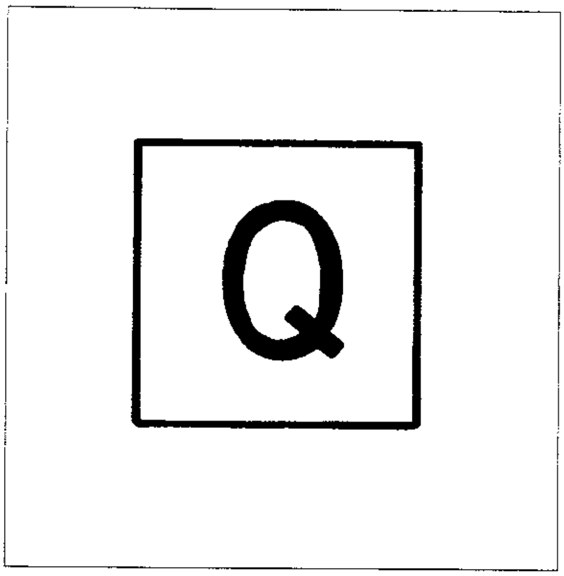
A program call can be programmed in the same manner as a cycle call, provided that the program number is specified in cycle definition 12.
This ensures that cycles created using parameter programming are handled in the same way as pre-programmed cycles (see "Cycle program call").

Parameters

Parameters

Numerical values in a program [nominal positions, feed rates, tool dimensions and entry data for cycles, software version 03 also tool numbers in the TOOL-CALL block, spindle slewing speeds and program marks (label numbers) with conditional jumps] can be replaced during program entry by a **variable parameter,** that is, by a "marker" for numerical values that are to be entered later or calculated by the control system.
During program execution, the control system then uses the numerical value provided by parameter definition.

Q parameter programs can not be switched from mm to inch nor from inch to mm because during switching the Q values (≙ label numbers) are converted by parameter comparisons.



Setting parameters

Parameters are designated with the letter Q and a number between 0 and 99. Parameters can also be entered with a negative sign. Positive signs do not have to be programmed. Parameters are entered (set) by pushing the Q key.

Parameter definition

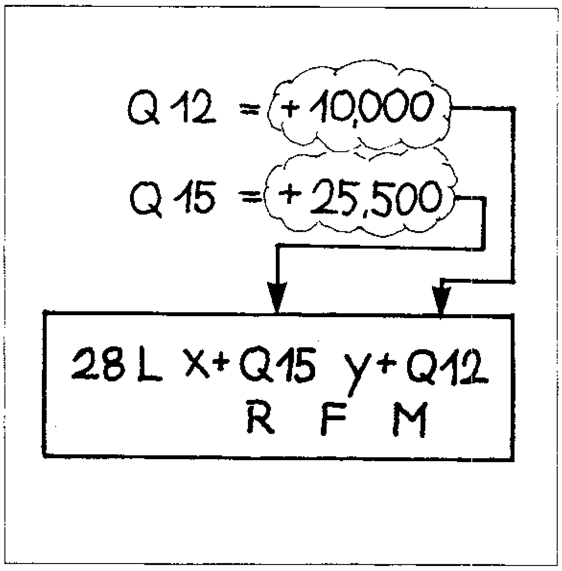
Certain numerical values can be assigned to parameters either directly or using mathematical and logical functions.

Parameter definition dialogue is initiated by pressing the Q DEF key. The **FN parameter functions** in the chart are selected using the ↑ or the ↓ key.

FN 0: ASSIGN
FN 1: ADDITION
FN 2: SUBTRACTION
FN 3: MULTIPLICATION
FN 4: DIVISION
FN 5: SQUARE ROOT
FN 6: SINE
FN 7: COSINE
FN 8: ROOT SUM OF SQUARES
FN 9: IF EQUAL, JUMP
FN 10: IF UNEQUAL, JUMP
FN 11: IF GREATER THAN, JUMP
FN 12: IF LESS THAN, JUMP
FN 13: ANGLE
FN 14: ERROR NUMBER

Parameter definition example

By specifying parameters instead of coordinates in a linear interpolation, you can create contours, e.g. ellipses, that are defined by mathematical functions. The contour is formed by several individual linear sections (see "Ellipse programming example").



In parameter programming, a step in a calculation can take between 3 and 20 ms. In cases of complicated mathematical functions and high feed rates, the tool may stop on the contour.

Parameters

Setting a parameter

Dialogue prompt, e.g.

COORDINATES ?

X	Select axis, e.g. X.
Q	Press parameter key.
□	Enter parameter number.
+/–	Enter sign if required.
ENT	Press ENT.

Sample display

```
27 L X+Q13 Y–Q2

      R F              M
```

Parameter Q13 is the marker for the numerical value of the X-coordinate. Parameter Q2 is the marker for the negative value of the Y-coordinate. For example, Q13 is assigned a value of +40.000 and Q2 a value of +19.000. The tool will move to position P (X+40.000/Y–19.000).

The parameters must be defined before they are called up. Parameters not defined at the beginning of program run will automatically be assigned a value of 0.
In the above display example, the tool would traverse to the position X0/Y0.

Selecting a parameter function

Operating mode ______________ ⇨

Dialogue initiation ______________ Q DEF

FN 0: ASSIGN ?

↓ or ↑ select required parameter function.

If the required function is in the display, e.g.

FN 9: IF EQUAL, JUMP

ENT Press ENT.

The first dialogue prompt appears in the display (see corresponding function for response).

Parameters
Parameter functions

FN 0:
ASSIGN

The function FN 0: "ASSIGN" assigns either a **numerical value** or another **parameter** to a certain parameter. Assignment is designated by an "=" sign.

Q5 = 65.432

Display:

```
18 FN 0: Q5 = + 65.432
```

FN 1:
ADDITION

The function FN 1: "ADDITION" defines certain parameter as the **sum** of two parameters, two numerical values, or a parameter and a numerical value.

Q17 = Q2 + 5.000

Display:

```
12 FN 1: Q17 = + Q2

              +     + 5.000
```

FN 2:
SUBTRACTION

The function FN 2: "SUBTRACTION" defines a certain parameter as the **difference** between two parameters, two numerical values, or a parameter and a numerical value.

Q11 = 5.000 – Q34

Display:

```
94 FN 2: Q11 = + 5.000

              –     + Q34
```

FN 3:
MULTIPLICATION

The function FN 3: "MULTIPLICATION" defines a certain parameter as the **product** of two parameters, two numerical values, or a parameter and a numerical value.

Q21 = Q1 x 60.000

Display:

```
85 FN 3: Q21 = + Q1

              *     + 60.000
```

FN 4:
DIVISION

The function FN 4: "DIVISION" defines a certain parameter as the **quotient** of two parameters, two numerical values, or a parameter and a numerical value.
(DIV: abbreviation for division).

Q12 = Q2 / 62

Display:

```
73 FN 4: Q12 = + Q2

              DIV   + 62.000
```

FN 5:
SQUARE ROOT

The function FN 5: "SQUARE ROOT" defines a certain parameter as the **square root** of a parameter or a numerical value.
(SQRT: abbreviaton for square root).

Q98 = $\sqrt{2}$

Display:

```
69 FN 5: Q98 = SQRT + 2.000
```

Parameters
Parameter functions

Program input
Example: FN 1

Operating mode ⇨

Dialogue initiation DEF ↓

FN 1: ADDITION	ENT	Press ENT to select function.

PARAMETER NUMBER FOR RESULT ?	□	Enter parameter number.
	ENT	Press ENT.

FIRST VALUE/PARAMETER ?

If a value is assigned:	□	Enter value.
	ENT	Press ENT.
If a parameter is assigned:	Q	Press parameter key.
	□	Enter parameter number.
	ENT	Press ENT.

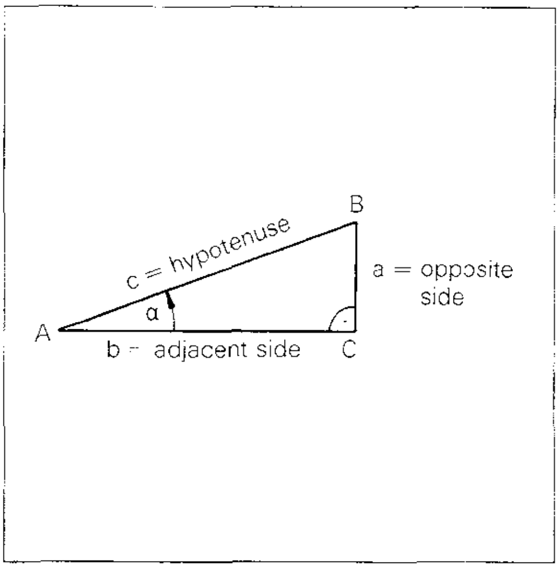
SECOND VALUE/PARAMETER ?

If a value is assigned:	□	Enter value.
	ENT	Press ENT.
If a parameter is assigned:	Q	Press parameter key.
	□	Enter parameter number.
	ENT	Press ENT.

Parameters
Parameter functions

Trigonometric functions

Sine and cosine functions establish a mathematical relationship between an angle and the side lengths of a right triangle. Trigonometric functions are programmed with
FN 6: sine, and
FN 7: cosine.
The parameter function FN13: "Angle" calculates the angle from sine and cosine values (see "Angle").



Defining trigonometric functions

sin α =

$$\frac{\text{length of side opposite}}{\text{length of hypotenuse (longest side)}} = \frac{a}{c}$$

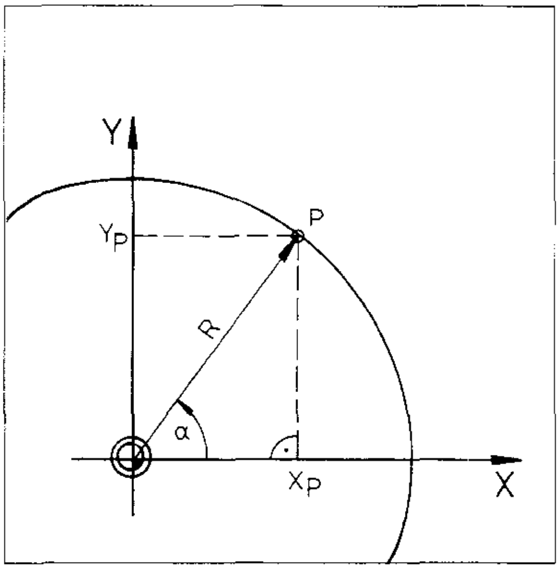
cos α =

$$\frac{\text{length of side adjacent to}}{\text{length of hypotenuse (longest side)}} = \frac{b}{c}$$

Trigonometric functions in a right triangle

XP = R x cos α

YP = R x sin α



FN 6: Sine

The function FN 6: "Sine" defines a certain parameter as the **sine** of an angle (in degrees (°)). The angle can be a numerical value or a parameter.

Q10 = sin Q8

Display:

113 FN 6: Q10 = SIN + Q8

FN 7: Cosine

The function FN 7: "Cosine" defines a certain parameter as the **cosine** of an angle (in degrees (°)). The angle can be a numerical value or a parameter.

Q81 = cos (– Q55)

Display:

911 FN 7: Q81 = COS – Q55

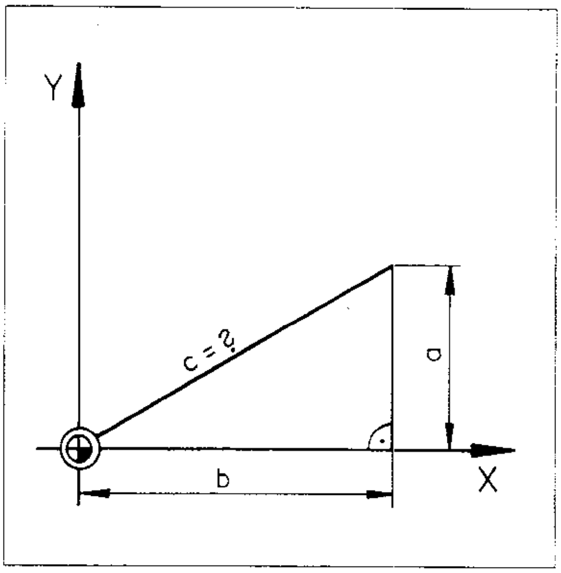
Parameters
Parameter functions

Length of a segment

The parameter function FN 8: "Root sum of squares" is used to **calculate lengths of segments** (sides) in right triangles.

According to the Pythagorean Theorem:

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



FN 8: Root sum of squares

The function FN 8: "Root sum of squares" defines a certain parameter as the **square root** of the sum of two squared values or parameters.

(**LEN** = abbreviation for length).

$$Q3 = \sqrt{30^2 + Q45^2}$$

Display:

```
56 FN 8: Q3 = + 30.000
                LEN      + Q45
```

Parameters
Parameter functions

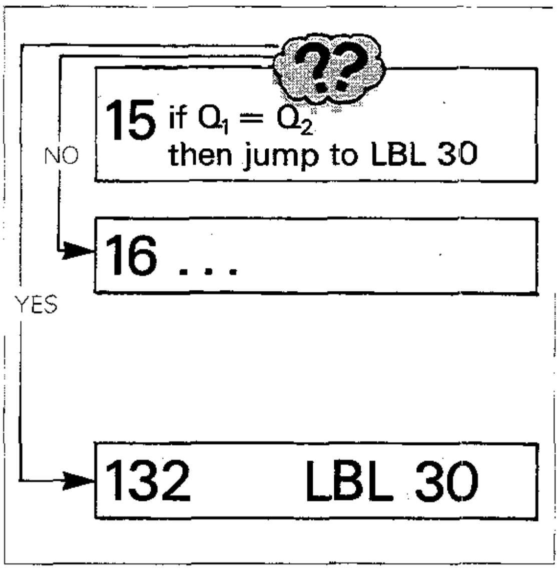
If-then jump

Parameter functions FN 9 through FN 12 can be used to compare a parameter with another parameter or with a numerical value.
Based on the result of this comparison, a jump (conditional jump) can be made to certain program marker (label).

The equations (or inequations) are:
- The first parameter is equal to a value or to a second parameter, e.g. **Q1 = Q2**
- The first parameter is not equal to a value or to a second parameter, e.g. **Q1 ≠ Q2**
- The first parameter is greater than a value or than a second parameter, e.g. **Q1 > Q2**
- The first parameter is less than a value or than a second parameter, e.g. **Q1 < Q2**

=	equal
≠	unequal
>	greater than
<	less than

If one of these equations is satisfied, a **jump** is made to a certain program marker.
If the equation is not satisfied, the program continues with the next block.



FN 9: If equal, then jump (go to)

When programming the function FN 9: "If equal, jump", a jump to a program marker is only made if a certain parameter is **equal to** another parameter or to a numerical value, then jump to LBL 30!

If: Q2 = 360,
Then jump to LBL 30!

Display:

```
47 FN 9: IF + Q2
   EQU + 360.000  GOTO LBL 30
```

Parameters
Parameter functions

Input
Example FN 9

Operating mode ⇨

Dialogue initiation Q DEF

FN 9: IF EQUAL, JUMP	▶ ENT	Press ENT to select function.

▼

FIRST VALUE/PARAMETER ?	▶ Q	Press parameter key.
	▼ □	Enter parameter number.
	▼ ENT	Press ENT.

▼

SECOND VALUE/PARAMETER ?		
If the parameter set above is to be compared with a value:	▶ □	Enter numerical value.
	▼ ENT	Press ENT.
If the parameter set above is to be compared with another parameter:	▶ Q	Press parameter key.
	▼ □	Enter parameter numbe:.
	▼ ENT	Press ENT.

▼

LABEL NUMBER ?	▶ □	Enter jump marker number.
	▼ ENT	Press ENT.

The on-screen displays are illustrated on the following page for the corresponding functions.

**FN 10:
If unequal,
jump (go to)**

When programming the function FN 10: "If unequal, jump", a jump to a program marker is only made if a certain parameter is **unequal to** another parameter or to a numerical value.

(**NE** = abbreviation for **not equal**).

**If Q3 ≠ Q10,
then jump to LBL 2!**

Display:

```
38 FN 10: IF + Q3

      NE + Q10  GOTO LBL 2
```

**FN 11:
If greater than,
jump (go to)**

When programming the function FN 11: "If greater than, jump", a jump to a program marker is only made if a certain parameter is **greater than** another parameter or to a numerical value.

(**GT** = abbreviation for **greater than**).

**If Q8 > 360,
then jump to label 17!**

Display:

```
28 FN 11: IF + Q8

      GT + 360.000  GOTO LBL 17
```

**FN 12:
If less than,
jump (go to)**

When programming the function FN 12: "If less than, jump", a jump to a program marker is only made if a certain parameter is **less than** another parameter or to a numerical value.

(**LT** = abbreviation for **less than**).

**If Q6 < Q5,
then jump to LBL 3!**

Display:

```
24 FN 12: IF + Q6

      LT + Q5  GOTO LBL 3
```

Parameters
Parameter functions

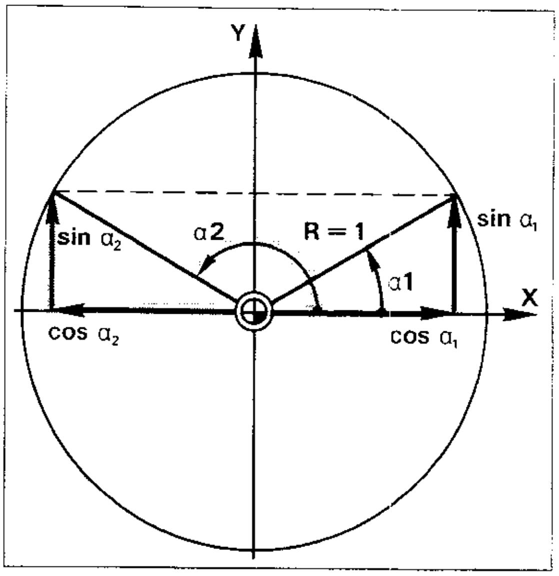
Angles from trigonometric functions

If the value of the trigonometric function sin α is known, there are always two angles that can satisfy the comparison.
Example: $\sin\alpha = 0.5$ $\alpha_1 = 30°$
$\alpha_2 = 150°$

The second trigonometric function cos α is needed to determine α. If the value of cos α is also known, then there is a plain solution for:
$\sin\alpha = + 0.5$
$\cos\alpha = + 0.866$ $\alpha = + 30°$
accordingly:
$\sin\alpha = + 0.5$
$\cos\alpha = - 0.866$ $\alpha = + 150°$
The control system calculates the angle α using the tangent function

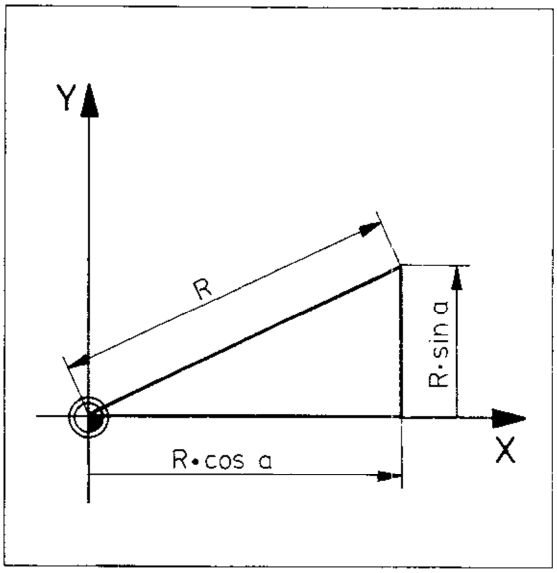
$$\tan\alpha = \frac{\sin\alpha}{\cos\alpha}\text{, and therefore}$$

$$\text{arc tan}\ \frac{\sin\alpha}{\cos\alpha} = \alpha$$



Angle of lines in a right triangle

In place of the angle functions sin α and cos α, the legs of a right triangle can also be used for angle determination. The legs of the right triangle correspond to the angle functions sin α and cos α multiplied with the length R of the hypotenuse.



FN 13: Angle

The function FN 13: "Angle" assigns an angle to a parameter using the values from the sine and cosine functions.
In place of the angle functions the legs of a right triangle can also be entered.

If the value 0 is entered for cos α, the control system calculates the angle α from the preprogrammed sin α. When sin α = 0 and cos α = 0 are entered, the following error message will appear:
= ARITHMETIC ERROR =

sin α = + 0.5
cos α = + 0.866

Display:

25 FN 13: Q11 = + 0.5

ANG + 0.866

k = 10 **10 x sin α = + 5**
10 x cos α = + 8.660

Display:

25 FN 13: Q11 = + 5

ANG + 8.660

Parameters
Parameter programming (Example)

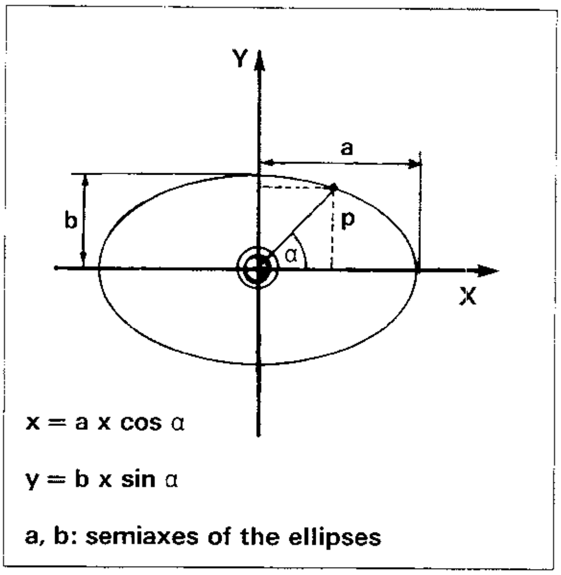
Programming with parameters will be demonstrated using an ellipse as an example.

Geometry

The **ellipse** is described according to the following shape (mathematical parameter shape of the ellipse):

$x = a \times \cos \alpha$
$y = b \times \sin \alpha$

This means that every angle α has both an X-coordinate and a Y-coordinate.
If you begin at α = 0° and increase α in small increments to 360°, you will get a large number of points on an ellipse. A closed contour is formed when these points are connected by straight lines.



Parameter definition

The program essentially consists of four parts:
- parameter definition,
- positioning (linear interpolation) for milling the ellipse,
- increasing the angular increment
- parameter comparison and continued program execution until the ellipse is complete.

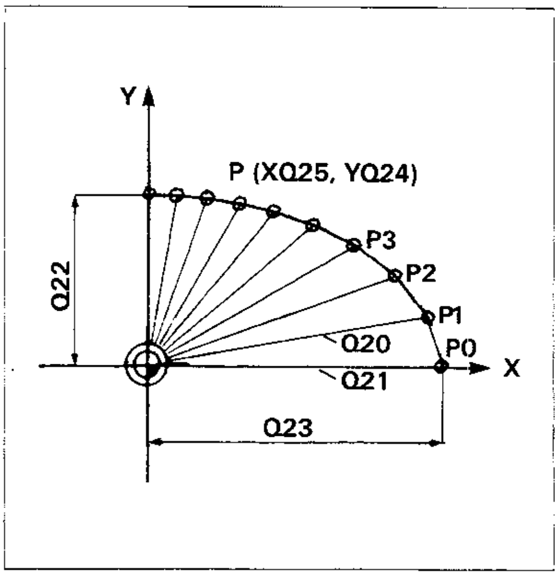
The following are defined as parameters:
- **Angular increment Q20:** the angle should increase in increments of 2°: Q20 = + 2.000
- **Starting angle Q21:** the first point on the contour has an angle of 0°: Q21 = 0.000
- **Semiaxis in X-direction Q23:** Q23 = +50.000
- **Semiaxis in Y-direction Q22:** Q22 = +30.000
- **X-coordinate Q25:** the numerical value of the X-coordinate is assigned to parameter Q25.
- **Y-coordinate Q24:** the numerical value of the Y-coordinate is assigned to parameter Q24.

Parameters Q25 and Q24 are defined according to the above formula:

(X) Q25 = Q23 ✱ cos Q21;
(Y) Q24 = Q22 ✱ sin Q21.

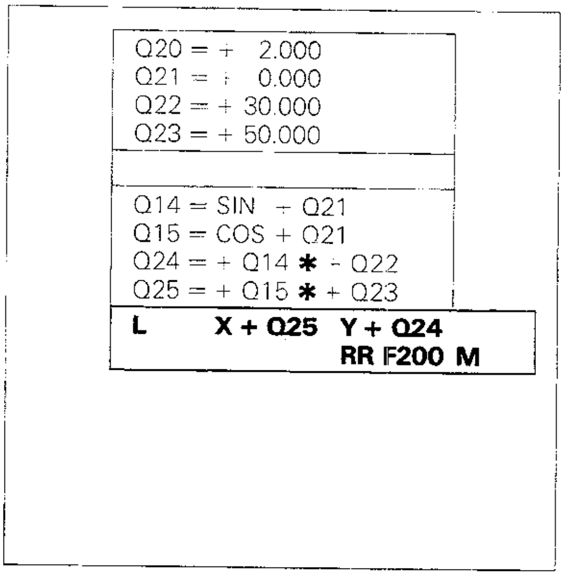
Both comparisons must be rewritten because they cannot be entered in this form, therefore:

first: Q14 = sin Q21
Q15 = cos Q21
then: Q24 = Q14 ✱ Q22
Q25 = Q15 ✱ Q23



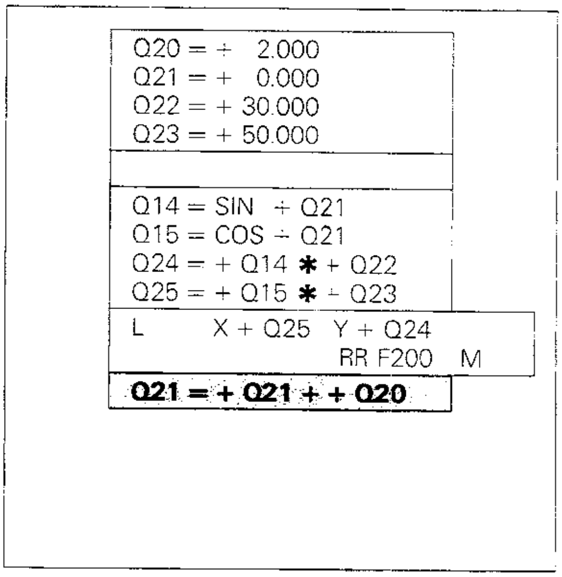
```
Q20 = +  2.000
Q21 = +  0.000
Q22 = + 30.000
Q23 = + 50.000

Q14 = SIN  + Q21
Q15 = COS  + Q21
Q24 = + Q14 ✱ + Q22
Q25 = + Q15 ✱ + Q23
```

Parameters
Parameter programming (Example)



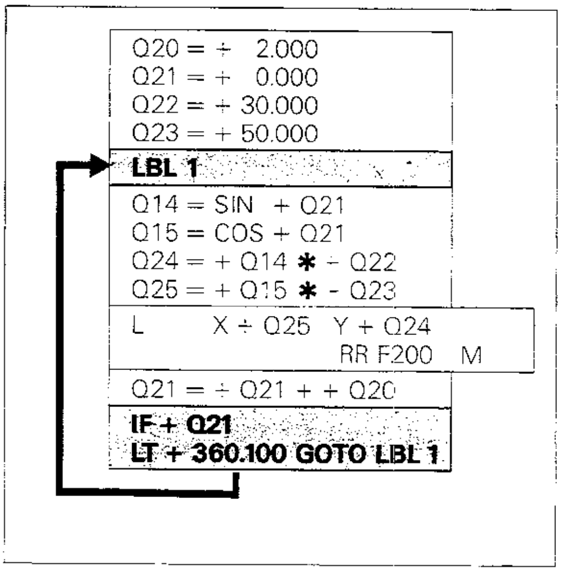
```
Q20 = + 2.000
Q21 = + 0.000
Q22 = + 30.000
Q23 = + 50.000

Q14 = SIN + Q21
Q15 = COS + Q21
Q24 = + Q14 * + Q22
Q25 = + Q15 * + Q23
L       X + Q25  Y + Q24
                 RR F200  M
```

Positioning block Milling of the ellipse is programmed in this block with linear interpolation.



```
Q20 = + 2.000
Q21 = + 0.000
Q22 = + 30.000
Q23 = + 50.000

Q14 = SIN + Q21
Q15 = COS + Q21
Q24 = + Q14 * + Q22
Q25 = + Q15 * + Q23
L       X + Q25  Y + Q24
                 RR F200  M
Q21 = + Q21 + + Q20
```

Increasing the angular increment New angle Q21 = previous angle Q21 + angular increment Q20



```
Q20 = + 2.000
Q21 = + 0.000
Q22 = + 30.000
Q23 = + 50.000
LBL 1
Q14 = SIN + Q21
Q15 = COS + Q21
Q24 = + Q14 * + Q22
Q25 = + Q15 * + Q23
L       X + Q25  Y + Q24
                 RR F200  M
Q21 = + Q21 + + Q20
IF + Q21
LT + 360.100 GOTO LBL 1
```

Parameter comparison and program repetition Program repetition requires that a jump marker (label) must be set prior to parameter definition for Q14 and Q15: LBL 1.

The following requirements must be met for program repetition:

If the angle Q21 is less than 360.1° (the angle must be larger then 360°, but less than 360° + angular increment), then jump (GOTO) LBL 1.

IF + Q21
LT + 360.100 GOTO LBL 1

Parameters
Special functions

FN 14: Error number

The parameter function FN 14: "Error number" is used to call up error messages and dialogue from the PLC-Eprom. You call up by entering an error number from 0 to 499.
The messages are allocated as follows:

Error number	On-screen display
0 ... 299	ERROR 0 ... ERROR 299
300 ... 399	PLC ERROR 01 ... PLC ERROR 99 (or dialogue specified by machine tool manufacturer)
400 ... 499	Dialogue 0 ... 99 for user cycles

Display:

28 FN 14: ERROR = 100

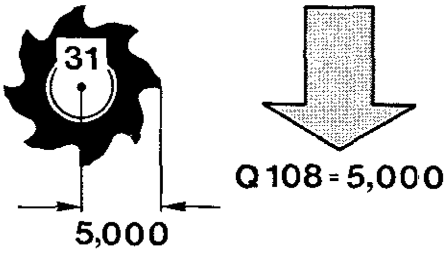
Q108 Tool radius

The control system stores the radius of the most recently activated tool under parameter Q108.

The radius can then be used for parameter calculations and comparisons.

The radius of the most recently activated tool is always assigned to parameter 108.

```
112 TOOL DEF 31 L = 0.700
                R = 5.000
113 TOOL CALL 31 Z
               S + 112.000
```



Q109 Tool axis
(Beginning with software version 02)

Various machines have the X-, Y- or Z-axis optionally as tool axis. With these machines it is advantageous to be able to extract the current tool axis in the processing program; in this way, for example, program branches are possible with manufacturer cycles.

The control sets the current tool axis under the parameter Q109.

Current tool axis	Parameter
no tool axis is called	Q109 = −1
X-axis is called	Q109 = 0
Y-axis is called	Q109 = 1
Z-axis is called	Q109 = 2
IV-axis is called	Q109 = 3

Parameters
Special functions

Q110
Spindle on/off
(Beginning with software version 03)

The parameter Q110 indicates the last M-function issued for the spindle direction of rotation.

M-function	Parameter
no M-function	Q110 = -1
M03 (spindle-on clockwise)	Q110 = 0
M04 (spindle-on counterclockwise)	Q110 = 1
M05, in case M03 was issued earlier	Q110 = 2
M05, in case M04 was issued earlier	Q110 = 3

Q111
Coolant on/off
(Beginning with software version 03)

The parameter Q111 indicates whether the coolant has been switched on or off.

It means:

M08 coolant switched on	Q111 = 1
M09 coolant switched off	Q111 = 0

Q112
Overlap factor
(Beginning with software version 03)

The program parameter Q112 contains the entry value of the pocket milling overlap factor (machine parameter 93). The overlap factor entered for pocket milling can be applied in Q parameter programs.

Q113
mm/inch measures
(Beginning with software version 03)

The parameter Q113 indicates whether the NC-program is in mm or inches.

It means:

in mm	Q113 = 0
in inches	Q113 = 1

Parameters
Special functions

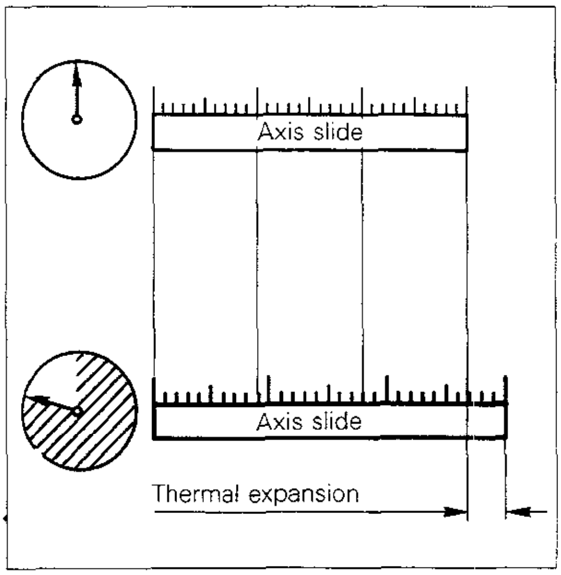
Transferring parameters to PLC-NC

TNC 151 B/Q and TNC 155 B/Q control systems can transfer Q-parameter values from an integrated PLC to an NC program. Parameters Q100 to Q107 are used for transferring values.

A possible application is compensating for the effects of temperature on the machine.

Compensating for thermal displacement

During periods of extended machine use, thermal displacement of the machine and the workpiece effects machining precision. Devices that measure thermal displacement transfer compensation values to the control system to remedy the situation. These values can be used in a machining program, e.g. to shift the datum. This type of measuring device is available from Firma Testoterm in 7825 Lenzkirch/Schwarzwald in the Federal Republic of Germany.



Example

Thermal expansion of the machine should be offset with a datum shift.
Thermal compensation values for the machine axes are stored under parameter numbers Q100 (X-axis) and Q101 (Y-axis) and Q102 (Z-axis). The control system requests compensation values via an M-function determined by the manufacturer (e.g. M70).

Your machine tool manufacturer can tell you if your machine is capable of transferring parameters from an integrated PLC.

```
84  L

              R    F    M70

85  CYCL     DEF  7.0  DATUM

86  CYCL     DEF  7.1  X + Q100

87  CYCL     DEF  7.2  Y + Q101

88  CYCL     DEF  7.3  Z + Q102
```

Parameters
Special functions

as of software version 07:

FN 15: PRINT

with the parameter function FN 15: PRINT current values of Q parameters can be output via the V.24 interface. A maximum of six parameters can be indicated depending on the PRINT command.

Instead of Q-Parameters, numerical values between 0 and 200 can also be entered.
These numbers call error messages and dialog texts that are stored in the PLC-EPROM or the ASCII sign ETX. The allocation of numerical values to the texts is as follows:

Numerical value	Output via the V.24 interface
0 ... 99	stored error messages in the PLC
100 ... 199	Texts/Dialogs for the user cycles
200	"ETX"

Display:

29 FN 15: PRINT Q1/Q2/Q3/Q4/Q5/Q6

Display:

120 FN 15: PRINT 12/18 8/4/10/55

Parameters for programmable touch probe function: Q115 ... Q118

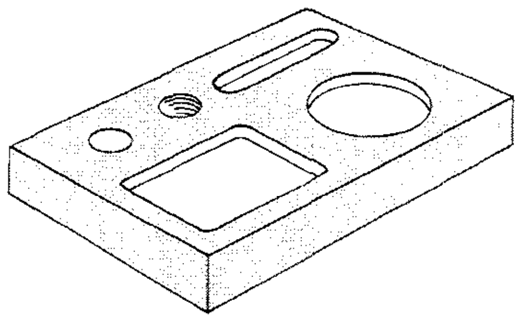
The parameters Q115 to Q118 contain the measured values that have been determined via the programmable touch probe function "workpiece surface as reference surface":

Q115	measured value X axis
Q116	measured value Y axis
Q117	measured value Z axis
Q118	measured value 4th axis

Canned cycles
Introduction

Canned cycles

In order to simplify and speed up programming, frequently re-occurring machining procedures and certain coordinate transformations can be pre-programmed in the form of fixed or "canned" cycles. Examples are the milling of pockets or zero point offsets.
Other programs can also be called up via cycles.



Cycle definition

Through cycle definition, the control system receives the data necessary to execute the cycle, e.g. the side length of the pocket etc.
The dialogue for cycle definition is initiated by pressing the CYCL DEF key. The cycle is then selected with the ↓ and ↑ keys, or (beginning with software version 02) with GOTO □ and the cycle number.

Available cycles

Cycles 1 to 6 and 14 to 16 are **machining cycles,** i.e. they are used to carry out machining procedures on a workpiece. Cycle 9 can be used to program a dwell time and cycle 12 to call up a program. A specified spindle orientation can be programmed with cycle 13 (optional). The remaining cycles are used for various **coordinate transformations.**

Cycles for coordinate transformation terminate path compensation.

CYCL DEF 1 Peck drilling	Machining cycles
CYCL DEF 2 Tapping	
CYCL DEF 3 Slot milling	
CYCL DEF 4 Pocket milling	
CYCL DEF 5 Circular pocket	
CYCL DEF 7 Datum shift	Coordinate transformation
CYCL DEF 8 Mirror image	
CYCL DEF 10 Rotation	
CYCL DEF 11 Scaling factor	
CYCL DEF 9 Dwell	
CYCL DEF 12 Program call	
CYCL DEF 13 Spindle orientation (optional)	
CYCL DEF 6 Roughing out	Cycles for machining pockets with various contours
CYCL DEF 14 Contour	
CYCL DEF 15 Pre-drilling	
CYCL DEF 16 Contour milling	

Manufacturer's cycles

Additional cycles can be stored at cycle numbers 68 to 99. Contact your machine-tool manufacturer or supplier for information.

Cycle call

A cycle call in a program causes the previously defined **machining cycle** to be run.
Coordinate transformations, dwell time and the **contour** cycle do not require a separately programmed cycle call, they are active immediately following cycle definition.
Three programming options are available for calling a cycle:

- via a CYCL CALL block,
- via auxiliary function M99,
- via auxiliary function M89 (depending on specified machine parameters).

A call via M89 is modal, meaning that the previously defined machining cycle is called up in each subsequent positioning block.
M89 is cancelled or deleted by entering M99 or by a CYCL CALL block.

Only the last defined machining cycle can be accessed via a cycle call.

Canned cycles
Cycle definition
Cycle call

Defining a cycle

Operating mode ______ ⇨

Dialogue initiation ______ CYCL DEF

CYCL DEF 1 PECKING

Look for cycle name	▶ ↓ ↑	
Select cycle via cycle number	▶ GOTO	with GOTO
	▼ □	Enter cycle number.
	▼ ENT	Transfer to memory.

If the desired cycle is in the display e.g.

CYCL DEF 4 POCKET MILLING	▶ ENT	transfer cycle.

The first dialog prompt for the cycle selected appears on the display.
(For the correct response see the cycle definition.)

Calling a cycle

Operating mode ______ CYCL DEF

Dialogue initiation ______ CYCL CALL

AUXILIARY FUNCTION M ?	▶ □	Specify auxiliary function if required.
	▼ ENT	Press ENT.

Sample display

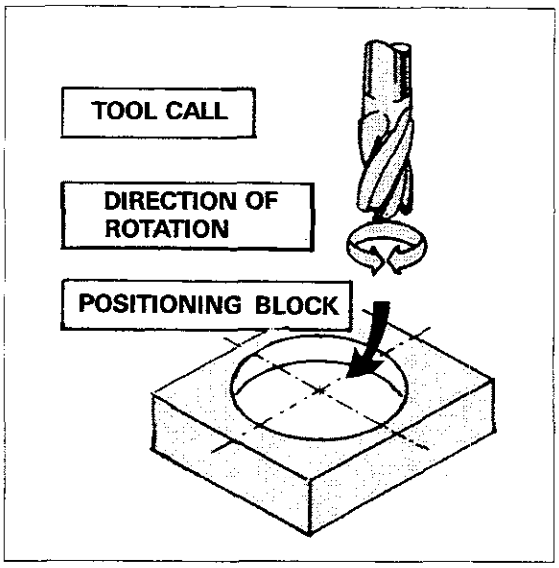
95 CYCL CALL	The last defined cycle is called.
M03	The spindle rotates clockwise.

Canned cycles
Machining cycles
Preparations

Requirements

The following functions must be programmed **before a cycle is called:**

- **tool call:** to define **spindle axis** and **spindle speed,**
- **auxiliary function:** to indicate the **direction of spindle rotation,**
- **positioning block for starting position:** for the machining cycle.



Error messages

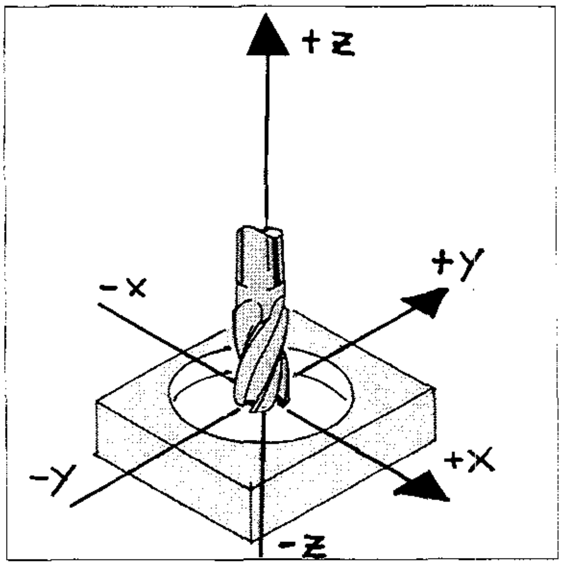
If **no tool call** is specified, the error message
= TOOL CALL MISSING =
is displayed.

If **no spindle direction** is specified, the error message
= SPINDLE ROTATES MISSING =
is displayed.

Dimensions

Tool-axis dimensions in cycle definition are always based on the **starting position** of the tool and interpreted as incremental dimensions.

It is not necessary to press the I key.

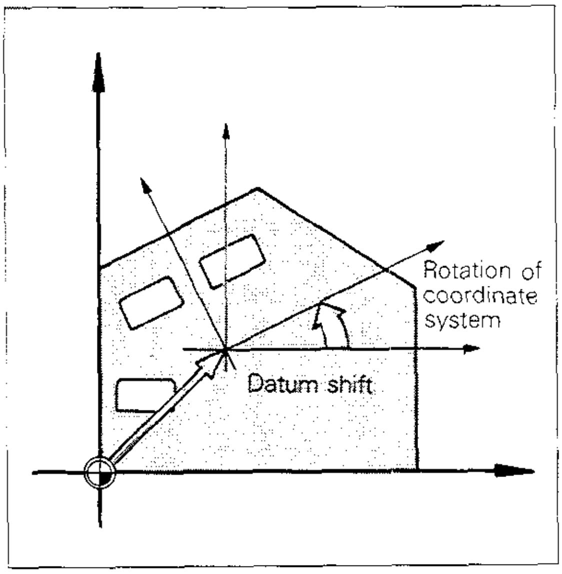


Machining cycles (in contrast to cycles for coordinate transformation) must always be called up for execution.

Canned cycles
Coordinate transformation

General information

A coordinate transformation modifies the coordinate system defined by the "Workpiece datum" function. These cycles are active immediately after definition and do not have to be called separately.



Cancelling a cycle

Coordinate transformations remain in effect until cancelled. This is done by defining a new cycle, in which the original condition is programmed, by programming the auxiliary function M02, M30 or via the last block
END PGM ... MM (depending on specified machine parameters).

Canned cycles
Peck drilling

Input data

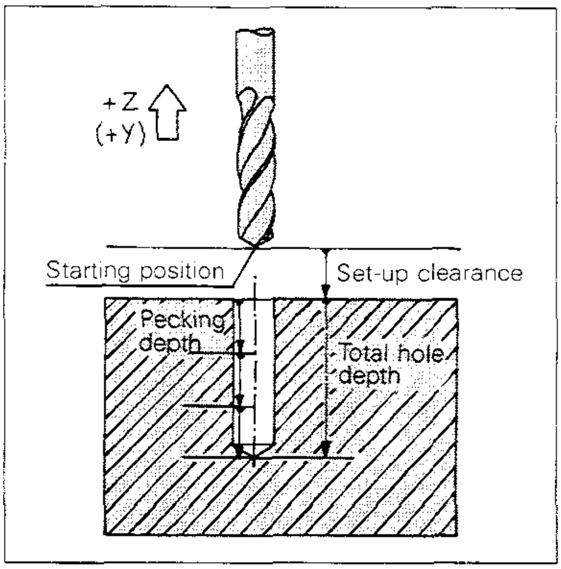
Set-up clearance: safety clearance between tool tip (at starting position) and workpiece surface.
Prefix sign:
- in positive axis direction +
- in negative axix direction –

Total hole depth: distance between workpiece surface and bottom of hole (tip of drill taper). See "Set-up clearance" for sign.

Pecking depth: infeed per cut, i.e. the amount by which the tool advances for each cut. See "Set-up clearance" for sign.

Dwell time: amount of time the tool remains at the total hole depth for chip breaking.

Feed rate: traversing speed of tool during machining operations.

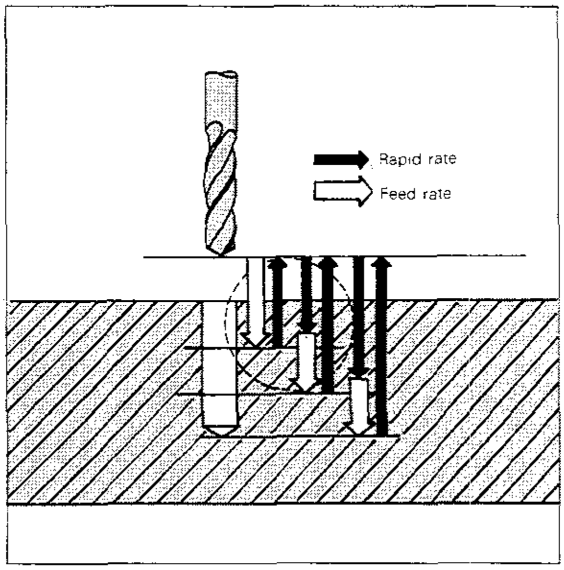


Procedure

From its **starting position,** the tool penetrates the workpiece to the first **pecking depth,** advancing at the programmed **feed rate.** Upon reaching the first pecking depth, the tool is retracted at rapid rate to its starting position and again advanced to the first pecking depth, taking the advanced stop distance into account.

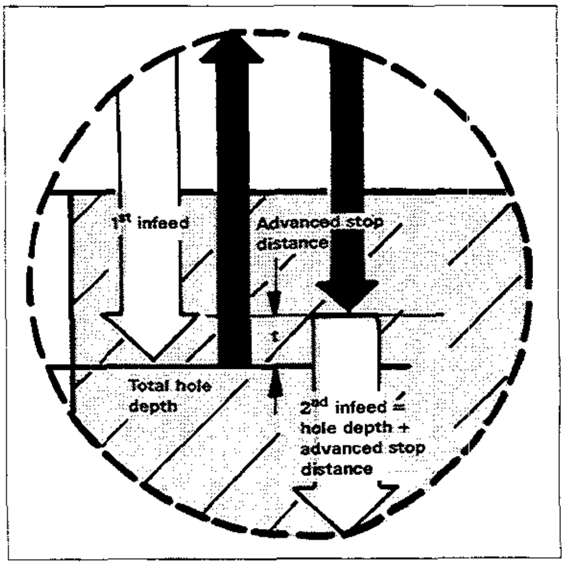
The tool then advances at the programmed feed rate to the next pecking depth, returns to the starting position etc.

The alternating drilling and retracting procedure is repeated until the programmed **total hole depth** is reached. At the end of the cycle, after the programmed dwell, the tool returns at rapid rate to the starting position.



Advanced stop distance

The advanced stop distance t is computed automatically by the control system:
- At a total hole depth of up to 30 mm: t = 0,6 mm;
- At a total hole depth exceeding 30 mm, the following equation applies:
  t = total hole depth/50; however, the maximum advanced stop distance is limited to 7 mm:
  t_{max} = 7 mm.



Canned cycles
Peck drilling

Cycle definition

Operating mode ______ ◇

Dialogue initiation ______ CYCL DEF or GOTO □ 1 ENT

CYCL DEF 1 PECKING	▶ ENT	Press ENT.
SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	+/−	with correct sign.
	ENT	Press ENT.
TOTAL HOLE DEPTH ?	▶ □	Specify hole depth.
	+/−	with correct sign.
	ENT	Press ENT.
PECKING DEPTH ?	▶ □	Specify infeed per cut.
	+/−	with correct sign.
	ENT	Press ENT.
DWELL TIME IN SECONDS ?	▶ □	Specify dwell at hole bottom.
	ENT	Press ENT.
FEED RATE ? F =	▶ □	Specify feed rate for pecking.
	ENT	Press ENT.

The set-up (safety) clearance, total hole depth and pecking depth (infeed per cut) must have the same sign, otherwise the error signal CYCL-PARAMETER SIGN FALSE will appear.

Remarks

- The total hole length can be programmed equal to the pecking depth. The tool then travels to the programmed depth in one operation (e.g. when centering).
- The total hole depth need not be a multiple of the pecking depth; in the final feed step only the remainder of the distance to the total hole depth is machined.
- The pecking depth can mistakenly (e.g. through a typing error) be entered as greater than the total hole depth. The control will in no case drill deeper than the programmed total hole depth.

This remark is also valid for all other machining cycles.

Canned cycles
Peck drilling

Sample display

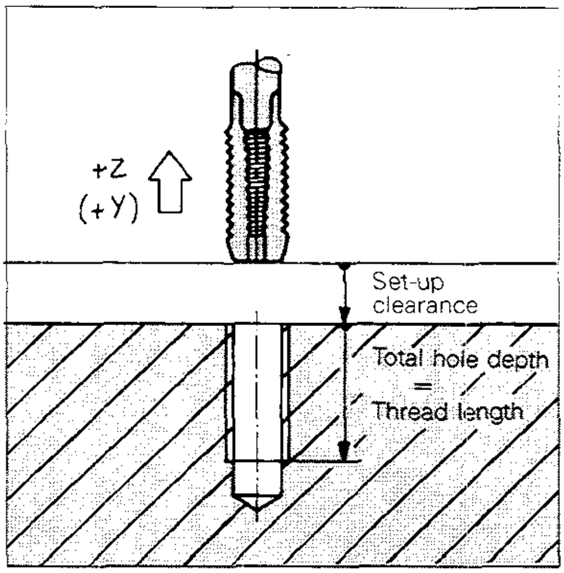
110 CYCL DEF 1.0 PECKING	Cycle definition "Pecking" occupies 6 program blocks.
111 CYCL DEF 1.1 SET-UP –2.000	Safety clearance
112 CYCL DEF 1.2 DEPTH –30.000	Total hole depth
113 CYCL DEF 1.3 PECKG –12.000	Pecking depth
114 CYCL DEF 1.4 DWELL 1.000	Dwell
115 CYCL DEF 1.5 F 80	Feed rate

Canned cycles
Tapping

Cycle

A **floating tap holder** is required for tapping. It must be able to offset the tolerances between feed rate and spindle speed, as well as spindle deceleration once the position has been reached.

When a cycle is called, **spindle speed override is inactive,** the **feed rate override** is active only within a **limited range.** The limits are determined by the machine parameters defined by the machine manufacturer.



Input data

Set-up clearance: (see Cycle 1)
(guide value: approx. 4 x thread pitch)

Total hole depth (= thread length): distance between workpiece surface and end of thread. See "Set-up clearance" for sign.

Dwell time: period of time between reversal of spindle rotation and retraction of the tool.

Contact your machine manufacturer to determine the input value for dwell.

Feed rate: traversing speed of the tool for thread cutting.

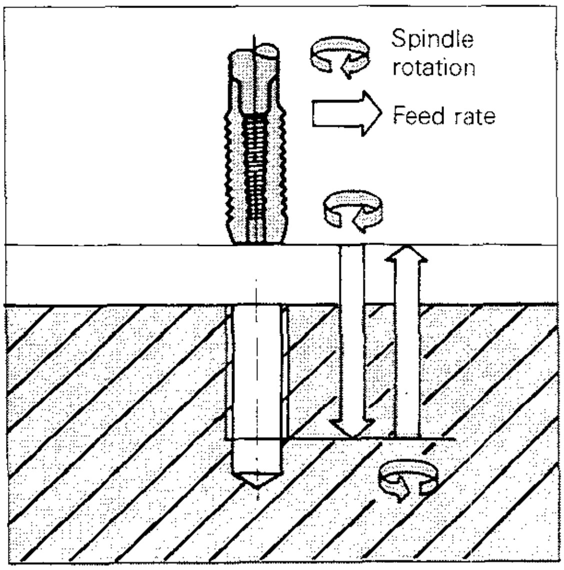
The feed rate for the tapping cycle is computed by the following equation:

F = S x P

F: feed rate
S: spindle speed in rpm
P: thread pitch

Procedure

The thread is cut in a single operation. Once the tool reaches the **total hole depth,** spindle rotation is reversed after a period of time specified in the machine parameters. At the end of the programmed **dwell time,** the tool is retracted to the starting position.



Canned cycles
Tapping

Cycle definition

Operating mode ______ ⇔

Dialogue initiation ______ CYCL DEF ↓ or GOTO □ 2 ENT

CYCLE DEF 2 TAPPING	▶ ENT	Press ENT.

SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	+/−	with correct sign.
	ENT	Press ENT.

TOTAL HOLE DEPTH ?	▶ □	Specify thread depth.
	+/−	with correct sign.
	ENT	Press ENT.

DWELL TIME IN SECONDS ?	▶ □	Specify dwell between spindle reversal and spindle retraction.
	ENT	Press ENT.

FEED RATE ? F=	▶ □	Enter calculated feed rate.
	ENT	Press ENT.

Enter set-up clearance and total hole depth with the same sign.

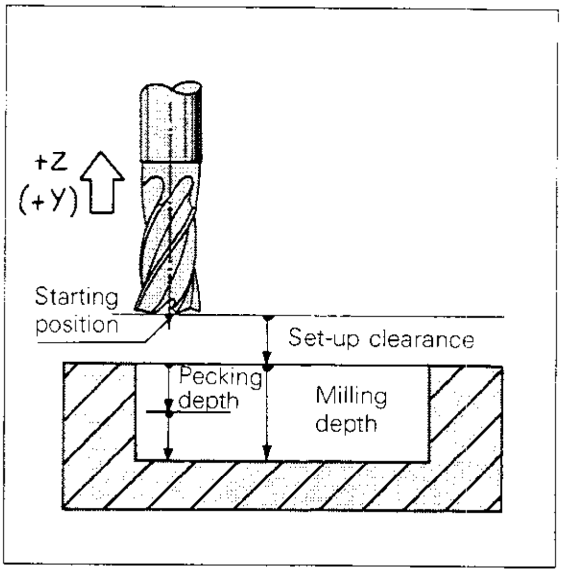
Sample display

80 CYCL DEF 2.0 TAPPING	Cycle definition "Tapping" occupies 5 program blocks.
81 CYCL DEF 2.1 SET-UP −2.000	Set-up clearance
82 CYCL DEF 2.2 DEPTH −30.000	Thread depth
83 CYCL DEF 2.3 DWELL 0.000	Dwell time
84 CYCL DEF 2.4 F 160	Feed rate

Canned cycles
Slot milling

Cycle

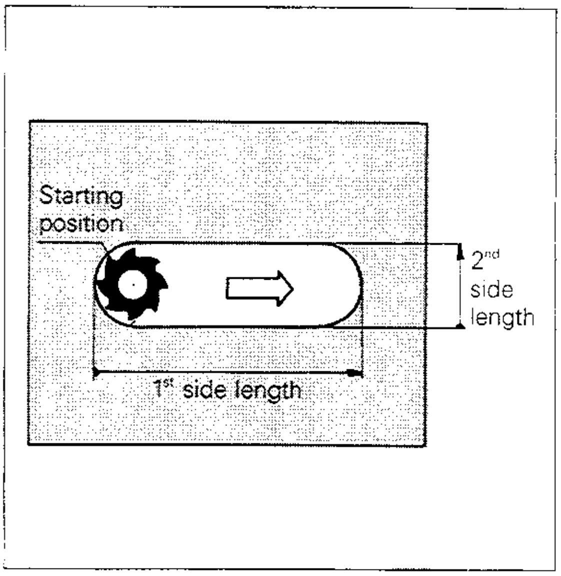
"Slot milling" is a combined roughing/finishing cycle.
The slot is parallel to an axis of the current coordinate system. The coordinate system may have to be rotated accordingly (see cycle 10: "Rotating the coordinate system").



Input data

Set-up clearance: see cycle 1.
Milling depth (= slot depth): distance between workpiece surface and bottom of slot. See "Set-up clearance" for sign.
Pecking depth: amount by which tool penetrates workpiece. See "Set-up clearance" for sign.
Feed rate for vertical feed: traversing speed of tool when penetrating workpiece.

1st side length: length of slot (finished size). The programmed sign must correspond to the milling direction:
To mill from the starting position in the positive axis direction: positive sign.
To mill from the starting position in the negative axis direction: negative sign.
2nd side length: width of slot (finished size). The sign is always positive.



The width of the slot must be larger than the diameter of the cutter.

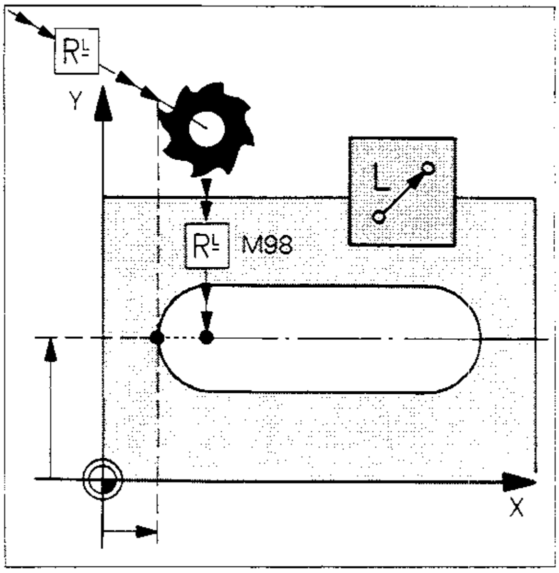
Feed rate: traversing speed of the tool in the machining plane.

Starting position

The starting position for the "Slot milling" cycle must be approached accurately, taking the tool radius into account.

Contour approach with a linear interpolation block

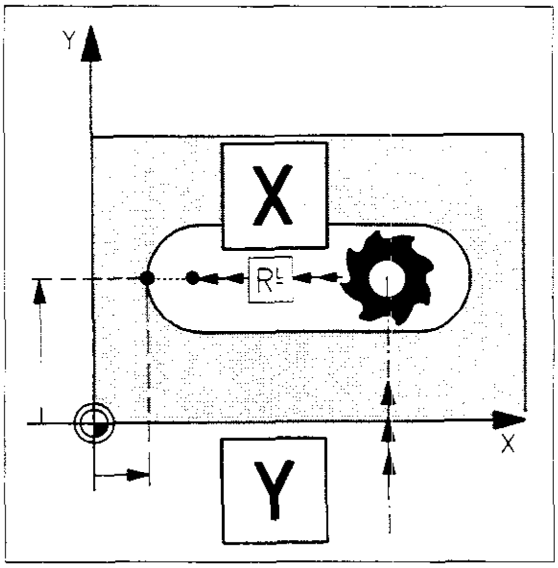
The slot contour is approached at right angles to the longitudinal, with radius compensation RL/RR and auxiliary function M98.



Canned cycles
Slot milling

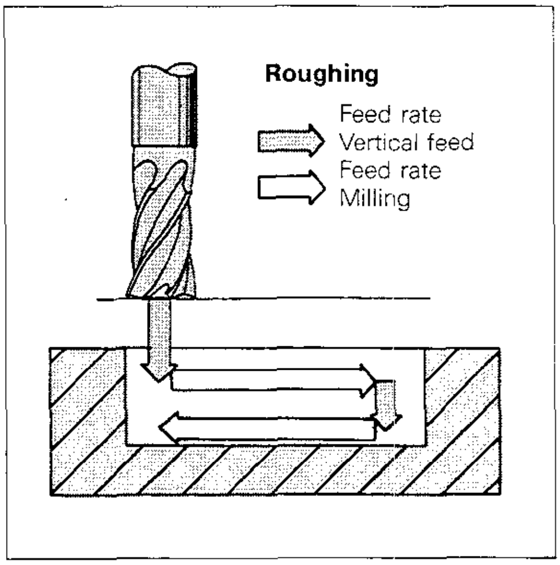
Contour approach with paraxial positioning blocks

The slot contour is approached in the longitudinal direction with radius compensation R–/R+.



Procedure

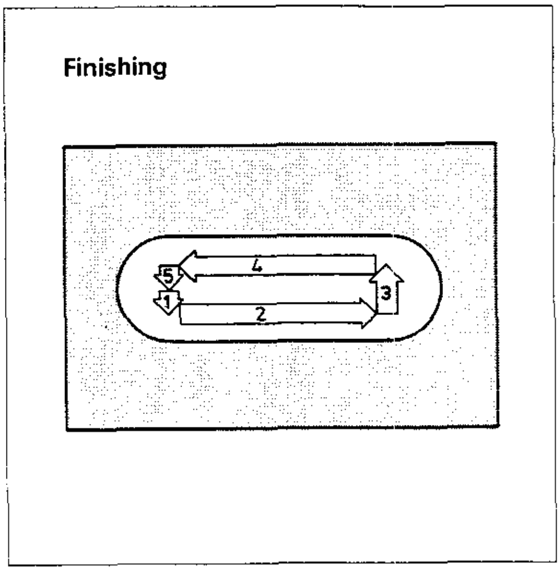
Roughing: The cutter penetrates the workpiece from the **starting position** and mils the slot in the longitudinal direction. After vertical feed at the end of the slot, milling is resumed in the opposite direction.
The procedure is repeated until the programmed **milling depth** is reached.



Procedure

Finishing: The control system advances the cutter laterally, at the bottom of the slot, by the amount of the remaining finishing cut and machines the contour with **down-cut** milling. The tool then returns at rapid rate to the set-up clearance. If the number of infeeds was odd, the cutter moves along the slot to the starting position, maintaining the set-up clearance.

The finishing cut produces a short straight section at the ends of the slot.



Notes:

Canned cycles
Slot milling

Cycle definition

Operating mode ______ ⇨

Dialogue initiation ______ CYCL DEF ↓ or GOTO □ 3 ENT

CYCL DEF 3 SLOT MILLING	▶ ENT	Press ENT.

▼

SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	▼ +/–	with correct sign.
	▼ ENT	Press ENT.

▼

MILLING DEPTH ?	▶ □	Specify milling depth.
	▼ +/–	with correct sign.
	▼ ENT	Press ENT.

▼

PECKING DEPTH ?	▶ □	Specify infeed per cut.
	▼ +/–	with correct sign.
	▼ ENT	Press ENT.

▼

FEED RATE FOR PECKING ?	▶ □	Specify feed rate for vertical feed.
	▼ ENT	Press ENT.

▼

FIRST SIDE LENGTH ?	▶ X	Specify longitudinal axis of slot, e.g. X.
	▼ □	Specify length of slot.
	▼ +/–	with correct sign.
	▼ ENT	Press ENT.

▼

Notes:

Canned cycles
Slot milling

SECOND SIDE LENGTH ?	Y	Specify axis for slot width, e.g. Y.
		Enter width of slot with positive sign.
	ENT	Press ENT.

FEED RATE ? F =		Specify feed rate for slot milling.
	ENT	Press ENT.

Enter set-up clearance, milling depth and infeed per cut (pecking depth) with the same sign.

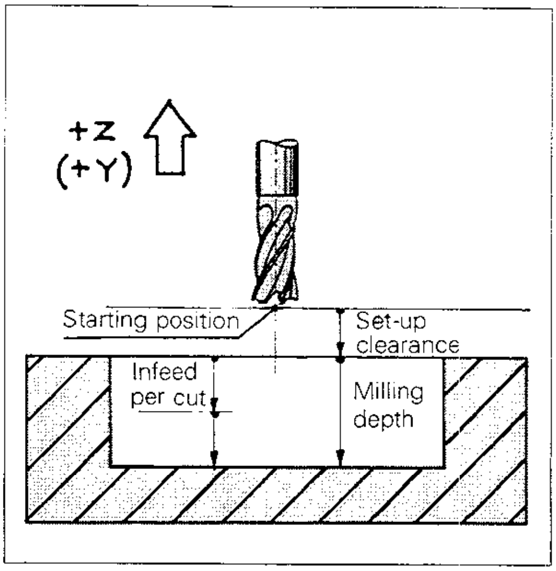
Sample display

100 CYCL DEF 3.0 SLOT MILLING	Cycle definition "Slot milling" occupies 7 program blocks.
101 CYCL DEF 3.1 SET-UP −2.000	Set-up clearance
102 CYCL DEF 3.2 DEPTH −40.000	Milling depth
103 CYCL DEF 3.3 PECKING −20.000	Infeed per cut
F 80	Feed rate for vertical feed
104 CYCL DEF 3.4 X −120.000	Length of slot
105 CYCL DEF 3.5 Y +21.000	Width of slot
106 CYCL DEF 3.6 F 100	Feed rate

Canned cycles
Pocket milling

Cycle

The machining cycle "Pocket milling" is a **roughing cycle.**
The sides of the pockets are parallel to the axes of the current coordinate system. The coordinates system may have to be rotated accordingly (see cycle 10: "Rotating the coordinate system").

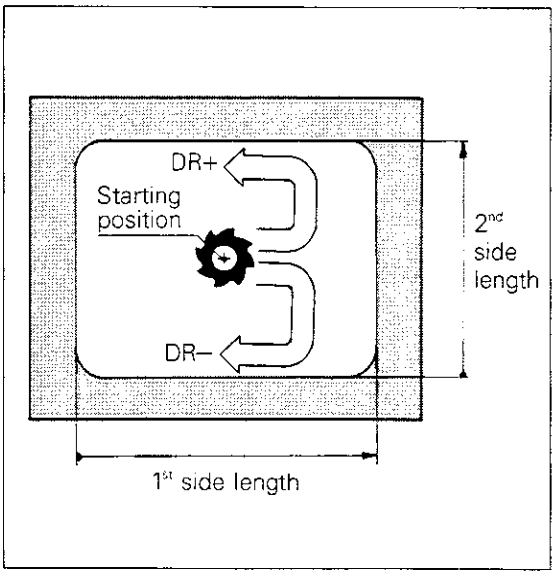


The radius at the corners of the pocket is determined by the cutter radius. There is no circular motion in the corners of the pocket.

Input data

Set-up clearance: see cycle 1.
Milling depth (= pocket depth): distance between workpiece surface and pocket bottom. See "Set-up clearance" for sign.

Pecking depth: infeed per cut, i.e amount by which tool penetrates workpiece. See "Set-up clearance" for sign.
Feed rate for vertical feed: traversing speed of tool when penetrating workpiece.
1st side length: length of pocket parallel to first main axis of machining plane. Sign is always positive.
2nd side length: width of pocket. The sign is also positive.
Feed rate: traversing speed of tool in machining plane.
Rotation: Direction of rotation of cutter path:
DR+: positive rotation (counterclockwise), down-cut milling;
DR–: negative rotation (clockwise), up-cut milling

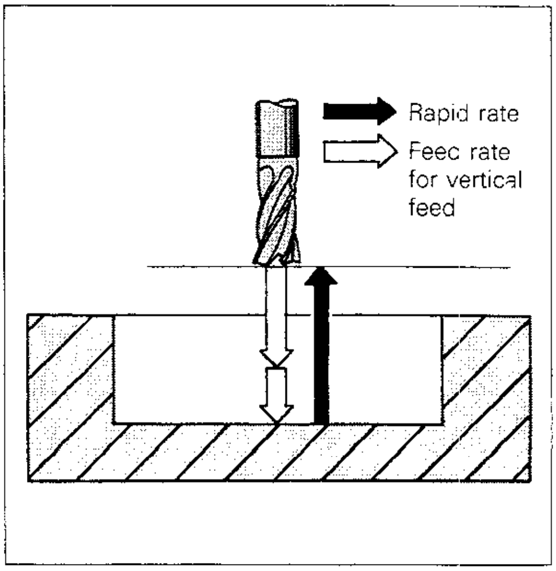


Start position

The start position must be approached in a previous positioning block without radius compensation.

Procedure

The tool penetrates the workpiece from the **starting position** (pocket centre) and then follows the path indicated. The starting direction of the cutter path is the positive axis direction of the longer side, i.e. if this side is parallel to the X-axis, the cutter starts off in the positive X-direction. When milling square pockets, the cutter will always start in the positive Y-direction.

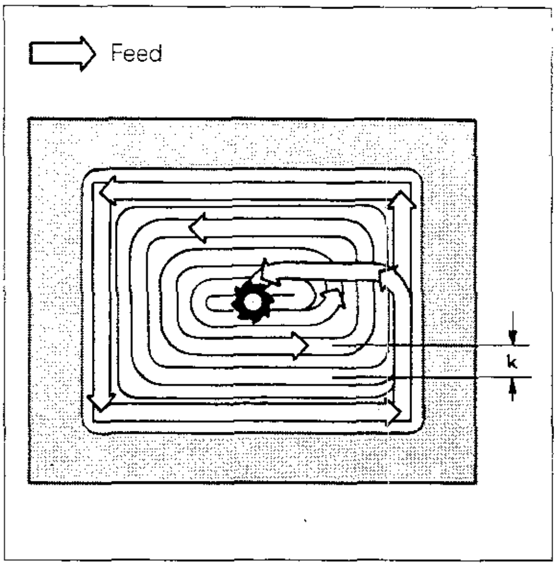


Canned cycles
Pocket milling

Procedure

The direction of rotation depends on the programmed **rotation** (in this case DR+). The maximum **stepover** is k.

The procedure is repeated until the programmed **milling depth** is reached; the tool then returns to the starting position.



Stepover

The control system calculates the stepover factor k according to the following equation:

$$k = K \times R$$

k: stepover
K: overlap factor determined by machine manufacturer (depends on specified machine parameters)
R: cutter radius

Notes:

Canned cycles
Pocket milling

Cycle definition

Operating mode ________ ◇

Dialogue initiation ________ CYCL DEF ↓ or GOTO □ 4 ENT

CYCL DEF 4 POCKET MILLING	▶ ENT	Press ENT to select cycle.

SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	+/−	with correct sign.
	ENT	Press ENT.

MILLING DEPTH ?	▶ □	Specify milling depth.
	+/−	with correct sign.
	ENT	Press ENT.

PECKING DEPTH ?	▶ □	Specify infeed per cut.
	+/−	with correct sign.
	ENT	Press ENT.

FEED RATE FOR PECKING ?	▶ □	Specify rate of vertical feed.
	ENT	Press ENT.

FIRST SIDE LENGTH ?	▶ X	Enter axis of side 1, e.g. X.
	□	Specify first side length with positive sign.
	ENT	Press ENT.

SECOND SIDE LENGTH ?	▶ Y	Enter axis of side 2, e.g. Y.
	□	Specify second side length with positive sign.
	ENT	Press ENT.

Notes:

Canned cycles
Pocket milling

FEED RATE ? F =	▶ □	Specify feed rate for milling pocket.
	▼ ENT	Press ENT.

ROTATION CLOCKWISE: DR– ?	▶ +/–	Specify rotation for cutter path.
	▼ ENT	Press ENT.

Enter set-up clearance, milling depth and infeed per cut with same sign.

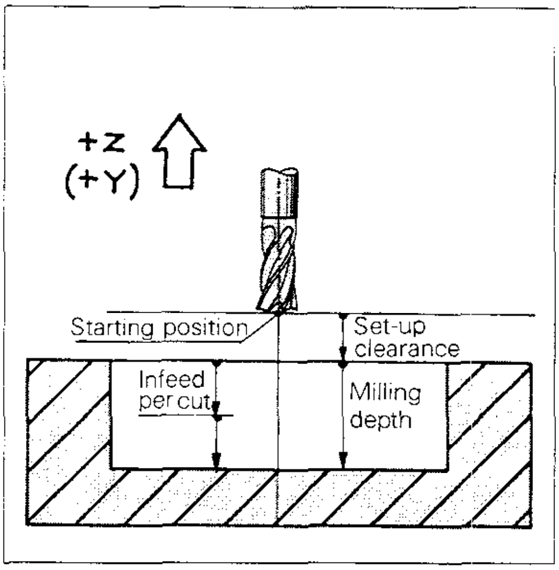
Sample display

250 CYCL DEF 4.0 POCKET MILLING	Cycle definition "Pocket milling" occupies 7 program blocks.
251 CYCL DEF 4.1 SET-UP –2.000	Set-up clearance
252 CYCL DEF 4.2 DEPTH –30.000	Milling depth
253 CYCL DEF 4.3 PECKING –10.000	Infeed per cut
F 80	Feed rate for vertical feed
254 CYCL DEF 4.4 X +80.000	Length of 1st pocket side
255 CYCL DEF 4.5 Y +40.000	Length of 2nd pocket side
256 CYCL DEF 4.6 F 100 DR+	Feed rate and cutter path rotation

Canned cycles
Milling a circular pocket

Cycle

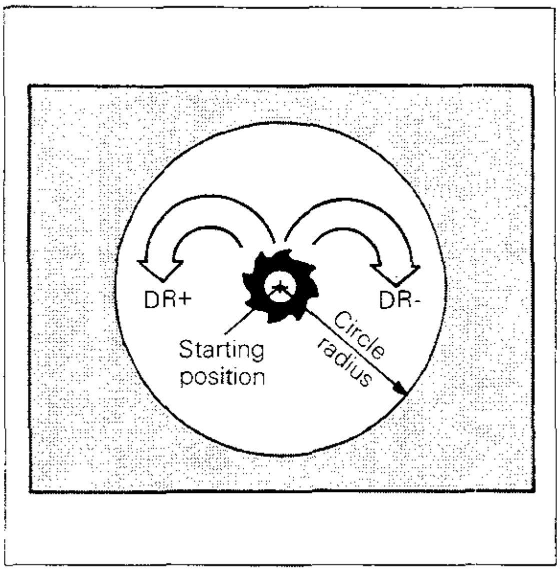
"Circular pocket" is a **roughing cycle.**



Input data

Set-up clearance: see cycle 1.
Milling depth (= pocket depth): distance between workpiece surface and pocket bottom.
See "Set-up clearance" for sign.
Pecking depth: infeed per cut, i.e. amount by which tool penetrates workpiece.
See "Set-up clearance" for sign.
Feed rate for vertical feed: traversing speed of tool when penetrating workpiece.
Circle radius: radius of circular pocket.
Feed rate: traversing speed of tool in machining plane.
Rotation: Direction of rotation of cutter path:

DR+: positive rotation (counterclockwise), down-cut milling;
DR–: negative rotation (clockwise), up-cut milling

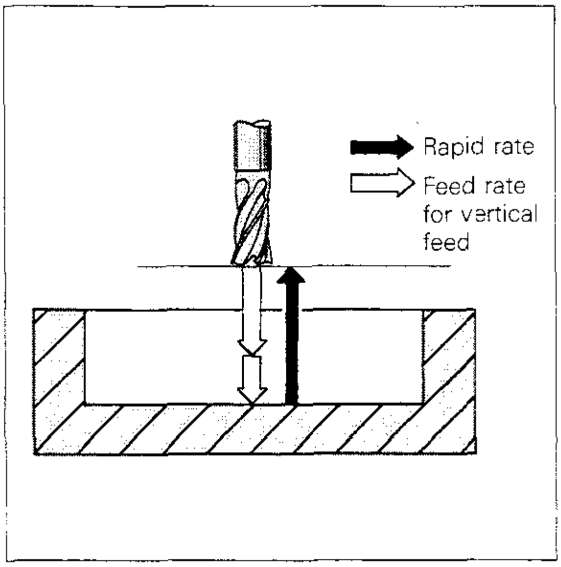


Start position

The start position must be approached in a previous positioning block without radius compensation.

Procedure

The tool penetrates the workpiece from the **starting position** (pocket centre).



Canned cycles
Milling a circular pocket

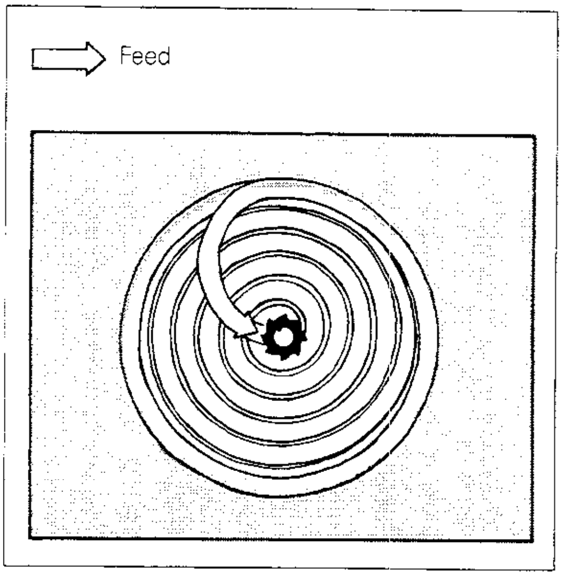
Procedure

The cutter then follows the illustrated spiral path; its direction depends on the programmed **rotation** (in this case DR+). The starting direction of the cutter is:

- the Y+ direction for the X, Y plane,
- the X+ direction for the Z, X plane,
- the Z+ direction for the Y, Z plane.

The maximum **stepover** is the amount of k (see "Pocket milling" cycle).

The procedure is repeated until the programmed **milling depth** is reached.
The tool then returns to the starting position.



Milling a circular pocket with the 4th axis

If the TNC's fourth axis controls an additional linear axis U, V or W, the fourth axis can also be used to mill a circular pocket.

To do this, the fourth axis must be programmed before the cycle call in the last positioning block.

Example:

```
15 L X+50.000  V+50.000

         R0 F              M

16 CYCL CALL

                           M
```

Notes:

Canned cycles
Milling a circular pocket

Cycle definition

Operating mode ______ ⇨

Dialogue initiation ______ CYCL DEF ↓ or GOTO □ 5 ENT

CYCL DEF 5 CIRCULAR POCKET	▶ ENT	Press ENT.
▼		
SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	▼ +/−	with correct sign.
	▼ ENT	Press ENT.
▼		
MILLING DEPTH ?	▶ □	Specify milling depth.
	▼ +/−	with correct sign.
	▼ ENT	Press ENT.
▼		
PECKING DEPTH ?	▶ □	Specify infeed per cut.
	▼ +/−	with correct sign.
	▼ ENT	Press ENT.
▼		
FEED RATE FOR PECKING ?	▶ □	Specify rate of vertical feed.
	▼ ENT	Press ENT.
▼		
CIRCLE RADIUS ?	▶ □	Specify radius of circular pocket.
	▼ ENT	Press ENT.
▼		
FEED RATE ? F =	▶ □	Specify feed rate for milling circular pocket.
	▼ ENT	Press ENT.
▼		

Notes:

Canned cycles
Milling a circular pocket

ROTATION CLOCKWISE: DR– ?	+/–	Specify rotation of cutter path.
	ENT	Press ENT.

Enter set-up clearance, milling depth and infeed per cut with same sign.

Sample display

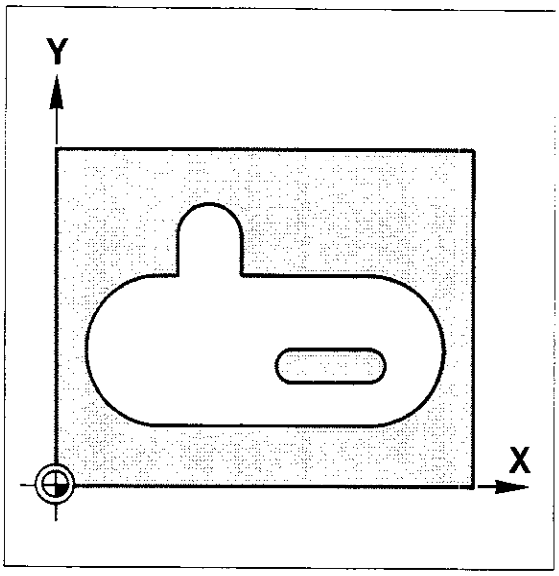
40 CYCL DEF 5.0 CIRCULAR POCKET	Cycle definition "Circular pocket" occupies 6 program blocks.
41 CYCL DEF 5.1 SET-UP –2.000	Set-up clearance
42 CYCL DEF 5.2 DEPTH –60.000	Milling depth
43 CYCL DEF 5.3 PECKING –20.000	Infeed per cut
F80	Feed rate for vertical feed
44 CYCL DEF 5.4 RADIUS 120.000	Circle radius
45 CYCL DEF 5.5 F100 DR–	Feed rate and cutter path rotation

Canned cycles
Variable-contour pockets

Introduction

Four cycles are required for milling pockets with variable contours:

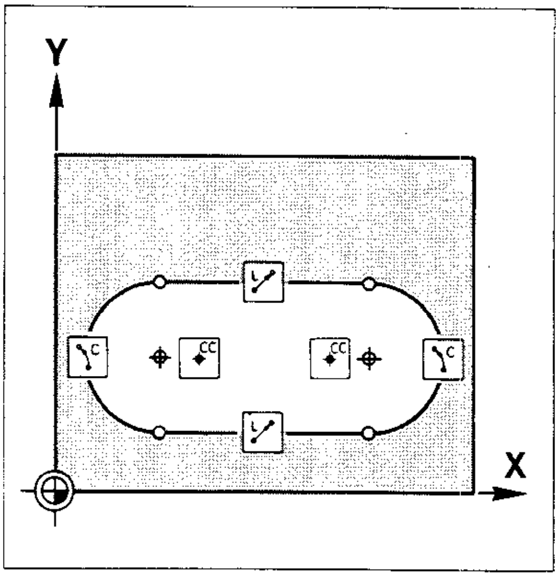
- **Cycle 14: CONTOUR GEOMETRY** (list of subroutines containing subcontours)
- **Cycle 15: PILOT DRILL** (rough drilling to pocket depth for all partial contours)
- **Cycle 6: ROUGH-OUT** (rough-milling of contour and clearing of pocket)
- **Cycle 16: CONTOUR MILL** (finish-milling of contour pocket).



Contour

The contour consists of one or more **pockets** and **islands** within the pocket. A total of up to **12 subcontours** is possible. Each subcontour must be programmed as a closed loop of contour elements. The following straight lines and arcs can be used as **contour elements:**

L	Line, end position programmed in Cartesian coordinates.
P L	Line, end position programmed in polar coordinates.
CC C	Circle (arc) defined by circle centre and end position; end position in Cartesian coordinates.
CC P C	Circle (arc) defined by circle centre and end position; end position programmed in polar coordinates.



Beginning with software version 03:
All the contour entry keys may be used to program the contour elements. Subroutines, program part repititions and Q parameter functions (FN) can also be programmed.
No coordinate conversions are permitted within contour definitions. Coordinate conversions can, however, be applied to the entire pocket.

Before machining, test the program with the help of the graphic simulation. The control can not calculate all geometries for pockets with various contours.

Canned cycles
Variable-contour pockets

Pocket

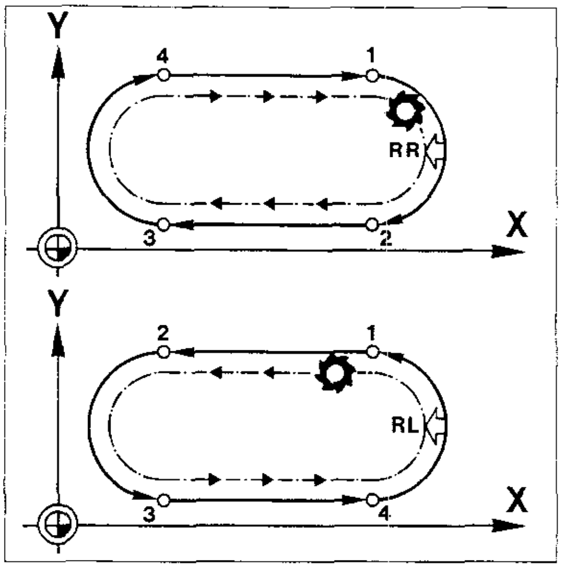
Pockets can be defined in two ways:

Option 1:

- Clockwise sequence of contour elements
- Radius compensation RR

Option 2:

- Counterclockwise sequence of contour elements
- Radius compensation RL.



Island

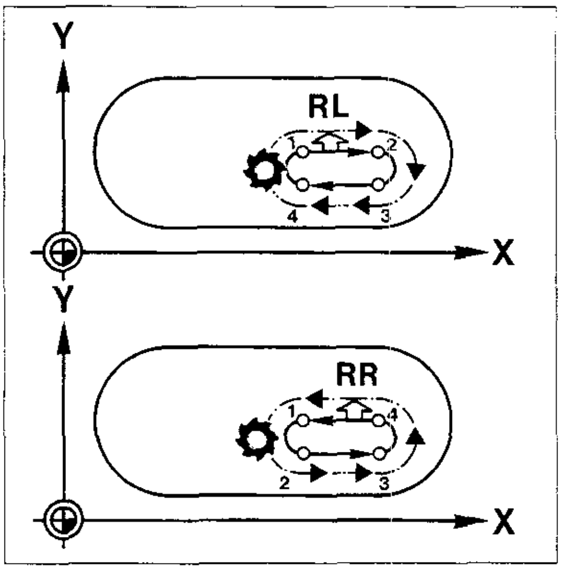
Islands can also be defined in two ways:

Option 1:

- Clockwise sequence of contour elements
- Radius compensation RL

Option 2:

- Counterclockwise sequence of contour elements
- Radius compensation RR.

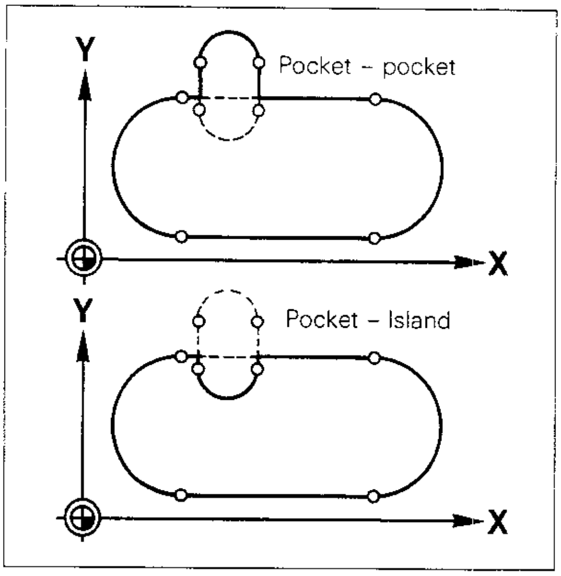


Canned cycles
Variable-contour pockets

Superimposing pockets and islands

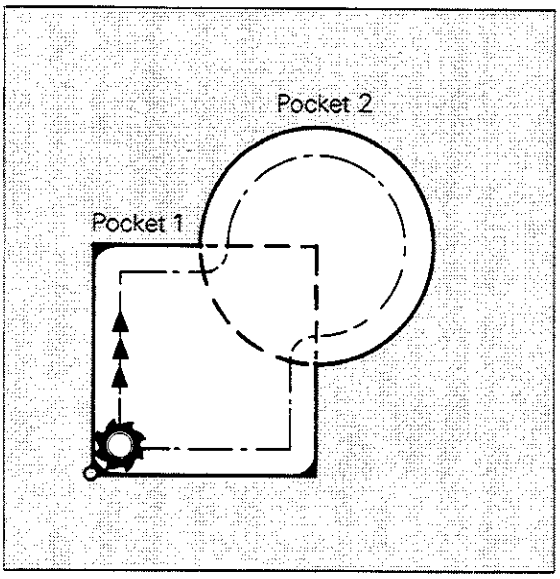
Pockets and islands can be superimposed (overlaid) on one another. The TNC computes the resulting contour automatically from the starting point of the first subcontour.

For this reason, the starting point of the subcontour is the determining factor of the resulting contour pocket.

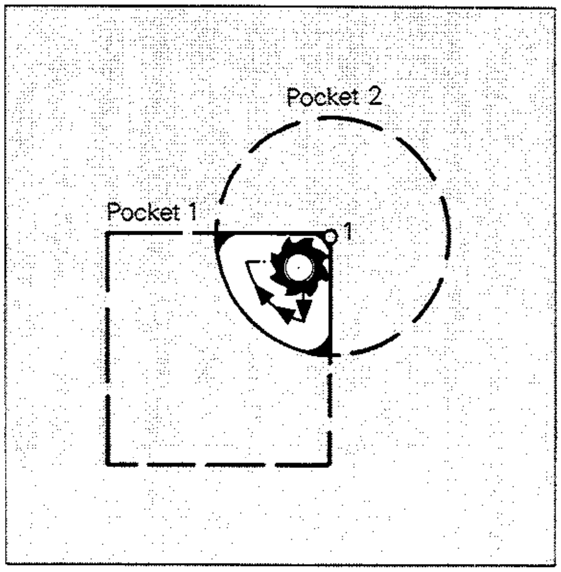


Superimposing pockets

The starting point of pocket contour 1 is located outside the area of pocket 2, the areas of both pockets will be cleared.



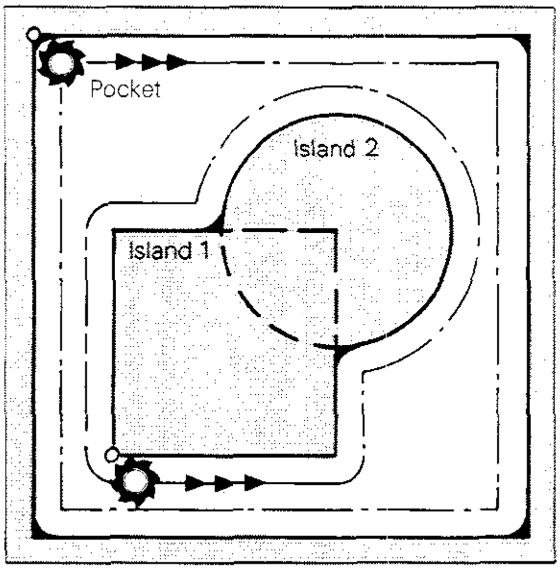
The starting point of pocket contour 1 is located within the area of pocket 2, only the common area of the two pockets will be cleared.



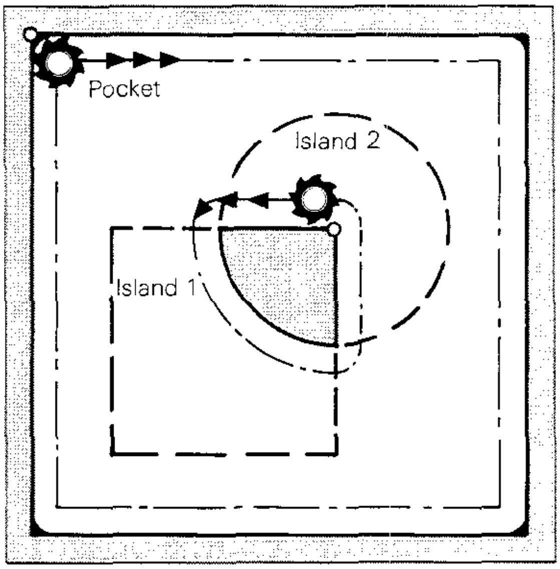
Canned cycles
Variable-contour pockets

Superimposing islands

The starting point of island contour 1 is located outside the area of island 2, neither of the areas of the two islands will be cleared.



The starting point of island contour 1 is located within the area of island 2, only the common areas of the two islands will remain.

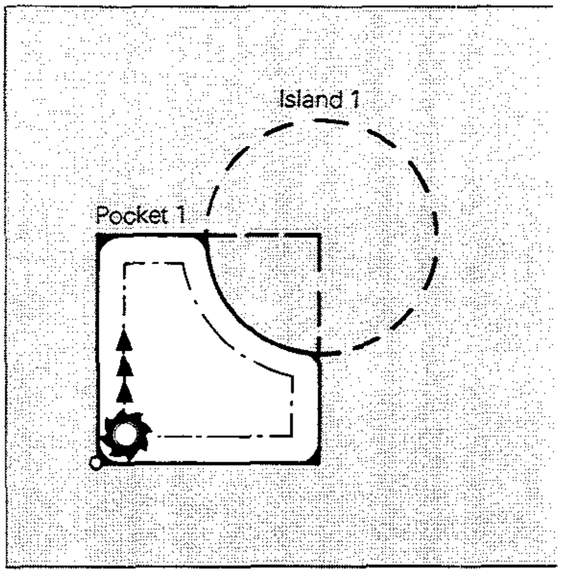


Canned cycles
Variable-contour pockets

Superimposing pockets and islands

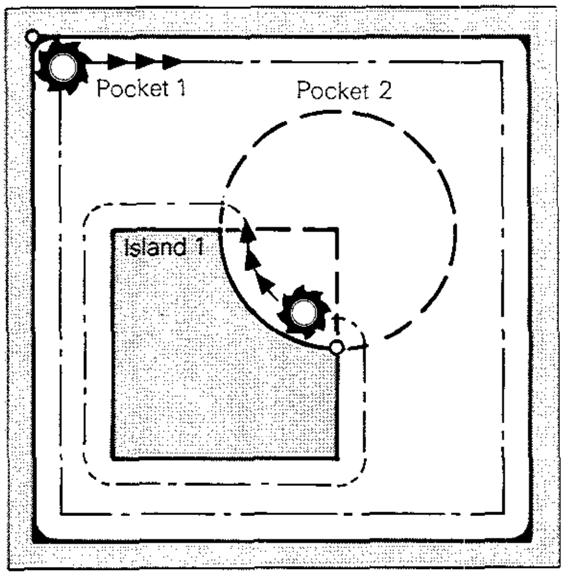
If pocket areas are reduced in size by superimposed islands, the starting point of pocket contour 1 must be located outside of island 1.

An island can also reduce several pocket areas. The starting points of the pocket contours must all lie outside the island.



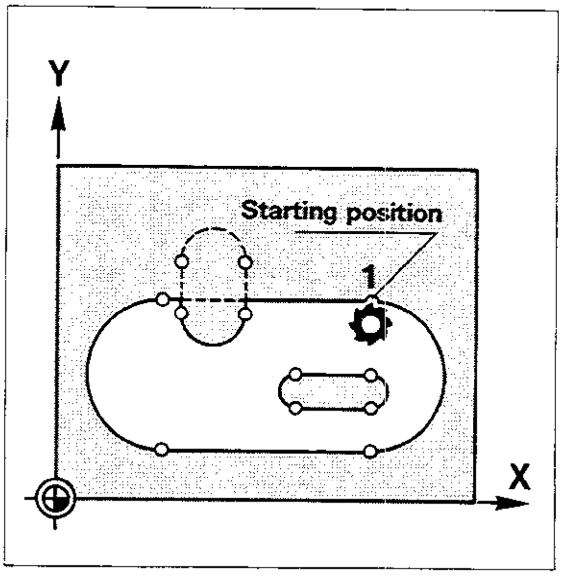
If island areas are reduced in size by superimposed pockets, the starting point of pocket contour 2 must be located outside of island 1.

A pocket can also reduce several island areas. The starting point of the superimposing pocket must lie within the first island.



Programming subcontours

Partial contours are saved and stored in **subroutines.** The first point of the subcontour is the **starting position,** where machining begins. The starting position of the first subcontour is also the penetration point for the cycle "Pilot drilling". The starting position is programmed via linear interpolation using the L key.



The first subcontour must be a pocket.

The starting position cannot be located on the contour of an island.

Radius compensation RL/RR should not be changed within a subcontour or a subroutine.

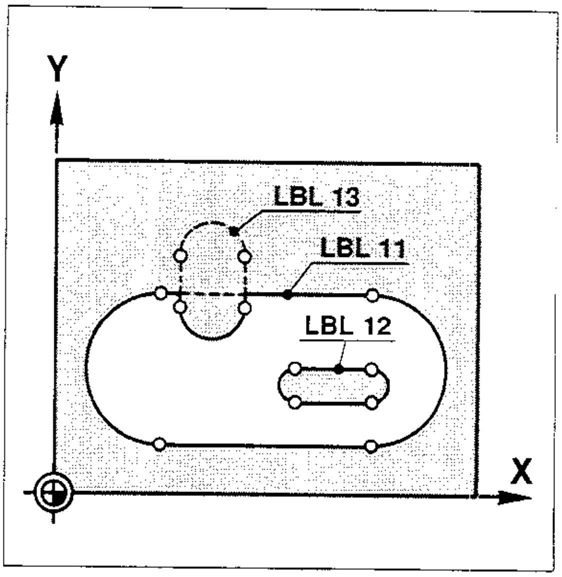
Canned cycles
Variable-contour pockets
Cycle 14: Contour geometry

Cycle

The label numbers (subroutines) of the subcontours are defined in cycle 14 "CONTOUR GEOMETRY". Up to 12 label numbers can be entered. The TNC computes the intersecting points of the resulting contour pocket from the programmed subcontours.

Cycle 14 "CONTOUR GEOMETRY" is active immediately following definition; no separate cycle call is necessary.

The first subcontour must be programmed as a pocket.



Canned cycles
Variable-contour pockets
Cycle 14: Contour geometry

Definition

Operating mode ______ ⇨

Dialogue initiation ______ CYCL DEF ↑ or GOTO □ 1 4 ENT

CYCLE DEF 14 CONTOUR GEOM.	▶ ENT	Press ENT to select cycle.
LABEL NUMBERS FOR CONTOUR ?	▶ □	Enter label number of first contour.
	ENT	Press ENT.
	□	Enter label number of second contour.
	ENT	Press ENT.
After entering final label number:	▶ ENT	Press ENT.
	END □	Press END to transfer cycle definition.

Sample display

```
5 CYCL DEF 14.0 CONTOUR GEOM.

6 CYCL DEF 14.1 CONTOUR LABEL

  11/12/13/
```

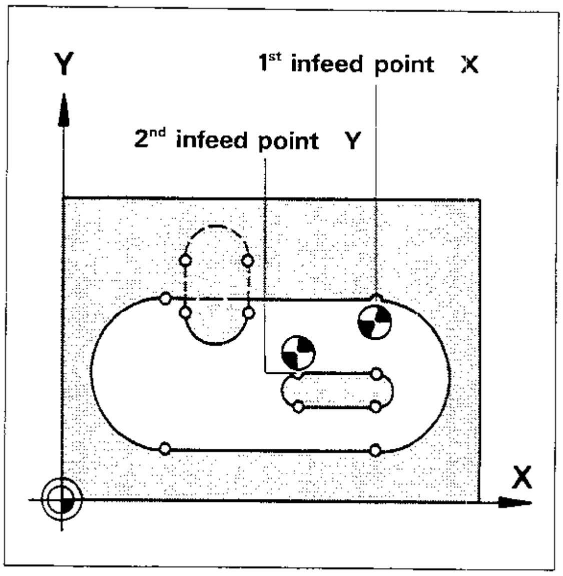
Cycle definition occupies up to 3 program blocks.

Subroutines with label numbers 11, 12 and 13 define the contour pocket.

Canned cycles
Variable-contour pockets
Cycle 15: Pilot drill

Cycle

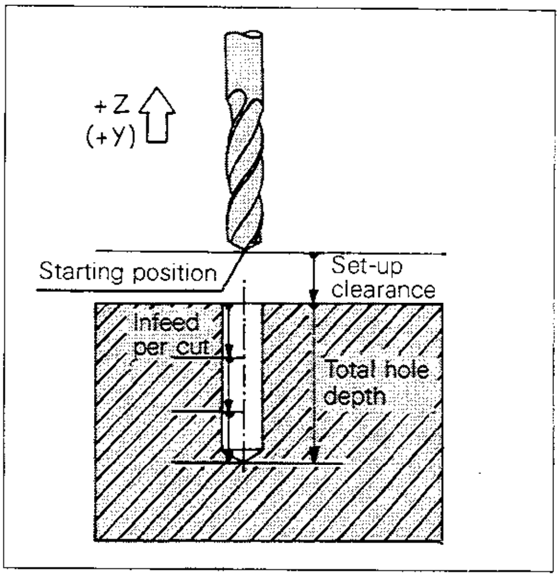
Cycle 15 is used to drill pilot holes at cutter infeed points.
The positions of the infeed points are identical to the starting positions of the subcontours. In the case of closed loops of contour elements, produced by superimposing several pockets and islands, the infeed point is the starting position of the first subcontour.



The cycle "Pilot drill" must be called separately.

Input data

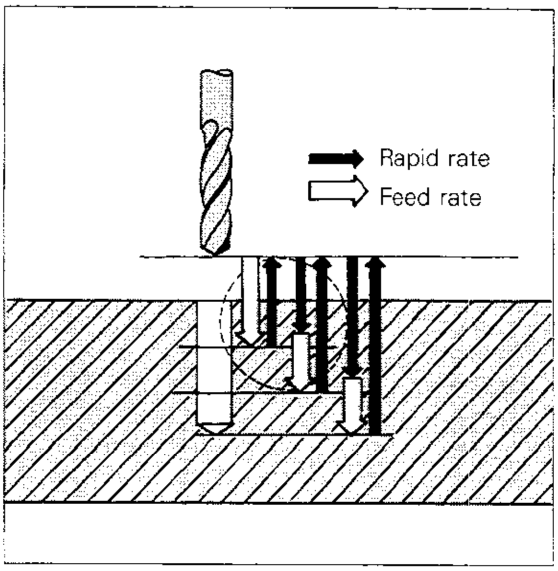
Set-up clearance: see cycle 1.
Total hole depth: distance between workpiece surface and bottom of pocket. See "Set-up clearance" for sign.
Pecking depth: infeed per cut, i.e. the amount by which the tool penetrates the workpiece for each cut. See "Set-up clearance" for sign.
Feed rate: traversing rate of tool when penetrating workpiece.
Contour mill allowance: allowance for the finishing procedure.



The tool must be located at the set-up clearance before the cycle is called.

Procedure

The control system positions the tool above the first infeed point at the programmed **set-up clearance,** taking the programmed **contour mill allowance** into account.
The tool, moving at the programmed **feed rate,** then penetrates to the first **pecking depth.** After drilling to this depth, the tool returns at rapid rate to the starting position and then plunges back to the first depth.
The tool then advances again at the programmed feed rate by the amount of the infeed increment, returns to the starting position and so on.
The alternating drilling/retracting action is repeated until the programmed **total hole depth** is reached.
Finally, the control system positions the tool at the programmed set-up clearance above the second infeed point and repeats the drilling operation.
The advanced halting distance corresponds to the set-up clearance.



Canned cycles
Variable-contour pockets
Cycle 15: Pilot drill

Definition

Operating mode ______ ◇

Dialogue initiation ______ CYCL DEF ↑ or GOTO ⊐ 1 5 ENT

CYCL DEF 15 PILOT DRILL	▶ ENT	Press ENT to select cycle.
SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	+/−	with correct sign.
	ENT	Press ENT.
TOTAL HOLE DEPTH ?	▶ □	Specify total hole depth.
	+/−	with correct sign.
	ENT	Press ENT.
PECKING DEPTH ?	▶ □	Specify infeed per cut.
	+/−	with correct sign.
	ENT	Press ENT.
FEED RATE ? F =	▶ □	Specify feed rate for infeed (positive numerical value).
	ENT	Press ENT.
CONTOUR MILL ALLOWANCE ?	▶ □	Enter finishing allowance.
	ENT	Press ENT.

Sample display

```
18 CYCL DEF 15.0 PILOT DRILL

19 CYCL DEF 15.1 SET-UP –2.000

                  DEPTH –20.000

20 CYCL DEF 15.2 PECKG –10.000

   F40            ALLOW +1.000
```

Cycle definition occupies up to 3 program blocks.

Set-up clearance

Total hole depth

Infeed per cut

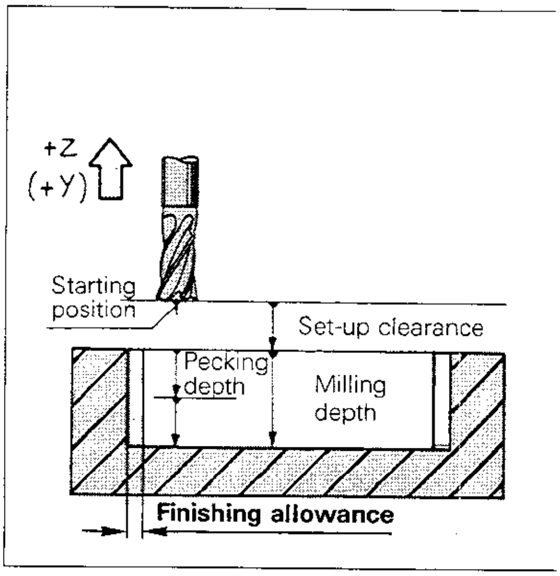
Rate of infeed and finishing allowance.

Canned cycles
Variable-contour pockets
Cycle 6: Rough-out

Cycle

Cycle 6 defines the roughing procedure for clearing the pocket.

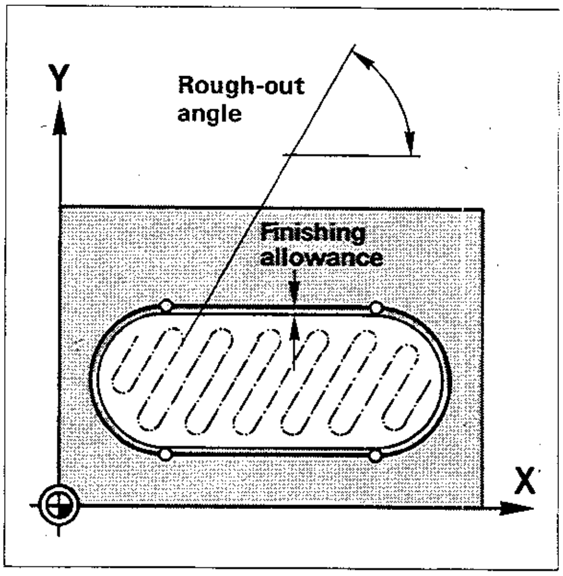
The cycle "Rough-out" must be called separately.



Input data

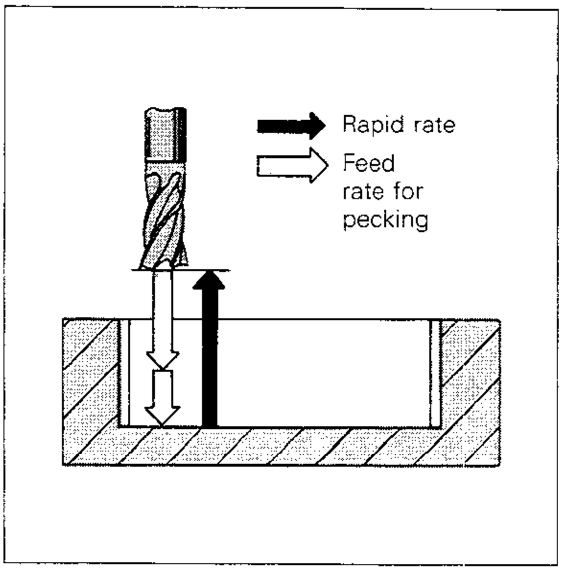
Set-up clearance: see cycle 1.
Milling depth: distance between workpiece surface and pocket bottom. See "Set-up clearance" for sign.
Pecking depth: infeed per cut, i.e. the amount by which the tool penetrates the workpiece for each cut. See "Set-up clearance" for sign.
Feed rate for pecking: traversing rate of tool when penetrating workpiece.
Contour mill allowance: allowance for the finishing procedure (positive numerical value).
Rough-out angle: direction for clearing pocket, based on angular reference axis of machining plane.
Feed rate: traversing speed of tool in machining plane.

The tool must be located at the set-up clearance before the cycle is called.



Procedure

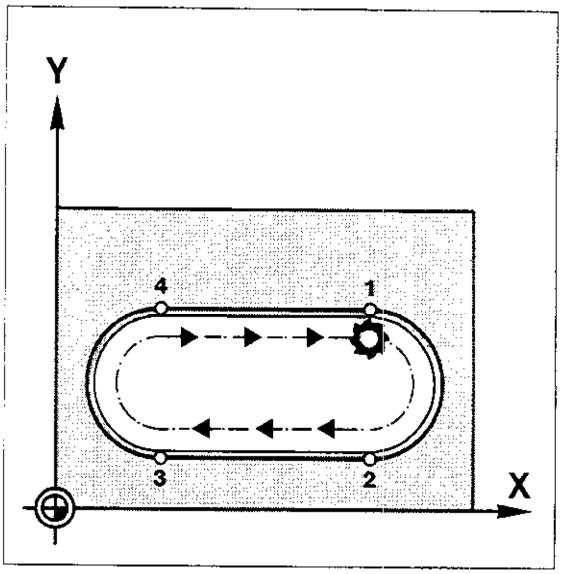
The control system positions the tool automatically above the first infeed point, taking the programmed **contour mill allowance** into account. **Beware of danger of collision with chucking device.**
The tool then penetrates the workpiece.
After reaching the first **pecking depth,** the tool mills the first subcontour at the programmed **feed rate,** taking the finishing allowance into account.
The direction of rotation for rough-milling is determined by a machine parameter defined by the machine manufacturer.



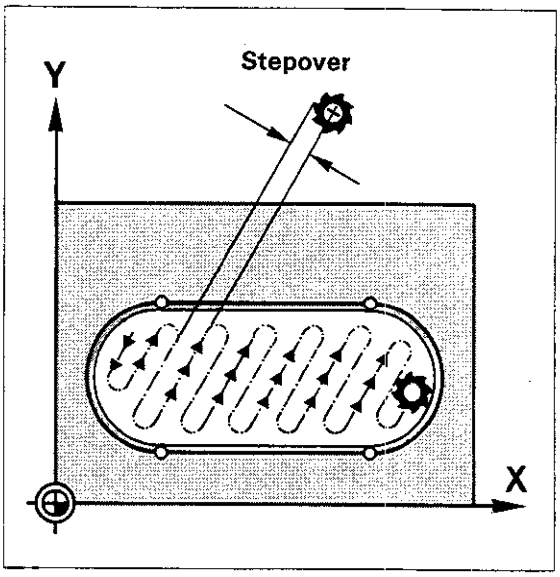
Canned cycles
Variable-contour pockets
Cycle 6: Rough-out

Procedure

At the infeed point, the control system advances the tool to the next pecking depth. The procedure is repeated until the programmed **milling depth** is reached.
The remaining subcontours are milled in the same manner.



The pocket is then cleared. The direction of feed corresponds to the programmed **rough-out angle.** The stepover per cut corresponds to the cutter radius.
The pocket can be cleared with multiple vertical feed motions.
At the end of the cycle, the control system retracts the tool to the set-up clearance.
When roughing out, islands are jumped over. For this, a minimum advanced halting distance is maintained. This halting distance corresponds to the maximum set-up clearance for the "Rough-out" and "Contour mill" cycles.



Notes:

Canned cycles
Variable-contour pockets
Cycle 6: Rough-out

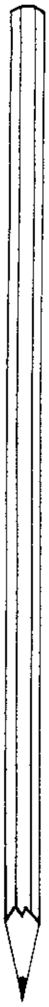
Definition

Operating mode ________________ ⟐

Dialogue initiation ________________ CYCL DEF ↓ or GOTO □ 6 ENT

CYCL DEF 6 ROUGH-OUT	▶ ENT	Press ENT to select cycle.
SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	▼ +/−	with correct sign.
	▼ ENT	Press ENT.
MILLING DEPTH ?	▶ □	Specify milling depth.
	▼ +/−	with correct sign.
	▼ ENT	Press ENT.
PECKING DEPTH ?	▶ □	Specify infeed per cut.
	▼ +/−	with correct sign.
	▼ ENT	Press ENT.
FEED RATE FOR PECKING ?	▶ □	Specify feed rate for vertical feed.
	▼ ENT	Press ENT.
CONTOUR MILL ALLOWANCE ?	▶ □	Enter finishing allowance (positive numerical value).
	▼ ENT	Press ENT.
ROUGH-OUT ANGLE ?	▶ □	Specify rough-out angle.
	▼ ENT	Press ENT.
FEED RATE ? F =	▶ □	Specify feed rate for milling pocket.
	▼ ENT	Press ENT.

Notes:



Canned cycles
Variable-contour pockets
Cycle 6: Rough-out

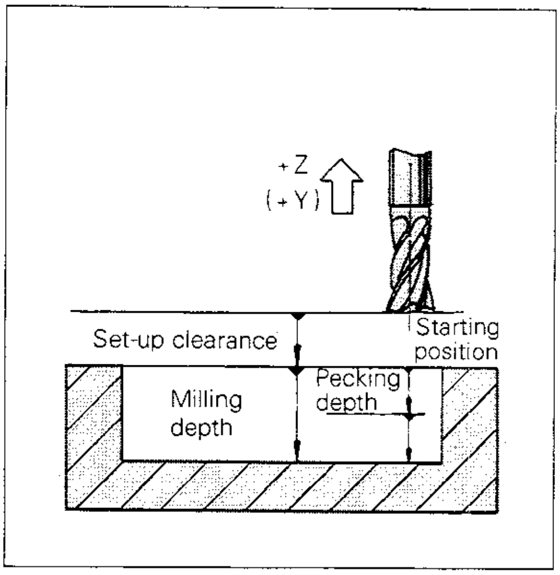
Sample display

16 CYCL DEF 6.0 ROUGH-OUT	Cycle definition occupies 4 program blocks.
17 CYCL DEF 6.1 SET-UP –2.000	Set-up clearance
DEPTH –20.000	Milling depth
18 CYCL DEF 6.2 PECKG –10.000	Infeed per cut
F40 ALLOW +1.000	Feed rate for tool infeed and finishing allowance
19 CYCL DEF 6.3 ANGLE +0.000	Rough-out angle
F60	Feed rate in machining plane.

Canned cycle
Variable-contour pockets
Cycle 16: Contour mill

Cycle

Cycle 16 "CONTOUR MILL" is used to finish-mill the contour pocket.

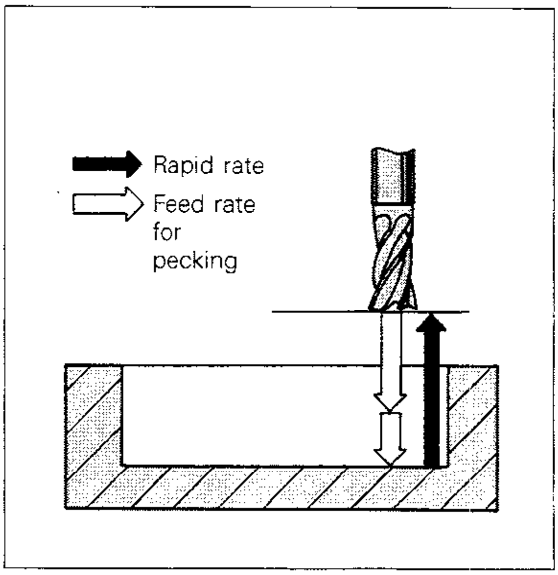


> The cycle can also be used for general milling of contours that are made up of subcontours.
> This provides the following advantages:
> - contour intersections are calculated.
> - collisions are prevented.

> The cycle "Contour mill" must be called separately.

Input data

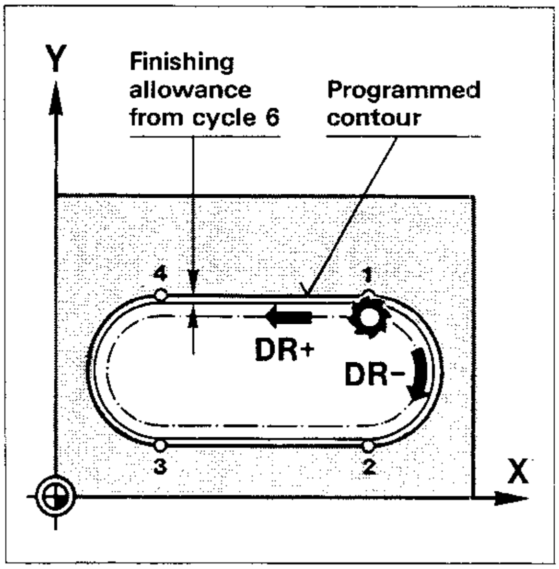
Set-up clearance: see cycle 1.
Milling depth: distance between workpiece surface and pocket bottom. See "Set-up clearance" for sign.
Pecking depth: infeed per cut, i.e. the amount by which the tool penetrates the workpiece for each cut. See "Set-up clearance" for sign.
Feed rate for pecking: traversing rate of tool when penetrating workpiece.
Direction of rotation for contour milling: cutting direction along the pocket contour (island contours: opposite cutting direction)
DR+: positive rotation,
down-cut milling for pocket and island
DR–: negative rotation,
up-cut milling for pocket and island.
Feed rate: traversing speed of tool in machining plane.



> The tool must be located at the set-up clearance before the cycle is called.

Procedure

The control system positions the tool automatically above the first infeed point. **Beware of danger of collision with chucking device.**
Moving at the programmed **feed rate,** the tool then penetrates to the first **pecking depth.**
When this depth is reached, the tool mills the first contour, moving at the programmed **feed rate** and taking the specified **direction of rotation** into account.
At the infeed point, the control system advances the tool to the next pecking depth, repeating the procedure until the programmed **milling depth** is reached.
The remaining subcontours are milled in the same manner.



Canned cycles
Variable-contour pockets
Cycle 16: Contour mill

Definition

Operating mode ______ ◇

Dialogue initiation ______ CYCL DEF ↑ or GOTO □ 1 6 ENT

CYCL DEF 16 CONTOUR MILL	▶ ENT	Press ENT to select cycle.

SET-UP CLEARANCE ?	▶ □	Specify set-up clearance.
	+/−	with correct sign.
	ENT	Press ENT.

MILLING DEPTH ?	▶ □	Specify milling depth.
	+/−	with correct sign.
	ENT	Press ENT.

PECKING DEPTH ?	▶ □	Specify infeed per cut.
	+/−	with correct sign.
	ENT	Press ENT.

FEED RATE FOR PECKING ?	▶ □	Specify feed rate for vertical feed.
	ENT	Press ENT.

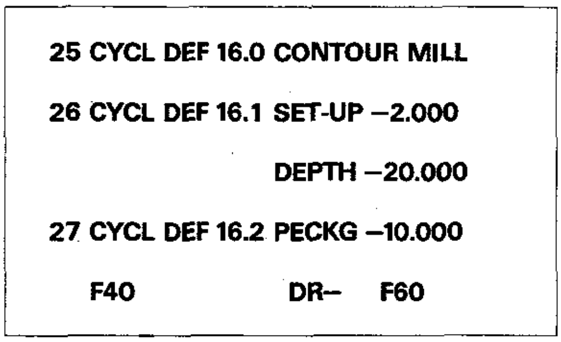
CW/CCW PATH FOR CONTOUR MILL ?	▶ □	Specify cutting direction.
	ENT	Press ENT.

FEED RATE ? F =	▶ □	Specify feed rate for milling pocket.
	ENT	Press ENT.

Notes:

Canned cycles
Variable-contour pockets
Cycle 16: Contour mill

Sample display



```
25 CYCL DEF 16.0 CONTOUR MILL

26 CYCL DEF 16.1 SET-UP –2.000

                 DEPTH –20.000

27 CYCL DEF 16.2 PECKG –10.000

   F40                DR–    F60
```

Cycle definition occupies 3 program blocks.

Set-up clearance

Milling depth

Infeed per cut

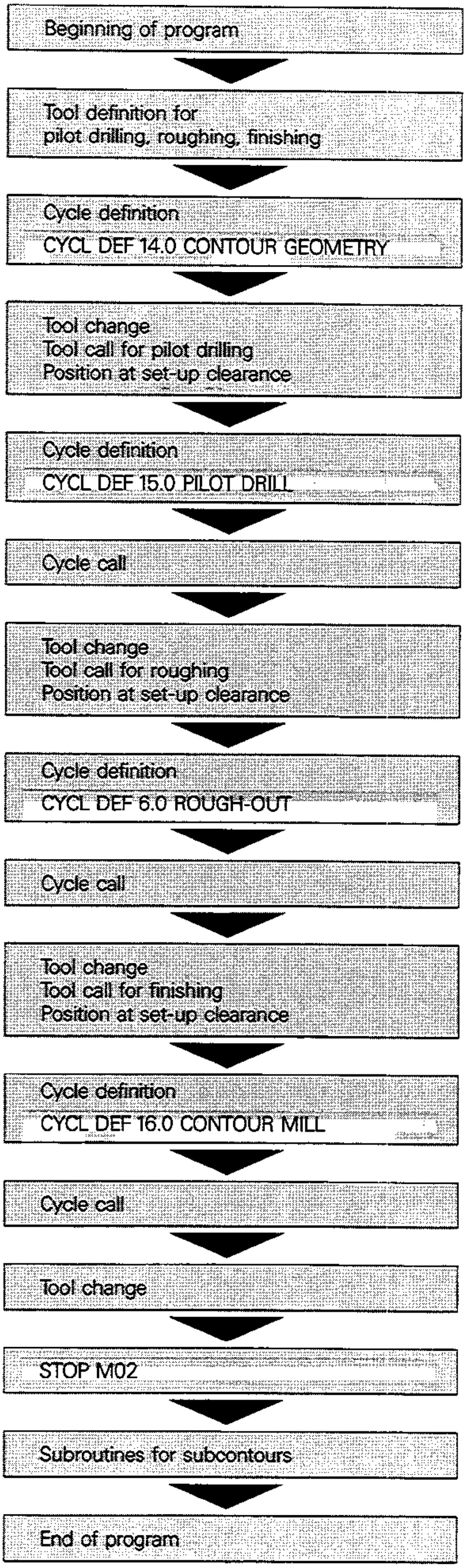
Feed rate for vertical feed,
cutting direction and feed rate in machining plane.

Canned cycles
Variable-contour pockets

Program format and cycle sequence

The program format shown at the right is recommended for programming a variable-contour pocket.

Use the graphics feature to check the contour pocket before running the program on the machine.



Beginning of program

Tool definition for
pilot drilling, roughing, finishing

Cycle definition
CYCL DEF 14.0 CONTOUR GEOMETRY

Tool change
Tool call for pilot drilling
Position at set-up clearance

Cycle definition
CYCL DEF 15.0 PILOT DRILL

Cycle call

Tool change
Tool call for roughing
Position at set-up clearance

Cycle definition
CYCL DEF 6.0 ROUGH-OUT

Cycle call

Tool change
Tool call for finishing
Position at set-up clearance

Cycle definition
CYCL DEF 16.0 CONTOUR MILL

Cycle call

Tool change

STOP M02

Subroutines for subcontours

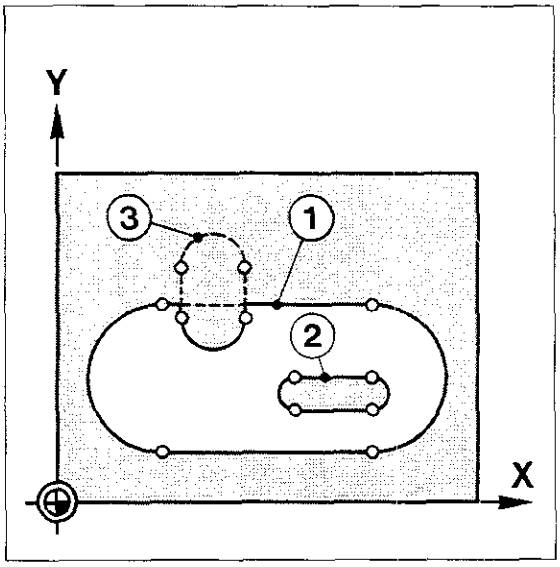
End of program

Canned cycles
Variable-contour pockets
Example

Pocket contour

The pocket shown in the illustration is made up of three subcontours:

Subcontour 1: pocket
Subcontour 2: island within pocket
Subcontour 3: island superimposed over pocket (subcontour 1)



Beginning of program

The program for milling the pocket is program number 40. The blank workpiece dimensions for graphics simulation on the TNC 155 are defined in the BLK FORM blocks.

```
0   BEGIN   PGM 40        MM
1   BLK FORM 0.1        Z X+0.000
            Y+0.000       Z-25.000
2   BLK FORM 0.2          X+80.000
            Y+60.000      Z+0.000
```

Tool definition

The tools are defined at the beginning of the program. Three tools are required to mill the pocket.

Tool 11: for pilot drilling
Tool 12: for roughing and clearing
Tool 13: for finishing.

```
3   TOOL   DEF 11      L+0.000
                       R+2.000
4   TOOL   DEF 12      L-4.900
                       R+2.000
5   TOOL   DEF 13      L-2.500
                       R+2.000
```

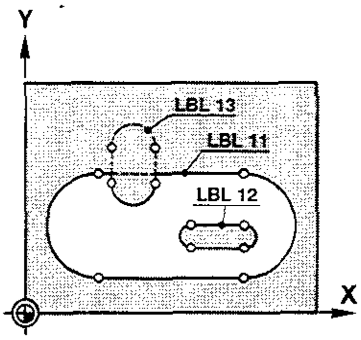
Canned cycles
Variable-contour pockets
Example

Contour definition

The label numbers of the subcontours are programmed during contour definition.

The TNC computes the intersecting points of the pocket contour from the programmed subcontours.

```
6 CYCL DEF 14.0 CONTOUR GEOM.

7 CYCL DEF 14.1 CONTOUR LABEL

           11 / 12 / 13 /    /    /
```



Tool change/ Set-up clearance

The tool change position is also programmed in a subroutine with the label number 1.

Tool No. 11 for pilot drilling is then called and positioned at the set-up clearance.

```
8 LBL  1
9 TOOL CALL 0                  Z
                    S
10 L  Z+100.000
                        R0 F15999    M
11 L  X-50.000           Y-50.000
                        R  F         M06
12 LBL  0
13 TOOL CALL 11                 Z
                    S 140.000
14 L  Z+2.000
                        R0 F15999    M
```

Pilot drilling

The cycle "Pilot drill" contains all the data required for vertical feed and penetration.
The cycle "Pilot drill" must be called separately.

```
15 CYCL DEF 15.0 PILOT DRILL

16 CYCL DEF 15.1 SET-UP -2.000

                 DEPTH -20.000

17 CYCL DEF 15.2 PECKG -10.000

   F40                ALLOW +0.500

18 CYCL CALL

                                M13
```

Canned cycles
Variable-contour pockets
Example

Tool change/ Set-up clearance

The next tool change takes place by calling the subroutine with the label number 1. Tool No. 12 is then called for rough-milling the pocket contour and positioned at the set-up clearance.

```
19 CALL LBL 1   REP

20 TOOL CALL 12               Z

                 S 140.000

21 L  Z+2.000

                   R  F           M
```

Roughing-out

The cycle "Rough-out" contains all the data required for rough-milling the pocket.
The cycle "Rough-out" must be called separately.

The TNC then mills the contour of the pocket, taking the finishing allowance into account.
The pocket is then cleared at the programmed angle.

```
22 CYCL DEF 6.0 ROUGH-OUT

23 CYCL DEF 6.1 SET-UP -2.000

                  DEPTH -20.000

24 CYCL DEF 6.2 PECKG -10.000

   F40                ALLOW 0.500

25 CYCL DEF 6.3 ANGLE +45.000

                  F140

26 CYCL CALL

                                  M13
```

Tool change/ Set-up clearance

Again the tool change takes place by calling the subroutine with the label number 1. Tool No. 13 is then called for finishing the pocket contour and positioned at the set-up clearance.

```
27 CALL LBL 1   REP

28 TOOL CALL 13               Z

                 S 140.000

29 L  Z+2.000

                   R  F           M
```

Canned cycles
Variable-contour pockets
Example

Contour milling

The cycle "Contour mill" contains all the data necessary for finish-milling the pocket contour. In addition, the milling direction can be specified, i.e. the pocket contour can be finished with down-cut or up-cut milling. In the example, DR– is programmed for down-cut milling.
A cycle call is required for cycle contour milling.

```
30 CYCL DEF 16.0 CONTOUR MILL

31 CYCL DEF 16.1 SET-UP –2.000

                     DEPTH –20.000

32 CYCL DEF 16.2 PECKG –10.000

   F80         DR–         F120

33 CYCL CALL

                                M13
```

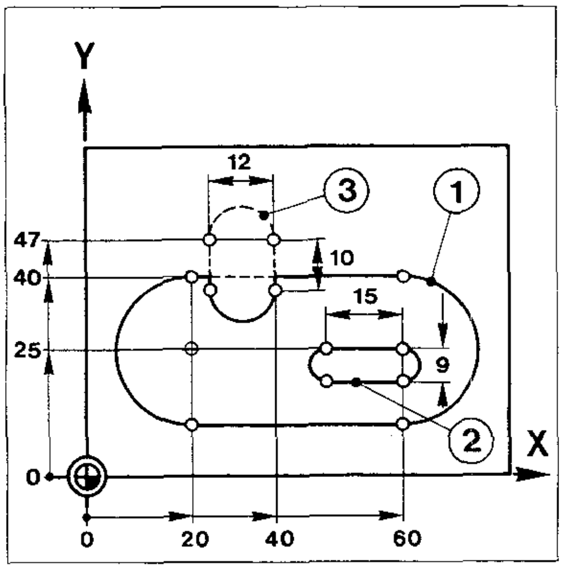
Tool change/ STOP

By calling the subroutine with the label number 1, the TNC moves the tool to the change position. The program is then interrupted with STOP; the auxiliary function M02 or M30 causes a return to the beginning of the program.

```
34 CALL LBL 1     REP

35 STOP

                                M02
```

Subroutines

After the programmed STOP, the subroutines for the three subcontours are programmed according to the dimensions shown in the drawing.



Canned cycles
Variable-contour pockets
Example

Subcontour 1

Subcontour 1 "Pocket" is programmed in the subroutine with the label number 11.

Because the contour elements are programmed clockwise, the radius compensation for the pocket contour is RR.

```
36 LBL   11
37 L   X+60.000    Y+40.000
                   RR            M
38 CC X+60.000    Y+25.000
39 CP IPA–180.000
              DR– R   F          M
40 L    X+20.000
                   R   F         M
41 CC X+20.000    Y+25.000
42 CP IPA–180.000
              DR– R   F          M
43 L    X+60.000
                   R   F         M
44 LBL 0
```

Subcontour 2

Subcontour 2 "Island" is programmed in the subroutine with the label number 12.

Because the contour elements are programmed counterclockwise, the radius compensation for the island contour is RR.

```
45 LBL   12
46 L    X+60.000    Y+25.000
                    RR           M
47 L    IX–15.000
                    R   F        M
48 CC IX+0.000     IY–4.500
49 CP IPA+180.000
              DR+ R   F          M
50 L    IX+15.000
                    R   F        M
51 CC IX+0.000     IY+4.500
52 CP IPA+180.000
              DR+ R   F          M
53 LBL 0
```

Subcontour 3

Subcontour 3 "Island" is programmed in the subroutine with the label number 13.

Because the contour elements are programmed counterclockwise, the radius compensation for the island contour is RR.

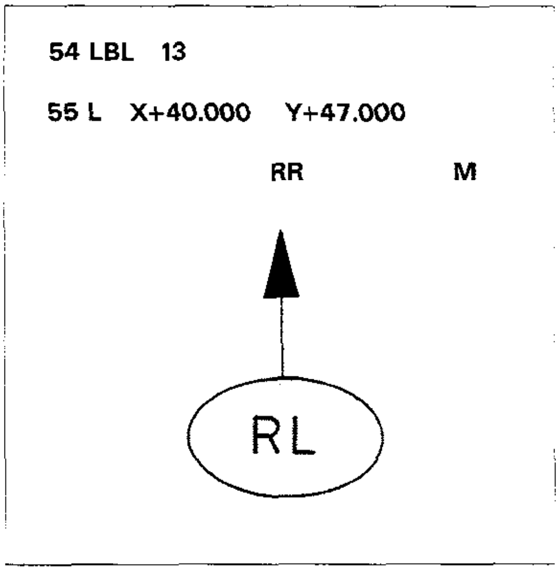
```
54 LBL   13
55 L    X+40.000    Y+47.000
                    RR           M
56 CC IX–6.000     Y+47.000
57 CP IPA+180.000
              DR+ R   F          M
58 L    IY–10.000
                    R   F        M
59 CC IX+6.000     IY+0.000
60 CP IPA+180.000
              DR+ R   F          M
61 L    X+40.000    Y+47.000
                    R   F        M
62 LBL 0
```

Canned cycles
Variable-contour pockets
Example

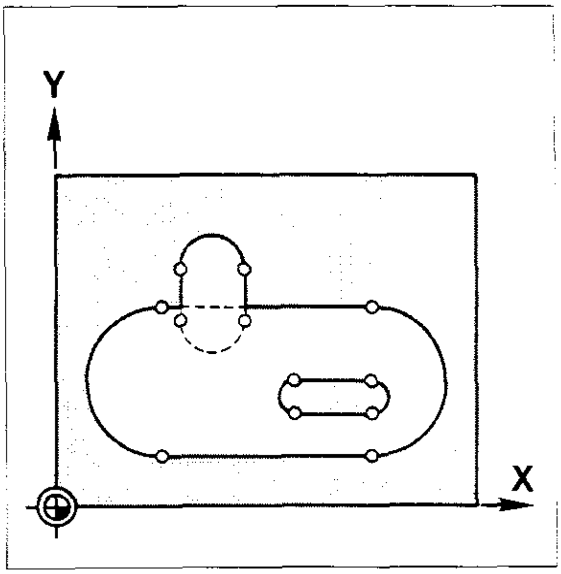
Modifying a pocket contour

In the preceding example, subcontour 3, with the sequence of contour elements and the radius compensation RR (block 55), is programmed as an island superimposed on the first subcontour.

By **changing the radius compensation** for subcontour 3 from RR to RL, the island becomes a pocket. The resulting pocket contour increases in size accordingly.



```
54 LBL   13

55 L   X+40.000   Y+47.000

                  RR              M
```

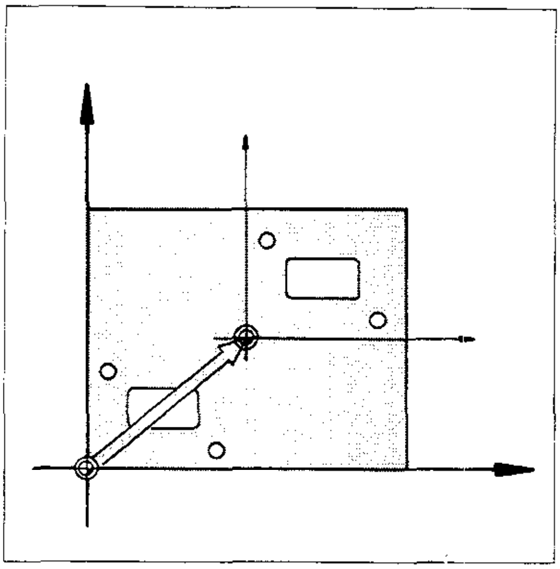


Notes:

Canned cycles
Datum shift

Cycle

The datum (zero point) can be shifted to any location within a program.
This feature allows you to carry out identical machining operations (e.g. milling slots or pockets) at various locations on a workpiece, without having to create and enter a new program for each job.



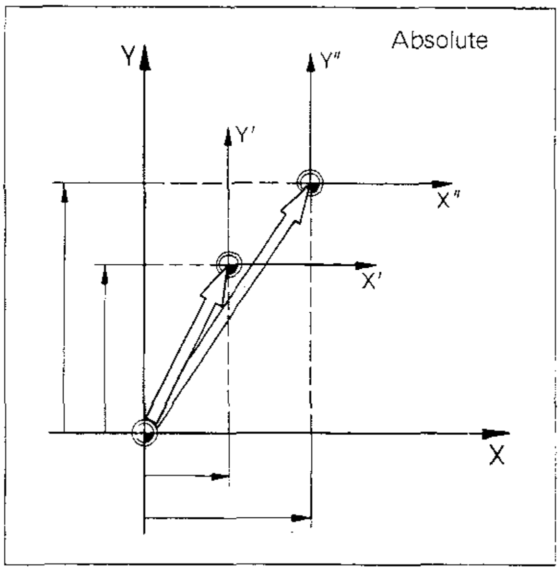
Datum shift

for a datum shift, also called a zero offset, you need only enter the coordinates of the new datum or zero point.
The control then shifts the coordinate system, with the axes **X, Y, Z** and the **4th axis,** to the new, offset datum. All subsequent coordinate data are then based on the new datum.

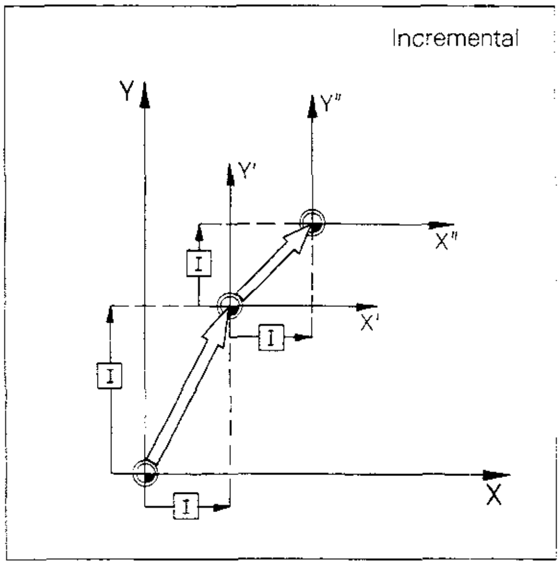
Incremental – absolute

The coordinates can be entered as follows when defining the cycle:

- **Absolute:** the coordinates of the new datum are based on the original workpiece datum ⊕.



- **Incremental:** the coordinates of the new datum are based on the previously valid datum, which may have already been shifted.



Cancelling a datum shift

To cancel a programmed datum shift:

- Enter an absolute datum shift with X 0.000/ Y 0.000/Z 0.000/IV. 0.000;
- Enter the auxiliary function M02, M30 or block END PGM ... MM (depending on specified machine parameters).

Canned cycles
Datum shift

Cycle definition

Operating mode ______ ⟐

Dialogue initiation ______ CYCL DEF ↓ or GOTO □ 7 ENT

CYCL DEF 7 DATUM SHIFT	▶ ENT	Press ENT to select cycle.

DATUM SHIFT ?	▶ X	Select axis.
	▼ I	Incremental – absolute?
	▼ □	Enter coordinates of new datum.
Numerical values can be assigned to **all axes X, Y, Z, IV.** for datum shift.	▼ Y	
After entering the coordinates of the new datum:	▶ ENT	Press ENT.

The cycle "Datum shift" is active immediately after cycle definition. The on-screen status display shows the shift, based on the workpiece datum.

Sample display

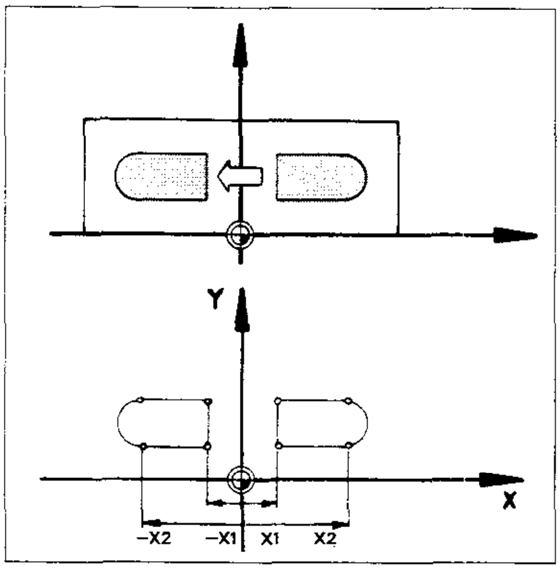
```
10 CYCL DEF 7.0 DATUM SHIFT

11 CYCL DEF 7.1 X + 20.000

12 CYCL DEF 7.2 Y + 10.000

13 CYCL DEF 7.3 Z + 10.000

14 CYCL DEF 7.4 C + 90.000
```

Cycle definition "Datum shift" occupies up to 5 program blocks.

Canned cycles
Mirror image

Cycle

Mirror-imaging an axis about the datum reverses the direction of the axis and changes the +/− sign of all coordinates of the axis. This produces a mirrored (reversed) image of a programmed contour or hole pattern. Mirror imaging is possible only in the machining plane, by reversing one or both axes simultaneously.



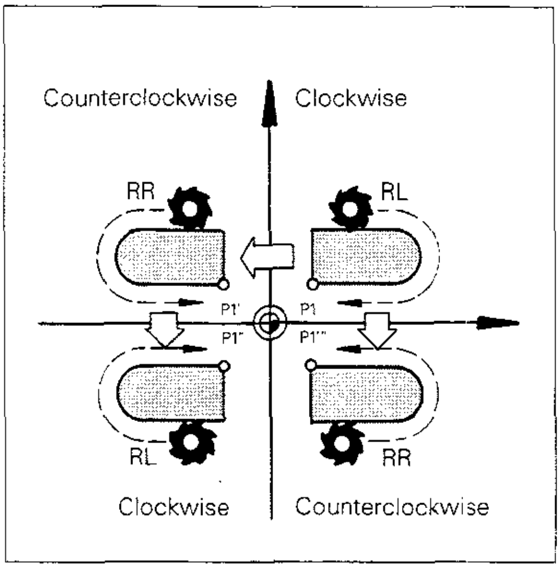
Mirror axis

The axis or axes to be mirrored are programmed for mirror imaging. When specifying coordinates in the program, the signs for the axes are reversed.
If the tool axis is mirror-imaged, the error message:
= MIRROR IMAGE ON TOOL AXIS =
is displayed.

Machining direction

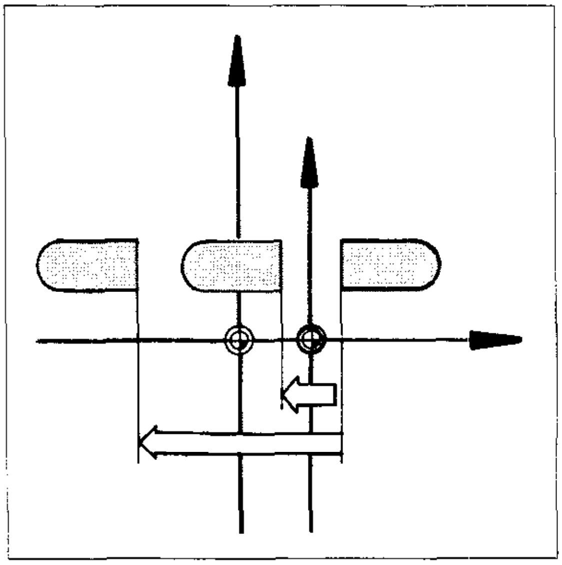
Mirror-imaging across one axis: The machining direction is reversed along with the signs of the coordinates. If a contour was originally milled counterclockwise, mirror-imaging will cause it to be machined clockwise. The milling direction does not change in canned cycles.

Mirror-imaging across two axis: The contour mirrored across one axis is reversed again across a second axis; the machining direction is also reversed a second time, maintaining the original direction.



Datum

When programming, make sure that the mirrored coordinate axis is located exactly between the mirrored contour and the contour to be produced by mirror-imaging. Program a datum shift prior to cycle definition if required.



Cancelling mirror-imaging

To cancel mirror-imaging:

- Program the cycle "Mirror-image", responding to the dialogue prompt by pressing NO ENT.
- Program the auxiliary function M02, M30 or the block END PGM ... MM (depending on specified machine parameters).

Canned cycles
Mirror image

Cycle definition

Operating mode ______________ ◇

Dialogue initiation ______________ CYCL DEF ↓ or GOTO □ 8 ENT

CYCL DEF 8 MIRROR IMAGE	▶ ENT	Press ENT to select cycle.

▼

MIRROR IMAGE AXIS ?	▶ X	Specify axis to mirror-image, e.g. X.
To mirror-image **across two axes simultaneously:**	▼ Y	Specify second axes to mirror-image, e.g. Y.
	▼ END □	Press END to select axes and terminate entry.

Always press END □ to terminate the entry of axis directions or axes without numerical values.
If the entry of the axis or axes is concluded by pressing ENT, the error message
= WRONG AXIS PROGRAMMED =
will be displayed.

The cycle "Mirror image" is active immediately after cycle definition. The mirrored axes are highlighted in the status display for datum shift.

Sample display

```
120 CYCL DEF 8.0 MIRROR IMAGE

121 CYCL DEF 8.1 X
```

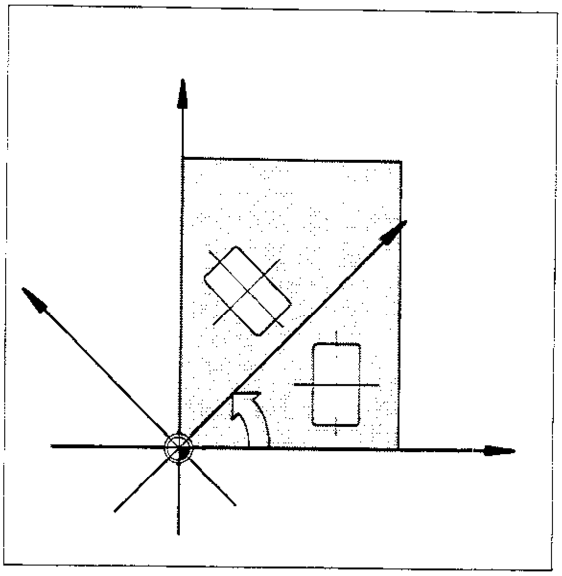
Cycle definition "Mirror image" occupies 2 program blocks.
Mirrored axis: X. Signs for X-coordinates are reversed in following program blocks.

Canned cycles
Rotating the coordinate system

Cycle

The coordinate system can be rotated about the datum in the machining plane within a program.

This feature makes it possible to mill pockets whose sides are not parallel to the original coordinate axes, without calculating effort.

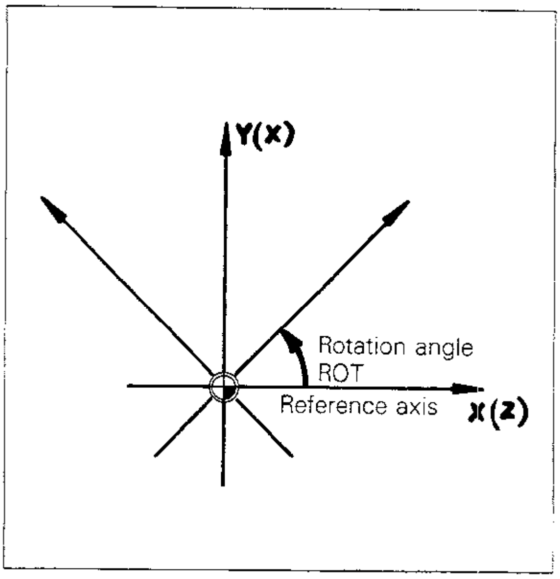


Rotation angle

Only the **rotation angle ROT** need be programmed for rotation.
The rotation angle is always based on the datum of the coordinate system – the centre of rotation.
The **reference axis** for programming in absolute dimension is:

- the **+ X-axis** in the **X, Y plane,**
- the **+ Y-axis** in the **Y, Z plane,**
- the **+ Z-axis** in the **Z, X plane.**

All coordinate data following the rotation are based on the datum with the rotated coordinate system.
The rotation angle can also be entered in incremental dimensions.



Input range

The rotation angle is entered in degrees (°). Input range: −360° to +360° (incremental and absolute).

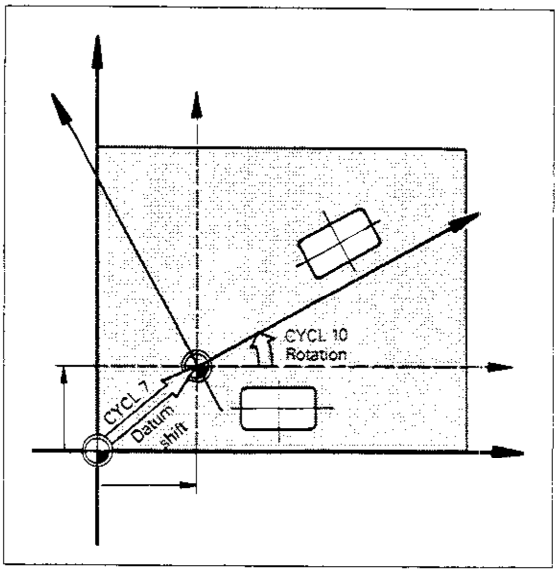
Rotation and datum shift

The "Rotation" and "Datum shift" cycles can be combined by programming them one after another. This makes a simultaneous shift and rotation of the coordinate system possible.

Cancelling rotation

To cancel coordinate system rotation:

- Program rotation with rotation angle 0° (ROT 0).
- Program the auxiliary function M02, M30 or the block END PGM ... MM (depending on specified machine parameters).



Canned cycles
Rotating the coordinate system

Cycle definition

Operating mode ______ ⇨

Dialogue initiation ______ CYCL DEF ↑ or GOTO □ 1 0 ENT

CYCL DEF 10 ROTATION	▶ ENT	Press ENT to select cycle.

▼

ROTATION ANGLE ?	▶ □	Specify rotation angle.
	▼ I	Incremental-Absolute.
	▼ ENT	Press ENT.

The cycle for coordinate sytem rotation is active immediately after cycle definition. The absolute rotation angle is indicated in the status display by "ROT ...".

Sample display

```
184 CYCL DEF 10.0 ROTATION

185 CYCL DEF 10.1 ROT + 45.000
```

Cycle definition "Rotation" occupies 2 program blocks.

Rotation angle in degrees (°).

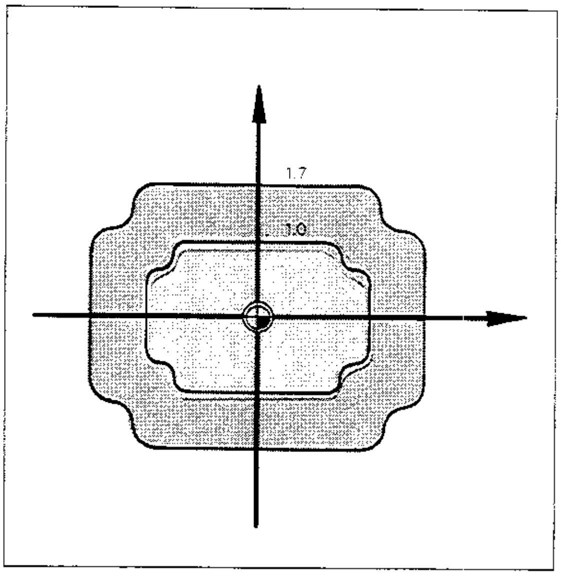
Canned cycles
Scaling factor

Cycle

Contours in the machining plane can be enlarged or reduced in size within a program.

This feature makes it possible to create similar geometrical contours without having to re-program them and to program shrinkage and oversize allowances.

Depending on the specified machined parameters, the scaling factor functions either in the machining plane or on the three main axes.
Contact your machine manufacturer or supplier for information.

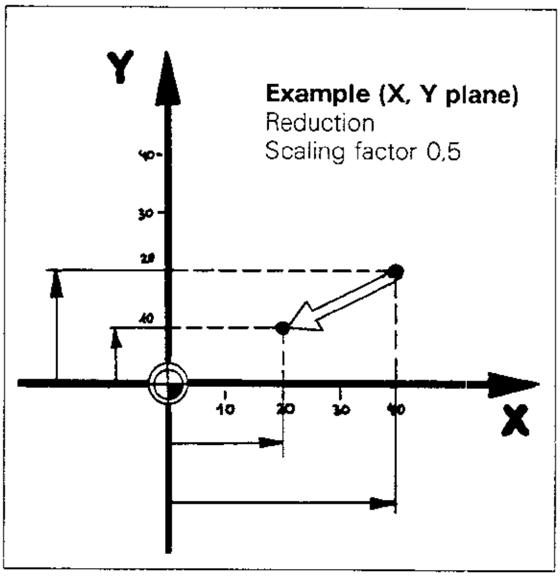


Scaling factor

To enlarge or reduce the size of a contour, program the scaling factor SCL.
The control system multiplies all coordinates and radii of the machining plane or all three axes X, Y and Z (independent of a machine parameter), by this factor that are executed following the cycle.
Input range: 0 to 99.999999.

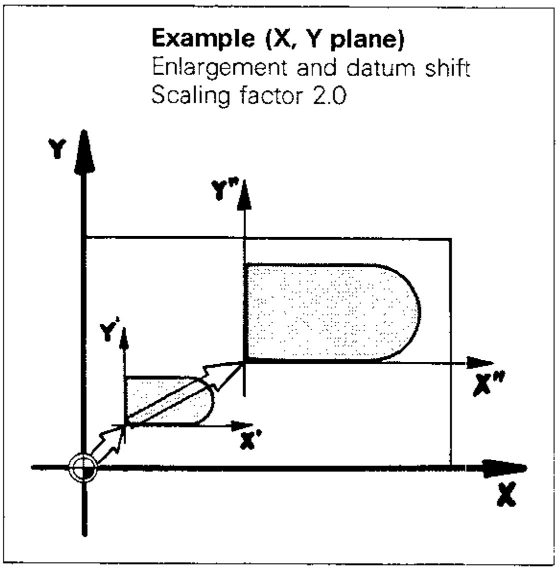
Location of datum

The position of the datum of the coordinate system does not change when the contour is enlarged or reduced in size.
To produce a geometrically similar contour at the intended location on the workpiece, it may be necessary to program a datum shift and/or a rotation of the coordinate system.
Before programming the scaling factor, werecommend locating the datum at a corner of the contour, which saves calculating effort.


Example (X, Y plane)
Reduction
Scaling factor 0,5

Cancelling the scaling factor

To cancel the cycle "Scaling":
- Program "Scaling" cycle with factor 1.0.
- Program the auxiliary function M02, M30 or the block END PGM … MM (depending on specified machine parameters).


Example (X, Y plane)
Enlargement and datum shift
Scaling factor 2.0

Canned cycles
Scaling factor

Cycle definition

Operating mode ______ ◇

Dialogue initiation ______ CYCL DEF ↑ or GOTO □ 1 1 ENT

CYCL DEF 11 SCALING	▶ ENT	Press ENT to select cycle.

▼

FACTOR ?	▶ □ ▼ ENT	Specify scaling factor. Press ENT.

The cycle "Scaling" is active immediately after cycle definition. The scaling factor is indicated in the status display by "SCL ...".

Sample display

```
12 CYCL DEF 11.0 SCALING

13 CYCL DEF 11.1 SCL 0.750000
```

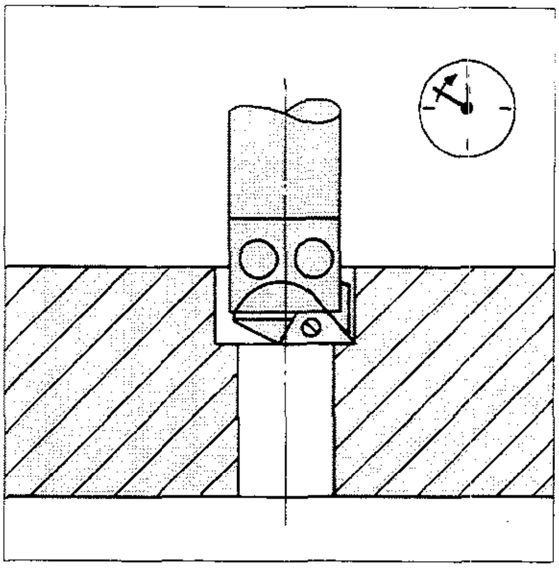
Cycle definition "Scaling" occupies 2 program blocks.

All subsequent coordinate data are reduced by 0.75 by programming the scaling factor 0.75.

Canned cycles
Dwell time

Cycle

The cycle "Dwell time" can be used within a program to interrupt the feed motion for a specified period of time while the spindle is still running, e.g. for chip breaking with single-point boring operations. The cycle "Dwell time" is run immediately after cycle definition.



Input range

Dwell time is indicated in seconds.
Input range: 0.000 s to 19,999.999 s.

Entering 19,999.999 s results in an operating pause of 5.5 hours!

Canned cycles
Dwell time

Cycle definition

Operating mode ______ ⟐

Dialogue initiation ______ CYCL DEF ↑ or GOTO □ 9 ENT

CYCL DEF 9 DWELL TIME	▶ ENT	Press ENT to select cycle.
DWELL TIME IN SEC. ?	▶ □	Specify required dwell.
	ENT	Press ENT.

The cycle "Dwell time" is run immediately after cycle definition.

Sample display

```
97 CYCL DEF 9.0 DWELL TIME

98 CYCL DEF 9.1 DWELL 10.000
```

Cycle definition "Dwell time" occupies 2 program blocks.

Canned cycles
Freely programmable cycles (Program call)

Cycle

The cycle "Program call" makes it more convenient to call programs, via CYCL CALL, M89 and M99, that were created with the aid of parameter functions, such as area clearance cycles. This gives these freely programmable (variable) cycles the same status as the pre-programmed canned cycles.

Canned cycles
Freely programmable cycles (Program call)

Cycle definition

Operating mode ______________________ ◇

Dialogue initiation ____________________ CYCL DEF ↓

CYCL DEF 12 PGM CALL	▶ ENT	Press ENT to select cycle.
PROGRAM NUMBER ?	▶ □	Enter program number.
	▼ ENT	Press ENT.

Sample display

```
5 CYCL DEF 12.0 PGM CALL
6 CYCL DEF 12.1 PGM 23
```

The called cycle is programmed in program 23.

Canned cycles
Spindle orientation (optional)

Introduction

Used as a 5th axis, the TNC can control the main spindle of a machine tool and rotate it into a specified position.
Applications of the spindle orientation feature include certain tool change systems that require the tool to be in a defined position relative to the changer, or for aligning the transmitter/receiver window of the HEIDENHAIN TS 510 infrared touch-probe system.

Defining position

Spindle positioning (orientation) is activated via an auxiliary function. The position can be defined by means of

- machine parameters, or
- cycle 13 "Spindle orientation".

Contact your machine manufacturer or supplier for more information on machine parameters and the auxiliary function.

Cycle

Cycle 13 "Spindle orientation" can be used to program a specified angular spindle position. An auxiliary function, defined by the machine manufacturer, must be programmed before spindle positioning is possible (not with CYCL CALL).

If spindle orientation is called via an auxiliary function, without having programmed a cycle definition, the TNC will align the main spindle according to a value defined in the machine parameters.

Input data

Orientation angle: angle relative to angular reference axis of machining plane.
Input range: 0 ... 360°
Input resolution: 0.5°

Display of the spindle position actual value

The spindle position actual value can be displayed in place of the fourth axis.
This is dependent on machine parameter 237 and is stipulated by your machine manufacturer.

Canned cycles
Spindle orientation (optional)

Definition

Operating mode ______________________ ◇

Dialogue initiation ______________________ CYCL DEF ↑

CYCLE DEF 13 ORIENTATION	▶ ENT	Press ENT to select cycle.
ORIENTATION ANGLE ?	▶ □	Specify required angular spindle position.
	▼ ENT	Press ENT.

Sample display

```
5 CYCL DEF 13.0 ORIENTATION

6 CYCL DEF 13.1 ANGLE 90.000
```

Cycle definition occupies 2 program blocks.

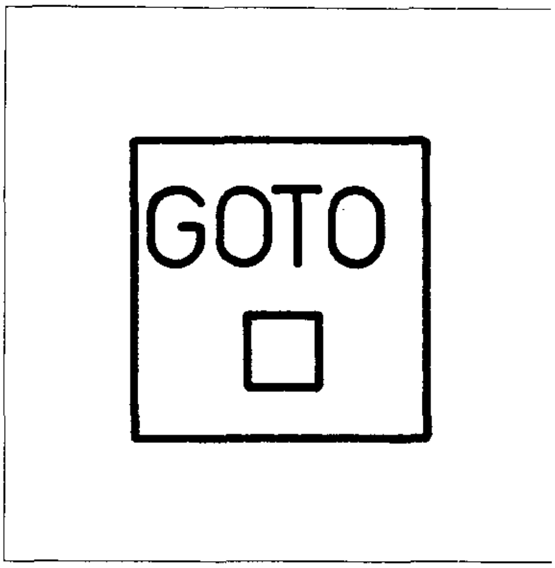
Editing a program

Editing

Editing is the term used to describe the process of checking, modifying or expanding a program. The editing functions help you to search for and modify program blocks and words. They are activated by pressing a key.

Calling a block

Call a specified block by pressing the GOTO □ key. □ is the symbol for program block.

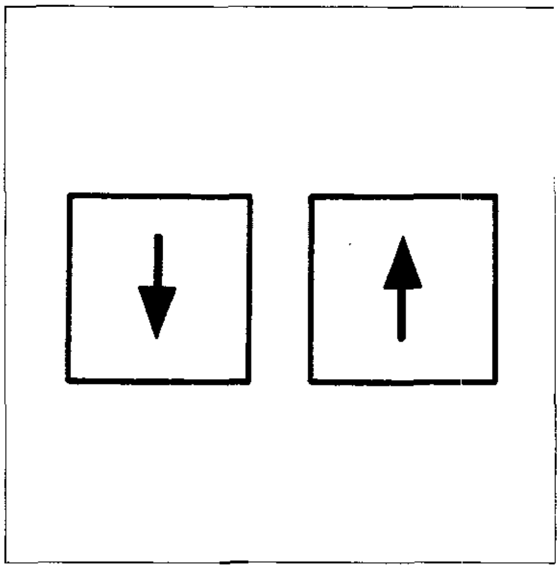


Paging through a program

You can "page" through a program block-by-block with the ↓ and ↑ keys.

↑ skip to next lower block number

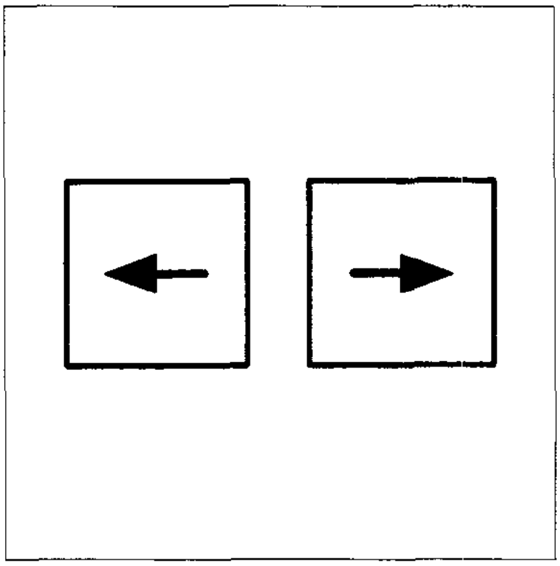
↓ skip to next higher block number



Editing words

Pressing the ← and → keys moves the **cursor** around within the current block. The cursor is an "editing pointer" in the form of a highlighted field on your screen. Use the two cursor movement keys to place the cursor on the program word you want to edit.

The cursor can be moved only in ◇ mode. Cursor movement must be started with the → key.



Editing a program
Calling a block

Calling a block number

Operating mode ______

Dialogue initiation ______ GOTO □

GOTO: NUMBER = ▶ □ Specify block number.

▼

ENT Press ENT.

Editing words

Operating mode ______

To edit a word in the current program block: ▶ → Place cursor on word to be edited.

▼

The dialogue prompt for the word appears highlighted, e.g.:

COORDINATES ? ▶ X Edit entry data.

▼

□

When all corrections are complete: ▶ END □ Press END to enter block (or move cursor off-screen to right or left).

To edit an additional word: ▶ ← → Place cursor on word to be edited.

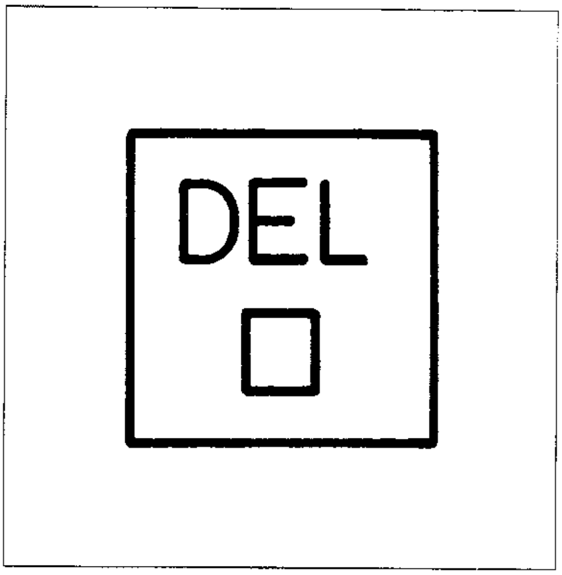
Editing a program
Deleting and inserting blocks

Deleting a block

The current block within a program is deleted by pressing the DEL □ key.
DEL is an abbreviation for **DELete,** which means to erase or take out.
Program blocks can be deleted only in ⇨ mode.

When deleting individual blocks, make sure the block you want to delete is the currently active block. To be on the safe side, call up the block by number.

After the block is deleted, the block with the next block number takes the place of the deleted block.
The blocks are re-numbered automatically.



Cycle definition or program part deletion

To delete cycle definitions or program sections, call up the last block of the cycle definition or program section. Then press the DEL □ key until all blocks in the definition or program section have been deleted.

Inserting a block

You can insert new blocks at any point in an existing program by calling up the block **preceding** the block you wish to insert.
The subsequent blocks will be re-numbered automatically.

If the storage capacity of the program memory is exceeded, the following error message will appear when the dialogue is initiated:
= PROGRAM MEMORY EXCEEDED =
This error message is also generated if you attempt to insert a block after the END block (end-of-program is displayed on current line).

Editing while programming

Incorrect entries made while programming can be corrected in two ways:

CE Entered data are erased and a highlighted "0" appears.

NO ENT Entered data are erased completely.

Editing a program
Deleting blocks

Deleting a block

Operating mode ______________________ ⇨

To delete the current program block:	▶ DEL □	Press	to delete block.

Editing a program
Search routines/Parameter display
Erasing a program

Searching for specified addresses

Blocks containing specified addresses can be found within a machining program by using the ↓ and ↑ keys.
To search for an address, use the → and/or ← keys to move the cursor to the word with the searched address and the ↓ and/or ↑ keys to page through the program. Only those blocks which contain the searched address will be displayed.
Search routines can be carried out only in ◈ mode.

Erasing a program

Press CL PGM to initiate the dialogue for erasing (clearing) a program.
After pressing this key, a program overview appears together with a highlighted pointer. Move the pointer with the ↓ ↑ ← → keys.

Only the program whose number is currently highlighted can be erased.

Editing a program
Search routines/Parameter display
Erasing a program

Searching for specified addresses

Operating mode __________ ⇨

To display all blocks with the address M:

▶ ↓ ↑ Select a block with searched address.

▼ → Move cursor to word with searched address.

▼

AUXILIARY FUNCTION M ?

▶ ↓ ↑ Call blocks containing searched address.

Always initiate cursor movement with the → key.

Erasing a program

Operating mode __________ ⇨

Dialogue initiation __________ CL PGM

CLEAR = ENT/END = NO ENT

To erase a program:

▶ ↓ ↑ ← → Move cursor to program number.

▼ ENT Press ENT to erase program.

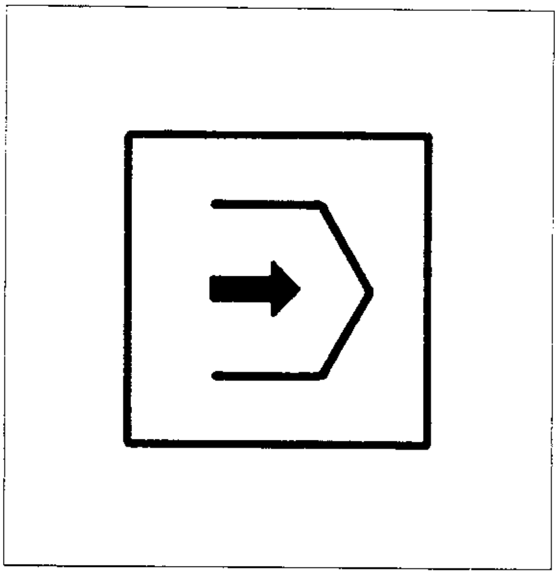
If erase is not desired or to terminate erase function:

▶ NO ENT

Test run
Parameter display

Testing a program

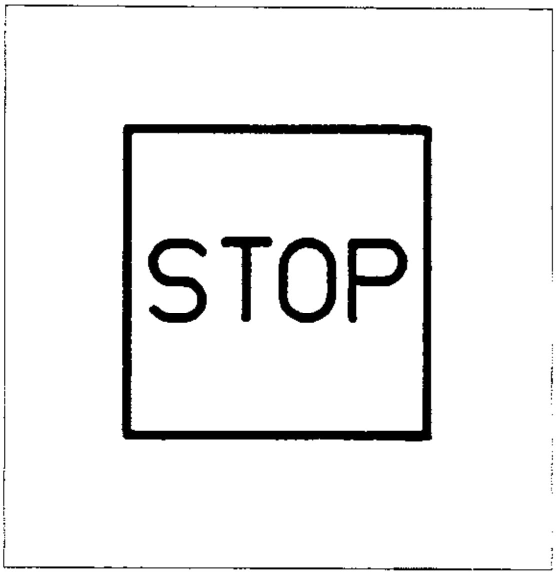
You can have the control system check a program for geometrical errors without machine movement before actually running the program. During a test run, the TNC calculates program sequences just like during an actual program execution. The test run will be interrupted if an error message is generated. The mode select key → also initiates the interactive input dialogue.



Stopping a test run

Interrupt and abort the test run at any desired location by pressing the STOP key.

The test run is interrupted automatically after each programmed STOP. The test run must be restarted to continue (see next page).



Displaying and setting Q parameters

In "Program run-single block" → or "Program run full sequence" → modes, you can display the current values of Q parameters and change them if desired. You have to interrupt the program run at the desired location to do this. Press the Q key and enter the parameter number to display the value on the screen on the dialogue line. You can then change the displayed value if you wish (e.g. for test runs).
The TNC will retain the set value until another one is programmed to replace it.
You can page through the parameter list with the ↓ and ↓ keys.
The display is erased with the DEL key.

Test run
Parameter display

Starting a program test run

Operating mode ______ →

TO BLOCK NUMBER =

To run test up to a specified block number:	▶ □	Enter block number.
	▼ ENT	Press ENT.
To test run entire program:	▶ NO ENT	

Displaying and setting Q parameters

Operating mode ______ → →

Dialogue initiation ______ Q

Q0 =	▶ □	Specify parameter number.
	▼ ENT	Press ENT.

▼

Q55 = 1112	▶ □	Enter parameter value if required.
	▼ ENT	Press ENT.

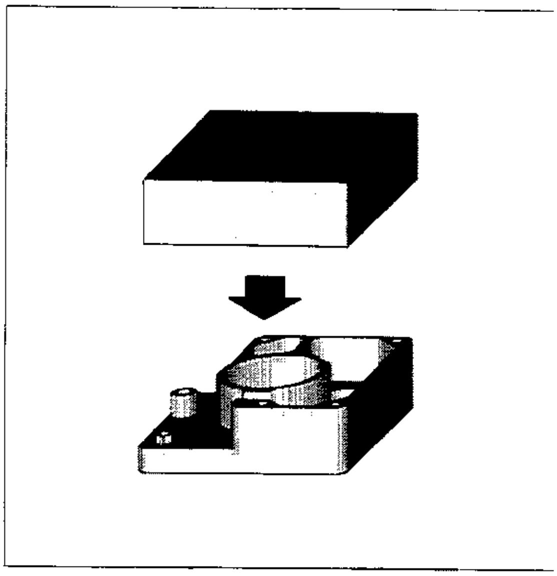
After the program starts, the TNC operates with the displayed or modified parameter value until it is replaced in the program by another value.

Graphics *
Blank form definition

Graphic simulation

The machining of a workpiece can be simulated graphically on the screen, to check the program without machine-slide movement before it is actually executed.

The workpiece blank is always of cuboid shape. Other workpiece shapes may be programmed separately if desired.



Machining operations can be simulated on the three main axes, with constant tool axis and a cylindrical end mill. Helical interpolation as well as interpolation on the 4th axis (e.g C-axis) cannot be simulated.

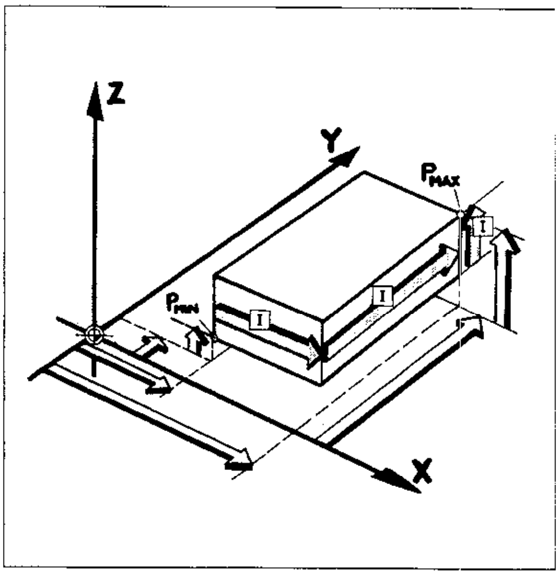
Defining a blank

The blank workpiece has to be defined for graphic simulation:

- its **position relative to the coordinate system** and
- its **dimensions** must be programmed.

It is necessary to specify only **corner points** when defining the cuboid. These points are identified as minimum point P_{MIN} and maximum point P_{MAX} (points with "minimum" and "maximum" coordinates).
P_{MIN} can only be entered in absolute dimensions.
P_{MAX} can be entered in either absolute or incremental dimensions.
The blank workpiece data are saved in a corresponding machining program and are available when the program is selected.
It is advisable to define the cuboid at the beginning or at the end of the program. This makes it easier to find the BLK FORM blocks when blank form dimensions change.

The interactive dialogue is initiated by pressing BLK FORM



The **maximum dimensions** of the blank may not exceed 14,000 x 14,000 x 14,000 mm.

* The graphics feature is available only on control system model TNC 155.

Graphics
Cuboid corner points – BLANK FORM

Entering corner points

Operating mode		
Dialogue initiation	BLK FORM	

SPINDLE AXIS PARALLEL X/Y/Z ?	Z	Specify spindle axis, e.g. Z.

DEF BLK FORM: MIN-CORNER ?	□	Enter numerical value for X-coordinate.
	ENT	Press ENT.
	□	Enter numerical value for Y-coordinate.
	ENT	Press ENT.
	□	Enter numerical value for Z-coordinate.
	ENT	Press ENT.

DEF BLK FORM: MAX-CORNER ?	I	Incremental – absolute?
	□	Enter numerical value for X-coordinate.
	ENT	Press ENT.
	I	Incremental – absolute?
	□	Enter numerical value for Y-coordinate.
	ENT	Press ENT.
	I	Incremental – absolute?
	□	Enter numerical value for Z-coordinate.
	ENT	Press ENT.

Sample display

```
1 BLK FORM 0.1    Z X+  0.000
        Y+  0.000    Z–15.000
2 BLK FORM 0.2      X+80.000
        Y+100.000    Z+  0.000
```

The blank is parallel to the main axes.
The coordinates of P_{MIN} are X 0.000, Y 0.000 and Z –15.000.
The coordinates of P_{MAX} are X 80.000, Y 100.000 and Z 0.000.

Graphics
Display options

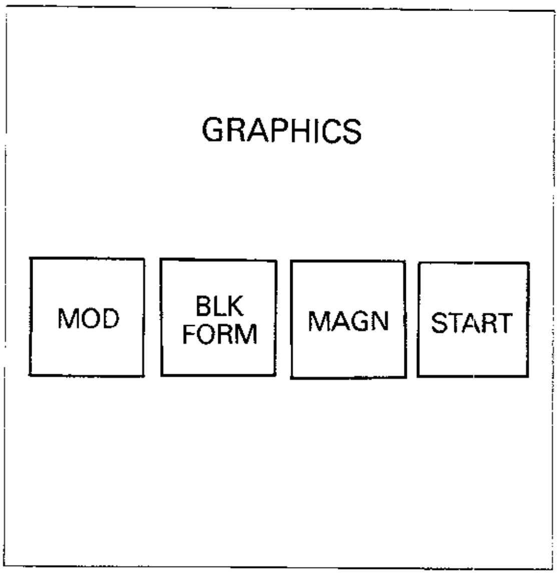
Graphics mode

A machining program can be simulated graphically in operating modes:

→ PROGRAM RUN – FULL SEQUENCE

⇛ PROGRAM RUN – SINGLE BLOCK

The machining program must be stored in the main memory before it can be displayed.
To call up the menu of display options on the screen, press MOD twice. Use ↑ and ↓ to move the highlighted pointer to the desired display option and press ENT to select.

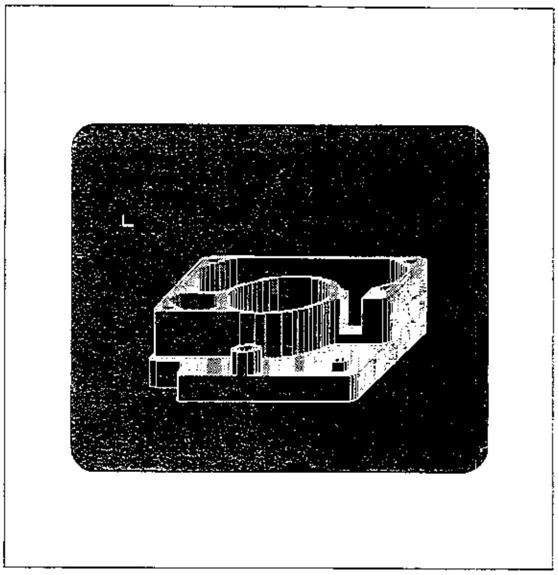


Display modes

Four different types of display are available.

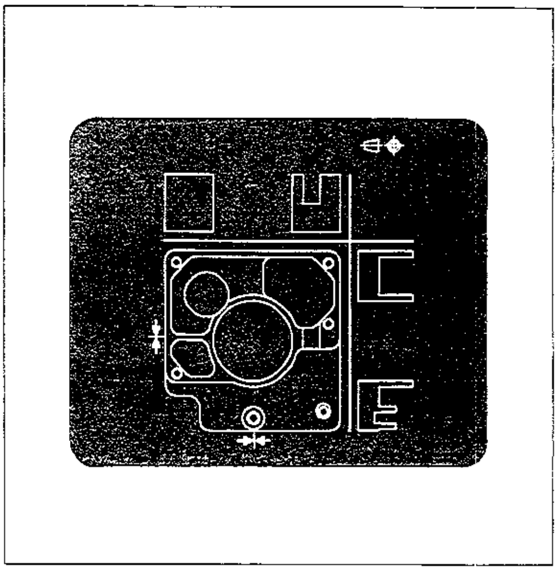
3D simulation

The program is run in 3D simulation. Use the → and ← keys to rotate the workpiece about its vertical axis and the ↑ and ↓ keys to tilt it about the horizontal axis.
The position of the coordinate system (machining plane) is indicated by an angle displayed at the upper left of the display.



Simulation in three planes

The program is run in plan view and two cross-sections, similar to a workpiece drawing. The sectional planes can be shifted by pressing the → ← ↑ ↓ keys.



The simulation in three planes can be switched from the standard German DIN display to the American standard third angle projection via machine parameters. Symbols conforming to the DIN 6 standard indicate the type of display:

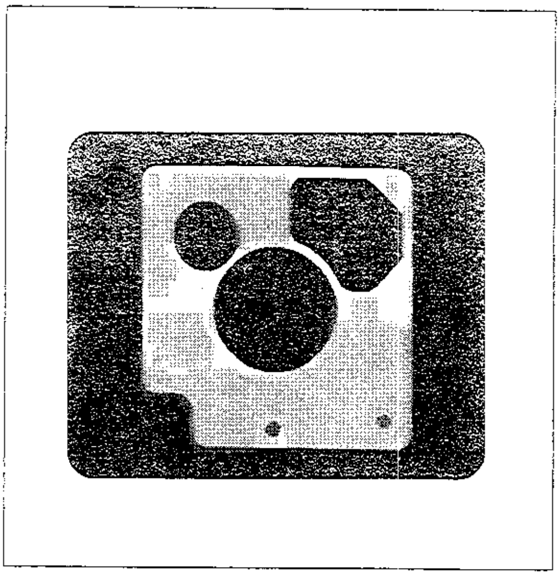
DIN standard ⊐⊕

U.S. standard ⊕⊐

Graphics
Display options

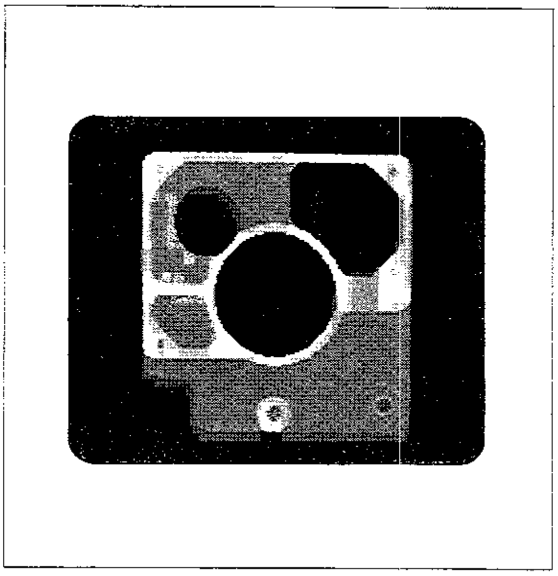
Plan view 1

The program is simulated in a plan view with **five levels of depth shading,** the deeper the level, the darker the shading.



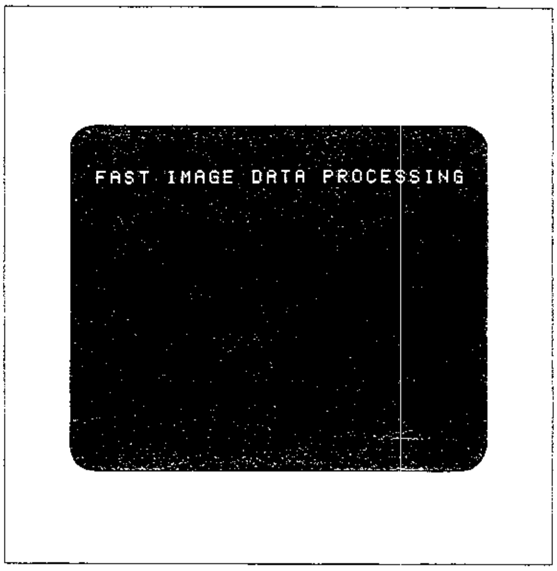
Plan view 2

Same as plan view 1, but features **17 levels of depth shading.** Image resolution on the other two axes is not as good.



Fast image generation

The finished workpiece can be displayed on the screen with the **fast image data processing** feature.
The TNC "develops" the workpiece as configured in the machining program, without graphically simulating individual production steps. Only the current block number is displayed on the screen.



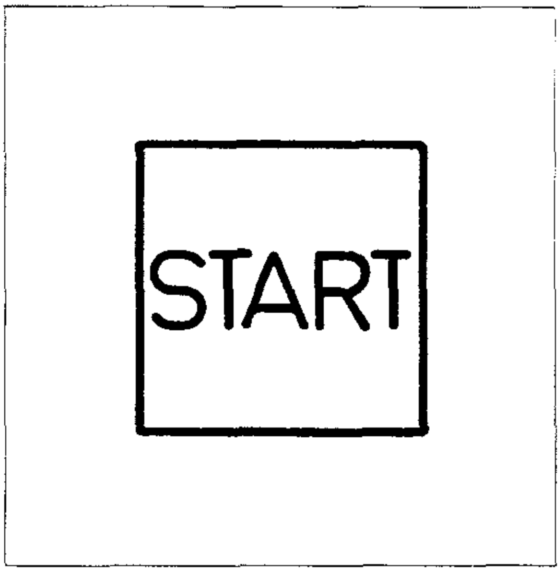
FAST IMAGE DATA PROCESSING

Graphics
Operation

Starting graphic simulation

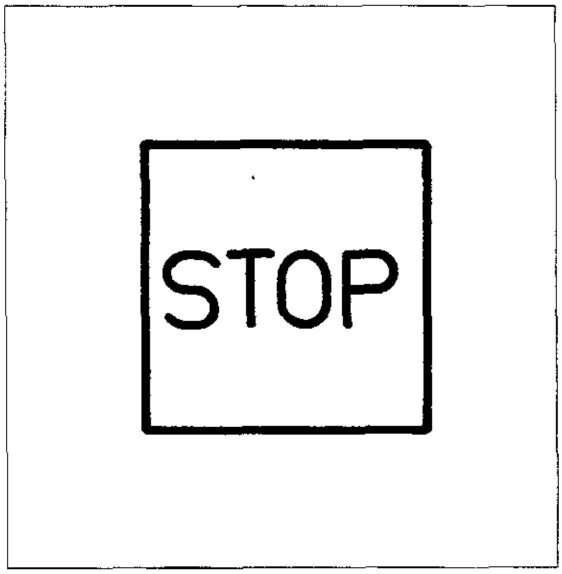
After selecting the desired graphics mode, start the program run by pressing START.

A TOOL CALL, defining the tool axis, must be programmed prior to the initial axis movement. Specifying the spindle axis during BLK FORM definition is not sufficient for running a program in graphics mode. If the tool axis is missing, the error message
= PGM SECTION CANNOT BE SHOWN =
appears after the graphics feature is started. This error message is also displayed if a fourth axis or helical interpolation was programmed.



Stopping graphic simulation

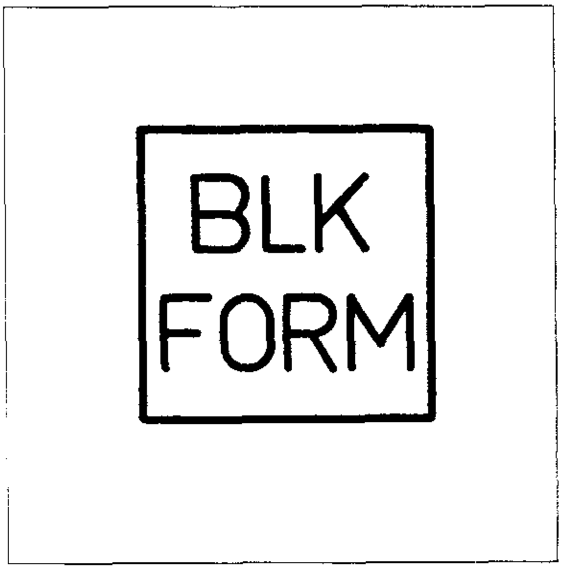
You can interrupt the graphic simulation at any time by pressing STOP. The current block will be completed.



Resetting the blank form

After interrupting the graphic simulation, reset the display to the blank workpiece (original cuboid shape) by pressing BLK FORM.

To restart the simulated machining of the workpiece, first return to the beginning of the program by pressing GOTO □.

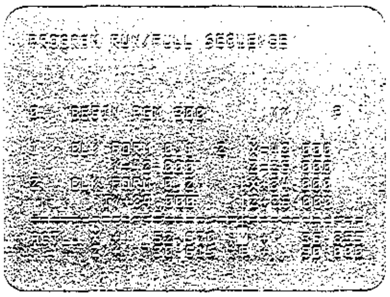


Graphics
Starting graphics mode

Starting graphics mode

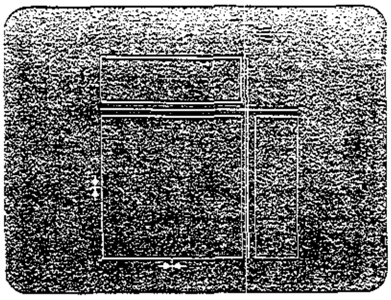
Operating mode ______________________ or

Select desired program. ▶ PGM NR ▼ □ ▼ ENT ▶

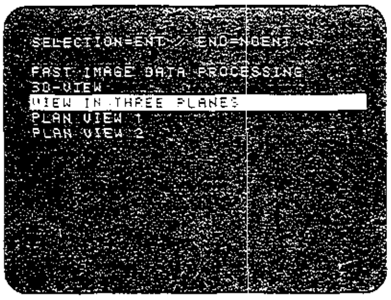

PROGRAM RUN/FULL SEQUENCE

[illegible]

Select graphics mode. ▶ MOD ▶

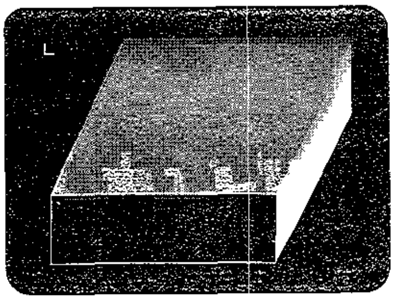


Call up graphics menu. ▶ MOD ▶

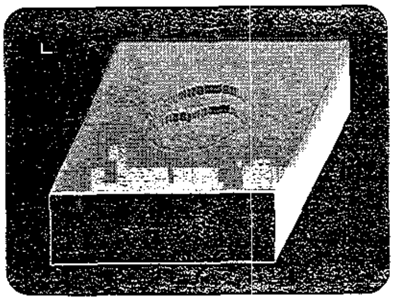

SELECTION=ENT / END=NOENT

FAST IMAGE DATA PROCESSING
3D-VIEW
VIEW IN THREE PLANES
PLAN VIEW 1
PLAN VIEW 2

Move cursor to desired display option, e.g. "3D simulation". ▶ ↑ ▼ ENT ▶



Start graphic simulation. ▶ START ▶

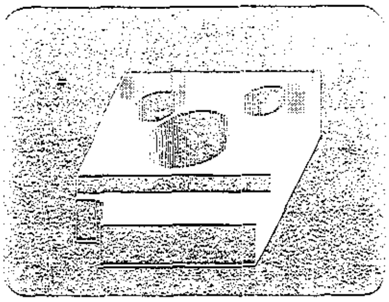


Notes:

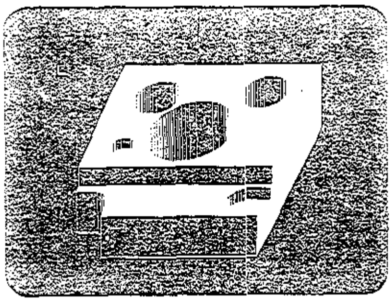
Graphics
Starting graphics mode

Graphics: stop/start

To interrupt graphic simulation: ▶ STOP ▶



To restart graphic simulation: ▶ START ▶



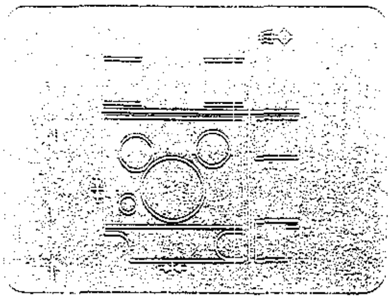
Notes:

Graphics
Simulation in three planes

Shifting planes

Interrupt graphic simulation.

▶ STOP ▶



▼

To shift horizontal sectional plane, e.g. up:

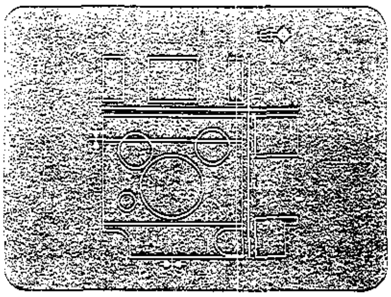
Press ↑ repeatedly (jog mode)

▶ ↑ ↑ ▶

or continually shift sectional plane.

▶ ↑ ENT ▶

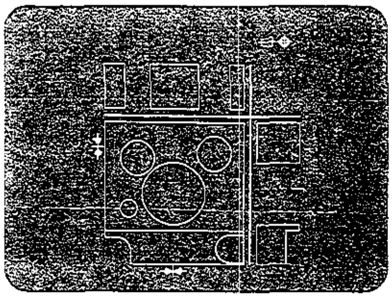
Press ENT repeatedly to shift plane faster.



▼

To stop shifting plane:

▶ STOP ▶



▼

To shift vertical plane, e.g. to right:

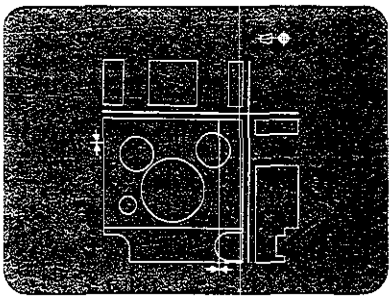
Press → repeatedly

▶ → → ▶

or continually shift sectional plane.

▶ → ENT ▶

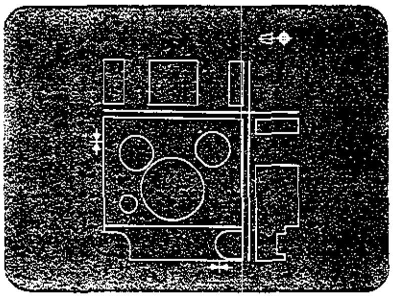
Press ENT repeatedly to shift plane faster.



▼

To stop shifting plane:

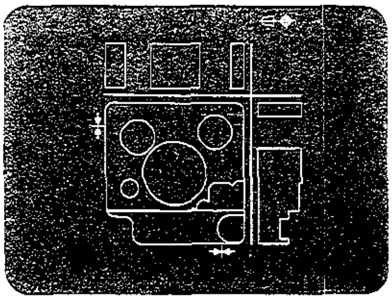
▶ STOP ▶



▼

Graphics
Simulation in three planes

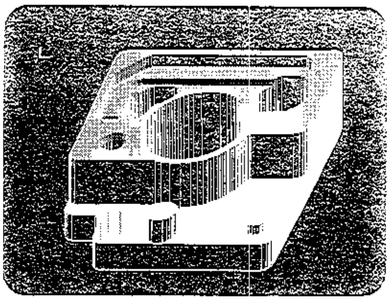
Restart graphic simulation. ▶ START ▶



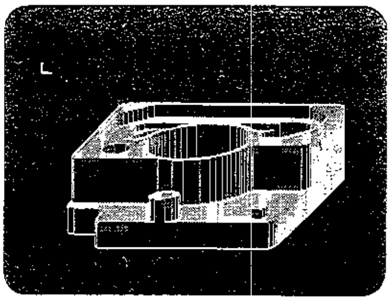
Graphics
3D simulation

Tilting and rotating

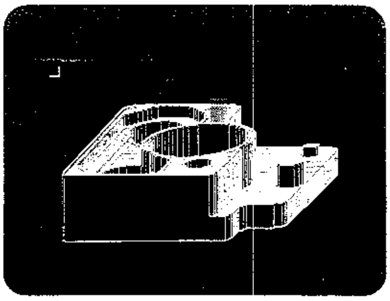
Interrupt graphic simulation. ▶ STOP ▶



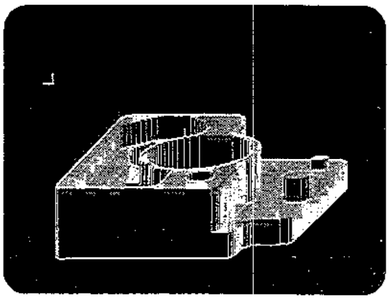
Tilt view, e.g. up: ▶ ↓ ▶



Rotate view, e.g. to right: ▶ ← ▶



Restart graphic simulation. ▶ START ▶



Graphics
Magnify

Magnify function

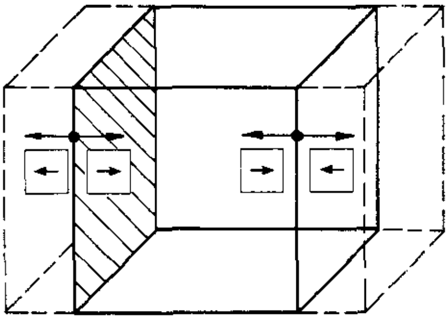
The magnify feature allows you to enlarge any desired detail of the workpiece.

The detail selected for magnification must be defined in the 3D graphics mode.

The simulation itself can be realized in all 4 graphic modes.

Defining limits of workpiece detail

You can define the limits of the selected detail with a wireframe model of a cuboid that appears in the upper left corner of the screen when you press MAGN.
You can use the → key to move the hatched area one point at a time toward the centre of the cuboid or, in conjunction with ENT to move it continuously. Press STOP to interrupt continuous movement.
Press ← to move the area back toward the outer edge.



Defining next boundary surface

Press ↑ to select the next boundary (right-hand surface).
In this way, you can select and move the left-hand, right-hand, front, rear, top and bottom surfaces one at a time.
Press ↓ to return to the preceding surface.

Saving the detail

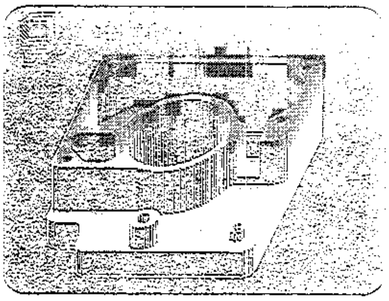
After the last boundary surface (top) has been defined, save the detail by pressing ↑ and then ENT. The blank is displayed on the screen in enlarged form. For a magnified detail of the actual contour, run a graphic simulation in any of the graphic display modes.

Graphics
Magnify

Limiting and enlarging a detail

The control system is in 3D graphic display mode.

Select MAGN function. ▶ MAGN ▶



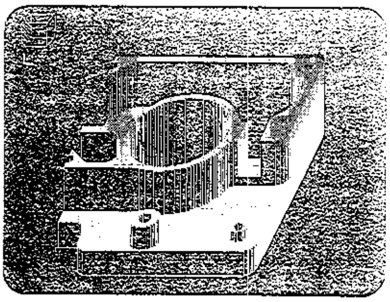
▼

To move left-hand boundary surface, e.g. to right:

Press → repeatedly (jog mode) ▶ → → ▶

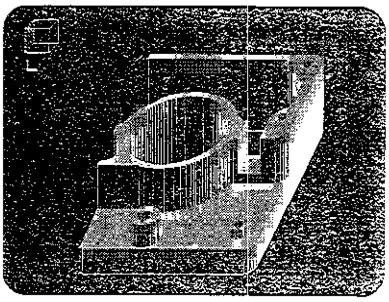
or continually shift boundary surface. ▶ → ENT ▶

Press ENT repeatedly to shift boundary surface faster.



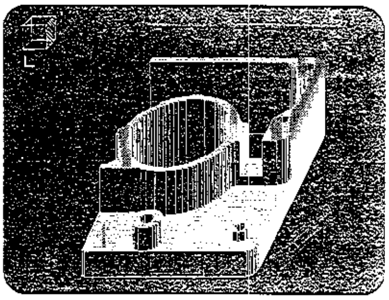
▼

To stop shifting surface and save: ▶ STOP ▶



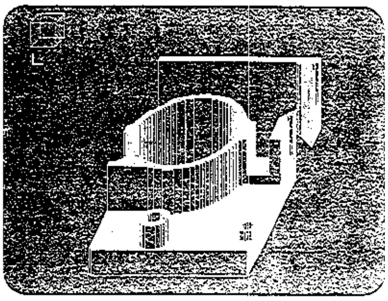
▼

Select next boundary surface (right). ▶ ↑ ▶



▼

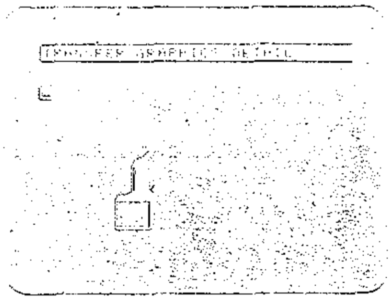
Move this and remaining surfaces as described above.



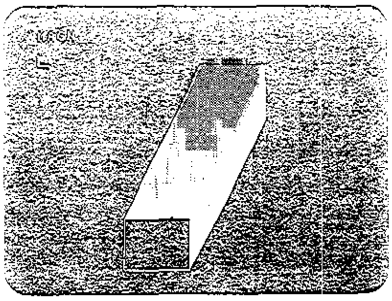
▼

Graphics
Magnify

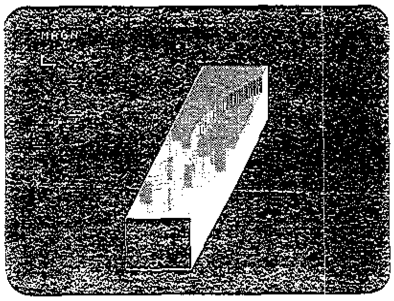
When the final (top) surface has been moved: ▶ ↑ ▶



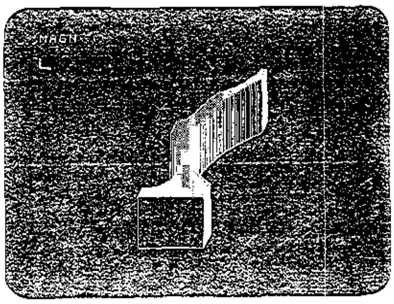
TRANSFER GRAPHICS DETAIL ▶ ENT ▶



Start program run. ▶ START ▶



The workpiece machining operation is simulated. Only the defined detail is displayed on the screen.

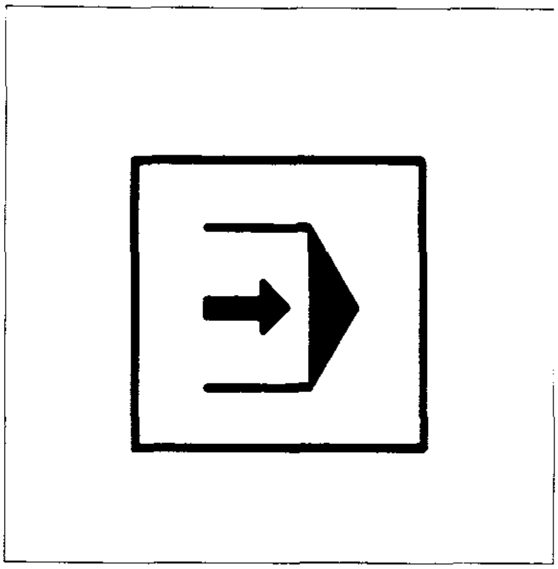


Notes:

Program run
Operating modes

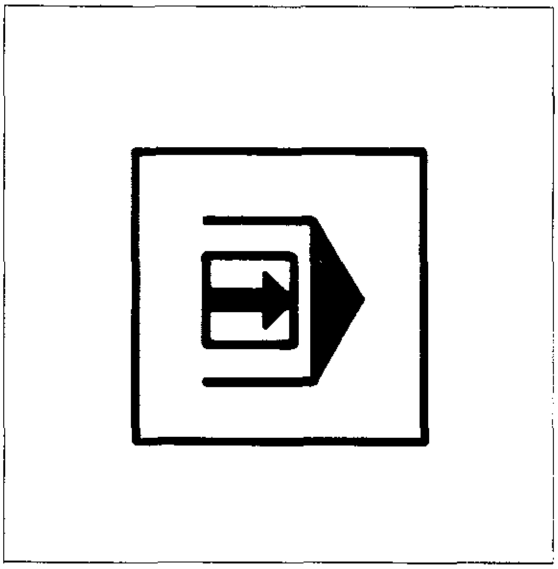
Program run – full sequence

In the operating mode ⇒ "Program run – full sequence", the TNC executes the program stored in memory up to a programmed stop or until the end of the program. After a programmed stop, the program run must be restarted to continue. The program run will also be stopped if the display indicates an error.



Program run – single block

In the operating mode ⇒ "Program run – single block", the TNC executes the program stored in memory block-by-block. The program must be restarted after each block is executed.

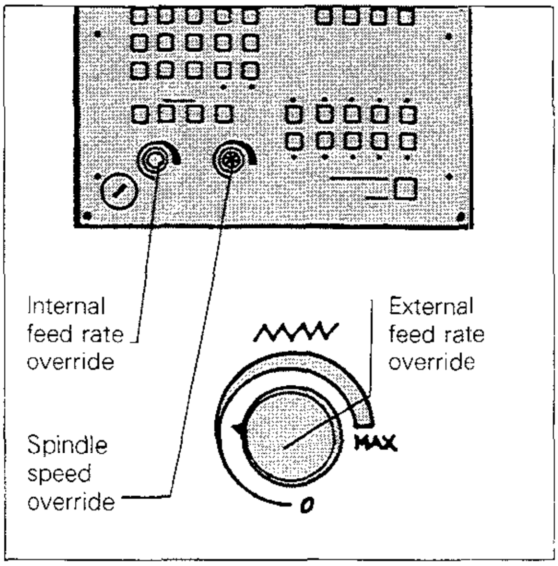


Feed rate

The programmed feed rate can be modified

- via the **internal feed rate override** and/or
- via the **external feed rate override** on the machine,

depending on how the control system was installed on the machine by the machine manufacturer.



Spindle speed

In the case of analogue output, the spindle speed can be modified via the **spindle override.**

Program run
Starting a program run

The workpiece datum must be set before machining the first workpiece.

Starting program run – single block

Operating mode ________________________

First program block displayed on current program line.	▶ START	Run first program block.
Second program block displayed on current program line.	▶ START	Run second program block.

Starting program run – full sequence

Operating mode ________________________

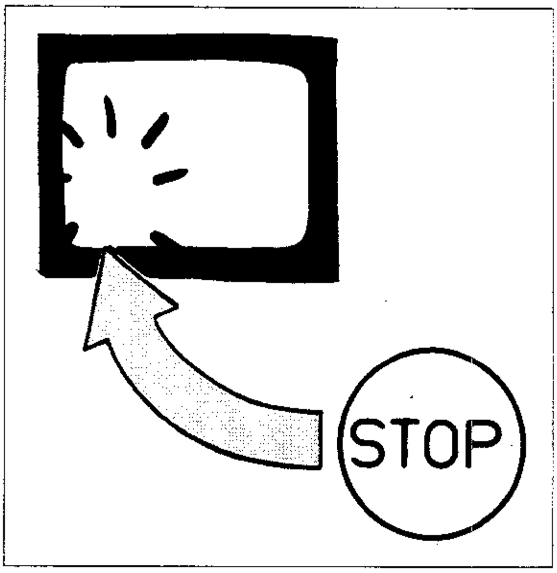
First program block displayed on current program line.	▶ START	Run program.

The TNC executes the program continually, until it reaches a programmed stop or until the end of the program.

Program run
Interrupting and aborting a program run

Interrupt program run

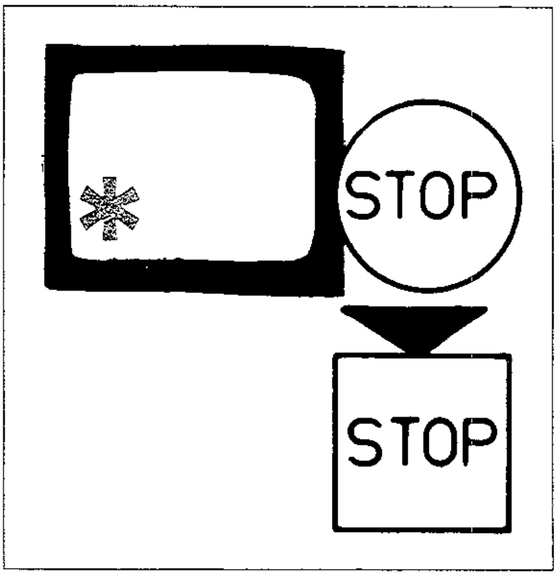
While the TNC is in → (Program run – full sequence) or ≡→ (Program run – single block) modes, you can interrupt program execution at any time by pressing the external stop button. An interruption is indicated on the screen by a flashing ✱ symbol (" ✱ " means control system in operation).



Abort program run

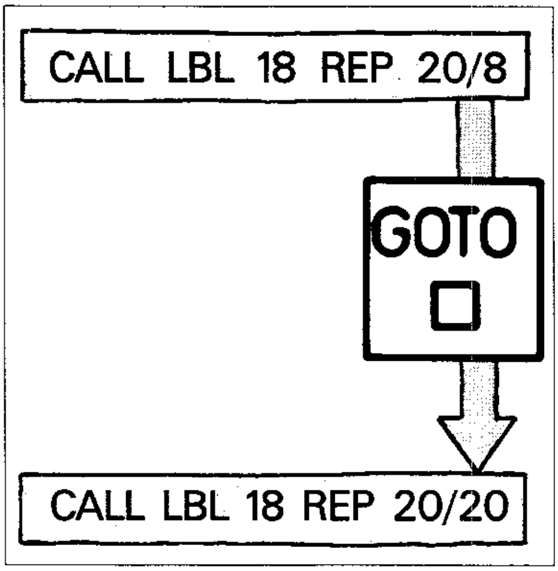
Program execution must be interrupted and aborted before switching to another operating mode (except when running a program with background programming).
To interrupt and abort a program run, press the external stop button and the stop button on the TNC. When a run is aborted, the ✱ symbol will disappear from the screen.

After aborting a program run, the TNC saves the following data:
- the **tool** that was activated last,
- **coordinate transformations** (datum, mirror-image, coordinate system rotation, scaling factor),
- the last programmed **circle centre/pole CC,**
- the last defined **canned cycle,**
- the current status of **program part repetitions,**
- the return jump address for **subroutines.**



If a program run is aborted during a **subroutine** or a **program part repeat** sequence and a block is then selected with GOTO □, the counter for the program part repeat is reset to the programmed number of repetitions. In the case of subroutines, the return jump address is erased.

To maintain the remaining number of repeats and/or the return jump address, use the ↓ ↑ keys to select program blocks.



Program run
Interrupting and aborting a program run

Interrupting program execution

Operating mode ________________ ≡▶ or →

To interrupt a running program:	▶ (STOP)	Interrupt program run.

▼

The "✱" symbol (TNC in operation) flashes.

Aborting program execution

Operating mode ________________ ≡▶ or →

To abort program execution:	▶ (STOP)	Interrupt program run.
	▼	
	[STOP]	Abort program run.

▼

The "✱" symbol (TNC in operation) disappears.

When running a program in ISO format, the [DEL □] key performs the function of the internal [STOP] key.

Program run
Interrupting and aborting a program run

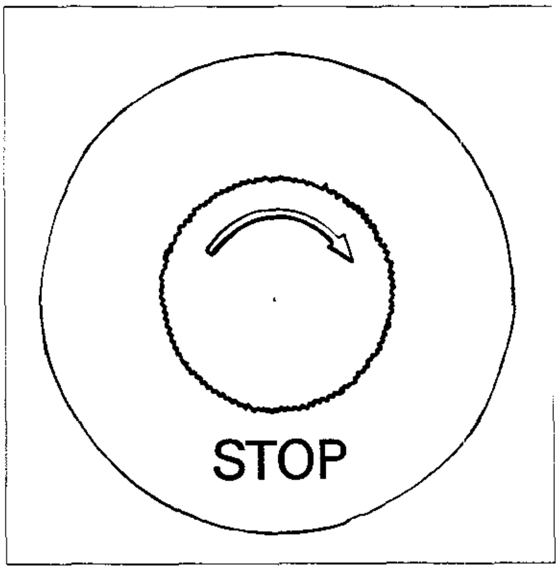
Emergency STOP

In an emergency, you can switch off the machine and the control unit by pressing the emergency STOP button. The control system indicates the interruption with

= EMERGENCY STOP =

To resume operation, first unlock the emergency STOP button by turning it clockwise. Then switch the power on again and clear the display by pressing CE.

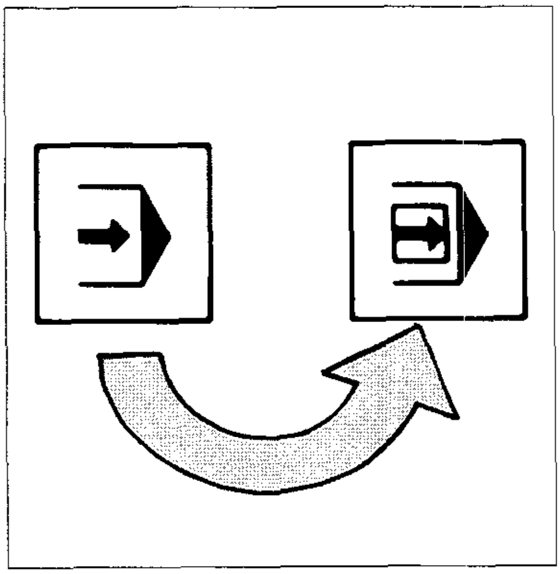
Caution when resuming program execution after aborting (see next page).



Changing from "full sequence" to "single block"

If you have selected ➔ (Program run – full sequence) mode, you can switch to ≣➔ (Program run – single block) mode while the program is running. Program execution will be stopped after the current block is run.

With the execution of continuous contours, the single block is not interrupted during the current block after switching to program run, but rather the machining of the continuous, precalculated contour (up to 14 blocks) is finished.
Beginning with software version 03:
One can select via machine parameter whether to stop the execution of continuous contours during the current block or whether the continuous, precalculated contour should be finished.



Program run
Resuming program execution

Resuming program execution after aborting is only possible under certain conditions!

Resuming program execution

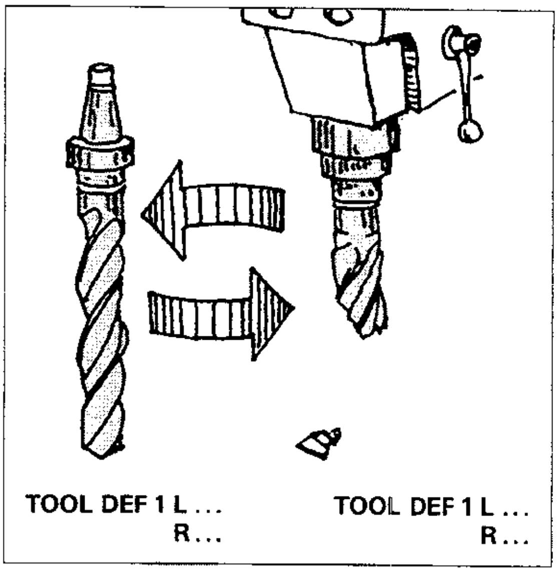
Resumption is only possible, where straight lines with L are programmed in absolute measures in Cartesian or polar coordinates, respectively.
A **resumption is not possible** for
- Straight lines with incremental dimensions (IX, IY, IZ. . .)
- Chamfers (L)
- Circular contours (C, CP, CT, CTP, CR, RND)
- Machining cycles.

Special caution is warranted when resuming:
- programs with Q parameters
- subroutines
- program part repititions.

If a program run is aborted in a **subroutine** or within a **program part repitition** and then a program block is selected with the GOTO □ key, then the counter for the program part repitition will be reset to the programmed number of repititions; with subroutines the return address will be erased.
If the unexecuted number of repititions or the return address is not erased, then the program blocks are to be selected only with the keys ↓, ↑.

Tool change

- In the case of a **tool change due to tool breakage,** new **tool compensation values** (tool definition) must be specified and called up in "MDI" mode. Then set the tool to the workpiece surface.



Error messages

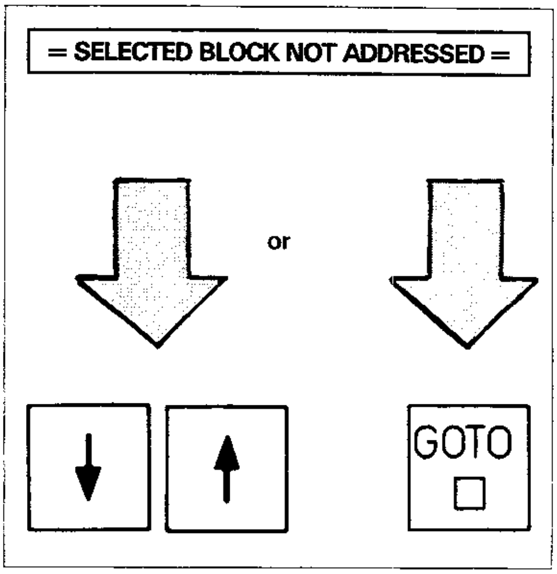
If, after aborting the program run, you paged through the program with the ↓ ↑ keys, did not select a block with GOTO □ and did not resume operation in the block in which the run was interrupted, the following error message will be displayed:

= SELECTED BLOCK NOT ADDRESSED =

Remedy

Select the block which was interrupted by
- using the ↑ and ↓ keys,
- pressing GOTO □ and entering the block number.

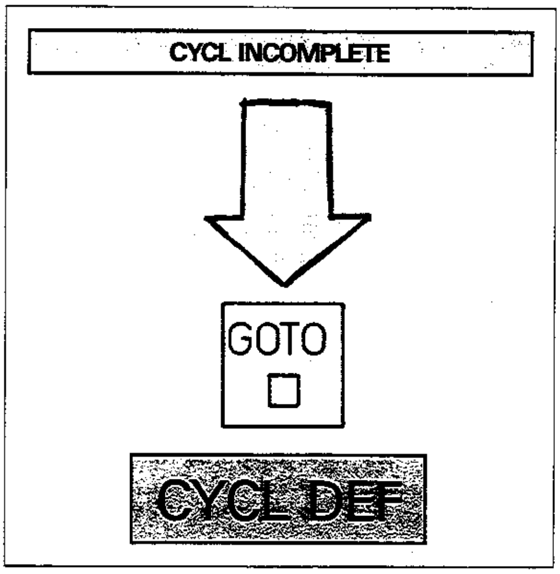
Caution when using GOTO □ (see above).



Program run
Resuming program execution

If a block is deleted or inserted after program execution has been interrupted, the previously read **cycle definition** is no longer active. When execution is resumed, the error message

= CYCL INCOMPLETE =
is displayed before the cycle call.



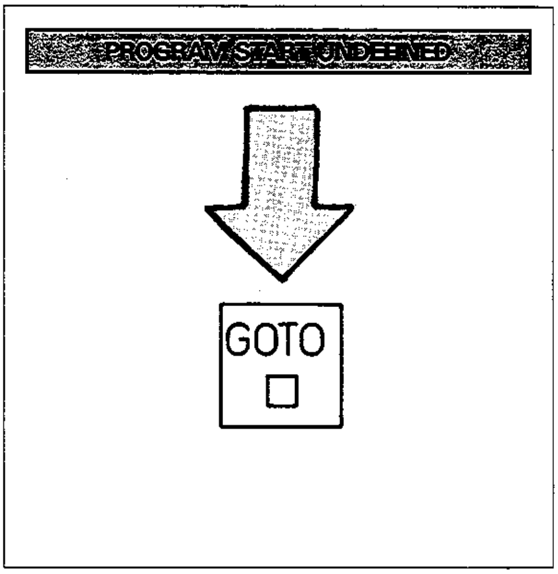
Remedy

The last cycle definition preceding the cycle call must be executed. You **must** select cycle definition with the GOTO □ key.

Caution when using GOTO □ (see "Aborting program execution").

If the program run is resumed after interruption in a canned cycle, the following error message will appear:

= PROGRAM START UNDEFINED =



Remedy

Either edit the program as required or select a previous block with the GOTO □ key.

Caution when using GOTO □ (see "Aborting program execution").

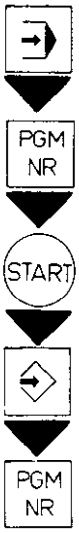
A canned machining cycle must be restarted. The canned cycle "Tapping" should **not** be repeated at the same position.

Program run with background programming

The TNC allows you to run a program in ⇥ mode while creating or editing another program in ◈ mode or while a program is being transferred via the V.24 interface. This feature is called "background programming".

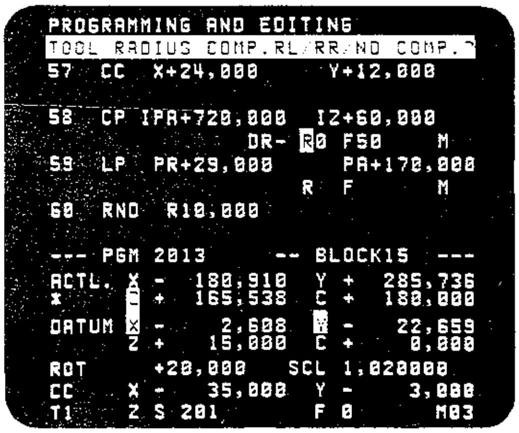
Procedure

First call up and start the program you wish to run (⇥ mode). Then, in operating mode ◈, call up the program you want to create or edit (see "Program call").
You can also transfer a program through the V.24 interface (see "External data transfer" section V).



Screen display

The program input procedure is displayed in the upper portion of the screen, while the lower half displays the current program run. In contrast to the normal display for program execution, only the program number and current block number are shown in this case. Position data and status displays (active cycles for coordinate transformations, tool, spindle speed, feed rate and auxiliary functions) are shown as usual.



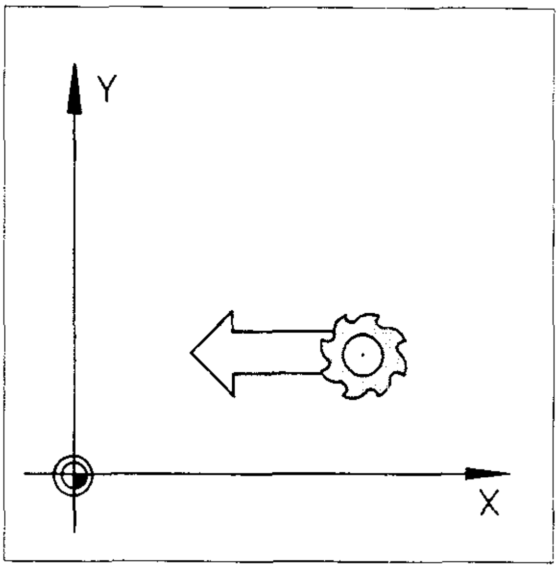
```
PROGRAMMING AND EDITING
TOOL RADIUS COMP.RL/RR/NO COMP.?
57  CC  X+24,000         Y+12,000

58  CP IPA+720,000  IZ+60,000
               DR- R0 F50     M
59  LP  PR+29,000       PA+170,000
                    R  F      M
60  RND  R10,000

--- PGM 2013      -- BLOCK15  ---
ACTL. X -  180,910  Y +  285,736
*     Z +  165,538  C +  180,000
DATUM X -    2,608  Y -   22,659
      Z +   15,000  C +    0,000
ROT      +20,000   SCL 1,020000
CC    X -   35,000  Y -    3,000
T1    Z S 201       F 0      M03
```

Notes:

Paraxial machining
Programming via axis address keys

Dialogue initiation

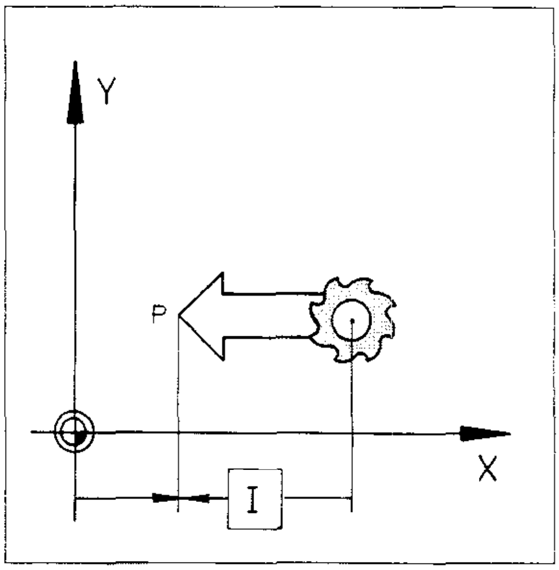
The entry of paraxial positioning blocks can be somewhat simplified:
Like the combined point-to-point and straight-cut control models TNC 131/TNC 135, the input dialogue is initiated directly using the axis address keys $\boxed{X}$ $\boxed{Y}$ $\boxed{Z}$ $\boxed{IV}$



Nominal position value

Enter the coordinate for the appropriate axis as the **nominal position.** The numerical value can be entered in absolute (based on workpiece datum) or incremental (based on previous nominal position) dimensions.

In either case, the tool will move from its current actual position, parallel to the selected axis, to the programmed target position.



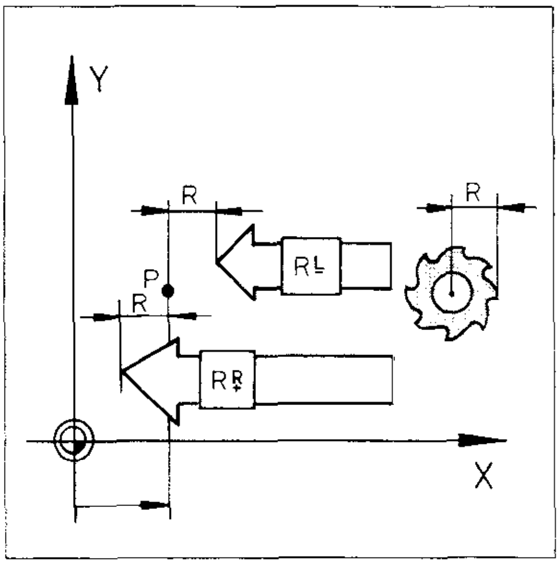
Radius compensation

When programming via axis address keys, tool radius compensation has the following significance:

- To **decrease** distance traversed by the tool radius: press $\boxed{R^{L}_{-}}$, screen displays **R–.**
- To **increase** distance traversed by the tool radius: press $\boxed{R^{R}_{+}}$, screen displays **R+.**
- The tool moves to the programmed nominal position, screen displays R0.

If a radius compensation R+/R– is also programmed when positioning the **spindle axis, no compensation** will be active on this axis.

When the **4th axis** is used as a **rotary table axis,** no radius compensation will be taken into account.



Paraxial machining
Programming via axis address keys

Do not enter paraxial positioning blocks containing a radius compensation R+/R– before or after positioning blocks containing a radius compensation RR/RL.

```
WRONG

16  L  X+15,000  Y+20,000
        RR   F     M03

17      Y+40,000
        R-   F100 M

18  L  X+50,000  Y+57,000
        RR   F     M
```

Within a program, paraxial positioning blocks entered via an axis address key can be inserted between positioning blocks containing R0 (no radius compensation) that were programmed via a contouring function key.

```
CORRECT

18  L  X+15,000  Y+20,000
        R0   F     M

19  L  X+10,000  Y+10,000
        R0   F     M

20      X+40,000
        R+   F     M

21  L  X+50,000  Y+20,000
        R0   F     M
```

Paraxial machining
Programming via axis address keys

Entering paraxial straight lines

Operating mode ______________________________

Dialogue initiation ______________________________ X or Y or Z or IV

POSITION VALUE ?	I	Incremental – absolute?
	□	Enter numerical value for selected axis.
	ENT	Press ENT.
TOOL RADIUS COMP.: R+/R–/NO COMP. ?	R-L R+R	Specify radius compensation if required.
	ENT	Press ENT.
FEED RATE ? F =	□	Specify feed rate if required.
	ENT	Press ENT.
AUXILIARY FUNCTION M ?	□	Specify auxiliary function if required.
	ENT	Press ENT.

Sample display

```
119 IX+46.000

          R+   F60    M03
```

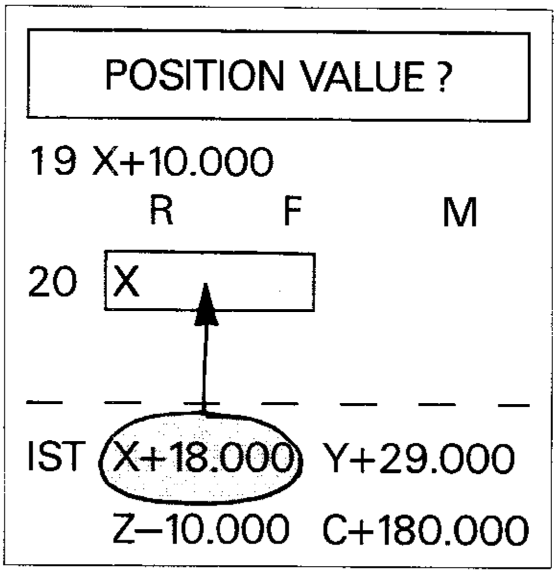
In block 119, the tool moves parallel to the X-axis by +46.000 plus the tool radius.
The feed rate is 60 mm/min., spindle rotation is clockwise.

Paraxial machining
Playback programming

Playback

If the tool has been moved in manual mode (via handwheel or axis address key), the actual position of the tool can be transferred to the machining program as a nominal position. This method of entering data is called "playback" programming.

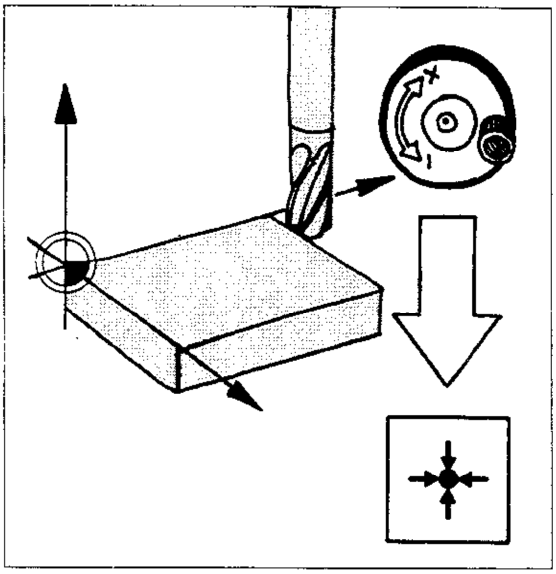
Playback programming is only practical for paraxial machining operations. Programming complex contours with this technique is not recommended.



```
POSITION VALUE ?

19 X+10.000
     R      F        M
20 X

IST X+18.000  Y+29.000
    Z–10.000  C+180.000
```

Procedure

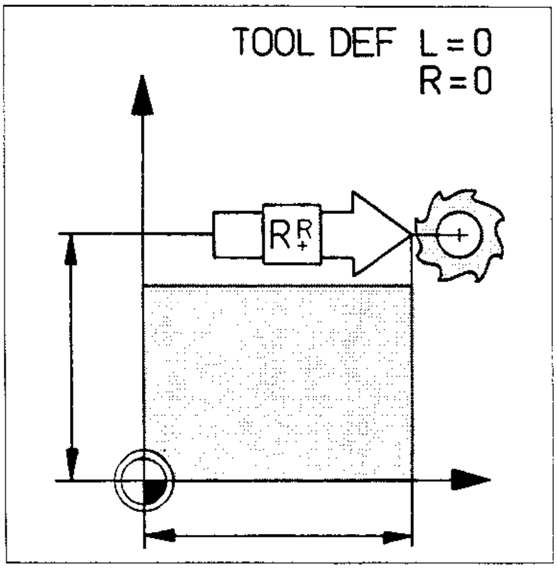
Move the tool manually, via handwheel or axis address key, to the required position.
In ◈ mode, within a positioning block, the actual value of the position is transferred as a nominal position value, by pressing the ⌖ key.



Radius compensation

The actual position value already contains the length and radius compensation data for the tool currently in use. For this reason, enter the compensation values L = 0 and R = 0 when defining this tool.

When programming positioning blocks in playback mode, enter the correct tool radius compensation R+ or R–. If a tool breaks or the original tool is replaced by another one, the new compensation values can then be taken into account.



Paraxial machining
Playback programming

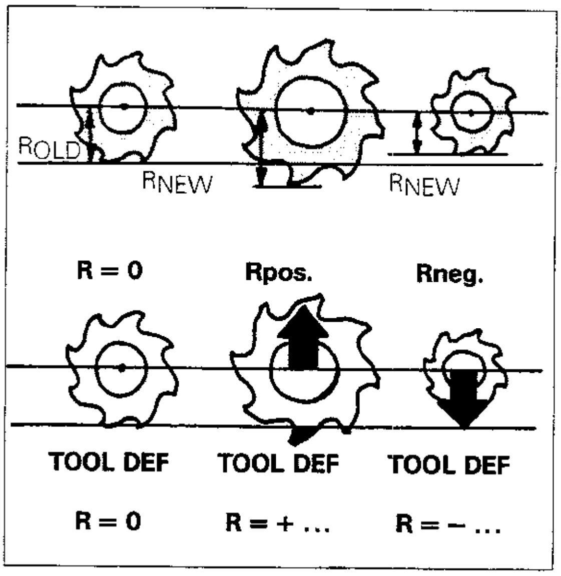
Tool compensation

The new compensation values are determined according to the following formula:

$R = R_{NEW} - R_{OLD}$

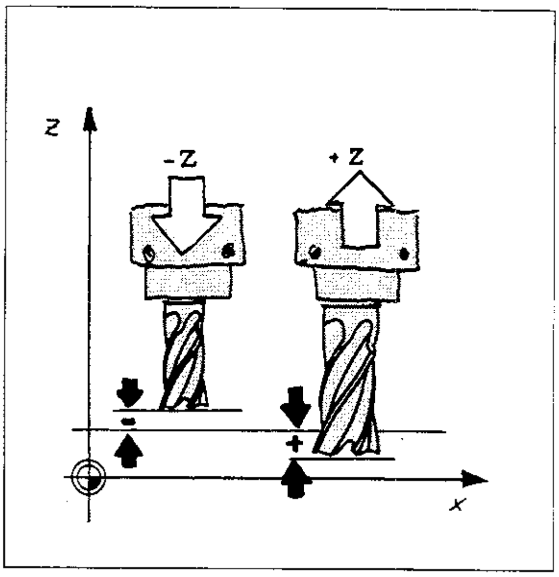
R	radius compensation value for TOOL DEF
R_{NEW}	radius of new tool
R_{OLD}	radius of original tool.

The new compensation values are entered during tool definition of the original tool (R = 0, L = 0). The compensation value R may be **positive or negative,** depending on whether the radius of the new tool is larger (+) or smaller (–) than the radius of the original tool.

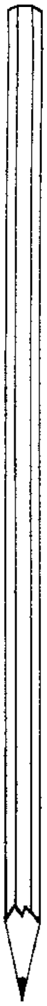


Length compensation

The compensation value for the new tool length is determined in the same way as for TOOL DEF. In this case, the zero tool is the one originally used.



Notes:



Paraxial machining
Playback programming

Input example

Operating mode ______________ ⇨

Dialogue initiation ______________ X or Y or Z or IV

POSITION VALUE ?	▶ ⓧ	Move tool manually to required position if necessary.
	▼ ⌖	Transfer nominal position value.
	▼ ENT	Press ENT.

▼

TOOL RADIUS COMP.: R+/R–/NO COMP. ?	▶ R$^{L}_{-}$ R$^{R}_{+}$	Specify radius compensation if required.
	▼ ENT	Press ENT.

▼

FEED RATE ? F =	▶ □	Specify feed rate if required.
	▼ ENT	Press ENT.

▼

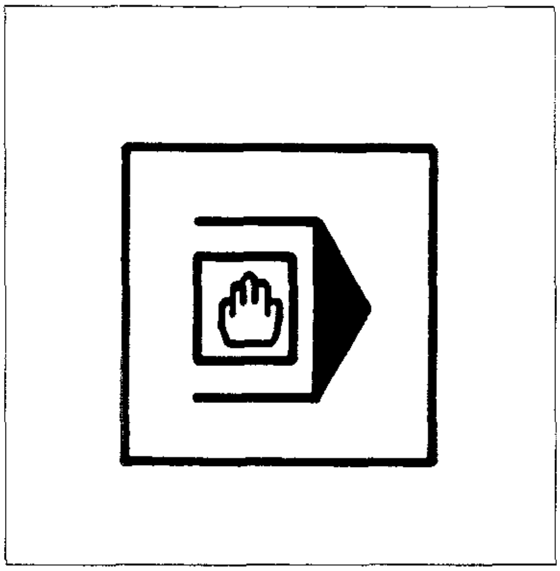
AUXILIARY FUNCTION M ?	▶ □	Specify auxiliary function if required.
	▼ ENT	Press ENT.

Program input can be terminated prematurely by pressing END □.

Paraxial machining
Positioning with MDI

Positioning

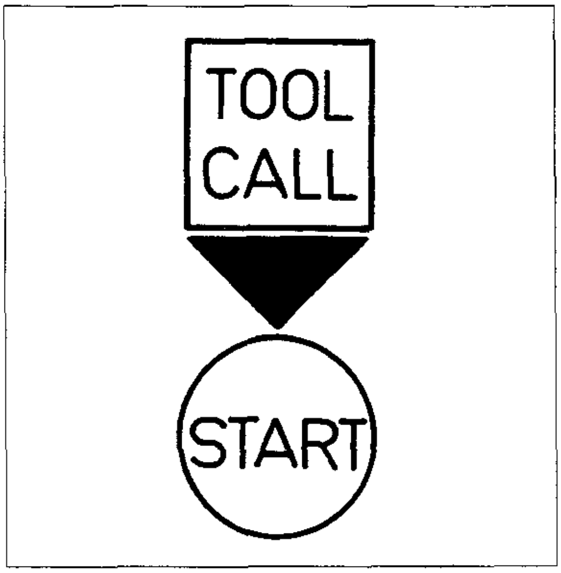
In ✋ mode "Positioning with MDI", **paraxial** positioning blocks can be entered and executed (without saving).
Each block must be run immediately after being entered by pressing the external start button.



If the positioning block contains data in incremental dimensions, the block can be run as often as required by pressing the external start button.

Tool call

If a tool definition TOOL DEF has been saved in the TNC's memory, a tool can be called via TOOL CALL in ✋ mode. This also activates the new tool compensation values.
The tool is called by pressing the external start button.

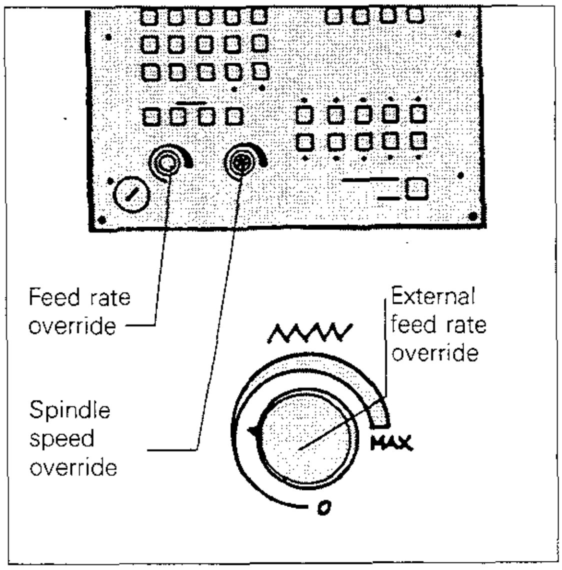


Feed rate

The programmed feed rate can be modified
- via the **internal feed rate override** and/or
- via the **external feed rate override** on the machine, depending on how the control system was installed on the machine by the machine manufacturer.

Spindle speed

In the case of analogue output, the spindle speed can be modified via the **spindle override.**



Paraxial machining
Positioning with MDI

Input example: position data

Operating mode ______

Dialogue initiation ______ X or Y or Z or IV

POSITION VALUE ?	▶ I	Incremental – absolute?
	▼ □	Enter numerical value for selected axis.
	▼ ENT	Press ENT.
▼		
TOOL RADIUS COMP.: R+/R–/NO COMP. ?	▶ R L R R	Specify radius compensation if required.
	▼ ENT	Press ENT.
▼		
FEED RATE ? F =	▶ □	Specify feed rate if required.
	▼ ENT	Press ENT.
▼		
AUXILIARY FUNCTION M ?	▶ □	Specify auxiliary function if required.
	▼ ENT	Press ENT.
▼		
BLOCK COMPLETE	▶ START	Run positioning block.

Program input can be terminated prematurely by pressing END □

Notes:

Paraxial machining
Positioning with MDI

Input example: tool call

Operating mode ______________________ →

Dialogue initiation ______________________ TOOL CALL

TOOL NUMBER ?	▶ □ ▼ ENT	Enter tool number. Press ENT.
WORKING SPINDLE AXIS X/Y/Z ?	▶ Z	Specify spindle axis, e.g. Z.
SPINDLE SPEED S RPM = ?	▶ □ ▼ ENT	Specify spindle speed. Press ENT.
BLOCK COMPLETE	▶ START	Start tool call.

If a central tool memory is not available, the program with the corresponding tool definition must be called in → mode.

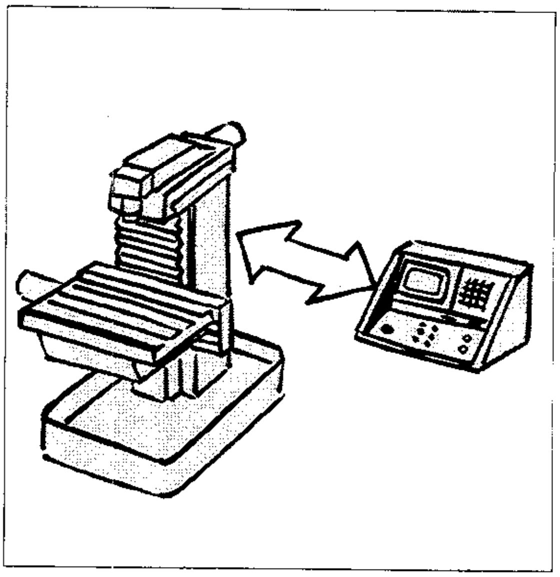
Machine parameters

Machine parameters

To enable the machine to carry out the commands issued by the control system, the TNC must "know" specific machine data, e.g. traverse paths, acceleration data etc. These data are defined by the machine manufacturer via so-called machine parameters.

User parameters

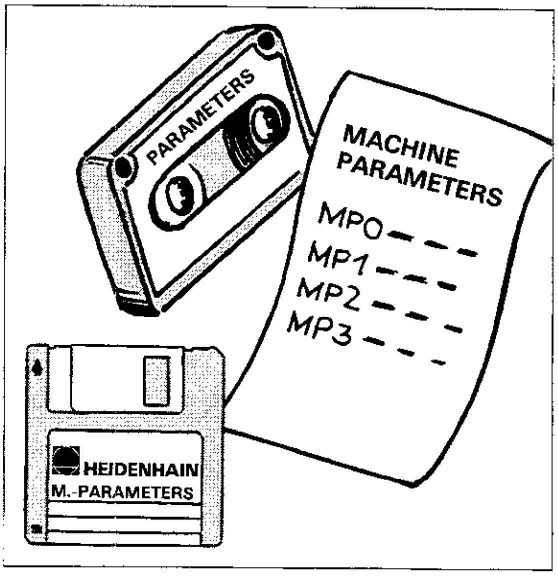
You can access certain machine parameters easily while in MOD operating mode, e.g. switching from HEIDENHAIN plain-language programming to standard ISO format. The user parameters available via the MOD mode are defined by the machine manufacturer, who can also provide further details on this subject.



Programming

The machine parameters must be programmed in the control system during initial commissioning. This can be done via an external data medium (e.g. ME cassette or FE disk containing stored machine parameters) or manually from the keyboard.

The machine parameters must be re-entered following an **interruption of power with discharged or missing buffer batteries.** The control system prompts you for the data in interactive dialogue.



Buffer battery

The buffer battery is the power source for the machine parameter memory and for the TNC's program memory. It is located beneath the cover on the front panel of the control unit.

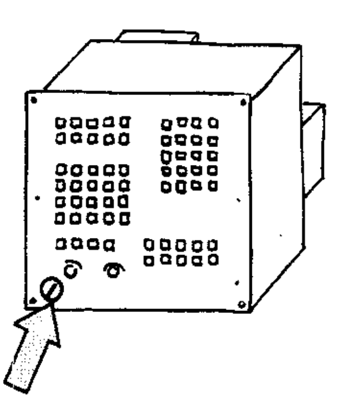
When the message

= EXCHANGE BUFFER BATTERY =

is displayed, it's time to replace the battery. (Memory contents will be saved for about one week after the above message is displayed.)

Replace battery with mains voltage connected. The TNC's memory units are then supplied with power from the mains supply.

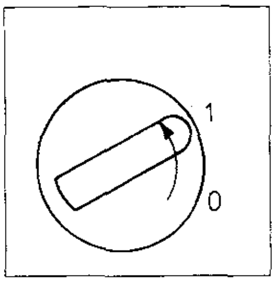
- All data memory units (RAM) will be erased if the buffer battery is replaced while the mains supply is switched off and machine parameters will have to be re-entered.



Battery type
Mignon cells, leak-proof
IEC designation: "LRG"
Recommended: VARTA type 4006

Machine parameters

Input from external data medium



Switch on power.

MEMORY TEST

The control system checks the internal electronic controllers. Display is cleared automatically.

EXCHANGE BUFFER BATTERY

Insert new battery.

CE Clear message.

OPERATING PARAMETERS ERASED

CE Clear error message.

For data input from magnetic tape:

MACHINE PARAMETER PROGRAMMING

MACHINE PARAMETER MP 0 ?

MP 0: 0

Insert cassette with parameters.

←8 TNC Select operating mode on ME unit.

EXT Start external data transfer.

MACHINE PARAMETER PROGRAMMING

EXTERNAL DATA INPUT

MP 0: 0

Machine parameters are programmed automatically.

Machine parameters

For data input from disk:

MACHINE PARAMETER PROGRAMMING

MACHINE PARAMETER MP 0 ?

MP 0: 0 ▶ Insert disk with parameters.

▼

EXT Select external data transfer.

▼

EXTERNAL DATA INPUT ? ▶ ENT Press ENT to confirm.

▼

PROGRAM NUMBER = ▶ □ Enter number of program containing machine parameters.

▼

ENT Press ENT.

▼

MACHINE PARAMETER PROGRAMMING

MACHINE PARAMETER MP 0 ?

MP 0: 0

Machine parameters are programmed automatically.

When all parameters have been entered:

▼

POWER INTERRUPTED ▶ CE Clear message.

▼

NC: PROGRAM MEMORY ERASED ▶ CE Clear message.

▼

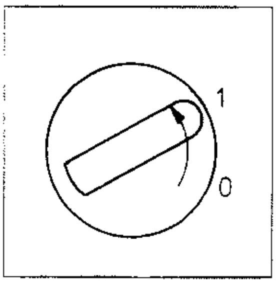
RELAY EXT. DC VOLTAGE MISSING ▶ I Switch on control voltage.

▼

After traversing the reference points, the control system is ready for operation.

Machine parameters

Manual input



Switch on power.

MEMORY TEST

The control system checks the internal electronic controllers. Display is cleared automatically.

EXCHANGE BUFFER BATTERY

Insert new battery.

CE Clear message.

OPERATING PARAMETERS ERASED

CE Clear error message.

MACHINE PARAMETER PROGRAMMING

MACHINE PARAMETER MP 0 ?

MP 0: 0

□ Enter machine paramete- MP 0 from table.

ENT Press ENT.

After each machine parameter is entered, the screen display advances automatically to the next parameter. Press ENT after entering each parameter.

When all parameters have been entered:

POWER INTERRUPTED

CE Clear message.

NC: PROGRAM MEMORY ERASED

CE Clear message.

RELAY EXT. DC VOLTAGE MISSING

I Switch on control voltage.

After traversing the reference points, the control system is ready for operation.

Machine parameters

Machine parameter number	Input value	Machine parameter number	Input value	Machine parameter number	Input value
MP 00					
MP 01		MP 51		MP 101	
MP 02		MP 52		MP 102	
MP 03		MP 53		MP 103	
MP 04		MP 54		MP 104	
MP 05		MP 55		MP 105	
MP 06		MP 56		MP 106	
MP 07		MP 57		MP 107	
MP 08		MP 58		MP 108	
MP 09		MP 59		MP 109	
MP 10		MP 60		MP 110	
MP 11		MP 61		MP 111	
MP 12		MP 62		MP 112	
MP 13		MP 63		MP 113	
MP 14		MP 64		MP 114	
MP 15		MP 65		MP 115	
MP 16		MP 66		MP 116	
MP 17		MP 67		MP 117	
MP 18		MP 68		MP 118	
MP 19		MP 69		MP 119	
MP 20		MP 70		MP 120	
MP 21		MP 71		MP 121	
MP 22		MP 72		MP 122	
MP 23		MP 73		MP 123	
MP 24		MP 74		MP 124	
MP 25		MP 75		MP 125	
MP 26		MP 76		MP 126	
MP 27		MP 77		MP 127	
MP 28		MP 78		MP 128	
MP 29		MP 79		MP 129	
MP 30		MP 80		MP 130	
MP 31		MP 81		MP 131	
MP 32		MP 82		MP 132	
MP 33		MP 83		MP 133	
MP 34		MP 84		MP 134	
MP 35		MP 85		MP 135	
MP 36		MP 86		MP 136	
MP 37		MP 87		MP 137	
MP 38		MP 88		MP 138	
MP 39		MP 89		MP 139	
MP 40		MP 90		MP 140	
MP 41		MP 91		MP 141	
MP 42		MP 92		MP 142	
MP 43		MP 93		MP 143	
MP 44		MP 94		MP 144	
MP 45		MP 95		MP 145	
MP 46		MP 96		MP 146	
MP 47		MP 97		MP 147	
MP 48		MP 98		MP 148	
MP 49		MP 99		MP 149	
MP 50		MP 100		MP 150	

Machine parameters

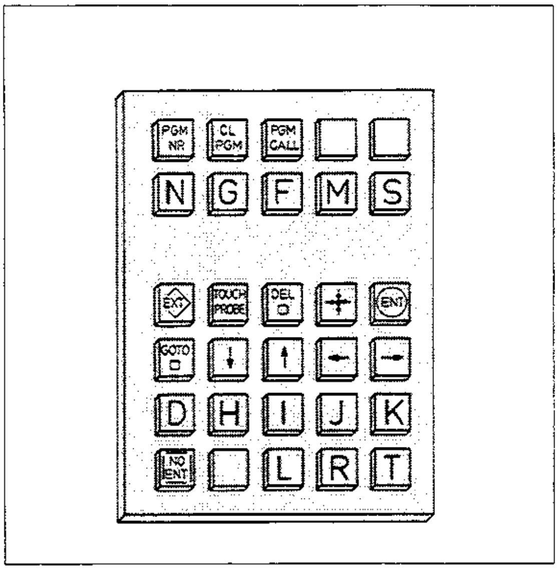
Machine parameter number	Input value	Machine parameter number	Input value	Machine parameter number	Input value
MP 151		MP 201		MP 251	
MP 152		MP 202		MP 252	
MP 153		MP 203		MP 253	
MP 154		MP 204		MP 254	
MP 155		MP 205		MP 255	
MP 156		MP 206		MP 256	
MP 157		MP 207		MP 257	
MP 158		MP 208		MP 258	
MP 159		MP 209		MP 259	
MP 160		MP 210		MP 260	
MP 161		MP 211		MP 261	
MP 162		MP 212		MP 262	
MP 163		MP 213		MP 263	
MP 164		MP 214			
MP 165		MP 215			
MP 166		MP 216			
MP 167		MP 217			
MP 168		MP 218			
MP 169		MP 219			
MP 170		MP 220			
MP 171		MP 221			
MP 172		MP 222			
MP 173		MP 223			
MP 174		MP 224			
MP 175		MP 225			
MP 176		MP 226			
MP 177		MP 227			
MP 178		MP 228			
MP 179		MP 229			
MP 180		MP 230			
MP 181		MP 231			
MP 182		MP 232			
MP 183		MP 233			
MP 184		MP 234			
MP 185		MP 235			
MP 186		MP 236			
MP 187		MP 237			
MP 188		MP 238			
MP 189		MP 239			
MP 190		MP 240			
MP 191		MP 241			
MP 192		MP 242			
MP 193		MP 243			
MP 194		MP 244			
MP 195		MP 245			
MP 196		MP 246			
MP 197		MP 247			
MP 198		MP 248			
MP 199		MP 249			
MP 200		MP 250			

Notes:

Programming in ISO format
Introduction

Snap-on keyboard

With the TNC 151/TNC 155 you can enter programs either in HEIDENHAIN format, featuring operator prompting in plain-language interactive dialogue, or in standard ISO 6983 format. Programming in ISO format can be an advantage when creating programs with external computer systems.
A snap-on overlay keyboard, with standard address keys, is available for ISO programming on the control unit. The keyboard is simply placed over the control unit keyboard and held in place magnetically.



The internal STOP key is assigned to the D key; for ISO programming, the DEL key performs the function of the internal STOP key.

Program input

The key assignment of the overlay keyboard is functional after **switching** from HEIDENHAIN plain-language dialogue prompting to standard programming.
Program input in ISO format is dialogue-prompted to some extent. Individual commands (words) can be entered into a block in any sequence. The control system sorts the programmed commands automatically when a block is complete and displays any errors made while programming or executing the program with plain-language error messages.

Block format: positioning blocks

Positioning blocks may include:
- 8 **G-codes** (preparatory functions) of various groups (see "G-codes") plus an additional G90 or G91 for each coordinate;
- 3 **coordinates** (from X, Y, Z, IV) plus two circle centre/pole coordinates (from I, J, K);
- 1 **feed rate** F (max. 5 digits);
- 1 **auxiliary function** M;
- 1 **spindle speed** S (max. 4 digits);
- 1 **tool number** of various groups (see "G-codes") (max. 3 digits).

Block format: canned cycles

Blocks with canned cycles may include:
- all **individual data** for the cycle (cycle parameter P);
- 1 **auxiliary function** M;
- 1 **spindle speed** S;
- 1 **tool number** of various groups (see "G-codes") (Tool call);
- 1 **positioning block;**
- 1 **feed rate** F;
- **cycle call.**

Error messages

The TNC displays block format errors while the block is being entered, e.g.
= G-CODE GROUP ALREADY ASSIGNED =
or after block entry is complete, e.g.
= BLOCK FORMAT INCORRECT =

Programming in ISO format
Changing programming modes
External programming

Changing from HEIDENHAIN to ISO programming

The changeover from HEIDENHAIN to ISO programming format is made via a machine parameter. This parameter can be modified by means of the MOD function "User parameters". User parameters are defined by the machine manufacturer, who can also provide you with further information.

Remarks on external programming

- At program start, in front of the % sign and after each programm block, CR LF or CR FF or FF must be programmed.
- After the block program end, CR LF or LF or CR FF or FF and additionally ETX (control C) must be programmed. In place of ETX a replacement sign can be determined via machine parameter (see data interface description TNC 151/155).
- Intervals (blank spaces) between the individual words can be left out.
- Zeros after a point can be left out.

Beginning with software version 03:

- During input of DIN blocks the sign "*" is no longer necessary at the end of a block.
- During output of DIN blocks the sign "*" is no longer put out by the control.
- During input of NC programs, commentaries that are marked with "*" or ";" are ignored.

Programming in ISO format
Changing programming modes

Operating mode ____ □ optional

Dialogue initiation ____ MOD

VACANT BLOCKS: 1638	▶ MOD ↑	Select MOD function "User parameters".
USER PARAMETERS	▶ ENT ▼ ↓ ↑	Select desired user parameter.
Dialogue defined by machine manufacturer		
Program input in HEIDENHAIN format:	▶ 0 ▼	
or	DEL □	Exit supplementary mode.
Program input in ISO format:	▶ 1 ▼	
	DEL □	Exit supplementary mode.
POWER INTERRUPTED	▶ CE	Clear error message.
RELAY EXT. DC VOLTAGE MISSING	▶ I	Switch on control voltage.

After traversing the reference points, the control system is ready for operation.

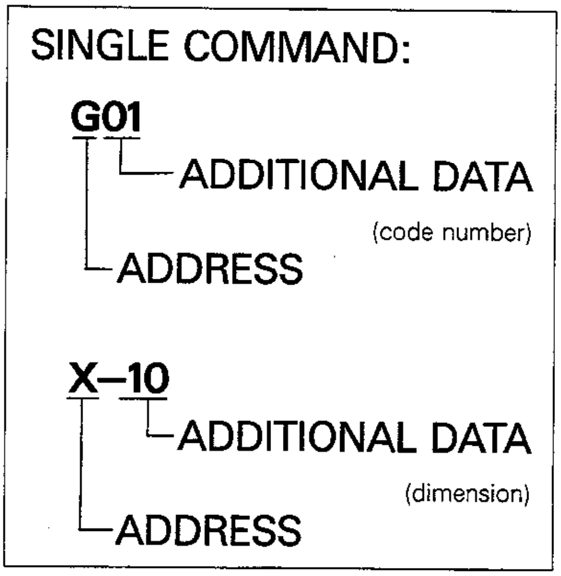
After switching programming modes, plain-language programs are automatically converted to ISO format and vice versa.

> Please note the following when switching from ISO format to plain-language format:
>
> - Modal functions (e.g. G01) are converted to the plain-language symbol (e.g. L) only in the block in which the function was programmed. The symbol ✱ then appears in subsequent blocks written in plain-language format.
> - K stands for Cartesian coordinates.
> - P stands for polar coordinates.
> - F MAX signifies rapid traverse.

Programming in ISO format
Control system operation

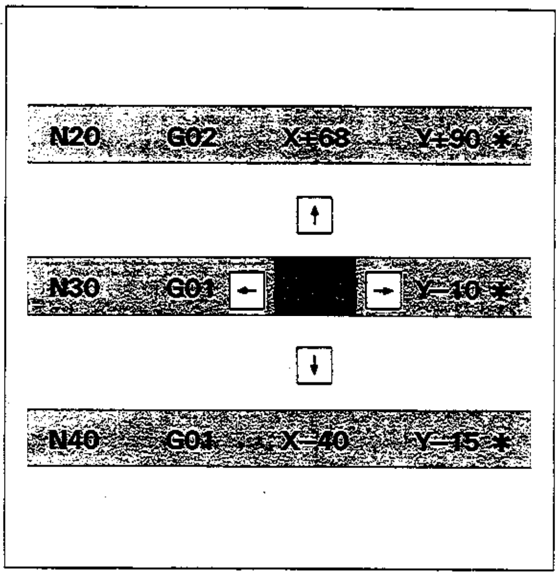
Entering single commands

Single commands consist of an **address** and **additional data.**
To enter a single command, first press the alpha address key and then enter the additional data from the numeric keypad.
Conclude the entry for the single command by pressing the alpha address key to enter the next command.
To conclude the block, press END □.



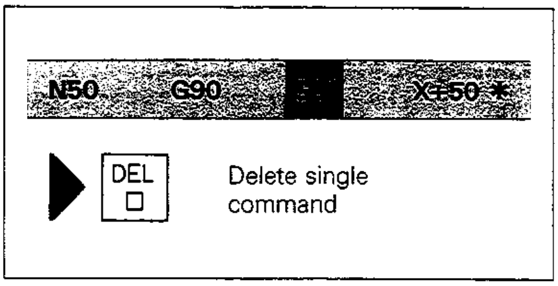
Editing

You can make changes to a program immediately while entering the block or later after program input is complete. The GOTO □ ↓ ↑ ← → keys are provided for this purpose (see "Editing").
In contrast to HEIDENHAIN plain-language format, you can move the cursor in ISO format with the → and ← keys.
When the **highlighted pointer** is located on a single command within a block, you can start a search routine by pressing the ↓ ↑ keys.
When you are finished editing, use the ← key to move the highlighted pointer beyond the beginning of the block, the → key to move it beyond the end of the block, or press END □.
Press CE to delete incorrectly entered **additional data.**

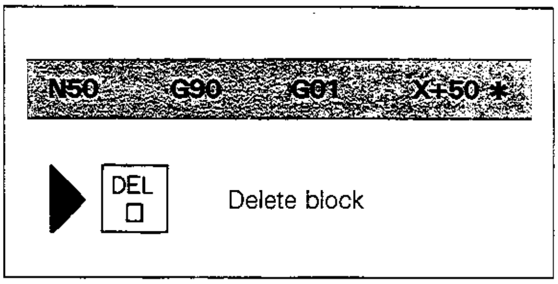


A zero, which can be overwritten, will appear in the pointer when you press CE.

Delete incorrectly entered **address letters** or **entire single commands** by pressing DEL □.



To do this, the highlighted pointer must be located over the command you wish to delete.
If no pointer is visible in the current block, pressing DEL □ will cause the entire block to be deleted!

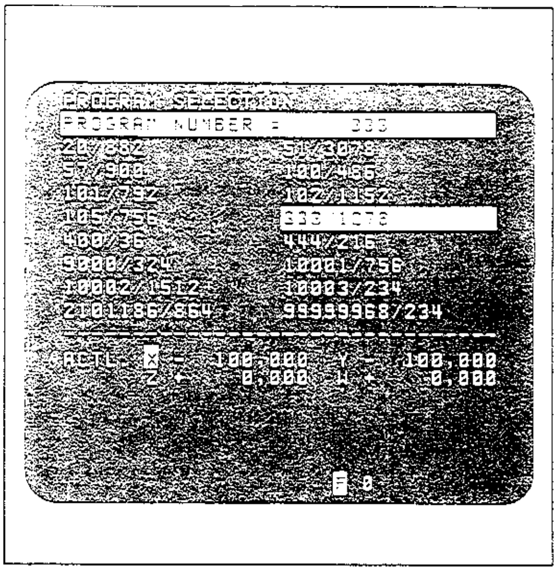


Programming in ISO format
Program management

Program management

The TNC's memory can accommodate up to **32 programs** with a total of **3,100 program blocks.** A program may contain up to 1,000 blocks. You can create a new program or call up an existing one by pressing PGM NR (see "Program call").

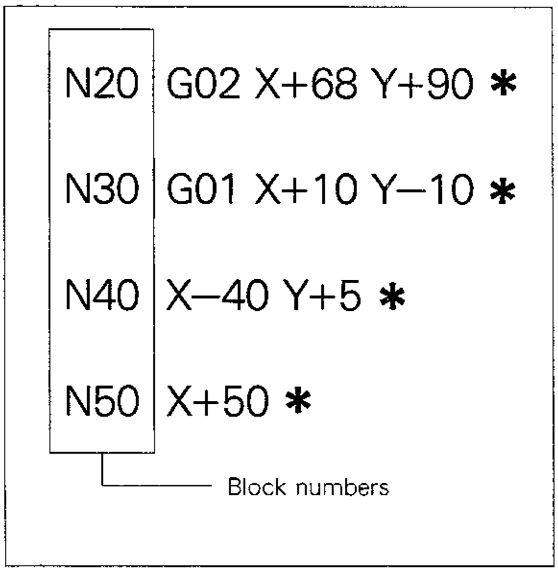
The number of characters (bytes) contained in the program is indicated after the program number in the program library, e.g. 333/1278.



```
PROGRAM SELECTION
PROGRAM NUMBER =        333
20/882              51/3078
57/900              100/466
101/792             102/1152
105/756             333/1278
400/36              444/216
9000/324            10001/756
10002/1512          10003/234
2101186/864         99999968/234
--------------------------------------
ACTL. X =  100,000   Y =  100,000
      Z +    0,000   W +    0,000

                       F 0
```

Block number

The block number consists of the **address N** and the actual block number. It can be entered **manually** via the N key or set **automatically** by the control system. The interval between individual block numbers is defined with the MOD function "Block number increment".
The TNC executes the program in the sequence in which the blocks were entered. The block number itself has no effect on the sequence in which the program is executed.
When **editing a program,** blocks with any block number may be inserted between two existing program blocks.

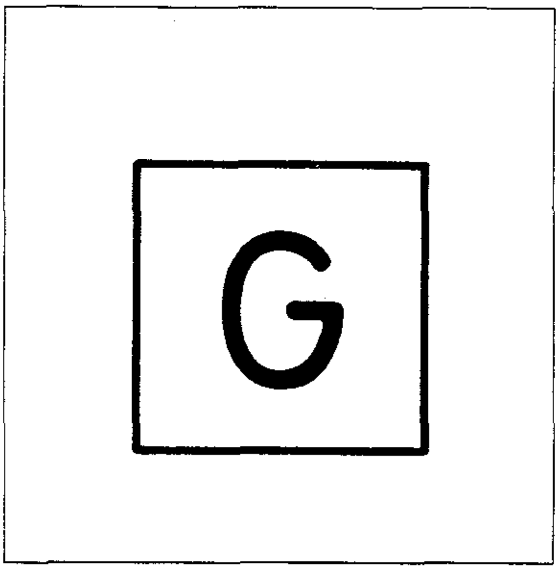


```
N20 G02 X+68 Y+90 *
N30 G01 X+10 Y–10 *
N40 X–40 Y+5 *
N50 X+50 *
```

Programming in ISO format
G-codes

Categories

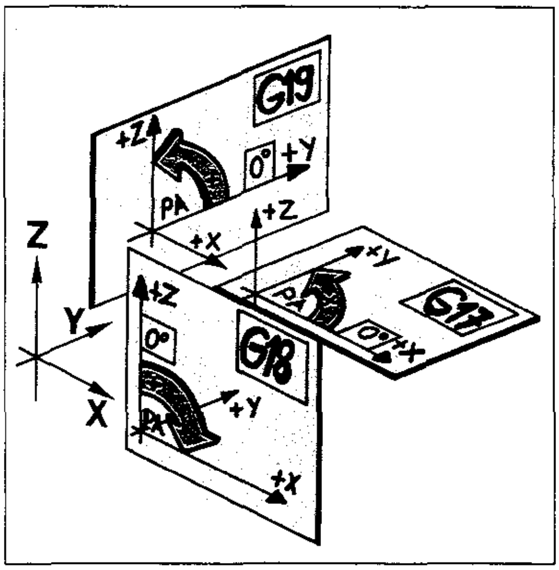
G-codes, also known as preparatory functions, mainly represent path characteristics for tool movement. They consist of the **address G** and a two-digit code number. The G-codes are subdivided into the following groups:



G-codes for positioning
Target position in Cartesian coordinates:
G00 – G07
Target position in polar coordinates: G10 – G16

G-codes for cycles
Machining cycles:
Drilling cycles G83 – G84
Milling cycles G37/G56 – G59/G74 – G78
Cycle call G79
Cycles for coordinate transformations:
Cycles G28/G54/G72/G73
Dwell time cycle: G04
Spindle orientation cycle (optional): G36
Freely programmable (variable) cycle:
(Program call) G39

G-codes for selecting machining plane
G17 Plane selection XY, tool axis Z, angular reference axis X
G18 Plane selection ZX, tool axis Y, angular reference axis Z
G19 Plane selection YZ, tool axis X, angular reference axis Y
G20 Tool axis IV



G-codes for milling and rounding corners and tangential contour approach
G24 – G27
G-codes for path compensation
G40 – G44

Miscellaneous G-codes

G29	Transfer of last nominal position value as pole
G30	Blank form definition for graphics, min. point
G31	Blank form definition for graphics, max. point
G38	Corresponds to STOP block in HEIDENHAIN format
G50	Erase/edit protection (at beginning of program)
G51	Next tool number when central tool memory is used
G55	Touch-probe function, workpiece surface as reference plane
G70	Dimensions in inches (at beginning of program)
G71	Dimensions in millimetres (at beginning of program)
G90	Absolute dimensions
G91	Incremental dimensions
G98	Set label number
G99	Tool definition

Programming in ISO format
G-codes

Entering G-codes

All the G-codes in a program block must be from different groups, e.g.:

N101 G01 G90 ... G41.

Multiple G-codes programmed from the same group would contradict each other, e.g.:

N105 G02 G03 ...

The TNC indicates this situation during program input by generating the error message
= G-CODE GROUP ALREADY ASSIGNED =

If a code number that is unrecognized by the control system is assigned to the address G, the error message
= ILLEGAL G-CODE =
is displayed.

The initial positioning block must include one G-code from each of the following groups:
G17, G18, G19, G20
G00, G01, G02, G03, G06 etc.
G40, G41, G42, G43, G44
G90, G91
There is **no standard default value!**

Programming in ISO format
Dimensions in inch/mm
Erase/edit protection

Dimensions in inch/mm

G70	Dimensions in inch (dialogue-prompted)
G71	Dimensions in mm (dialogue-prompted)

After dialogue initiation via PGM NR and response to the prompt

PROGRAM NUMBER

the following dialogue prompt appears:

MM = G71/INCH = G70

Respond to the prompt by entering G71 or G70.

Block format (example)

```
%2 G71
```

%	Beginning of program
2	Program number
G71	Dimensions in mm

Erase/edit protection

G50	Erase/edit protection (dialogue-prompted)

If the ← → keys are used in the initial program block (e.g. %2 G71), after program input is complete, to select the dialogue prompt

PGM PROTECTION ?

the program can be safeguarded from being erased or altered by entering G50.

Block format (example)

```
%2 G71 G50
```

%	Beginning of program
2	Program number
G71	Dimensions in mm
G50	Erase/edit protection.

The erase/edit protection is cancelled by entering the code number 86357.

Please see "Erase/edit protection" for explanation.

Programming in ISO format
Tool definition/Tool call

Tool definition

G99 Tool definition (dialogue-prompted)

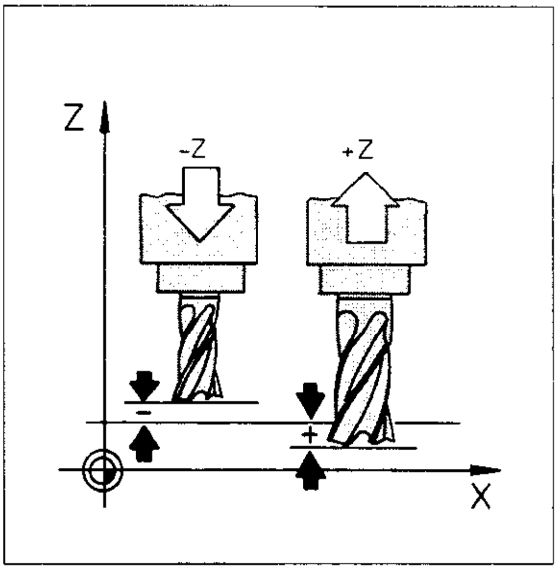
Block format (example)

G99 T1 L+0 R+20

G99	Tool definition
T ...	Tool number
L ...	Tool length compensation
R ...	Tool radius compensation

Please see "Tool definition" for explanation.

Tool definition occupies one program block.



Tool call

T Tool call

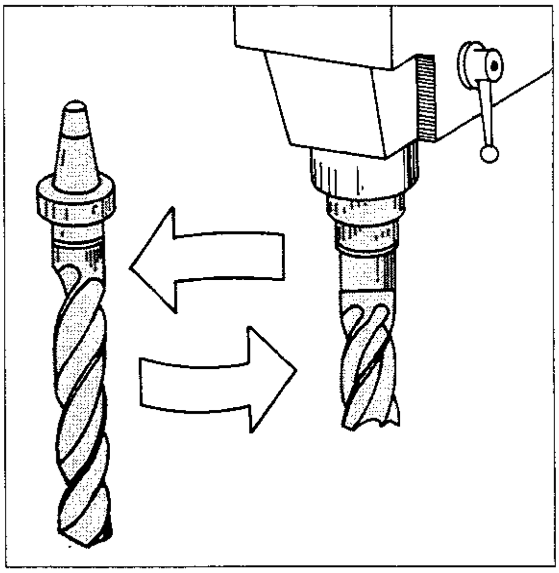
The machining plane (G17/G18/G19/G20) and the spindle speed S must be defined in addition to the tool call. Because G17/G18/G19/G20 automatically terminates path compensation, it should not be programmed within a contour.

Block format (example)

T1 G17 S1000

T ...	Tool call + tool number
G17	Selection of plane XY, tool axis Z
S ...	Spindle speed

Please see "Tool definition" for explanation.



Next tool

G51 Next tool when central tool memory is used.

Block format (example)

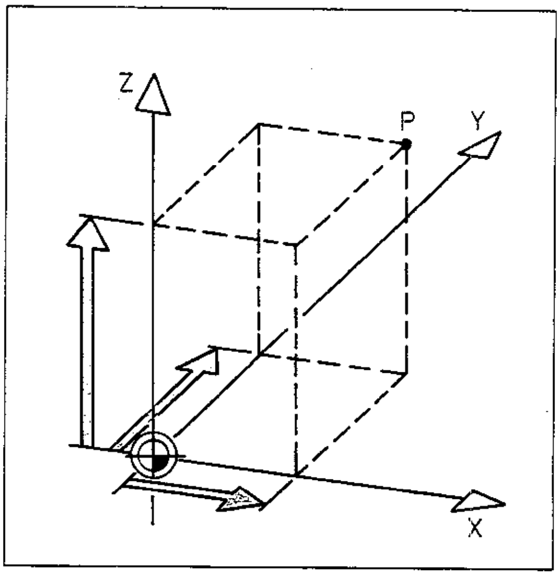
G51 T1

G51	Next tool
T ...	Tool number

Programming in ISO format
Dimensions

Cartesian coordinates

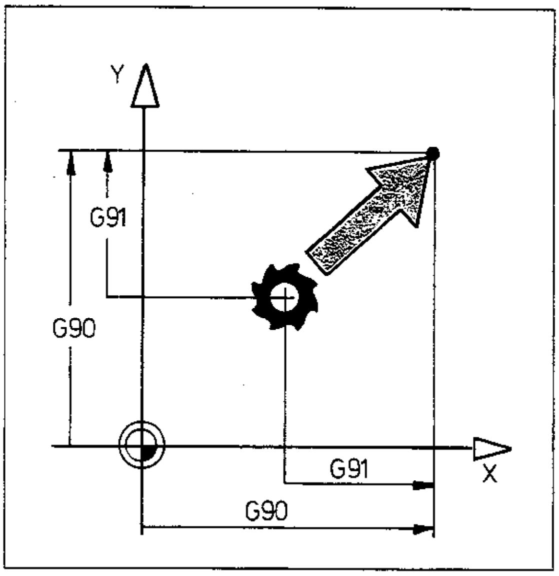
Cartesian coordinates are programmed via the X Y Z IV keys. Up to 3 target point coordinates can be specified for linear interpolation and up to 2 target point coordinates for circular interpolation.



Incremental/ absolute dimensions

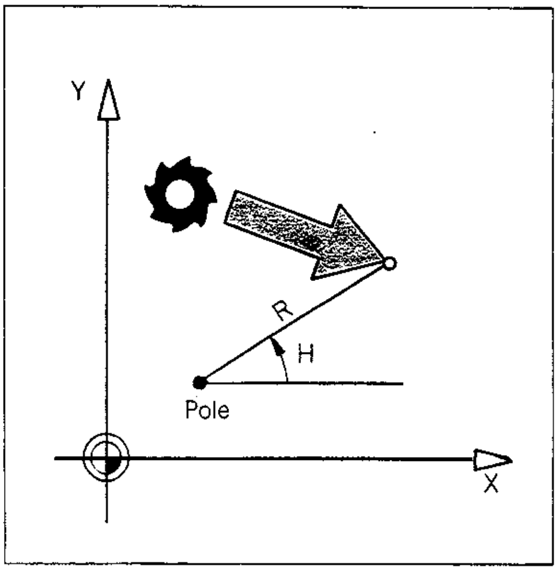
The G-code G90 "Absolute dimensions" and G91 "Incremental dimensions" are **modal** commands, i.e. each remains in effect until cancelled by the other G-code (G91 or G90).
To specify **coordinates in absolute dimensions,** the G-code **G90 – Absolute dimensions** must be entered before the coordinate or already be active.
To specify **coordinates in incremental dimensions,** the G-code **G91 – Incremental dimensions** must be entered before the coordinate or already be active.

G90 or G91 must be programmed before the first coordinate at the beginning of the machining program. Otherwise, this error message is displayed:
= PROGRAM START UNDEFINED =



Polar coordinates

Polar coordinates are programmed via the H key (polar coordinate angle H) and the R key (polar coordinate radius R).
The pole must be defined before entering the polar coordinates.



Pole/circle centre

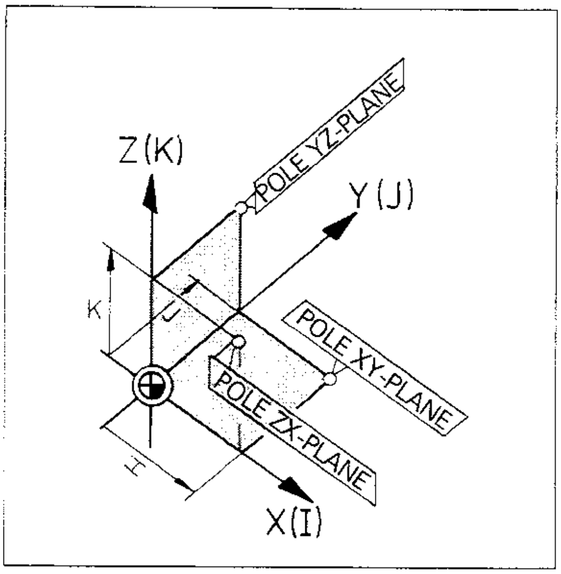
The pole/circle centre is always defined by two Cartesian coordinates. The axis designations for these coordinates are:

- I for the X-axis
- J for the Y-axis
- K for the Z-axis

The pole/circle centre must be located in the appropriate machining plane:

Machining plane	Coordinates of pole/ circle centre
X, Y plane	I, J
Y, Z plane	J, K
Z, Y plane	K, I

Use the I J K keys to enter the coordinates.

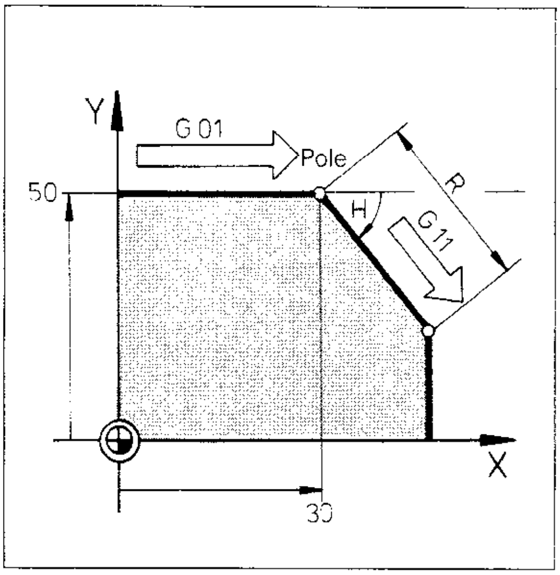


Pole definition G29

To transfer the last nominal position value as a pole, simply enter the code G29.

Example:

```
N30 G01 G90 X+30 Y+50

N40 G29 G11 R+50 H-45
```



Programming in ISO format
Linear interpolation

Target position in Cartesian coordinates

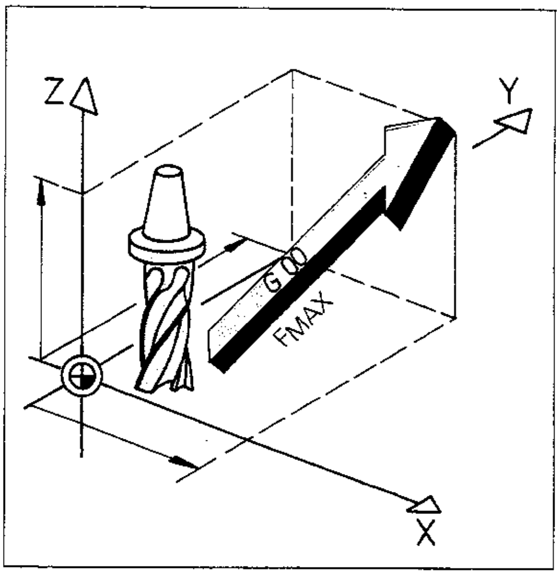
G00 Linear interpolation, Cartesian, **in rapid traverse.**

Block format (example)

```
G00 G90 X+80 Y+50 Z+10
```

G00	Linear interpolation, Cartesian, in rapid traverse
G90	Absolute dimensions
X ...	X-coordinate of target position
Y ...	Y-coordinate of target position
Z ...	Z-coordinate of target position

Simultaneous traversing of three machine axes in a straight line is not available on models TNC 151 F/TNC 155 F/TNC 151 W/ TNC 155 W.

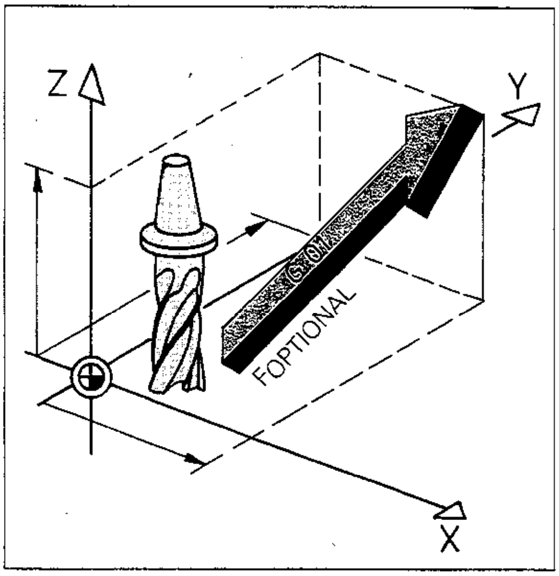


G01 Linear interpolation, Cartesian.

Block format (example)

```
G01 G90 X+80 Y+50 Z+10 F150
```

G01	Linear interpolation, Cartesian
G90	Absolute dimensions
X ...	X-coordinate of target position
Y ...	Y-coordinate of target position
Z ...	Z-coordinate of target position
F ...	Feed rate



Paraxial positioning

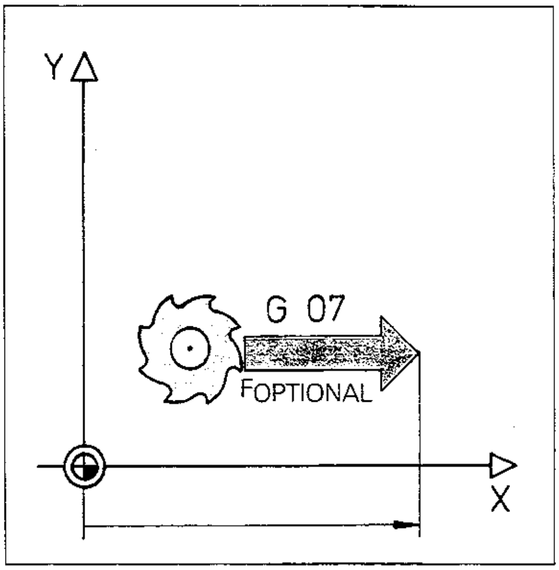
G07 Traverse in paraxial straight line.

Block format (example)

```
G07 G90 X+40 F190
```

G07	Paraxial positioning block
G90	Absolute dimensions
X ...	Coordinate of target position
F ...	Feed rate

G07 is active only in the block in which it is programmed (non-modal).



Programming in ISO format
Linear Interpolation

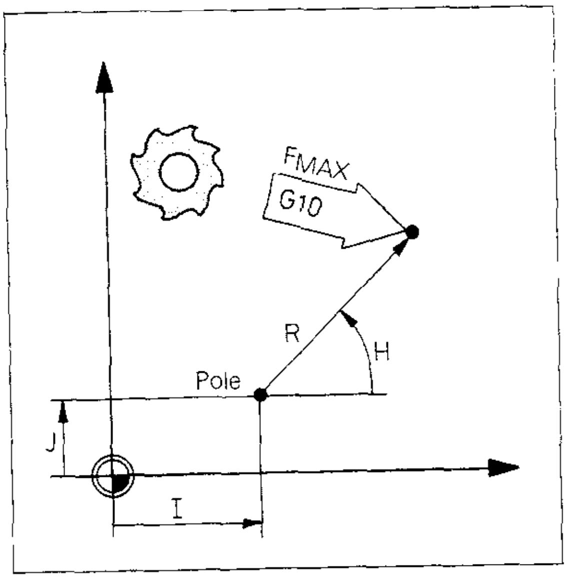
**et position
olar
dinates**

G10 Linear interpolation, polar, **in rapid traverse.**

Block format (example)

G90 I+20 J+10 G10 R+30 H+45

G90	Absolute dimensions
I ...	X-coordinate of pole
J ...	Y-coordinate of pole
G10	Linear interpolation, polar, in rapid traverse
R ...	Polar coordinate radius to end position
H ...	Polar coordinate angle to end position

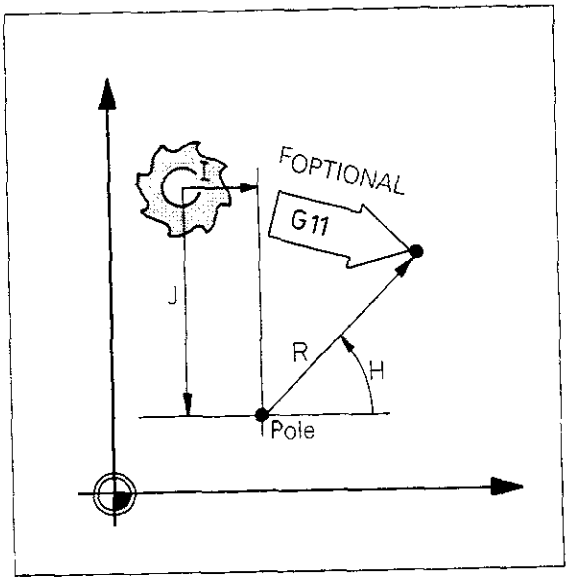


G11 Linear interpolation, polar.

Block format (example)

G91 I+10 J–30 G11 G90 R+30 H+45 F150

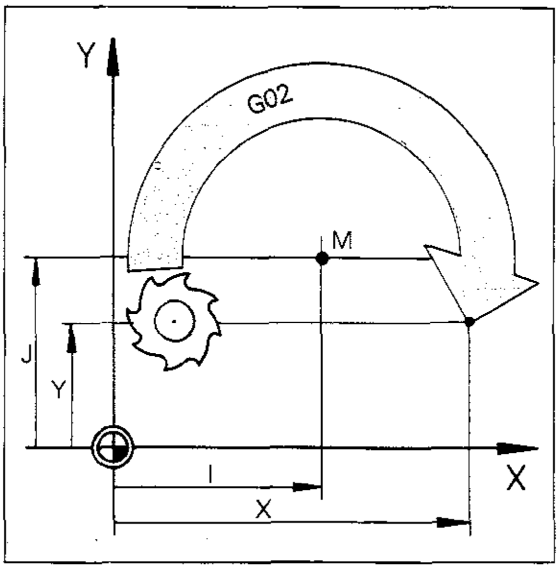
G91	Incremental dimensions
I ...	X-coordinate of pole
J ...	Y-coordinate of pole
G11	Linear interpolation, polar
G90	Absolute dimensions
R ...	Polar coordinate radius to end position
H ...	Polar coordinate angle to end position
F ...	Feed rate



Programming in ISO format
Circular interpolation

Target position in Cartesian coordinates

G02 Circular interpolation, Cartesian, **clockwise,** defined via **centre point** and **target position.**

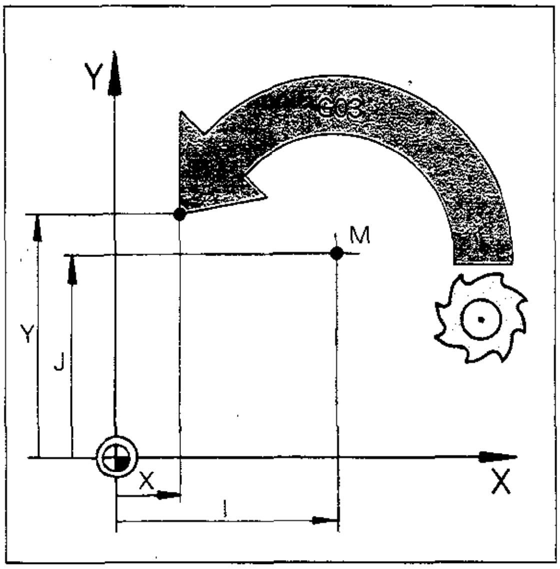


Block format (example)
Preceding block: Approach to starting point of arc

```
G90 I+30 J+30 G02 X+69 Y+23 F150
```

G90	Absolute dimensions
I ...	X-coordinate of circle centre
J ...	Y-coordinate of circle centre
G02	Circular interpolation, Cartesian, clockwise
X ...	X-coordinate of target position
Y ...	Y-coordinate of target position
F ...	Feed rate

G03 Circular interpolation, Cartesian, **counterclockwise,** defined by **centre point** and **end position.**

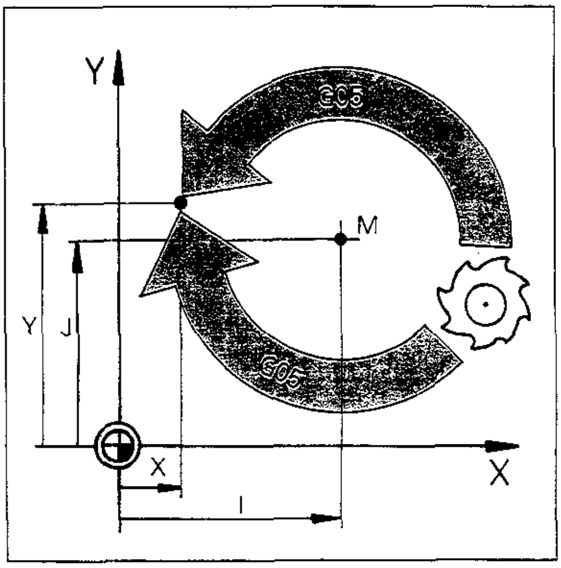


Block format (example)
Preceding block: Approach to starting point of arc

```
G90 I+30 J+28 G03 X+12 Y+32 F150
```

G90	Absolute dimensions
I ...	X-coordinate of circle centre
J ...	Y-coordinate of circle centre
G03	Circular interpolation, Cartesian, counterclockwise
X ...	X-coordinate of target position
Y ...	Y-coordinate of target position
F ...	Feed rate

G05 Circular interpolation, Cartesian, **no specified rotation direction,** defined by **centre point** and **end position.**



Block format (example)
Preceding block: Approach to starting point of arc

```
G90 I+22 J+20 G05 X+5 Y+30 F150
```

G90	Absolute dimensions
I ...	X-coordinate of circle centre
J ...	Y-coordinate of circle centre
G05	Circular interpolation, Cartesian, no specified rotation
X ...	X-coordinate of target position
Y ...	Y-coordinate of target position
F ...	Feed rate

If circular interpolation with specified rotation has not already been executed before circular interpolation with G05/G15, this message will appear:
= PROGRAM START UNDEFINED =

Programming in ISO format
Circular interpolation

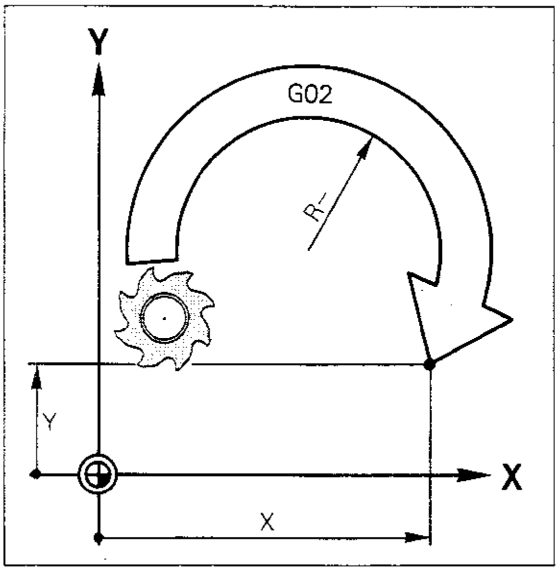
Target position in Cartesian coordinates

G02 Circular interpolation, Cartesian, **clockwise,** defined by **radius** and **end position.**

Block format (example)
Preceding block: Approach to starting point of arc

G02 G90 X+69 Y+23 R–20 F150

G02	Circular interpolation, Cartesian, clockwise
G90	Absolute dimensions
X ...	X-coordinate of end position
Y ...	Y-coordinate of end position
R–...	Circle radius, central angle greater than 180°
F ...	Feed rate

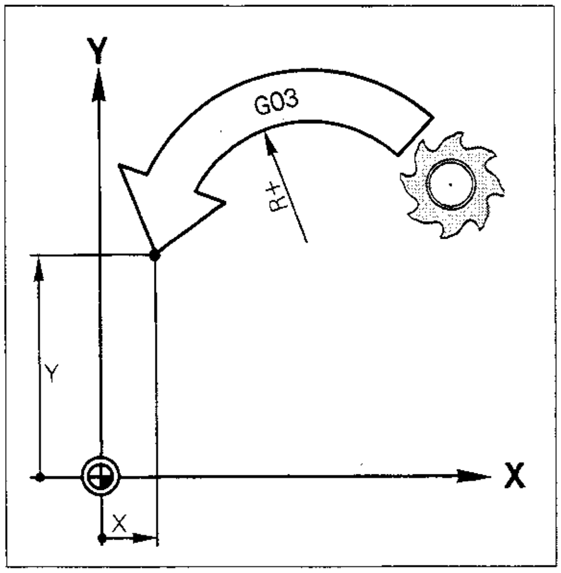


G03 Circular interpolation, Cartesian, **counterclockwise,** defined by **radius** and **end position.**

Block format (example)
Preceding block: Approach to starting point of arc

G03 G90 X+12 Y+32 R+20 F150

G03	Circular interpolation, Cartesian, counterclockwise
G90	Absolute dimensions
X ...	X-coordinate of end position
Y ...	Y-coordinate of end position
R+...	Circle radius, central angle less than 180°
F ...	Feed rate

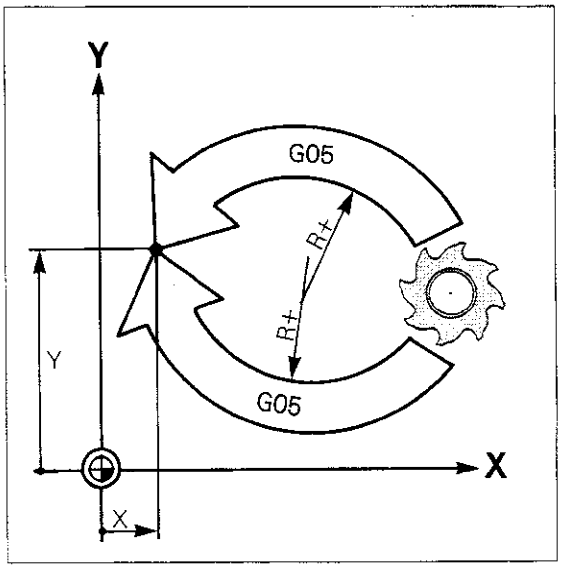


G05 Circular interpolation, Cartesian, **no specified rotation** defined by **radius** and **end position.**

Block format (example)
Preceding block: Approach to starting point of arc

G05 G90 X+5 Y+30 R+20 F150

G05	Circular interpolation, Cartesian, no specified rotation
G90	Absolute dimensions
X ...	X-coordinate of end position
Y ...	Y-coordinate of end position
R+...	Circle radius, central angle less than 180°
F ...	Feed rate



Programming in ISO format
Circular interpolation

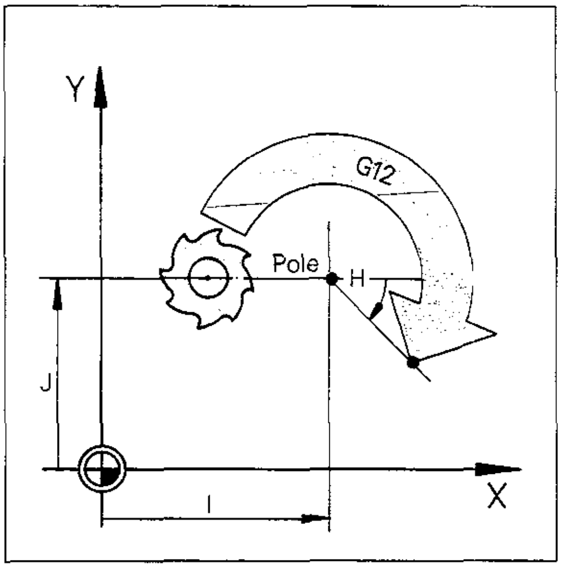
End position in polar coordinates

G12 Circular interpolation, polar, **clockwise.**

Block format (example)
Preceding block: Approach to starting point of arc

```
G90 I+50 J+40 G12 H–45 F150
```

G90	Absolute dimensions
I ...	X-coordinate of pole/circle centre
J ...	Y-coordinate of pole/circle centre
G12	Circular interpolation, polar, clockwise
H ...	Polar coordinate angle to end position
F ...	Feed rate

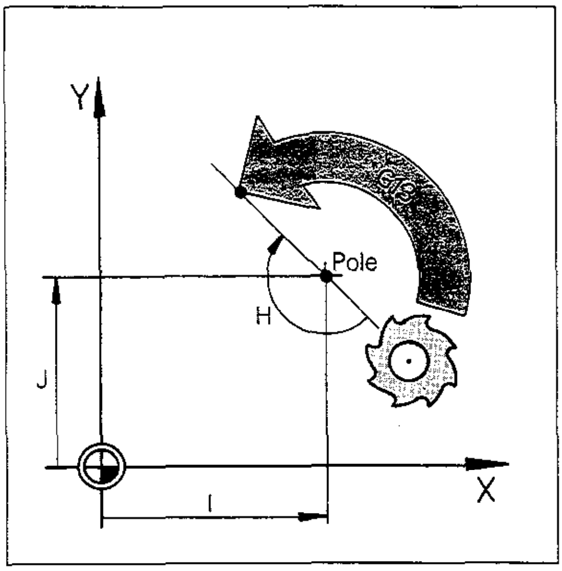


G13 Circular interpolation, polar, **counterclockwise.**

Block format (example)
Preceding block: Approach to starting point of arc

```
G90 I+30 J+25 G13 G91 H–180 F150
```

G90	Absolute dimensions
I ...	X-coordinate of pole/circle centre
J ...	Y-coordinate of pole/circle centre
G13	Circular interpolation, polar, counterclockwise
G91	Incremental dimensions
H ...	Polar coordinate angle to end position
F ...	Feed rate

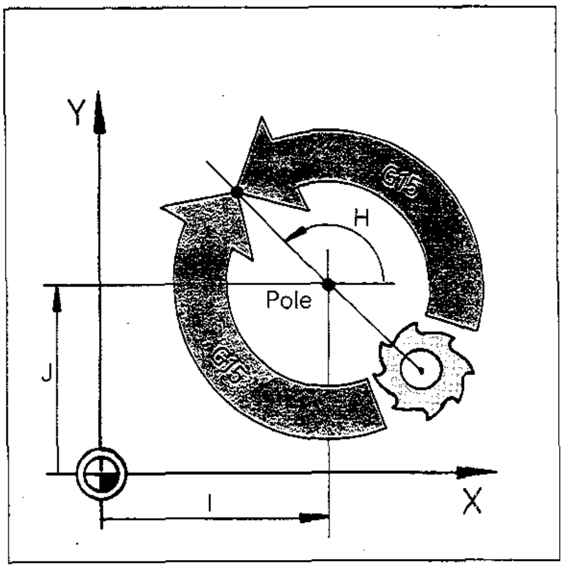


G15 Circular interpolation, polar, **no specified rotation** (also see function G05).

Block format (example)
Preceding block: Approach to starting point of arc

```
G90 I+50 J+40 G15 G91 H+120 F150
```

G90	Absolute dimensions
I ...	X-coordinate of pole/circle centre
J ...	Y-coordinate of pole/circle centre
G15	Circular interpolation, polar, no specified rotation
H ...	Polar coordinate angle to end position
F ...	Feed rate



Programming in ISO format
Circular interpolation
Tangential transition arc

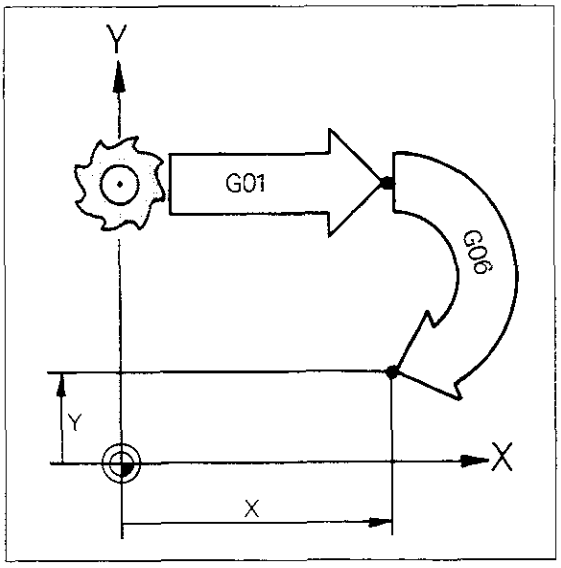
End position in Cartesian coordinates

G06 Circular interpolation, Cartesian, **tangential transition to contour,** defined by end position.

Block format (example)

G06 G90 X+50 Y+10

G06	Circular interpolation, Cartesian, tangential transition to contour
G90	Absolute dimensions
X ...	X-coordinate of end position
Y ...	Y-coordinate of end position



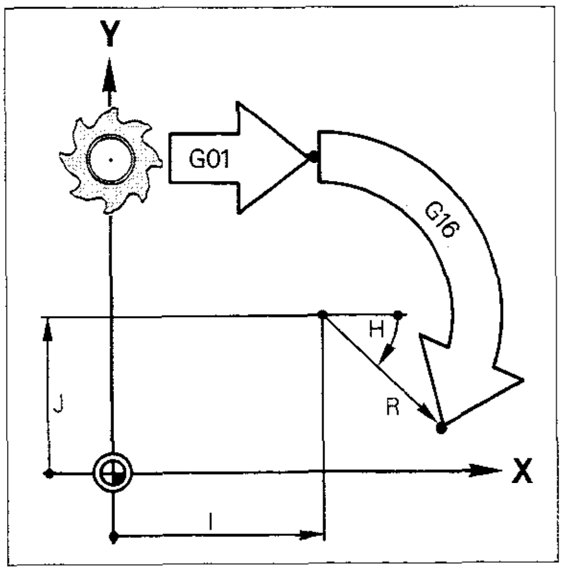
End position in polar coordinates

G16 Circular interpolation, polar, **tangential transition to contour,** defined by end position.

Block format (example)

G90 I+50 J+30 G16 R+15 H–60

G90	Absolute dimensions
I ...	X-coordinate of pole
J ...	Y-coordinate of pole
G16	Circular interpolation, Cartesian, tangential transition to contour
R ...	Polar coordinate radius to end position
H ...	Polar coordinate angle to end position

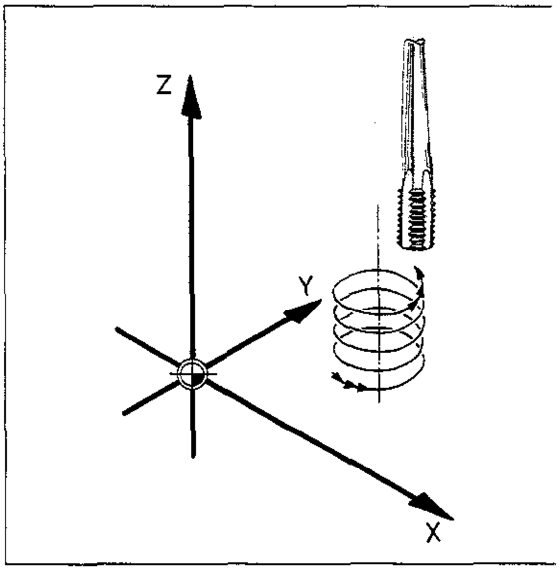


Programming in ISO format
Helical interpolation

Helical interpolation

Helical interpolation is the combination of a circular interpolation in the machining plane and superimposed linear motion on the tool axis. Please see "Helical interpolation" for further information.

Helical interpolation is not available on models TNC 151 F/TNC 155 F/TNC 151 W/ TNC 155 W.



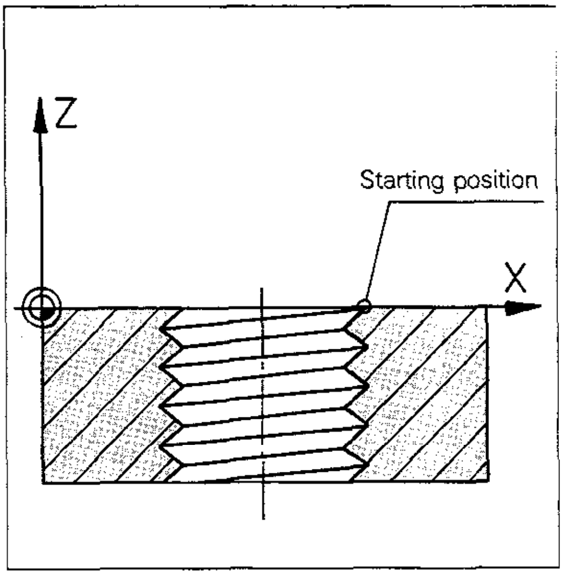
G12 ... Z Helical interpolation, **clockwise.**

G13 ... Z Helical interpolation, **counterclockwise.**

Block format (example)

```
G90 I+15 J+45 G12 G91 H+1080 Z-5
```

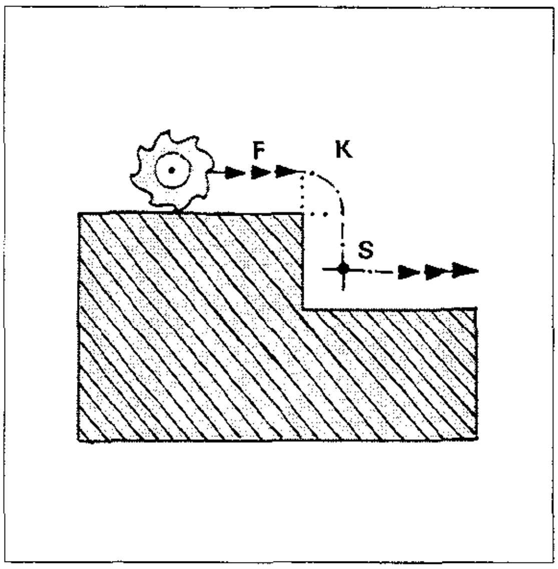
G90	Absolute dimensions
I ...	X-coordinate of pole/circle centre
J ...	Y-coordinate of pole/circle centre
G12	Circular interpolation, polar, clockwise
G91	Incremental dimensions
H ...	Polar coordinate angle = angle of rotation
Z ...	Height coordinate of helix



Programming in ISO format
Tool path compensation

Compensated tool path

Tool path compensation means that the tool moves to the left or right of the programmed contour, with the cutter axis offset by the amount of the **tool radius,** thus producing the actual programmed contour. A **transition arc K** is inserted into the tool path automatically on **outer corners.** On **inner corners** the TNC automatically calculates a **path intersection S** to prevent back-cutting on the contour.



Tool path compensation

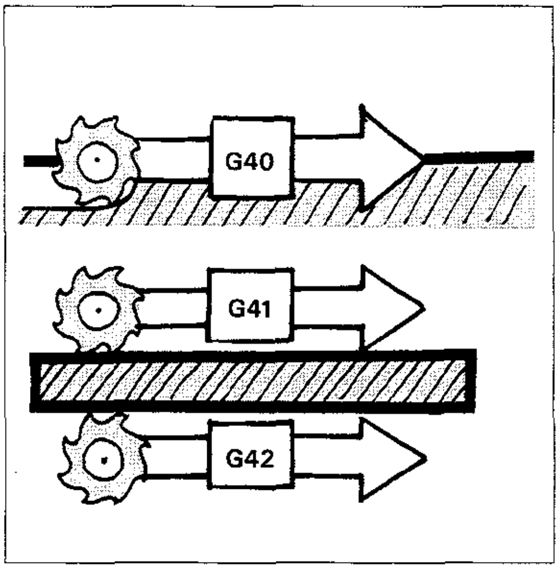
Tool path compensation is programmed via G-codes, which are **modal commands,** i.e. they remain active until cancelled or replaced by another G-code.

You can enter a tool path compensation in any **positioning block.**

G40 The tool moves precisely **on** the programmed contour. (Cancel path compensation with G41/G42/G43/G44).

G41 The tool moves on a path to the **left** of the contour.

G42 The tool moves on a path to the **right** of the contour.

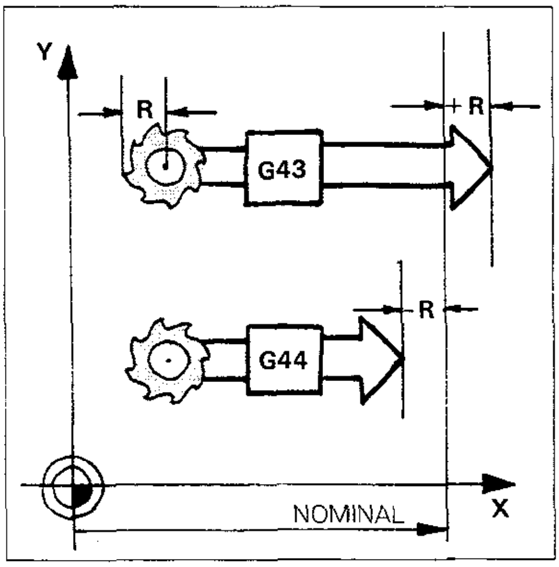


Tool radius compensation with paraxial positioning blocks

In the case of paraxial positioning blocks, the tool path can be shortened or extended by the amount of the tool radius.

G43 Tool path is extended

G44 Tool path is shortened



Programming in ISO format
Chamfers/Rounding corners

Chamfers

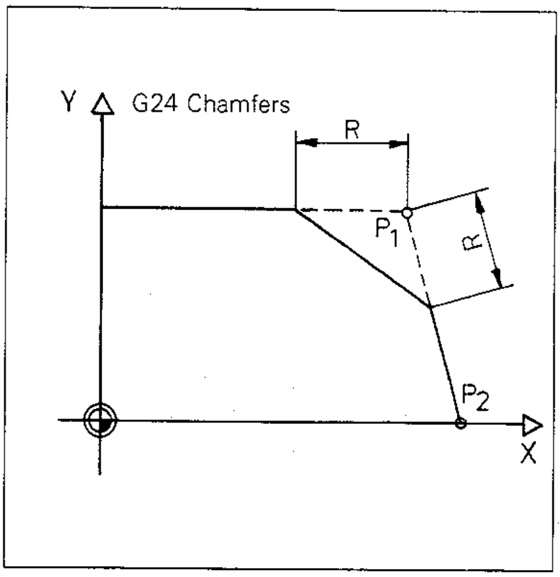
G24 Chamfers

Program format

```
N25 G01 X ... Y ... (position P1)
N26 G24 R ... (chamfer)
N27 X ... Y ... (position P2)
```

The function G24 can also be programmed in the positioning block for the corner P1 to be chamfered.

Please see "Chamfers" for further explanation.



Rounding corners

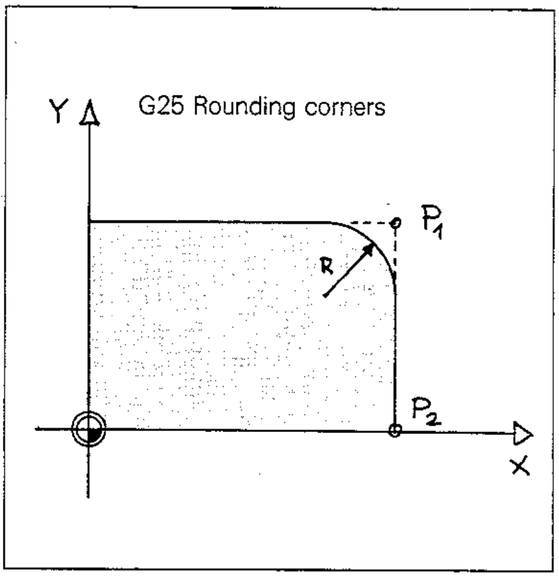
G25 Rounding corners

Program format

```
N15 G01 X ... Y ... (position P1)
N16 G25 R ... (corner radius)
N17 X ... Y ... (position P2)
```

The function G25 can also be programmed in the positioning block for the corner P1 to be rounded.

Please see "Rounding corners" for further explanation.



A positioning block with both coordinates of the machining plane must be programmed before and after the rounding corners/ chamfer function.

Programming in ISO format
Tangential contour approach and departure

Tangential contour approach

G26 Contour approach on an arc with tangential transition to first contour element (dialogue-prompted).

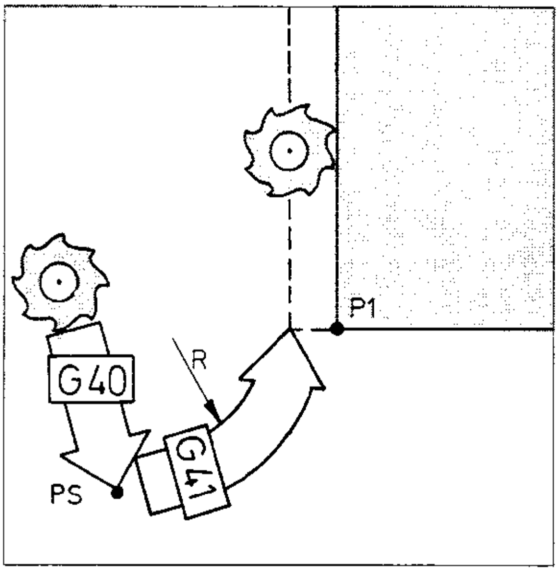
Program format

N25 G40 G01 X ... Y ... (position PS)

N26 G41 X ... Y ... (position P1)

N27 G26 R ... (arc)

The function G26 can also be programmed in the positioning block for the first contour position P1. See "Contour approach on an arc" for explanation.



Tangential contour departure

G27 Contour departure on an arc with tangential transition to the previously finished contour element (dialogue-prompted).

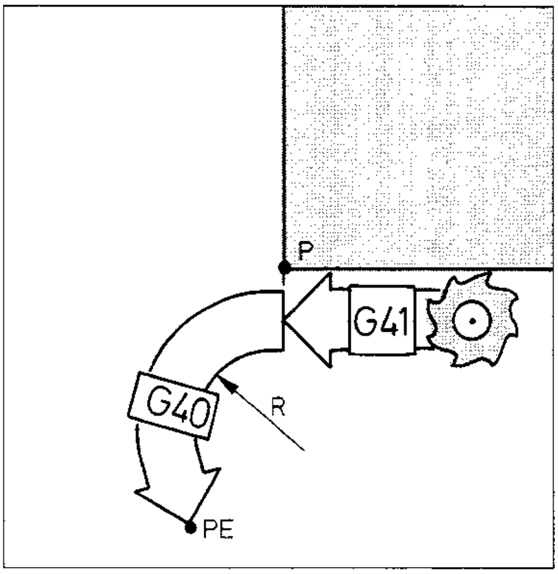
Program format

N35 G41 G01 X ... Y ... (position P)

N36 G27 R ... (arc)

N37 G40 X ... Y ... (position PE)

The function G27 can also be programmed in the positioning block for the last contour position P. See "Contour departure on an arc" for explanation.



Programming in ISO format
Canned cycles

Categories

Cycles are subdivided into the following categories:
- **Machining cycles** (for workpiece machining).
- **Coordinate transformations** (for altering the coordinate system).
- **Dwell time**
- **Freely programmable (variable) cycles**
- **Spindle orientation** (optional)

Machining cycles are defined via the G-codes and must be called up separately following cycle definition via G79 – "Cycle call" or M99 – "Cycle call" or M89 – "Modal cycle call". This also applies to the **freely programmable (variable) cycles** and **spindle orientation.**

Coordinate transformations are effective immediately following cycle definition via G-codes and do not require a separate cylce call. This is also true of the **Dwell time** and **Contour** cycles.

Programmable **machining cycles** (dialogue-prompted):

G83	Peck drilling
G84	Tapping

G74	Slot milling
G75	Rectangular pocket milling, clockwise
G76	Rectangular pocket milling, counterclockwise

G37	Definition of pocket contour
G56	Pilot drilling of contour pocket
G57	Rough-out contour pocket
G58	Contour milling (finish), clockwise
G59	Contour milling (finish), counterclockwise

Programmable **coordinate transformations** (semi-dialogue-prompted)

G28	Mirror image
G54	Datum shift
G72	Scaling factor
G73	Coordinate system rotation

Additional cycles (dialogue-prompted)

G04	Dwell time
G36	Spindle orientation (optional)
G39	Freely programmable cycle (Program call)

Peck drilling

G83 Peck drilling (dialogue-prompted)

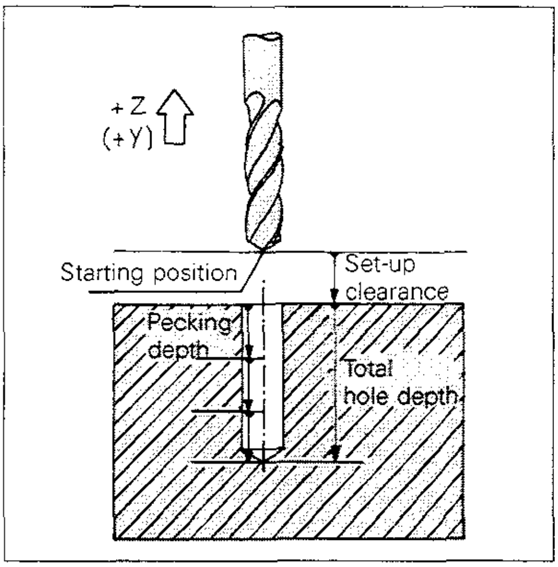
Block format (example)

G83 P01 –2 P02 –20 P03 –10

P04 0 P05 150

G83	Peck drilling
P01	Set-up clearance
P02	Total hole depth
P03	Pecking depth
P04	Dwell
P05	Feed rate

See "Peck drilling" for explanation of cycle parameters and cycle procedure.



The cycle parameters P01/P02/P03 must have the same sign.

Tapping

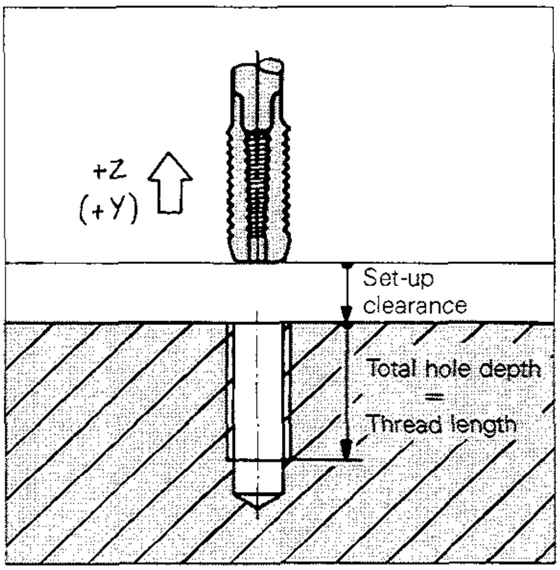
G84 Tapping (dialogue-prompted)

Block format (example)

G84 P01 –2 P02 –20 P03 0 P04 80

G84	Tapping
P01	Set-up clearance
P02	Total hole depth (thread length)
P03	Dwell
P04	Feed rate

See "Tapping" for explanation of cycle parameters and cycle procedure.



The cycle parameters P01/P02 must have the same sign.

Slot milling

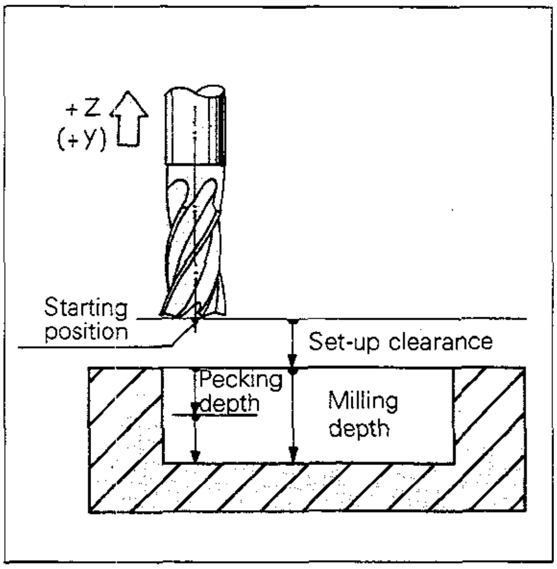
G74 Slot milling (dialogue-prompted)

Block format (example)

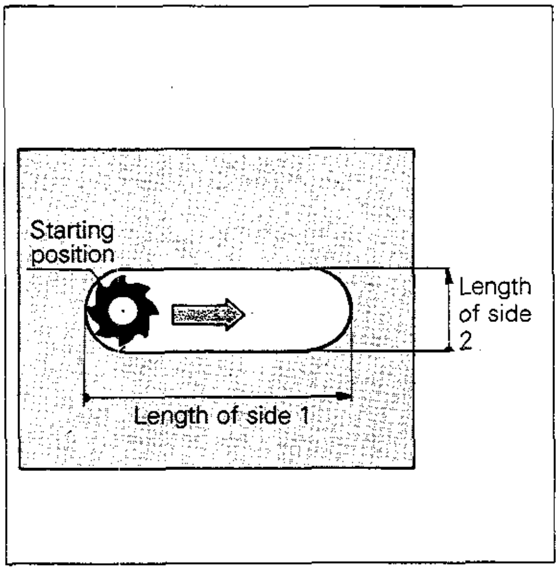
```
G74 P01 -2 P02 -20 P03 -10 P04 80
P05 X+50 P06 Y+10 P07 150
```

G74	Slot milling
P01	Set-up clearance
P02	Milling depth
P03	Pecking depth
P04	Feed rate for vertical feed
P05	Longitudinal axis and length of slot
P06	Transverse axis and width of slot
P07	Feed rate

See "Slot milling" for explanation of cycle parameters and cycle procedure.



Cycle parameters P01/P02/P03 must have the same sign.



Programming in ISO format
Machining cycles

Milling rectangular pockets

G75 Rectangular pocket milling, **clockwise** (dialogue-prompted)

G76 Rectangular pocket milling, **counterclockwise** (dialogue-prompted)

Block format (example G76)

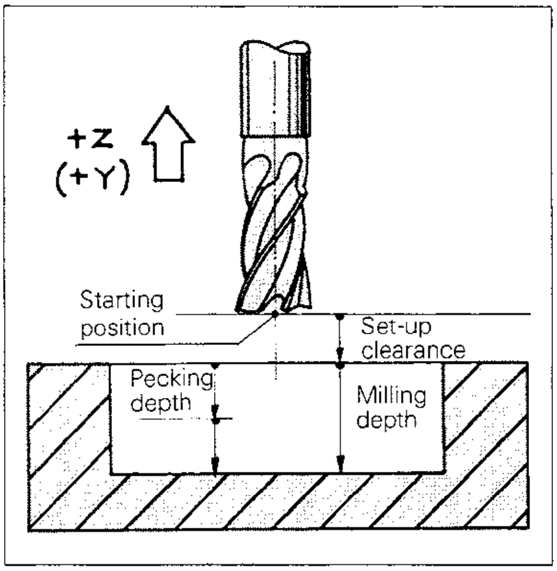
G76 P01 –2 P02 –20 P03 –10 P04 80

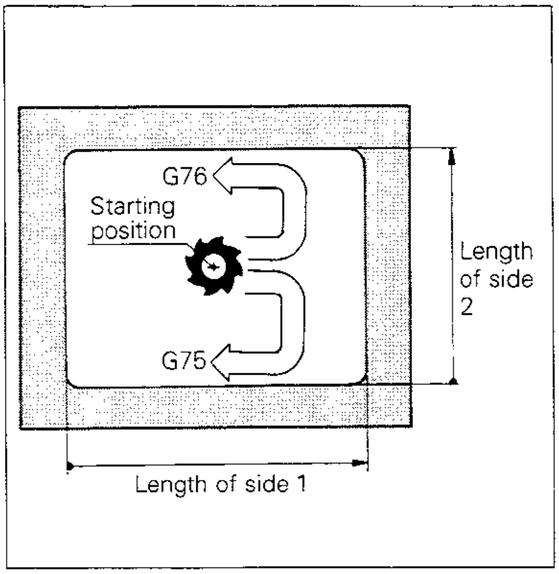
P05 X+90 P06 Y+50 P07 150

G76	Rectangular pocket milling, counterclockwise
P01	Set-up clearance
P02	Milling depth
P03	Pecking depth
P04	Feed rate for vertical feed
P05	1st axis and side length of pocket
P06	2nd axis and side length of pocket
P07	Feed rate

See "Pocket milling" for explanation of cycle parameters and cycle procedure.

Cycle parameters P01/P02/P03 must have the same sign.
Cycle parameters P05 and P06 must have a positive sign.





Programming in ISO format
Machining cycles

Milling circular pockets

G77 Circular pocket milling, **clockwise** (dialogue-prompted)

G78 Circular pocket milling, **counterclockwise** (dialogue-prompted)

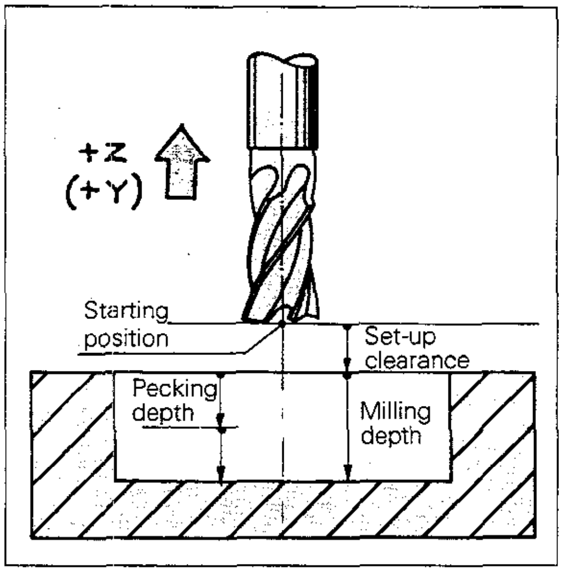
Block format (example G78)

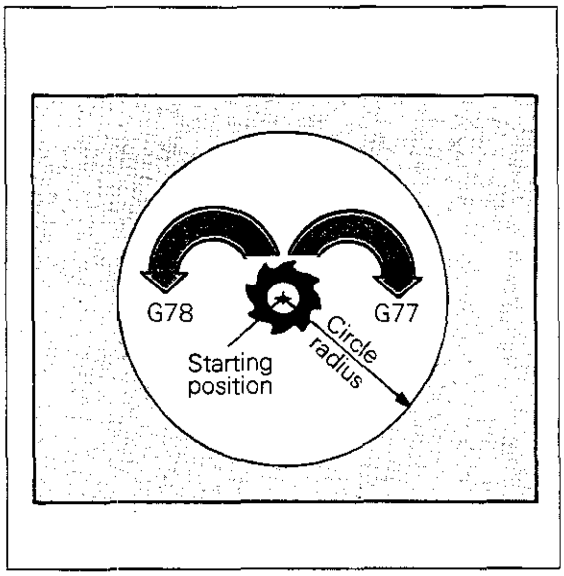
```
G78 P01 -2 P02 -20 P03 -10 P04 80
P05 90 P06 150
```

G78	Circular pocket milling, counterclockwise
P01	Set-up clearance
P02	Milling depth
P03	Pecking depth
P04	Feed rate for vertical feed
P05	Circle radius
P06	Feed rate

See "Circular pocket" for explanation of cycle parameters and cycle procedure.

Cycle parameters P01/P02/P03 must have the same sign.





Contour

G37 Definition of pocket contour (dialogue-prompted)

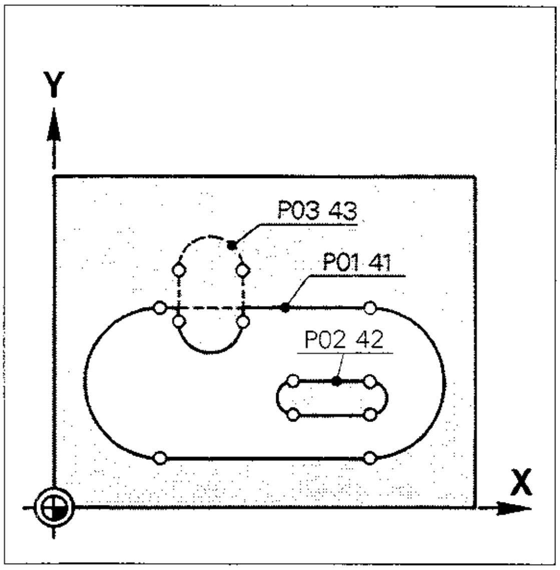
Block format (example)

G37 P01 41 P02 42 P03 43 P04

P05 P06 P07 P08 P09 P10 P11 P12

G37 Definition of pocket contour
P01 First subcontour (must be programmed as pocket)
P02 Second subcontour
.
.
.
P12 Twelfth subcontour

See "Contour cycle" for explanation of cycle.



Pilot drilling

G56 Pilot drilling of contour pocket (dialogue-prompted)

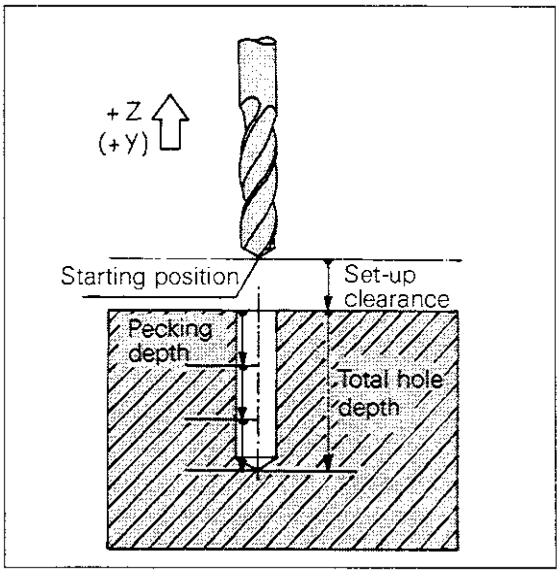
Block format (example)

G56 P01 –2 P02 –18 P03 –10

P04 40 P05 1,5

G56 Pilot drilling of contour pocket
P01 Set-up clearance
P02 Total hole depth
P03 Pecking depth
P04 Feed rate
P05 Finishing allowance

See "Pilot drilling" for explanation of cycle parameters and cycle procedure.



Cycle parameters P01/P02/P03 must have the same sign.

Rough-out

G57 Rough-out contour pocket (dialogue-prompted)

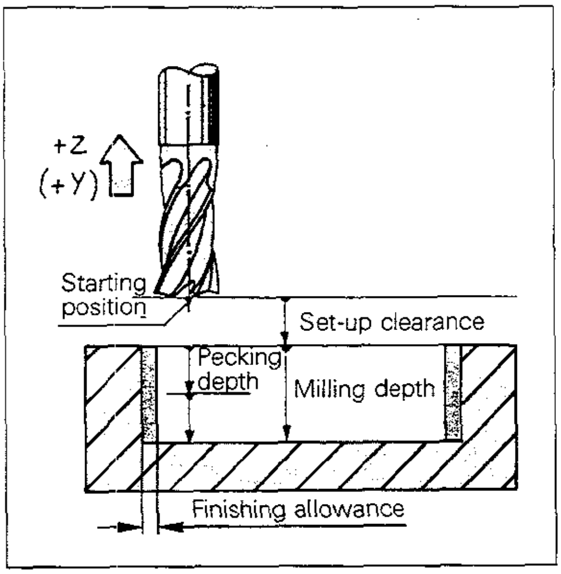
Block format (example)

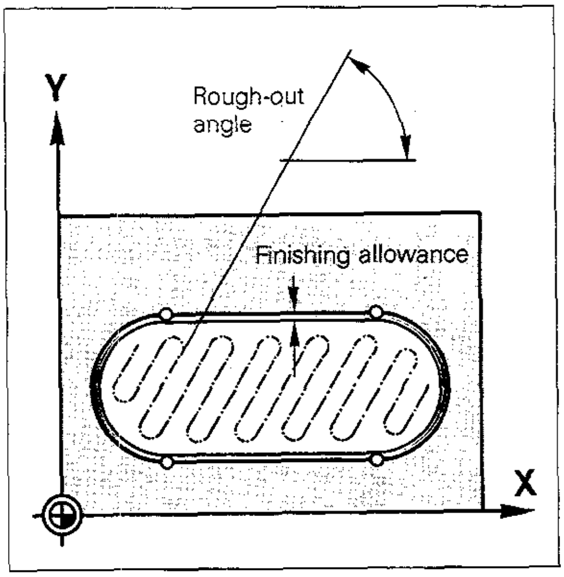
```
G57 P01 –2 P02 –18 P03 –10
P04 40 P05 2 P06 +45 P07 120
```

P01	Set-up clearance
P02	Milling depth
P03	Pecking depth
P04	Feed rate for vertical feed
P05	Finishing allowance
P06	Rough-out angle
P07	Feed rate

See "Rough-out" for explanation of cycle parameters and cycle procedure.

Cycle parameters P01/P02/P03 must have the same sign.





ntour milling

G58 Contour milling (finish), clockwise (dialogue-prompted)

G59 Contour milling (finish), counter-clockwise (dialogue-prompted)

Block format (example G58)

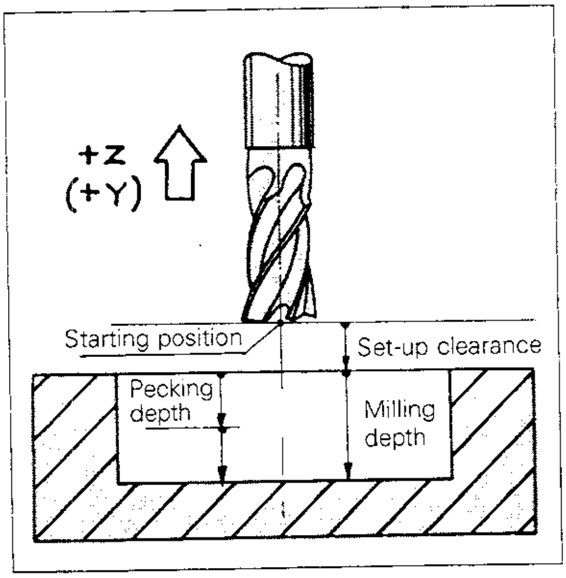
G58 P01 –2 P02 –18 P03 –10

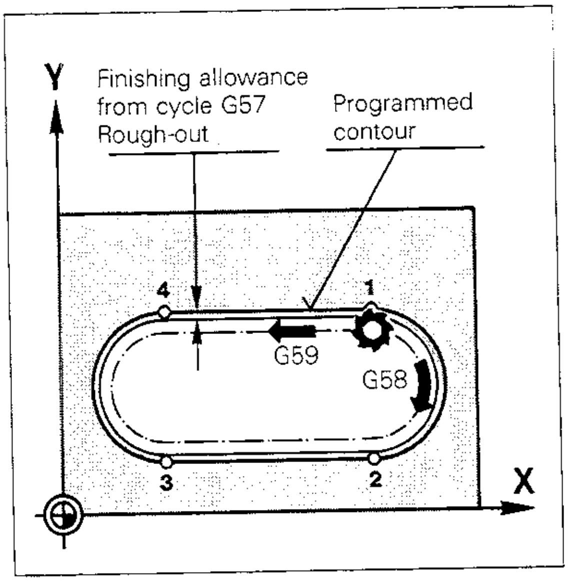
P04 80 P05 120

G58	Contour milling, clockwise
P01	Set-up clearance
P02	Milling depth
P03	Pecking depth
P04	Feed rate for vertical feed
P05	Feed rate

See "Contour milling" for explanation of cycle parameters and cycle procedure.

Cycle parameters P01/P02/P03 must have the same sign.





Programming in ISO format
Coordinate transformations

Mirror image

G28 Mirror image

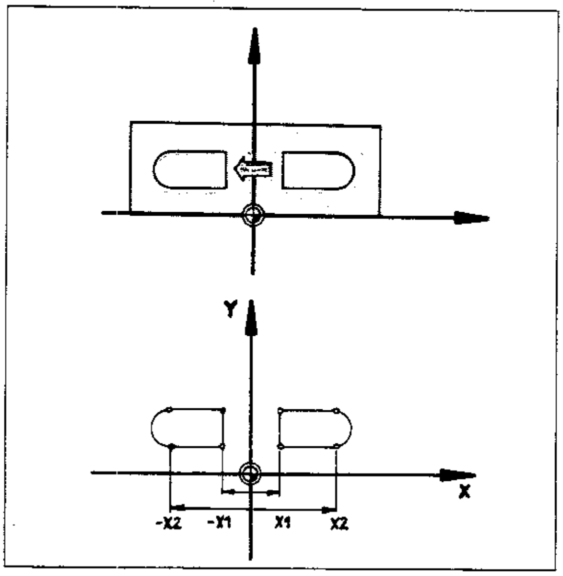
Block format (example)

G28 X

G28	Mirror image cycle
X	Mirrored axis

Two axes can also be mirror-imaged simultaneously; the tool axis cannot be mirror-imaged.

See "Mirror image" for explanation of cycle.



Datum shift

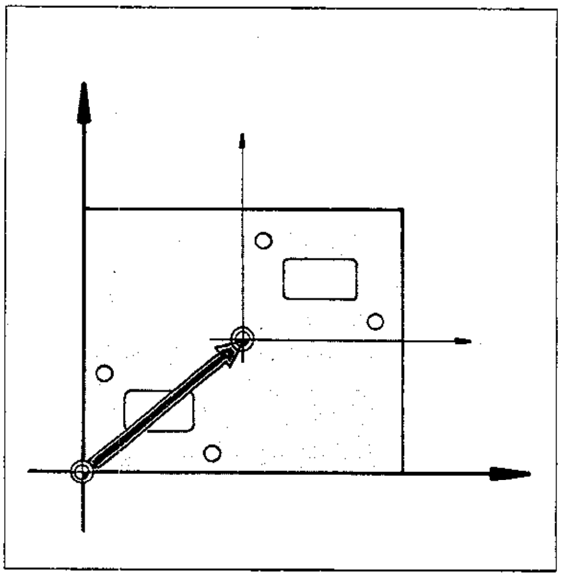
G54 Datum shift

Block format (example)

G54 G90 X+50 G91 Y+15 Z–10

G54	Datum shift cycle
G90	Absolute dimensions
X ...	Shift of X-axis
G91	Incremental dimensions
Y ...	Shift of Y-axis
Z ...	Shift of Z-axis

See "Datum shift" for explanation of cycle.



Scaling factor

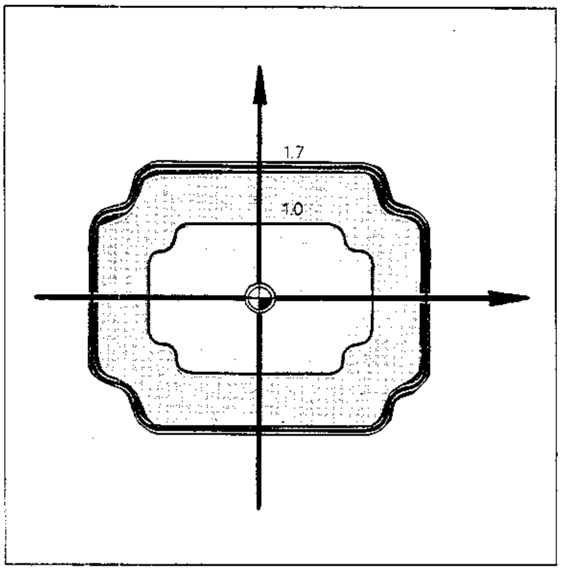
G72 Scaling factor (dialogue-prompted)

Block format (example)

G72 F1.7

G72	Scaling factor (cycle)
F ...	Scaling factor

See "Scaling factor" for explanation of cycle.



Programming in ISO format
Coordinate transformations
Dwell time cycle, freely programmable cycle

Coordinate system rotation

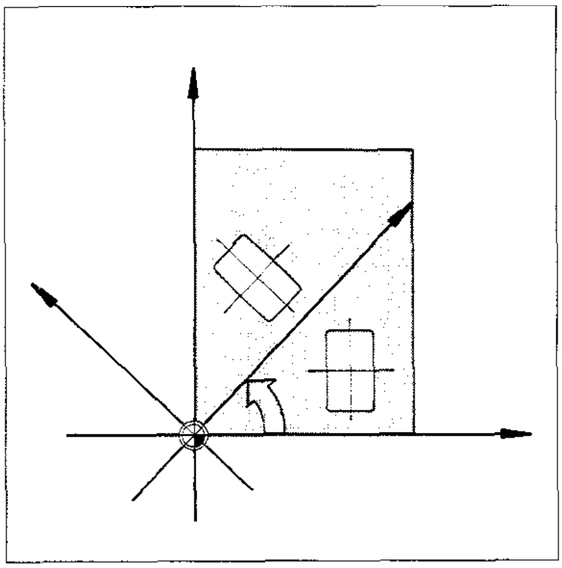
G73 Rotation of coordinate system (dialogue-prompted)

Block format (example)

G90 G73 H+120 G17

G90	Absolute dimensions
G73	Coordinate system rotation (cycle)
H ...	Angle of rotation
G17	Selection of plane for angular reference axis

See "Rotation of coordinate system" for explanation of cycle.



Dwell time cycle

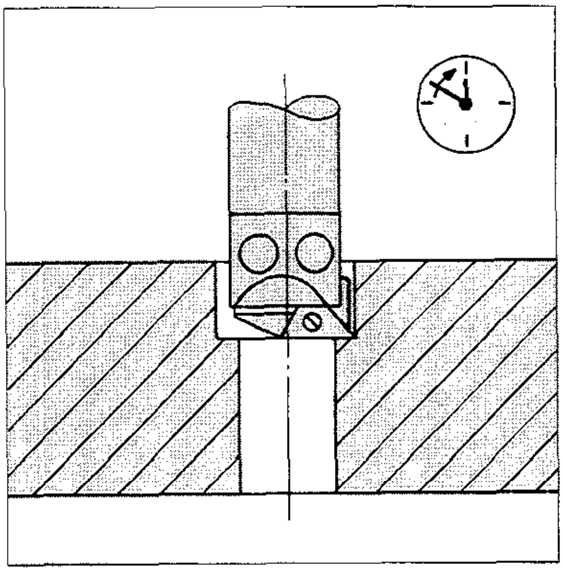
G04 Dwell time (dialogue-prompted)

Block format (example)

G04 F5

G04	Dwell time (cycle)
F ...	Dwell time in seconds

See "Dwell time" for explanation of cycle.



Freely programmable cycle (Program call)

G39 Freely programmable cycle (dialogue-prompted)

Block format (example)

G39 P01 12

G39	Freely programmable cycle (Program call)
P01	Program number

See "Freely programmable (variable) cycle" for explanation of cycle.

Notes:

Programming in ISO format
Touch-probe function
Spindle orientation cycle

Spindle orientation (optional)

G36 Spindle orientation (optional, dialogue-prompted)

Block format (example)

G36 S+45

G36	Spindle orientation cycle
S ...	Angular position of spindle

See "Spindle orientation" for explanation of cycle.

Workpiece surface as reference plane

G55 Touch-probe function, workpiece surface as reference plane (dialogue-prompted)

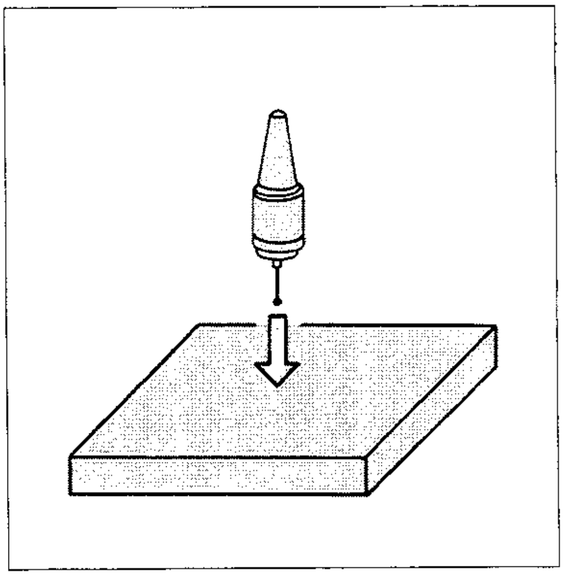
Block format (example)

G55 P01 10 P02 Z– P03 G90

X+50 Y+50 Z–20

G55	Workpiece surface as reference plane
P01	Parameter number for measurement
P02	Approach axis and direction
P03	Probing point

See chapter "Touch-probe" for explanation of probe function.



Programming in ISO format
Subroutines and program part repeats

Label number

A **label number** (program marker) is programmed with the command G98 L . . . This label number can be included in any desired program block that does not contain a **label call.**

A **jump command** is programmed with the address "L" followed by the label number.

A jump command with G98 L . . . should not be programmed in the same block as a label call "L . . .".

Program label:

N35 G98 L15 G01 . . .

Label number 15

Label call:

N45 L15 . . .

Program part

The program part is identified by G98 L . . . (label number) at the beginning of the program.

The label call "L . . ., . . ." forms the end of a program part repeat. When programming a **program part repeat,** enter the number of repetitions after the label number. Separate the label number from the number of repetitions with a decimal point [•], e. g.:

L 15.8: call label 15
repeat program part 8 times.

Program part:

N35 G98 L15 G01 . . .

⋮

Program part repeat:

N70 L15.8

Subroutine

The beginning of a subroutine is identified by G98 L . . . (label-number). The end is formed by entering G98 L0 (label number 0).

A **subroutine call** is also programmed by entering the address L followed by the label number.

Do not program repetitions together with a subroutine call.

Subroutine:

N75 G98 L19 G00 . . .

⋮

N90 G98 L0

Subroutine call:

N150 L19

Programming in ISO format
Program jump/STOP block

Jump to another program

Use the PGM CALL key to program a jump to another program.

Block format (example)

%29

%... Program call

See "Program call" for further information.

STOP block

G38 Corresponds to STOP block in HEIDENHAIN format.

Block format (example)

G38

Programming in ISO format
Parameter programming

Setting parameters

Parameters are markers for numerical values that are based on units of measurement. They are identified by the letter "Q" and a number and are entered (set) using the Q key.

Defining parameters

Parameter definition is the process of assigning a given numerical value or allocating a numerical value via mathematical or logical functions. Parameter definition consists of the **address D** and a code number (see table at right). Parameter definition is dialogue-prompted.

D00:	Assignment
D01:	Addition
D02:	Subtraction
D03:	Multiplication
D04:	Division
D05:	Square root
D06:	Sine
D07:	Cosine
D08:	Root sum of squares
D09:	IF equal, THEN jump
D10:	IF not equal, THEN jump
D11:	IF greater than, THEN jump
D12:	IF less than, THEN jump
D13:	Angle
D14:	Error number

Block format

Program definition requires a program block. The individual **block components** of parameter definition are identified by the **letter P** and a **number** (also see cycle parameters for machining cycles). The significance of these components depends on their sequence in the block, which in turn, depends on the input dialogue. To **check** this, we recommend moving the highlighted pointer in the block with the → and ← keys. The corresponding dialogue prompt for each block component will be displayed.

Programming in ISO format
Parameter programming

Example 1: Q98 = $\sqrt{+2}$

D05 Q98 P01 +2

D05	Square root
Q98	Parameter to which result is assigned
P01	Parameter or numerical value in square root

Example 2: Q12 = Q2 x 62

D03 Q12 P01 +Q2 P02 +62

D03	Multiplication
Q12	Parameter to which result is assigned
P01	Factor 1 (parameter or numerical value)
P02	Factor 2 (parameter or numerical value)

Example 3: IF Q6 less than Q5, THEN jump to LBL 3

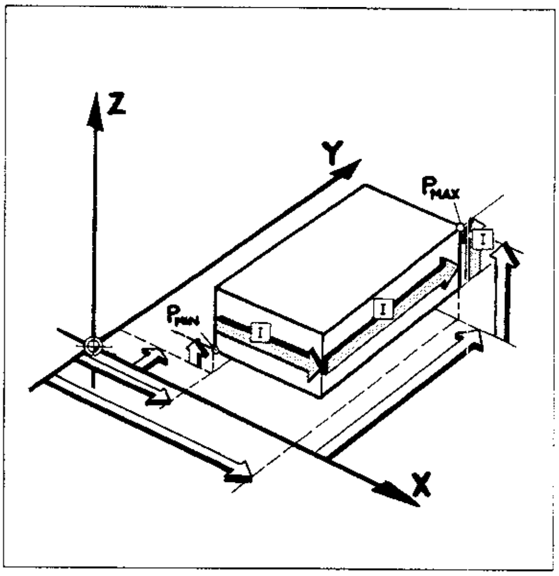
D12 P01 +Q6 P02 +Q5 P03 3

D12	IF less than, THEN jump
P01	First comparative value or parameter
P02	Second comparative value or parameter
P03	Label number

Programming in ISO format
Graphics – Blank form definition

Definition of blank

The blank workpiece (BLANK FORM) is defined by points P_{MIN} and P_{MAX} – see "Blank form" (Graphics).
The tool axis must be specified via G17/G18/G19, in addition to P_{MIN}.
Otherwise, this error message will appear:
= BLK FORM DEFINITION INCORRECT =



Entering P_{MIN}

G30 Definition of point P_{MIN} (input in absolute dimensions only)

Block format (example)

G30 G17 X+5 Y+5 Z–10

G30	Definition of P_{MIN}
G17	Plane selection and tool axis
X ...	X-coordinate of P_{MIN}
Y ...	Y-coordinate of P_{MIN}
Z ...	Z-coordinate of P_{MIN}

The function G90 (absolute dimensions) can be omitted if G30 is programmed.

Entering P_{MAX}

G31 Definition of point P_{MAX} (input in absolute or incremental dimensions)

Block format (example)

G31 G91 X+95 Y+95 Z+10

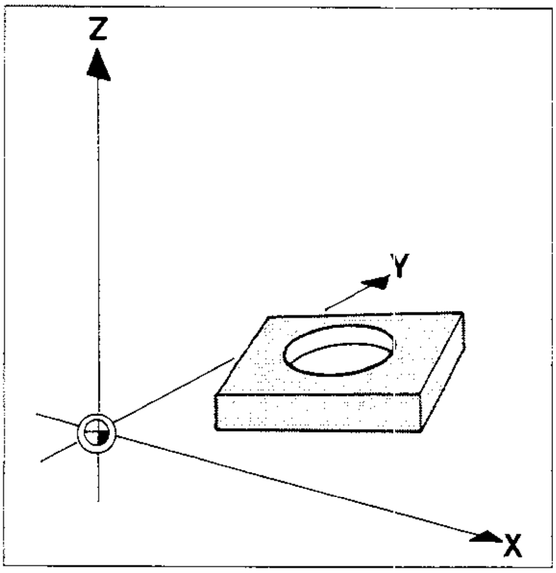
G31	Definition of P_{MAX}
G91	Incremental dimensions
X ...	X-coordinate of P_{MAX}
Y ...	Y-coordinate of P_{MAX}
Z ...	Z-coordinate of P_{MAX}

You can interrupt the graphic simulation of the machining procedure by pressing DEL □.

Touch-probe
Introduction

Touch-probe

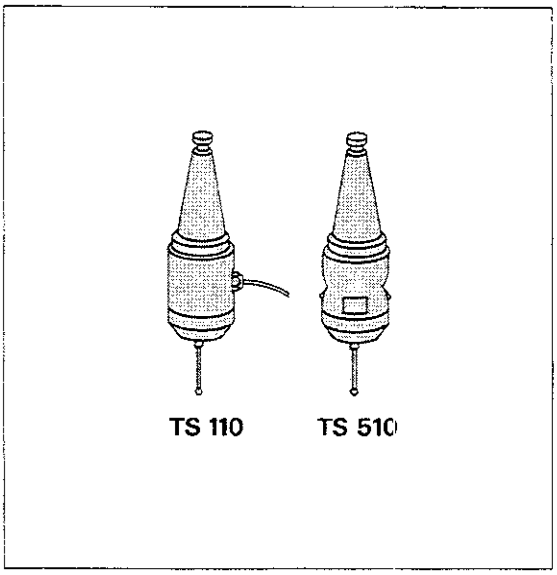
Operated in conjunction with a HEIDENHAIN touch-probe system, the **TNC control systems** can automatically detect misalignment in clamped workpieces. The misalignment is computed, stored and automatically compensated for when the workpiece is machined. This makes accurate alignment of the workpiece during set-up unnecessary. The programmable probing function permits workpiece inspection before or during the machining procedure. In the case of castings with varying elevations, for example, the surface can be probed before machining, allowing the correct depth to be reached when machined later. In the same way, changes in position caused by a rise in machine temperatures can be monitored at specified intervals and compensated for.



Versions

The touch-probe system is available in two versions:
Touch-probe 111 system with cable;
probe signal transmission and power supply via cable connector. The touch-probe 111 system consists of the TS 111 probe head and APE interface electronics.
Touch-probe 511 system featuring infrared transmission and battery power supply.
The touch-probe 511 system consists of the TS 511 probe head, APE 510 or APE 511 (for connecting two SE 510's) interface electronics and the SE transmitter/receiver unit.

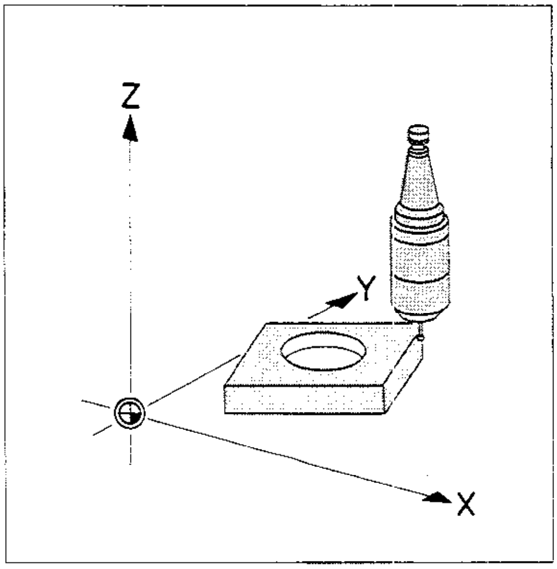
Both versions feature a standard tool shank and can be clamped in the spindle like an ordinary tool. The stylus can be replaced. The batteries of the TS 511 probe head with infrared transmission have a service life of 8 hours in probing operation and 1 month in standby mode.



The TS 511 probe head features a transmitter/receiver window on one side (for the triggering signal) and a transmitter window offset by 180°. The side with the transmitter/receiver window must face the SE transmitter/receiver unit when probing the workpiece. The transmitter window on the other side is not required for use with the HEIDENHAIN control systems.

Operation

The probe head moves to the side or upper surface of the workpiece. The feed rate for probing and the maximum stylus overtravel are determined by the machine parameters defined by the machine manufacturer. The probe signals the control system when it contacts the workpiece and the TNC saves the coordinates of the probed points. With the touch-probe function, workpiece surfaces, corners and circle centres can be easily determined and set for use as reference surfaces or reference points.



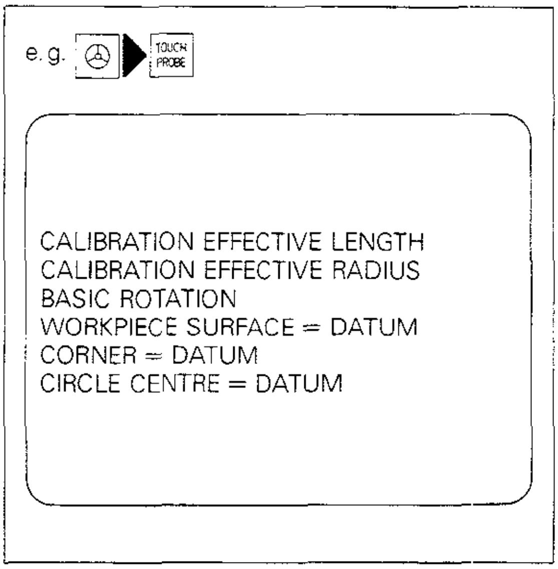
Touch-probe
Dialogue initiation/Error messages

Dialogue initiation

The touch-probe system operates in the following modes:

- Electronic handwheel
- Manual mode
- single block/full sequence program run.

Initiate the input dialogue by pressing TOUCH PROBE.
If you are currently in "Electronic handwheel" or "Manual" mode, the menu of **touch-probe functions** shown at the right will be displayed. Select the desired touch-probe function with the ↓ ↑ keys and press ENT.
In "Programming/editing" mode, the interactive dialogue for programming the touch-probe function "workpiece surface = datum" appears after the dialogue is initiated with TOUCH PROBE.


e.g. ▶ TOUCH PROBE

CALIBRATION EFFECTIVE LENGTH
CALIBRATION EFFECTIVE RADIUS
BASIC ROTATION
WORKPIECE SURFACE = DATUM
CORNER = DATUM
CIRCLE CENTRE = DATUM

Exiting touch-probe functions

You can exit the touch-probe functions at any time by pressing NO ENT. The control system will return to the previously selected operating mode.

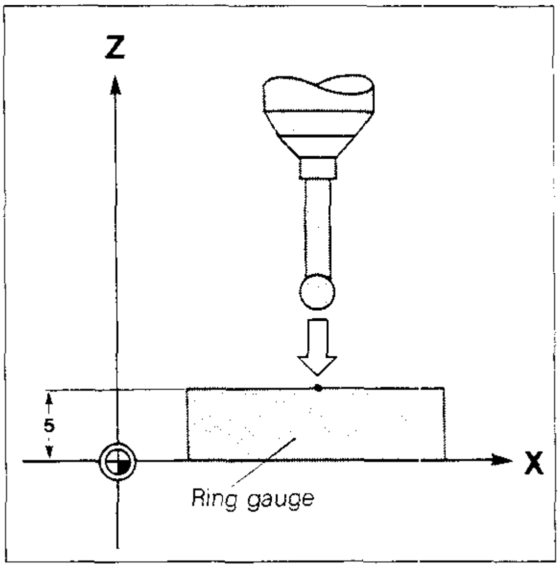
Error messages

If the probe cannot locate a probing point within the gauging distance defined by the machine parameters or if the probing point has already been reached when the touch-probe function is initiated, the following error message is displayed:
= TOUCH POINT INACCESSIBLE =.
If when starting the touch-probe function the touch point is already reached, then the error message is displayed:
= STYLUS EXTENDED =
When using touch-probe systems featuring **infrared transmission,** the transmitter/receiver window (the side with two windows) must be aligned with the evaluator electronics. If it is poorly aligned or if the transmission gap is obstructed (e.g. by the splash shield), the following error message is displayed:
= PROBE SYSTEM NOT READY =.
If the battery voltage in touch-probe system with infrared transmission drops below a specified value, this error message appears:
= EXCHANGE TOUCH PROBE BATTERY =

Touch-probe
Calibration effective length

troduction

The effective length of the stylus and the effective radius of the stylus tip can be determined with the aid of the TNC.
The control system automatically computes the necessary data via the touch-probe functions "CALIBRATION EFFECTIVE LENGTH" and "CALIBRATION EFFECTIVE RADIUS".
The length and radius data are saved and stored and taken into account when gauging the workpiece.
The compensation data can also be entered at any time from the control unit keyboard.



alibrating ids

A ring gauge of known height and internal radius is required for calibrating the effective radius of the stylus tip ball. The ring gauge is clamped to the machine table.

:ffective ength

When gauging the effective length of the stylus tip ball, the probe moves to a reference plane. After touching the surface, the probe is retracted in rapid traverse to its original position. The effective length is displayed when calibration is selected again.

Before calibrating the effective length of the stylus tip ball, set the reference plane with the zero tool.

Touch-probe
Calibrating effective length

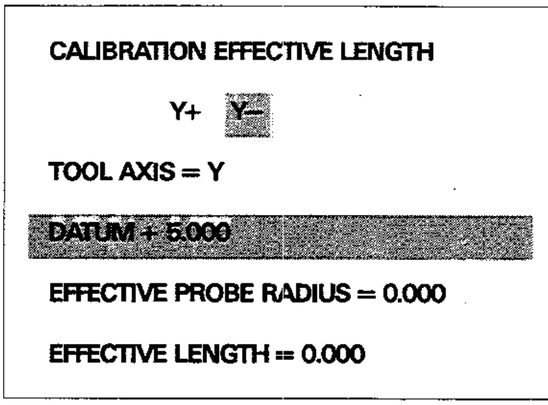
Input

Operating mode ____________ ✋ or ⊛

Dialogue initiation ____________ TOUCH PROBE ↑

CALIBRATION EFFECTIVE LENGTH	▶ ENT	Press ENT to select probe function.
CALIBRATION EFFECTIVE LENGTH Z+ Z– TOOL AXIS = Z DATUM + 0.000 EFFECTIVE PROBE RADIUS := 0.000 EFFECTIVE LENGTH = 0.000	▶ Y	Specify tool axis if required.
CALIBRATION EFFECTIVE LENGTH Y+ Y– TOOL AXIS = Y DATUM + 0.000 EFFECTIVE PROBE RADIUS := 0.000 EFFECTIVE LENGTH = 0.000	▶ X Y Z ▼ ↓ ▼ □ ▼ ENT	Move probe system to vicinity of reference plane. Enter datum if required: Select "Datum". Enter datum in tool axis, e.g. + 5.0 mm. Press ENT.
CALIBRATION EFFECTIVE LENGTH Y+ Y– TOOL AXIS = Y DATUM + 5.000 EFFECTIVE PROBE RADIUS := 0.000 EFFECTIVE LENGTH = 0.000	▶ ← →	Select traverse direction of probe if required, here Y–.

Touch-probe
Calibrating effective length



CALIBRATION EFFECTIVE LENGTH

Y+ Y–

TOOL AXIS = Y

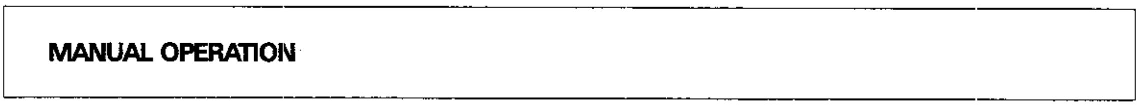
DATUM + 5.000

EFFECTIVE PROBE RADIUS = 0.000

EFFECTIVE LENGTH = 0.000

▶ START Move probe in negative Y direction.

After contacting the surface, the touch probe returns in rapid traverse to its original position.



MANUAL OPERATION

The TNC switches automatically to the display "Manual operation" or "Electronic handwheel".

The gauged length is displayed when calibration is selected again.

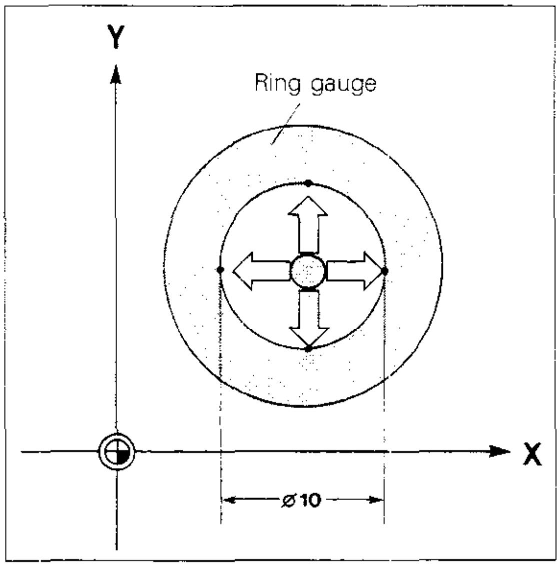
Notes:

Touch-probe
Calibrating effective radius

Effective radius

The probe must be located within the bore of the ring gauge. The effective radius of the stylus tip ball is determined by probing four points on the bore. The directions of traverse are specified by the control system, e.g. X+, X–, Y+, Y– (tool axis = Z).
After contacting each point, the probe moves in rapid traverse back to its original position; the TNC displays the coordinates of the contact points.

The effective radius is displayed when calibration is selected again.



Touch-probe
Calibrating effective radius

Input

Operating mode ______ or

Dialogue initiation ______ TOUCH PROBE ↑

CALIBRATION EFFECTIVE RADIUS	ENT	Select touch-probe function.
CALIBRATION EFFECTIVE RADIUS **X+ X– Y+ Y–** **TOOL AXIS = Z** **RADIUS RING GAUGE = 0.000** **EFFECTIVE PROBE RADIUS = 0.000** **EFFECTIVE LENGTH = 8.455**	↓ □ ENT	Select "Radius ring gauge". Enter radius of ring gauge, e.g. 10.0 mm. Press ENT. Enter another tool axis if required (see "Effective length").
CALIBRATION EFFECTIVE RADIUS **X+ X– Y+ Y–** **TOOL AXIS = Z** **RADIUS RING GAUGE = 10.000** **EFFECTIVE PROBE RADIUS = 0.000** **EFFECTIVE LENGTH = 8.455**	X Y Z ← →	Move to approximate center of ring gauge. Select probe traverse direction, e.g. X+.
CALIBRATION EFFECTIVE RADIUS **X+ X– Y+ Y–** **TOOL AXIS = Z** **RADIUS RING GAUGE = 10.000** **EFFECTIVE PROBE RADIUS = 0.000** **EFFECTIVE LENGTH = 8.455**	START	Move probe in positive X-direction.

Touch-probe
Calibrating effective radius

After probing the ring gauge, the probe returns in rapid traverse to its original position.

CALIBRATION EFFECTIVE RADIUS

X+ X– Y+ Y–

X (probe point) Y (probe point)

Z (probe point) C (probe point)

▶ ← → Select next probe traverse direction, e.g. X–.

CALIBRATION EFFECTIVE RADIUS

X+ X– Y+ Y–

X (probe point) Y (probe point)

Z (probe point) C (probe point)

▶ START Move probe in negative X-direction.

After probing the ring gauge, the probe returns in rapid traverse to its original position.

The TNC displays the actual values of the second probe point below the values of the first contact point.

Then probe the ring gauge in the positive and negative Y-directions.

When the procedure is completed:

MANUAL OPERATION

The TNC switches automatically to "Manual operation" or "Electronic handwheel".

The radius of the measured probe tip is displayed on the appropriate line when calibration is selected again.

Notes:

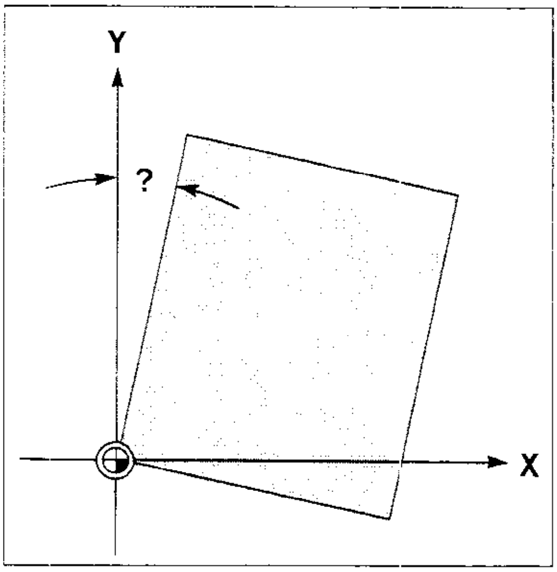
Touch-probe
Basic rotation

Description

The touch-probe function "Basic rotation" can be used to determine the amount of angular misalignment of a clamped workpiece.

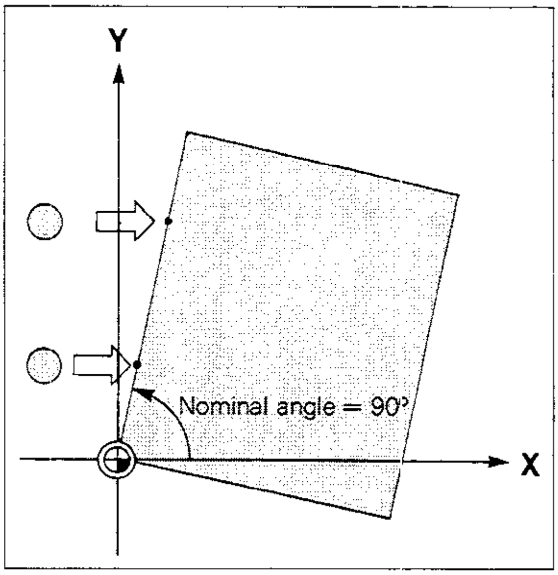
The TNC compensates for the angular deviation by means of a basic rotation of the coordinate system.

The **basic rotation** must be carried out **in advance** if you want to set the datum using the functions
= CORNER = DATUM = or
= CIRCLE CENTRE = DATUM =



Procedure

The touch-probe moves to the side face of the workpiece from two different starting positions. The directions of traverse are specified e.g. X+, X–, Y+, Y– (tool axis = Z).
After contact with the side faces, the probe returns in rapid traverse to the respective original position.
The TNC saves the coordinates of the contact points and uses them to compute the angular deviation. In order to compensate for the deviation, the control system must know the "nominal angle" of the side face.
Enter the nominal angle on the line after
= ROTATION ANGLE =.



Touch-probe
Basic rotation

Input

Operating mode ______ or

Dialogue initiation ______ TOUCH PROBE ↑

BASIC ROTATION	▶ ENT	Press ENT to select touch-probe function.

▼

BASIC ROTATION **X+ X– Y+ Y–** **ROTATION ANGLE = 0.000**	▶ □ ▼ ENT	Specify angular position of probed side face, e.g. Y-axis: + 90°. Press ENT.

▼

BASIC ROTATION **X+ X– Y+ Y–** **ROTATION ANGLE = + 90.000**	▶ X Y Z ▼ ← →	Move to first starting position. Select direction of traverse, e.g. X+.

▼

BASIC ROTATION **X+ X– Y+ Y–** **ROTATION ANGLE = + 90.000**	▶ START	Move probe in positive X-direction.

▼

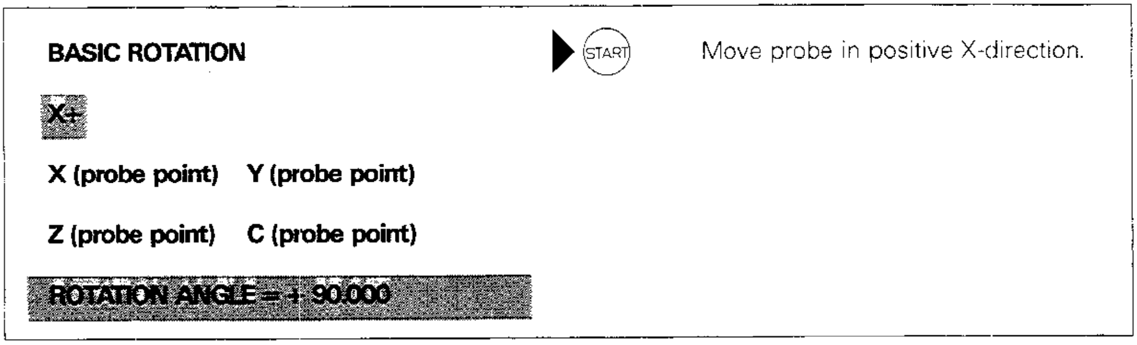
After touching the side face, the probe returns in rapid traverse to the first starting position.

▼

BASIC ROTATION **X+** **X (probe point) Y (probe point)** **Z (probe point) C (probe point)** **ROTATION ANGLE = + 90.000**	▶ X Y	Move to second starting position.

▼

Touch-probe
Basic rotation



BASIC ROTATION

X+

X (probe point) **Y (probe point)**

Z (probe point) **C (probe point)**

ROTATION ANGLE = + 90.000

▶ START Move probe in positive X-direction.

After touching the side face, the probe returns in rapid traverse to the second starting position.

MANUAL OPERATION

The TNC switches automatically to the previously selected operating mode "Manual operation" or "Electronic handwheel".

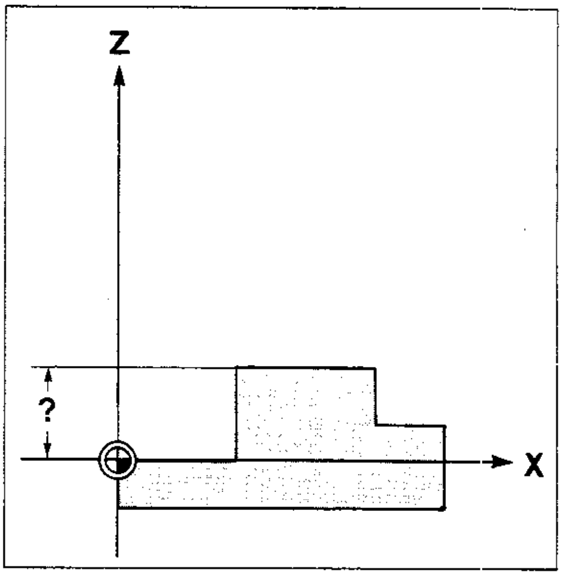
The measured angle of rotation is displayed when "Basic rotation" is selected again.

ROT appears highlighted in the on-screen status display if "Basic rotation" has been programmed and remains displayed as long as a basic rotation is stored in memory. A "Basic rotation" is not cancelled by turning off the power supply. To cancel the command, select the touch-probe function "Basic rotation" and enter the angle of rotation 0° from the keyboard.

Touch-probe
Workpiece surface = datum

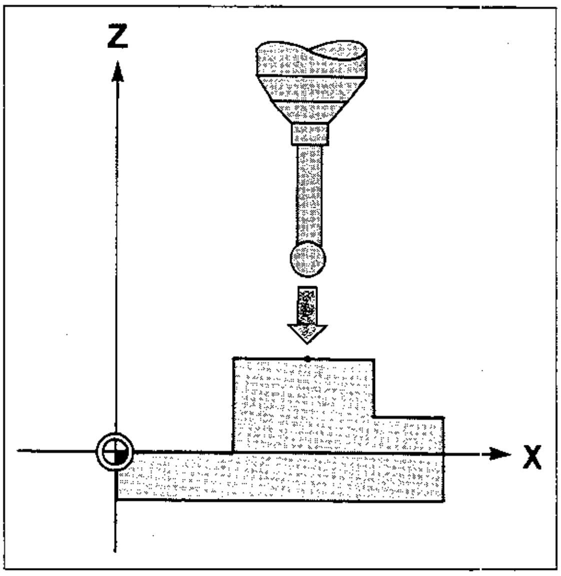
Description

In the case of workpieces clamped paraxially to the table, you can use the touch-probe function "Surface = datum" to define the workpiece surface or side face on any axis as datum. The TNC then bases all nominal position values for subsequent machining on that surface.



Procedure

The probe moves to the surface of the workpiece. After contact with the surface, the probe is retracted in rapid traverse to its original position. The TNC saves the coordinates of the contact point on the traversed axis and displays the value on the line "DATUM".
Any desired value can be assigned to the contact point by entering it from the keyboard.



Touch-probe
Workpiece surface = datum

Input

Operating mode ____________________ or

Dialogue initiation ____________________ TOUCH PROBE

SURFACE = DATUM	▶ ENT	Press ENT to select probe function.

▼

SURFACE = DATUM	▶ X Y Z	Move to starting position.
X+ X– Y+ Y– Z+ Z– C+ C–	▼ ← →	Select direction of traverse, e.g. Z–.

▼

SURFACE = DATUM	▶ START	Move probe in negative Z-direction.
X+ X– Y+ Y– Z+ Z– C+ C–		

▼

After contacting the surface, the touch probe returns in rapid traverse to its original position.

▼

X (probe point) Y (probe point)		
Z (probe point) C (probe point)		
DATUM Z – 18.125		Enter any desired datum if required.
	▶ □ ▼ ENT	Press ENT.

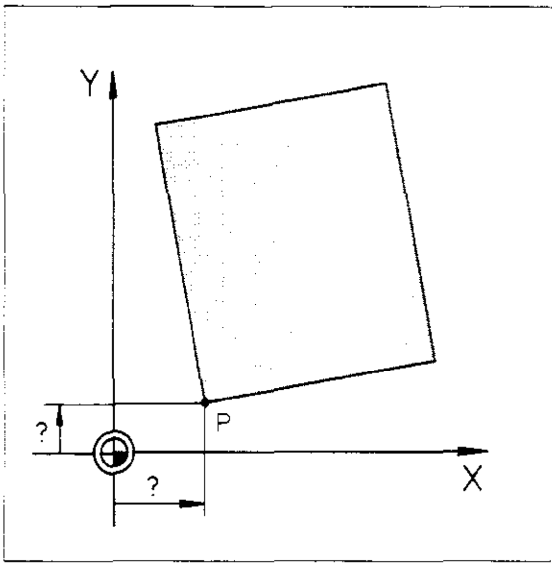
Notes:

Touch-probe
Corner = datum

Description

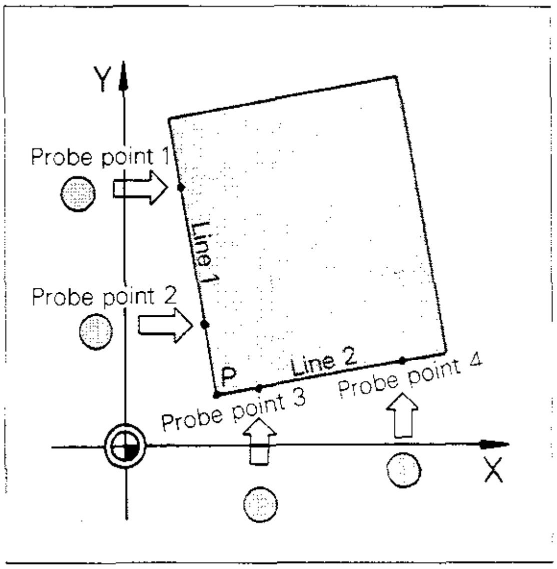
With the touch-probe function "Corner = datum", the TNC computes the coordinates of a corner point of the clamped workpiece. The computed value can be used as the reference point for the subsequent machining procedure: all nominal position values wil' be based on this point.

The function
= BASIC ROTATION =
must be carried out before
= CORNER = DATUM =

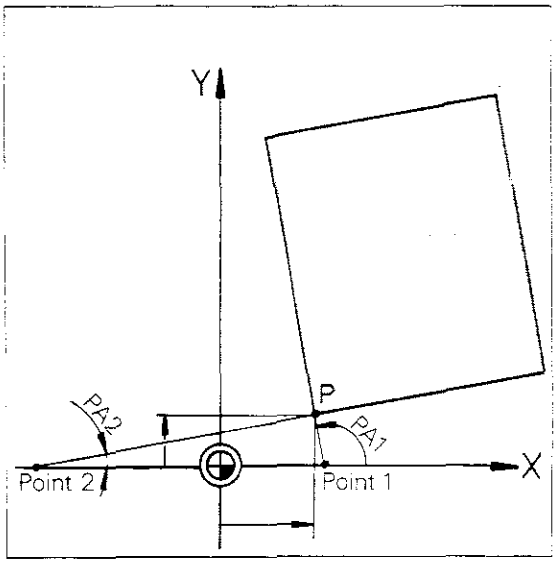


Procedure

The touch-probe moves to two side faces of the workpiece from two different starting positions per face. The directions of traverse are specified: X+, X–, Y+, Y– (tool axis = Z).
After contact with the surface, the probe is retracted in rapid traverse to its original position. The TNC saves the coordinates of the contact points and uses them to calculate two straight lines. The missing corner point is the intersection of these lines.



The screen displays the coordinates of the corner point. The computed lines are displayed beneath them by a point on each line and the corresponding angle PA.
You can enter any desired datum from the input keyboard, instead of the calculated corner point. If "Basic rotation" was defined before the touch-probe function "Corner = datum", the straight line computed for "Basic rotation" may be used for the touch-probe function "Corner = datum" as well.



Touch-probe
Corner = datum

Input

Operating mode ______________ ✋ or ⊗

Dialogue initiation ______________ TOUCH PROBE ↓

CORNER = DATUM	▶ ENT	Press ENT to select probe function.

CORNER = DATUM **X+ X– Y+ Y–**	▶ Ⓧ Ⓨ Ⓩ ▼ ← →	Move to first starting position. Select direction of traverse, e.g. X+.

CORNER = DATUM **X+ X– Y+ Y–**	▶ START	Move probe in positive X-direction.

After contacting the side surface, the touch probe returns in rapid traverse to its original position.

CORNER = DATUM **X+** **X (probe point 1) Y (probe point 1)** **Z (probe point 1) C (probe point 1)**	▶ Ⓧ Ⓨ	Move probe to next starting position.

CORNER = DATUM **X+** **X (probe point 1) Y (probe point 1)** **Z (probe point 1) C (probe point 1)**	▶ START	Move probe in positive X-direction.

After contacting the side surface, the touch probe returns in rapid traverse to its original position.

The control system displays the actual values of the second probe point beneath the values of the first point. The first line is also indicated by a random point on the line and the angle of direction.

Touch probe
Corner = datum

The second side face is then probed from two different starting positions.

When this procedure is complete:

CORNER = DATUM

X (corner point) Y (corner point)

X (line 1) Y (line 1)

PA (angle of line 1)

X (line 2) Y (line 2)

PA (angle of line 2)

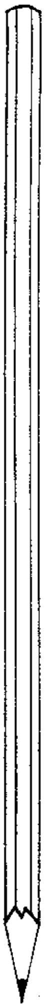
DATUM X (corner point)

DATUM Y (corner point)

□ ↓ □ Specify any corner point coordinates for X and Y if required.

ENT Press ENT.

Notes:



Touch-probe
Corner = datum

Input immediately following "Basic rotation"

Operating mode ______________________ ✋ or ⎈

Dialogue initiation ______________________ TOUCH PROBE ↓

CORNER = DATUM	▶ ENT	Press ENT to select probe function.

▼

CORNER = DATUM

TOUCH POINTS OF BASIC ROTATION ?

X (line 1) **Y (line 1)**

PA (angle of line 1)

To transfer probe points used for basic rotation:	▶ ENT	Press ENT.
If you do not wish to transfer probes points used for basic rotation:	▶ NO ENT	Press NO ENT.

▼

Then probe the second side face as described above.

▼

CORNER = DATUM **X+ X– Y+ Y–**	▶	. . . .

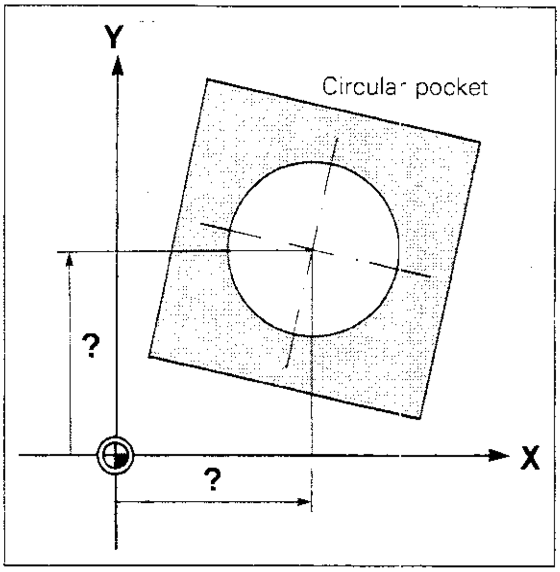
Notes:

Touch-probe
Circle centre = datum

Description

In the case of clamped workpieces with cylindrical features (bore, circular pocket or external cylinder), the touch-probe function "Circle centre = datum" can be used to determine the coordinates of the circle centre.
The calculated circle centre can be used as the datum for the subsequent machining procedure. All nominal position values will be based on this point.

> The function
> = BASIC ROTATION =
> must be carried out before
> = CIRCLE CENTRE = DATUM =.



Procedure

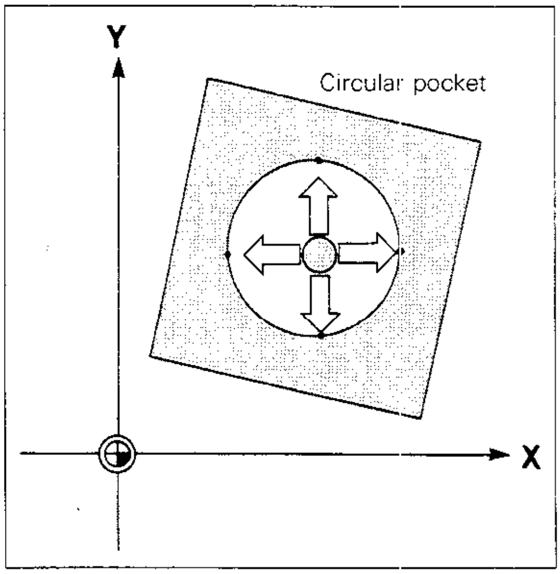
In the case of bores and circular pockets, the probe must be located within the bore or pocket.

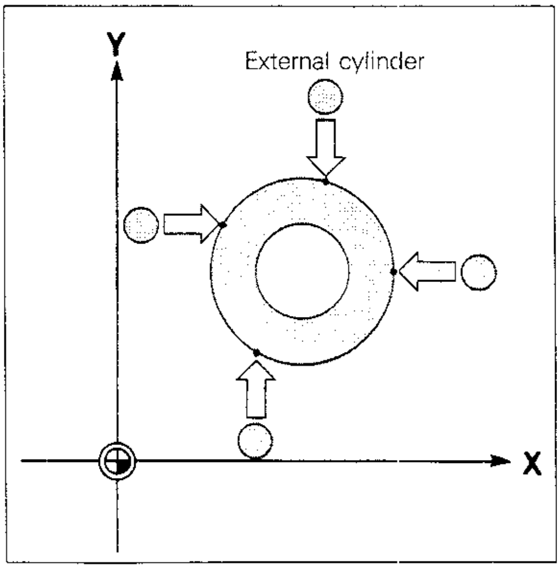
To determine the circle centre, probe four points of the external cylinder or bore. The directions of traverse are specified, e.g. X+, X–, Y+, Y–(tool axis = Z).

After each contact, the probe is retracted in rapid traverse to its original position. The TNC saves the coordinates of all computed contact points and uses them to calculate the circle centre.

The coordinates of the circle centre are displayed on the screen with the specified radius PR.

You can enter any desired values from the input keyboard, instead of the calculated circle centre coordinates.





Touch-probe
Circle centre = datum

Input

Operating mode ______________________ ✋ or ⊛

Dialogue initiation ______________________ TOUCH PROBE ↓

CIRCLE CENTRE = DATUM	▶ ENT	Press ENT to select probe function.

▼

CIRCLE CENTRE = DATUM	▶ X Y Z	Move to first starting position.
X+ X– Y+ Y–	▼ ← →	Select direction of traverse, e.g. X+.

▼

CIRCLE CENTRE = DATUM	▶ START	Move probe in positive X-direction.
X+ X– Y+ Y–		

▼

After touching the cylindrical surface, the probe is retracted in rapid traverse to its starting position.

▼

CIRCLE CENTRE = DATUM	▶ ← →	Select next direction of traverse, e.g. X–.
X+ X– Y+ Y–		
X (probe point 1) Y (probe point 1)		
Z (probe point 1) C (probe point 1)		

▼

CIRCLE CENTRE = DATUM	▶ START	Move probe in negative X-direction.
X+ X– Y+ Y–		
X (probe point 1) Y (probe point 1)		
Z (probe point 1) C (probe point 1)		

▼

After touching the cylindrical surface, the probe is retracted in rapid traverse to its starting position.

The TNC displays the actual values of probe point 2.

▼

Touch-probe
Circle centre = datum

Then probe two additional points on the cylindrical surface, in positive and negative Y-direction.

When this procedure is complete:

CIRCLE CENTRE = DATUM

X (midpoint) Y (midpoint)

PR (circle radius)

DATUM X (midpoint) ▶ □ — Specify any circle centre coordinates for X and Y if required.

DATUM Y (midpoint)

↓

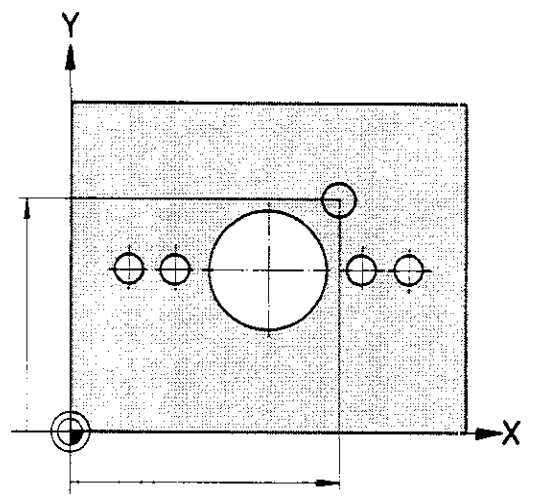
□

ENT — Press ENT.

Touch-probe
Programmable touch-probe function: "Surface = datum"

Description

You can probe a surface of a workpiece with program control, both before and while machining the part. In the case of castings with varying elevations, for example, the TNC can probe the surface before machining, allowing the correct depth to be reached during the subsequent machining procedure. In the same way, changes in position caused by a rise in machine or workpiece temperatures can be monitored and compensated for.



Coordinates of probe point

Programming

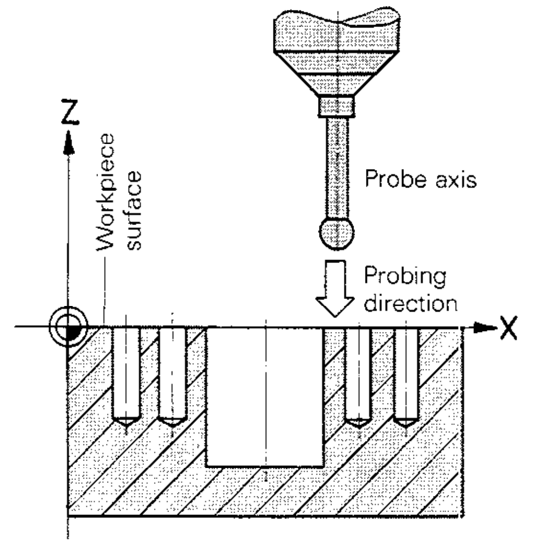
Initiate programming with the TOUCH PROBE key.

The TNC will then prompt you for the parameter number at which the results of the measurement will be saved. After entering the probe axis and direction, specify the nominal position for the touch-probe cycle. The programmed touch-probe cycle requires two program blocks.

Procedure

Travelling at rapid rate, the probe moves to the advanced stop distance above the programmed nominal position (probe point). The advanced stop position is determined by the machine manufacturer via a machine parameter.
The probe then moves to the workpiece, on the probe axis and in the probing direction, travelling at the feed rate specified for measuring and touches the surface. After contact, the probe is retracted in rapid traverse to its original position.

To compensate for deviations in the position of the workpiece surface, the datum must be shifted on the probe axis, using the "Datum shift" cycle, by the amount of the value saved under Q. The gauged value can also be used in a tool definition as a length compensation factor, for example.



Touch-probe
Programmable touch-probe function: "Surface = datum"

Input

Operating mode ________ ⇨

Dialogue initiation ________ TOUCH PROBE

PARAMETER NUMBER FOR RESULT	▶ □	Specify parameter number.
	▼ ENT	Press ENT.

▼

PROBE AXIS/PROBING DIRECTION ?	▶ Z	Specify probe axis, e. g. Z.
	▼ +/−	Specify probing direction.
	▼ ENT	Press ENT.

▼

POSITION VALUE ?	▶ X	Specify coordinates of probe point: select axis, e. g. X.
	▼ I	Incremental – absolute?
	▼ □	Enter numerical value.
	▼ Y	Select next axis, e. g. Y.
	⋮	
After all coordinates have been entered:	▶ ENT	Press ENT.

Sample display

```
32 TCH PROBE 0.0 REF. PLANE
                   Q10 Z–
33 TCH PROBE 0.1 X + 10.000
          Y + 20.000 Z + 0.000
```

The X, Y plane is probed in the negative Z-direction. The gauged value is saved at parameter Q10. The coordinates of the nomina probe point are X 10.000/Y 20.000/Z 0.000.

Notes:

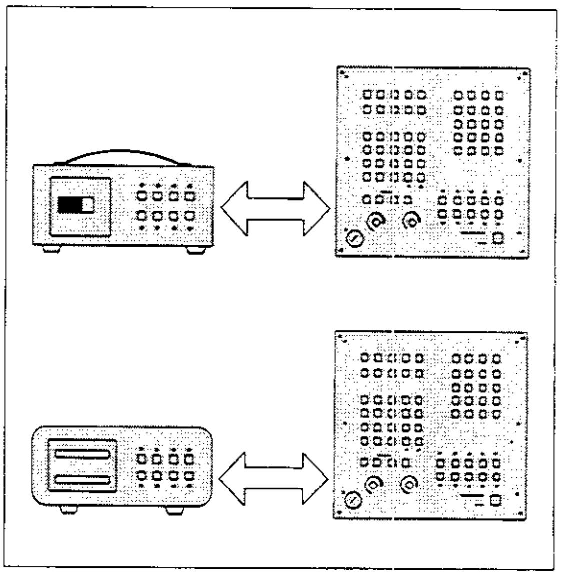
External data transfer
The TNC data interface

V.24/RS-232-C interface

The TNC control system is equipped with a **V.24 (RS-232-C) data interface** for input and output of programs in plain-language or ISO formats. This means that you can use the interface to transfer programs from the TNC's memory to an **external storage unit,** e.g. a magnetic tape unit or floppy disk unit, or to some other **peripheral device,** such as a printer. You can also transfer data from an external storage unit to the control unit.

The interface port is located at the rear of the control unit.

The interface operating mode (ME magnetic tape, FE floppy disk or operation with other external devices) must be specified in advance.



Operating mode

The TNC's V.24 interface can be switched to three different **interface operating modes:**

ME mode: for connecting a HEIDENHAIN ME magnetic tape unit or a HEIDENHAIN FE floppy disk unit. Commands are entered from the keypad of the external unit.
FE mode: for connecting a HEIDENHAIN FE floppy disk unit. Commands are entered via TNC menu.
EXT mode: for connecting other peripheral equipment.

The interface operating mode is defined via the supplementary operating mode (MOD) **V.24 INTERFACE** (see "Interface definition").

Baud rate

The **data transmission speed** (baud rate) at the TNC interface depends on the interface operating mode:

ME-mode: 2400 baud
FE-mode: 9600 baud
EXT-mode: 2400 baud; the baud rate can be set to one of the values shown in the table at the right via the supplementary operating mode (MOD) **BAUD RATE** (see "Interface definition").

Operating mode: EXT

Possible baud rates:
110 baud
150 baud
300 baud
600 baud
1 200 baud
2 400 baud
4 800 baud
9 600 baud

1 baud = 1 bit per sec

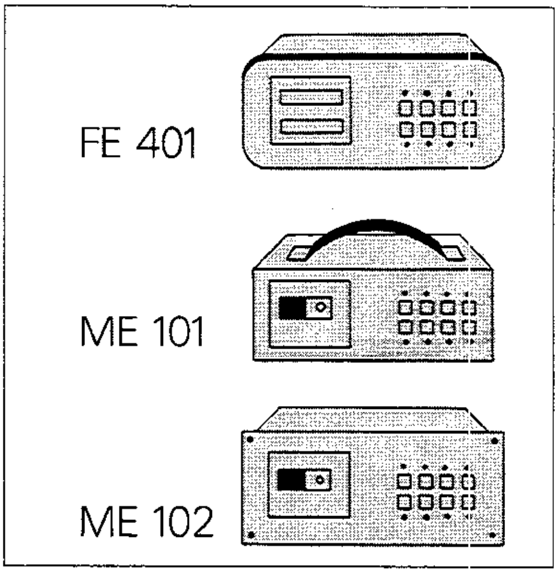
Transfer blockwise

The TNC 151/TNC 155 can load machining programs in plain-language or ISO format from an external programming station or floppy disk unit via the V.24 data interface and simultaneously execute these programs (see "Transfer blockwise").

External data transfer
Floppy disk unit/Magnetic tape unit

Disk and magnetic tape units

HEIDENHAIN offers a floppy disk unit and two magnetic tape units for saving and storing machining programs or transferring programs that have been created at an external programming station.
FE 401: Portable floppy disk unit for use with multiple machines.
ME 101: Portable magnetic tape unit for use with multiple machines.
ME 102: Magnetic tape unit for permanent installation at the machine.

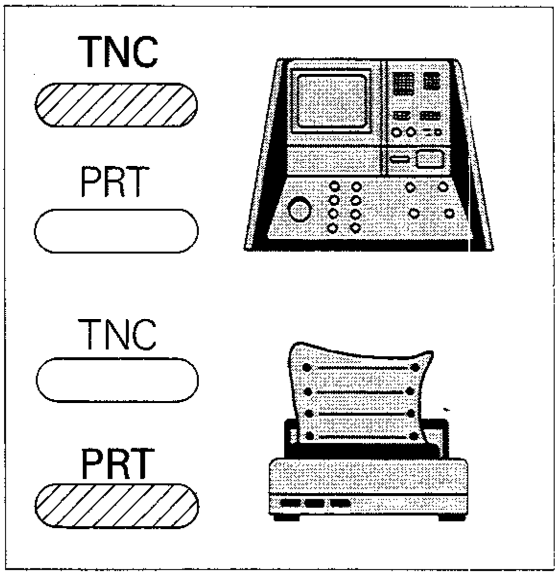


Connection options

Each of the external storage units is equipped with two V.24 data interfaces identified by **TNC** and **PRT**.

TNC port: for connection to the control unit.
PRT port: for connection to a peripheral device.

These ports make it possible to connect a second device to the external storage unit, in addition to the TNC.



Operating modes

The **FE 401** can transfer data either in ME mode or in FE mode. The mode can be defined via a switch located on the unit.
The **ME 101/ME 102** can transfer data in ME mode only.

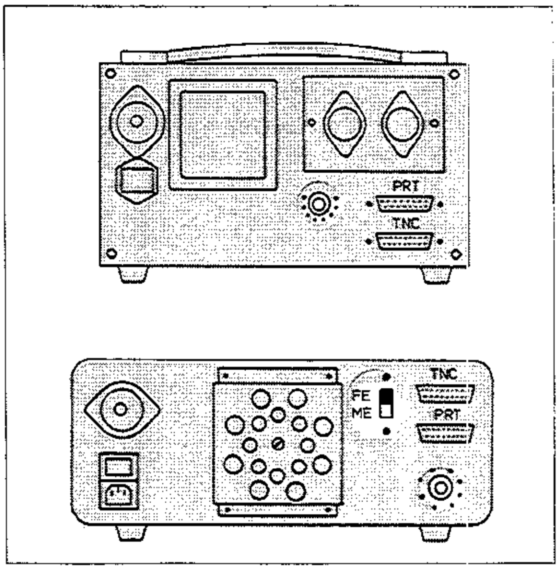
Baud rate

The baud rate at the **TNC port** is defined as follows:

ME mode:	2400 baud
FE mode:	9600 baud

The baud rate at the **PRT port** can be adjusted with the aid of a switch located at the rear of the external unit.

ME 101/ME 102:	110/150/300/600/1200/2400 bd
FE 401:	110/150/300/600/1200/2400/ 4800/9600 bd.



External data transfer
Interface definition

V.24 interface definition

Operating mode ______________ □ optional except →

Dialogue initiation ______________ MOD

VACANT BLOCKS = 1112	▶ ↓	Page through supplementary mode menu until V.24 INTERFACE appears.

▼

V.24 INTERFACE = ME		
To define for ME mode:	DEL	Press DEL to confirm ME mode.
To select FE interface or operation with other external unit:	▶ ENT	Page until FE or EXT appears.
	▼ DEL	Press DEL to confirm and exit supplementary mode.

The V.24 interface can be defined via machine parameters for operation with other external devices.
For further information, see "TNC 151/TNC 155 Mounting and interface description".

Baud rate definition for EXT

Operating mode ______________ □ optional except →

Dialogue initiation ______________ MOD

VACANT BLOCKS = 1112	▶ ↓	Page through supplementary mode menu until BAUD RATE appears.

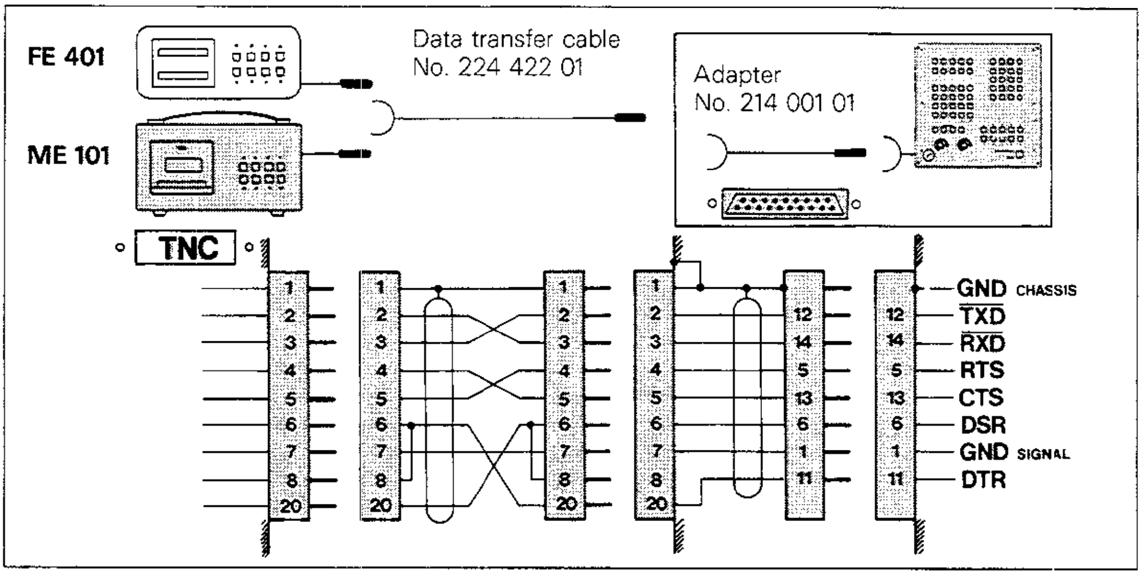
▼

BAUD RATE = 2400	▶ □	Enter desired baud rate from table.
	▼ ENT	Press ENT.

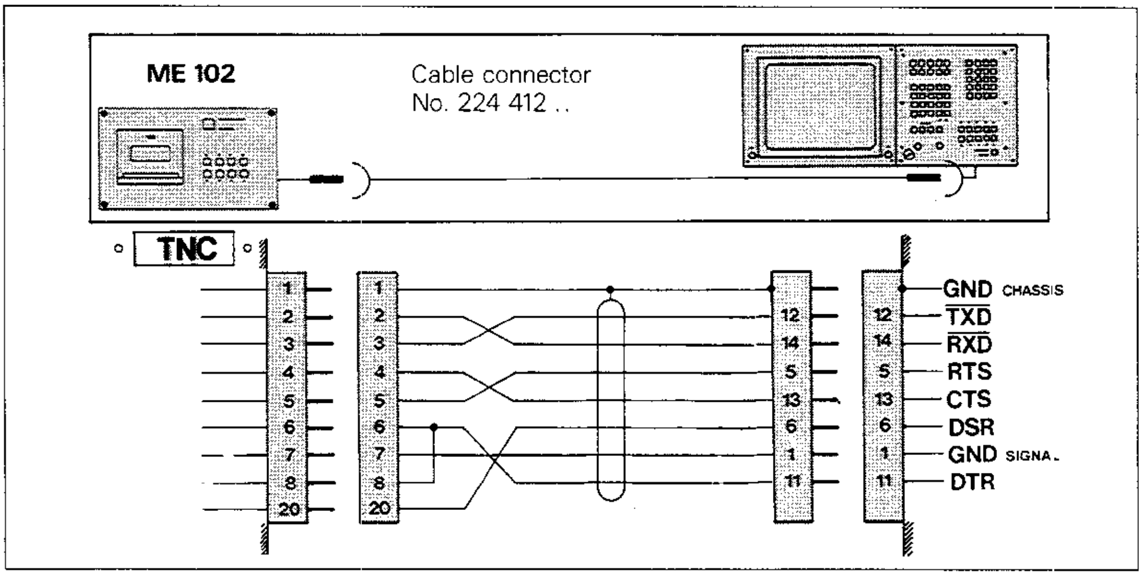
You can also save the new baud rate by pressing MOD or using the ↓ ↑ keys.

External data transfer
Cables and connector pin assignment

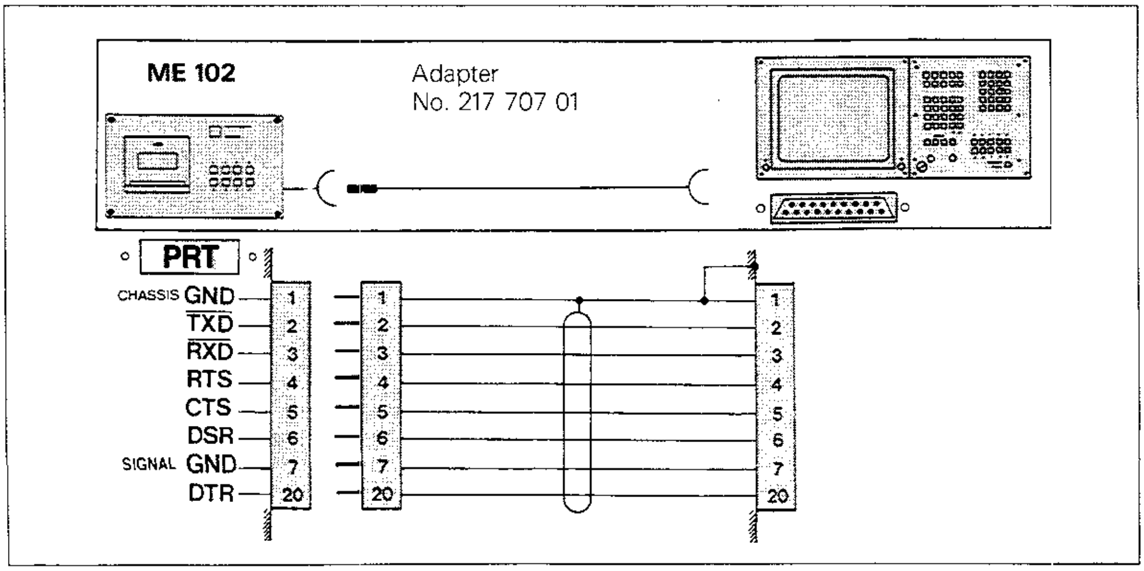
**ME 101 magnetic tape unit/
FE 401 disk unit
↔ TNC**



**ME 102 magnetic tape unit
↔ TNC**

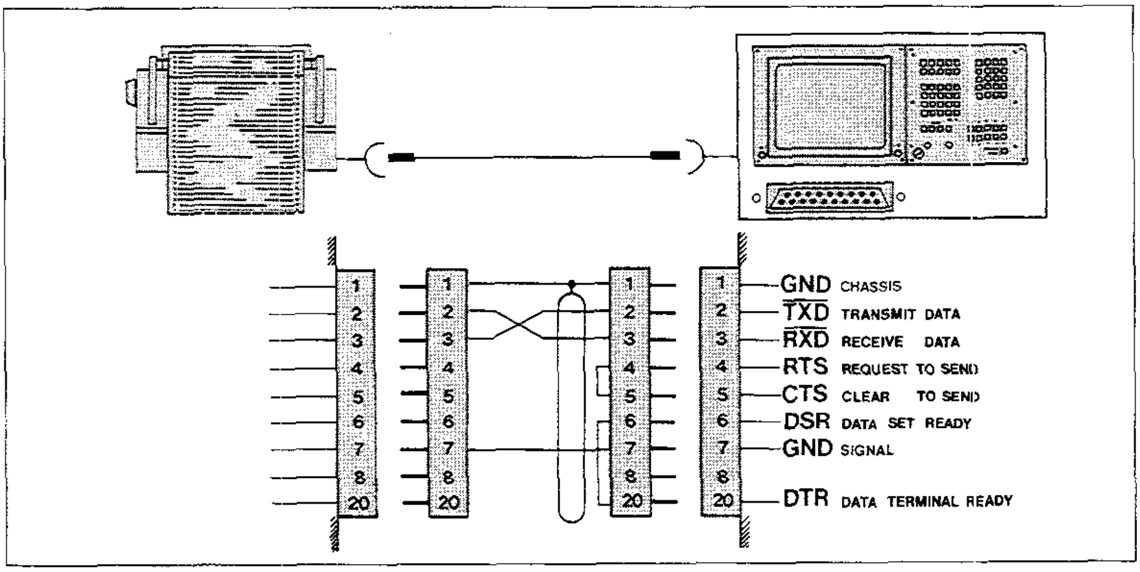


**ME 102 magnetic tape unit
↔ PRT**



External data transfer
Cables and connector pin assignment

Magnetic tape unit/floppy disk unit/TNC ↔ peripheral device



External data transfer
General information

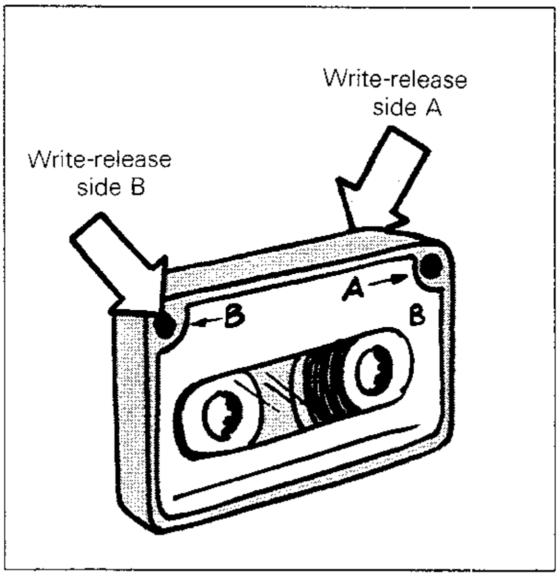
Data media

The ME 101/ME 102 **magnetic tape units** use **minicassettes** for data storage. They can store up to 32 different programs with a total of 1,000 program blocks (approx. 35 kilobytes) per tape side.
The FE 401 **floppy disk unit** uses 3.5" **disks** (double-sided, 135 TPI), with a storage capacity of maximum 256 different programs with a total of 25,000 program blocks (approx. 790 kilobytes). The FE 401 is equipped with two disk drives. Simultaneous disk access via the "TNC" and "PRT" interfaces is possible, e.g. for running a program and printing out hardcopy on a printer at the same time. The second disk drive is designed for data back-up (disk copy).

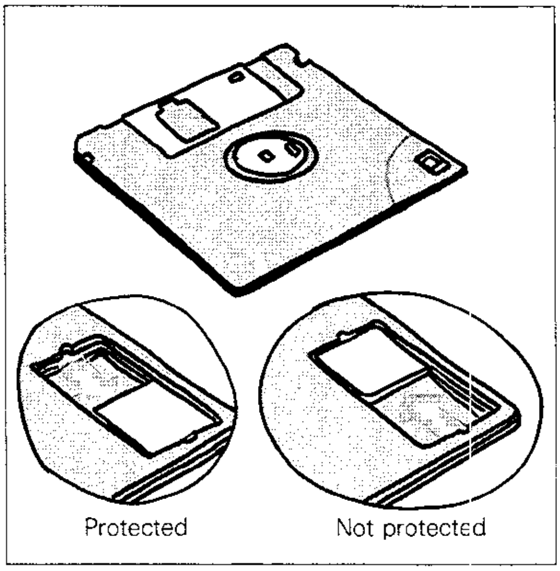
Write protection

The minicassettes and disks can be safeguarded against accidental erasure or write-over.

The **write-release tabs** must be inserted in the magnetic tape cassette for transferring data.



The small **sliding tab** on the reverse side of the disk must cover the opening at the corner of the disk for transferring data.



External data transfer
Procedure for ME, FE and EXT operation

Data transfer

Data can be transferred between the TNC and an external unit in ⟐ PROGRAMMING/EDITING mode. In addition, you can transfer a program to the TNC and run it simultaneously in ⇥ PROGRAM RUN mode (see "Transfer blockwise"). The TNC interface must be adapted to the external unit (ME, FE or other peripheral, e.g. printer) with respect to operating mode.

ME mode

In **ME mode,** commands are entered from the keypad of the magnetic tape unit or disk unit (switch in ME position) and via the TNC menu (see illustration).

FE mode

In **FE mode,** commands are entered only from the TNC menu. You do not need to press any keys on the FE unit.

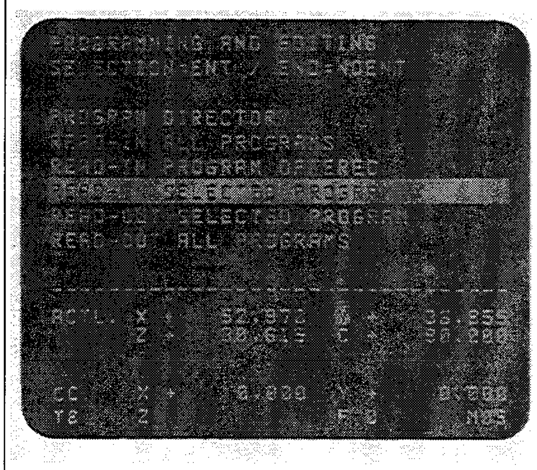
EXT mode

For information on entering commands in **EXT mode,** please refer to the manufacturer's instructions for the external unit in question.

Desired data transfer option	Press ME or FE keys (red lamp must light)
TNC => ME	TNC →8
ME => TNC	←8 TNC
TNC => FE	TNC ⊡↑
FE => TNC	⊡↓ TNC

Dialogue initiation

The dialogue for transferring data in any direction (tape/disk => TNC or TNC => tape/disk) is initiated by pressing EXT. The transfer mode options shown at the right are displayed on the screen. Use the ↓ ↑ keys to move the highlighted pointer to the desired mode and press ENT to select and start the operating mode. To exit the menu, press NO ENT.



```
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

ACTL. X +    52.972  B + [illegible]
      Z +    33.615  C + [illegible]

CC    X +     0.000  Y +     0.000
T2    Z              F 0       M05
```

Interrupting data transfer

Once data transfer has begun, it can be interrupted by pressing DEL □ on the TNC or STOP on the ME/FE unit. If data transfer is interrupted, the error message

= ME: PROGRAM INCOMPLETE =

appears. After this message is cleared with the CE key, the menu of operating mode options for data transfer is displayed.

External data transfer
External data storage unit → TNC

Program directory

Operating mode ______ ⇨

Dialogue initiation ______ EXT ↑

PROGRAM DIRECTORY ▶ ENT Press ENT to select mode.

EXTERNAL DATA INPUT

Magnetic tape/disk starts.

END = NOENT

10 600

All programs stored on the tape or disk are displayed but not transferred.

To exit the operating mode: ▶ NO ENT Press NO ENT to exit mode.

PROGRAMMING AND EDITING

The TNC is now in PROGRAMMING AND EDITING mode.

External data transfer
External data storage unit → TNC

Read-in all programs

Operating mode ______________ ⇒

Dialogue initiation ______________ EXT ↓

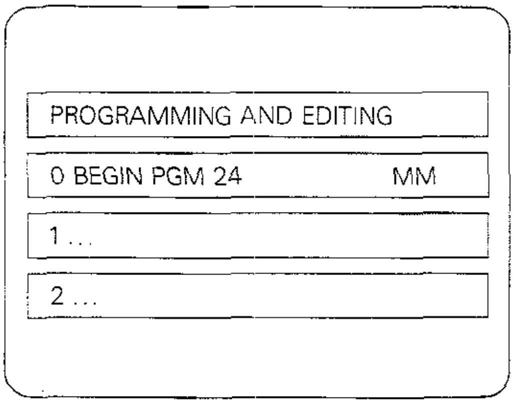
READ-IN ALL PROGRAMS	▶ ENT	Press ENT to select mode.

▼

EXTERNAL DATA INPUT

Magnetic tape/disk starts.

▼



PROGRAMMING AND EDITING

0 BEGIN PGM 24 MM

1 ...

2 ...

All programs stored on the tape/disk are now in the TNC's memory. The program with the highest number is displayed.

External data transfer
External data storage unit → TNC

Read-in program offered

Operating mode ______________________ ◇

Dialogue initiation ______________________ EXT ↑

READ-IN PROGRAM OFFERED	▶ ENT	Press ENT to select mode.

▼

EXTERNAL DATA INPUT

Magnetic tape/disk starts.

▼

ENTRY = ENT/OVERREAD = NOENT

22

To transfer offered program:	▶ ENT	Press ENT to transfer program.
To **skip** offered program:	▶ NO ENT	Press NO ENT to skip to next program.

▼

ENTRY = ENT/OVERREAD = NOENT

24

The TNC displays all programs stored on the tape or disk, one after another.
After displaying the program with the highest number, the TNC automatically returns to PROGRAMMING AND EDITING mode.

External data transfer
External data storage unit → TNC

Read-in selected program

Operating mode ______ ◇

Dialogue initiation ______ EXT ↑

READ-IN SELECTED PROGRAM ▶ ENT Press ENT to select mode.

▼

PROGRAM NUMBER = ▶ □ Specify desired program number.

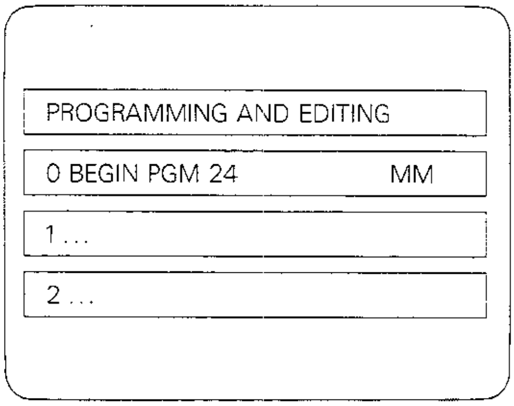
▼ ENT Press ENT.

▼

EXTERNAL DATA INPUT

Magnetic tape/disk starts.

▼



```
PROGRAMMING AND EDITING
0 BEGIN PGM 24          MM
1 ...
2 ...
```

The selected program is in the TNC's memory and is displayed.

External data transfer
TNC → external data storage unit

Read-out selected program

Operating mode ______ ◇→

Dialogue initiation ______ EXT ↓

READ-OUT SELECTED PROGRAM ▶ ENT — Press ENT to select mode.

▼

EXTERNAL DATA OUTPUT

Magnetic tape/disk starts, then stops after leader output.

▼

OUTPUT = ENT/END = NOENT ▶ ↓ ↑ ← → — Move cursor to desired program number.

1 13

14 24

▼ ENT — Press ENT to transfer selected program to tape/disk.

▼

EXTERNAL DATA OUTPUT

Magnetic tape/disk starts, then stops after program transfer is complete.

▼

OUTPUT = ENT/END = NOENT

1 **13**

14 24

Cursor positioned at next program number.

To exit operating mode: ▶ NO ENT — Press NO ENT to exit mode.

▼

PROGRAMMING AND EDITING

The TNC is now in PROGRAMMING AND EDITING mode.

External data transfer
TNC → external data storage unit

Read-out all programs

Operating mode ________ ⇨

Dialogue initiation ________ EXT ↓

READ-OUT ALL PROGRAMS ▶ ENT Press ENT to select mode.

▼

EXTERNAL DATA OUTPUT

Magnetic tape/disk starts and data transfer begins.

After data transfer is complete, the TNC returns to PROGRAMMING AND EDITING mode.

External data transfer
Transfer blockwise

Program run from external storage unit

In "Transfer blockwise" mode, machining programs can be transferred via the V.24 (RS-232-C) serial interface from an external storage unit or the FE unit and executed simultaneously. This makes it possible to run machining programs that exceed the TNC's RAM memory capacity.

Data interface

The data interface can be programmed via machine parameters. Please refer to the "TNC 151/TNC 155 Mounting Instructions and Interface Description" for a detailed description of interface signals of the transfer protocol and the software installation required by your computer. The V.24 interface of the TNC must be defined for external data transfer or FE mode.

Starting "Transfer blockwise"

You can start the transfer of data from an external storage unit in "Single block" and "Full sequence" modes by pressing EXT. The TNC loads the program blocks in available memory and interrupts data transfer when memory capacity is reached.

No program blocks are displayed on the screen until available memory is full or the program has been completely transferred.

Although program blocks are not displayed, program execution can be started by pressing the external START button.

Short positioning blocks are usually run when transferring data from an external storage medium. To avoid unnecessary interruption of a program run after is has started, a large number of program blocks should be saved as a buffer. For this reason, it is a good idea to wait until available memory space is full.

After the program run has started, the executed blocks are deleted as further blocks are called from the external storage unit.

External data transfer
Transfer blockwise

Skipping program blocks

If you press the GOTO □ key in "Transfer blockwise" mode before initiating the start, and enter a block number, all blocks preceding the specified block number will be skipped.

Interrupting program execution

To interrupt a program run:

- press the external STOP button and the internal STOP key.

The display "TRANSFER BLOCKWISE" remains on the screen even after execution has been interrupted. The message disappears when you

- call up a new program number

or

- switch from program run "Single block" or "Full sequence" to another operating mode.

Program format

The following conditions apply to program format in "Transfer blockwise" mode:

- Program calls, subroutine calls, program part repeats and conditional program jumps cannot be executed.
- Only the last defined tool can be called (except for operation with central tool storage).

Block number

The program destined for transfer may contain blocks numbered higher than 999.
The block need not be numbered consecutively, but must not exceed 65,534. Four-digit block numbers in plain-language programs are displayed on two lines on the screen.

Graphics
(as a software version 07)

The TNC can graphically simulate on the screen programs that are transferred blockwise from an external memory. It is only necessary to program the workpiece definition BLK FORM behind the BEGIN PGM block.

External data transfer
Transfer blockwise

Starting "Transfer blockwise"

Operating mode ______________ ▤ →

Dialogue initiation ______________ EXT

PROGRAM NUMBER	▶ □	Enter desired program number.
	▼ ENT	Press ENT.

▼

TRANSFER BLOCKWISE		
Wait until initial program blocks are displayed on screen.	▶ START	Press START to run program.

Interrupting "Transfer blockwise"

TRANSFER BLOCKWISE		
To interrupt the program run:	▶ STOP	Press STOP to interrupt run.
	▼ STOP	Press STOP to abort program run.

In → mode, you can also abort program execution by switching to ▤ mode.

External data transfer
TNC 155 graphics output to a printer

You can check a machining program on the TNC 155 with the aid of the graphics feature. The image displayed on the screen can be output via the V.24 interface (EXT mode) and sent to a printer for hardcopy print-out.
The external printer is interfaced to the TNC 155 via machine parameters 226 to 233. To start printing, press EXT while the graphic image is displayed on the screen.
The following input values for machine parameters 226 to 233 apply to the **Texas Instruments OMNI 800/Model 850 printer:**

Parameter No.	Input value
226	1 819
227	17 200
228	6 977
229	2 060
230	1 290
231	6 990
232	2
233	0

The following input values apply to the **EPSON matrix printer:**

Parameter No.	Input value
226	795
227	13 080
228	0
229	0
230	1 805
231	2 587
232	10 757
233	2

Beginning with software version 03:
When printing out the graphic the control automatically switches onto the interface operating mode "EXT", if "ME" or "FE" operation is engaged via the MOD function.

External Data Transfer
Transfer of TNC 145 programs

TNC 145 program management

The TNC 145 can manage only one program at a time in its main memory. In contrast to the TNC 150-/TNC 151-/TNC 155- and TNC 355 programs this program has no program number and can therefore not be managed by the above mentioned controls.

Remedial action

Before transfer of TNC 145 programs, a **service number** must be entered. The TNC then stores the transferred TNC 145 program under this number.

External data transfer
Transfer of TNC 145 programs

All programs read in

Operating mode TNC ____________________

Operating mode ME ____________________ TNC

Dialog initiation ____________________ PGM NR

PROGRAM SELECTION

PROGRAM NUMBER = ▶ ENT

Enter service program number. (maximum of 8 digits).

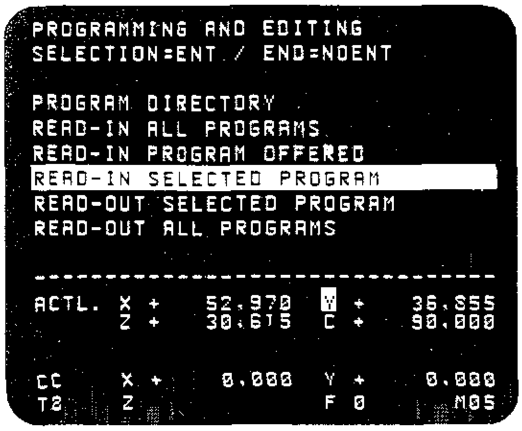
Transfer to memory.

MM = ENT / INCH = NO ENT ▶ ENT for **dimensions in mm.**

or

▶ NO ENT for **dimensions in inch.**

0 BEGIN PGM 12345678 MM ▶ EXT External data transfer.



```
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

--------------------------------
ACTL.  X +   52.970  Y +   36.555
       Z +   30.615  C +   90.000

CC     X +    0.000  Y +    0.000
T2     Z             F 0      M05
```

▶ ↑ ↓ Select "read in all programs."

READ IN ALL PROGRAMS ▶ ENT Start transfer.

The cassette contents with the TNC 145 programs are now stored in the TNC main memory under the service program number 12345678.

Notes:

Technical description Specifications

Control system versions

TNC 151 with BE 111 (9" monochrome) or BE 211 (12" monochrome) video terminal.
Incorporating programmable logic controller (PLC).

TNC 155 with BE 411 video terminal (12" monochrome).
Incorporating programmable logic controller (PLC)

TNC ... B = without separate input/output boards
TNC ... Q = inputs and outputs on 1 or 2 separate PC boards

Type of control

Contouring control system for 4 axes. Optional: *spindle orientation* as 5th axis (no interpolation with remaining axes). *Linear interpolation* on 3 of 4 axes, *circular interpolation* on 2 of 4 axes (only if 4th axis is parallel to one linear axis, limited contour programming with 4th axis), *Helical interpolation.*

Program input and display in HEIDENHAIN plain-language programming format or per ISO 6983 standard.
mm/inch conversion for input values and displays.
Display increment: 0.005 mm or 0.0002 in. or 0.001 mm/0.0001 in.
Nominal positions (absolute or incremental dimensions) in Cartesian or polar coordinates.
Input resolution up to 0.001 mm or 0.0001 in. or 0.001°.

Operator prompting and displays

Plain-language interactive dialogues and error messages (in eight languages).
Display of current, preceding and next two program blocks.
Display of actual position/nominal position/distance to go/distance from reference mark/trailing error and status indicator for all major program data.

Program memory

Semiconductor memory with backup battery for 32 NC programs, total 3,100 blocks.
Programmable erase/edit protection.

Central tool magazine

Up to 99 tools. Suitable for tool changer with variable pocket allocation.

Operating modes

Manual/electronic handwheel: control system functions as numerical position indicator.

Positioning via manual data input: each positioning block is run after being entered; block is not saved.

Program run – single block: Program is executed block-by-block after key is pressed.
Program run – full sequence: Program started by pressing key, executed to programmed STOP or end of program.

Programming and editing: (also during program run):
a) for linear or circular interpolation,
   manually per program listing or part drawing, or
   externally via V.24-/RS-232-C data interface (e.g. via FE 401 floppy-disk unit or ME 101/102 magnetic tape unit from HEIDENHAIN or other peripheral devices)
b) for paraxial operation additionally by transfer of current position data (actual values) with conventional workpiece machinig (playback mode).

Transfer blockwise: Access to programs from master computer or FE 401 floppy-disk unit. Programs exceeding control system memory capacity can be transferred and run simultaneously.

Supplementary operating modes: mm/inch, character height for position display, axis software limits, user parameters (defined by machine manufacturer).
Displays: vacant blocks, actual position/nominal position/distance to go/trailing error/distance from reference mark.
V.24 interface: ME/FE/EXT, baud rate.
ISO programming format: block number increment.

Technical description
Specifications

Programmable functions	Straight line, chamfer Circle (defined by centre and end position of arc or radius and end position of arc), with tangential transition from preceding contour (input: arc end position) Rounded corners (enter radius) Tangential contour approach and departure Tool number, tool length and radius compensation Spindle orientation (optional) Spindle speed Rapid traverse Feed rate Program calls from within other programs Subroutines/program part repeats Canned machining cycles for peck drilling, tapping, slot milling, milling rectangular pockets, milling circular pockets, cycles for milling pockets with variable contours (up to 12 subcontours; intersections computed by control system) Coordinate system offset and rotation Mirror-imaging, scaling factor Dwell time/Auxiliary function M/Program STOP Manufacturer-specific cycles.
Parameter programming	Mathematical functions (=, +, −, x, ÷, sin, cos, angle α from R x sinα and R x cosα, $\sqrt{\ }$, $\sqrt{a^2+b^2}$); parameter comparison (=, ≠, >, <).
Test run without machine movement	**TNC 151 and TNC 155:** analytical program test **TNC 155 only:** graphic simulation of machining program: Simulation modes: in 3 planes, plan view with depth shading, 3D simulation, magnification.
Program editing	Modification of program words, insertion and deletion of program blocks, search routine for finding program blocks with specific characteristics within a program.
Program continuation after interruption	Control system facilitates resumption of program after interruption by saving all important program data.
Touch-probe functions	Programmable: determine actual position of workpiece surface for setting up in "Manual" and "Electronic handwheel" modes: calibrate, define angular clamping position of workpiece, define workpiece corner and circle centre, define workpiece surface as reference plane.

Data interface	Serial interface per CCITT recommendation V.24 or EIA standard RS-232-C;
	Baud rates: 110/150/300/600/1200/2400/4800/9600 baud;
	Expanded interface with control characters and block check characters (BCC) for "Transfer blockwise".

Error control and monitoring	Control system displays programming and operating errors in plain language. It monitors the functioning of major electronic assemblies, positioning and measuring systems and important machine functions. If an error is detected, a plain-language error message is generated and the machine is shut down via emergency STOP.
Reference mark analysis	Reference values are transferred automatically following power failure by passing over transducer reference marks (includes interval-coded reference marks).
Maximum traverse path	± 30,000 mm or ± 30 m/1181 inches.
Maximum traversing speed	16 m per min./630 in. per min.
Feed rate and spindle override	0 to 150% via two potentiometers on the control unit console.
Position transducers	HEIDENHAIN incremental linear transducer (also available with interval-coded reference marks) or rotary encoder; grating pitch: 0.02/0.01 or 0.1 mm.

Technical description
Specifications

Limit switches	Software-controlled limit switch for machine axes (X+/X–/Y+/Y–/Z+/Z– and IV+/IV–); each traverse range is specified as machine parameter; additional programmable traverse range limits.
Integrated PLC (programmable logic interface controller)	2048 commands 1000 user flags (not power-failure protected) 1000 user flags (power-failure protected) 1024 permanently assigned flags 16 counter, 32 timers **Inputs/outputs for TNC 151 B/TNC 155 B:** 23 inputs (24 V =, approx. 10 mA); 24 outputs (24 V =, max. 50 mA) **PLC input/output board for TNC 151 Q/TNC 155 Q:** 63 inputs (24 V =, approx. 10 mA) **PL 100:** 31 outputs (24 V =, max. 1.2 A) **PL 110:** 25 outputs (24 V =, max. 1.2 A) + 3 bipolar output pairs (15 V =, 300 mA) **External voltage supply** for PLC input/output board 24 V = +10% to –15% **Macro programs** for tool change (random or fixed addressing). Input/output capacity can be doubled with a second power board.
Control unit inputs TNC 151/TNC 155 (with standard PLC program)	Transducers X/Y/Z/IV/spindle Electronic handwheel (HR 150 or HR 250) or unit with 2 electronic handwheels (HE 310) Touch-probe systems (TS 510/TS 110) Start, stop, rapid traverse Feedback "Auxiliary function complete", feed rate release Manual operation (opens position control loop); feedback emergency STOP test Reference end position X/Y/Z/IV Reference pulse inhibitor X/Y/Z/IV Axis direction buttons X/Y/Z/IV External feed rate potentiometer
Control unit outputs TNC 151/TNC 155 (with standard PLC program)	One each analogue output for X/Y/Z/IV (with automatic offset calibration), one analogue output for spindle Axis release for X, Y, Z, IV Control system in operation M strobe signal S strobe signal T strobe signal 8 outputs for M, S and T functions, coded "Coolant OFF", "Coolant ON" "CCW spindle rotation" "Spindle STOP" "CW spindle rotation" Spindle interlock Control system in "Automatic" mode Emergency STOP
Supply voltage	Multirange 100/120/140/200/220/240 V, +10% to –15%, 48 to 62 Hz
Power consumption	**TNC 151:** approx. 60 W (with 9" or 12" video display unit) **TNC 155:** Logic and control unit approx. 45 W, BE 411 video display unit approx. 40 W
Ambient temperature	Operation: 0 to 45° C (0 to 118° F) Storage: –30 to 70° C (–22 to 158° F)
Type of enclosure: control console	IP 54
Weight	Control unit: 12 kg (26 lb) BE 111 9" video display unit: 6.8 kg (15 lb); BE 211/BE 411 12" video display unit: 10 kg (22 lb) PL 100/PL 110 PLC input/output board: 1.2 kg (3 lb) (TNC ... Q).

Technical description
Specifications

Electronic handwheels

For connection to TNC 151/TNC 155	**HR 150:** for installation in machine control console (only one handwheel possible) **HR 250:** portable unit with 1 handwheel Attaches magnetically to machine. **HE 310:** portable unit with 2 handwheels 4 axis-control keys for switching both handwheels to individual axes, where simultaneous motion on X–Y, X–Z, Y–IV, IV–Z is possible. Safety switch Emergency STOP switch Attaches magnetically to machine.
Traverse per handwheel revolution	10/5/2.5/1.25/0.625/0.313/0.156/0.078/0.039/0.02 mm (selectable via TNC keyboard)
Maximum traversing rate	2.4 m per min. (≙ 4 rps) if not limited by TNC parameters
Power supply	from TNC
Cable length	**HR 150:** 1 m (3 ft), max. 10 m (33 ft) **HR 250:** 3 m (10 ft), max. 10 m (33 ft) **HE 310:** 3 m (10 ft), max. 20 m (66 ft)
Enclosure	IP 64 (HR 250 and HE 310 only)
Ambient temperature	Operation: 0 to 45° C (0 to 118° F) Storage: −30 to 70° C (−22 to 158° F)
Weight	**HR 150:** 0.3 kg (0.66 lb) (without rotary knob/handwheel) **HR 250:** 1.1 kg (2.4 lb) **HE 310:** 3.9 kg (8.6 lb)

Technical description Specifications

Floppy-disk unit

	FE 401: compact portable unit for use on multiple machines (can also be used with TNC 131/TNC 135/TNC 145 and TNC 150)
Data interfaces	2 interfaces per CCITT recommendation V.24 or EIA standard RS-232-C Baud rates: with 1 interface: 2400/9600 baud with 1 interface: 110/150/300/600/1200/2400/4800/9600 baud
Disk drives	2 disk drives, including one for copying Panasonic JU 343
Floppy disks	BASF 3 1/2 inch, double-sided 135 TPI Storage capacity: approx. 790 kilobyte (approx. 25,000 program blocks), max. 256 different programs
Supply voltage	Multirange 100/120/140/200/220/240 V +10 % to −15 %, 48 to 62 Hz
Power input	Max. 18 W
Ambient temperature	Operation: 15 to 45° C (59 to 113° F) (approx. 10 min. after starting: 10 to 45° C [50 to 113° F]) Storage: −40 to +60° C (−40 to 140° F)
Weight	4.9 kg (11 lb)

Magnetic tape unit

	ME 101: compact portable unit for use on multiple machines **ME 102:** integrated unit for permanent installation in machine control console
Data interfaces	2 interfaces per CCITT recommendation V.24 or EIA standard RS-232-C Baud rates: with 1 interface: 2400 baud, fixed with 1 multirange interface: 110/150/300/600/1200/2400 baud
Cassette drive	Philips Mini-DCR
Cassettes	Philips Digital Mini-Cassette, No. 8920 440 10101, with write-protect tab Storage capacity: approx. 35 kilobyte per side (approx. 1000 program blocks per side) Erasure time: approx. 180 sec.
Supply voltage	Multirange 100/120/140/200/220/240 V +10 % to 15 %, 48 to 62 Hz
Power input	16 W
Ambient temperature	Operation: 4 to 45° C (39 to 113° F) Storage: −30 to 70° C (−22 to 158° F)
Weight	4.7 kg (10 lb)

Technical description
Specifications

3D Touch-probe systems

TS 511
with infrared
transmission

Touch trigger 3D probe
Probing repeatability better than 1 µm (40 µinch)
Probing speed: max. 3 m/min. (9.8 ft per min.)
Stylus with shear point
Ruby probe tip
Shank and stylus shape available to customer specifications

Infrared transmission path:
2 signal transmitters (0° and 180°)
1 starting signal receiver (for 0°)
Optional signal radiation direction to spindle axis (specify when ordering): 90/60/30°
Distance between 3D probe and transmitter/receiver: 500 to 2000 mm (20 to 79 in.)

Power supply:
4 "micro-sized" NiCd accumulator batteries
Maximum operating time per charge:
measuring mode 8 hr., standby mode 1 month
Delivery includes second battery set and external charging unit (220 V, 50 Hz)

Enclosure: IP 55

Interface to CNC control unit
The interface consists of transmitter/receiver and evaluator electronics system.
SE-Transmitter/receiver:
Diameter: 80 mm (3 in.) Length: 49 mm (2 in.)
Cable length: 3 m (10 ft)
Enclosure: IP 66
APE-Evaluator electronics system:
In die-cast aluminium housing: (LxWxH) 175 mm x 80 mm x 57 mm (7 x 3 x 2 in.)
Maximum cable length: 20 m (65 ft)
Enclosure IP 64

TS 110
with cable

3D Touch-trigger probe
Technical data same as 3D probe with infrared transmission, but without infrared transmitter/receiver.
Cable length: 3 m (10 ft)

Evaluator electronics system:
In die-cast aluminium housing: (LxWxH) 175 mm x 80 mm x 57 mm (7 x 3 x 2 in.)
Maximum cable length: 20 m (65 ft)
Enclosure: IP 64

Notes:

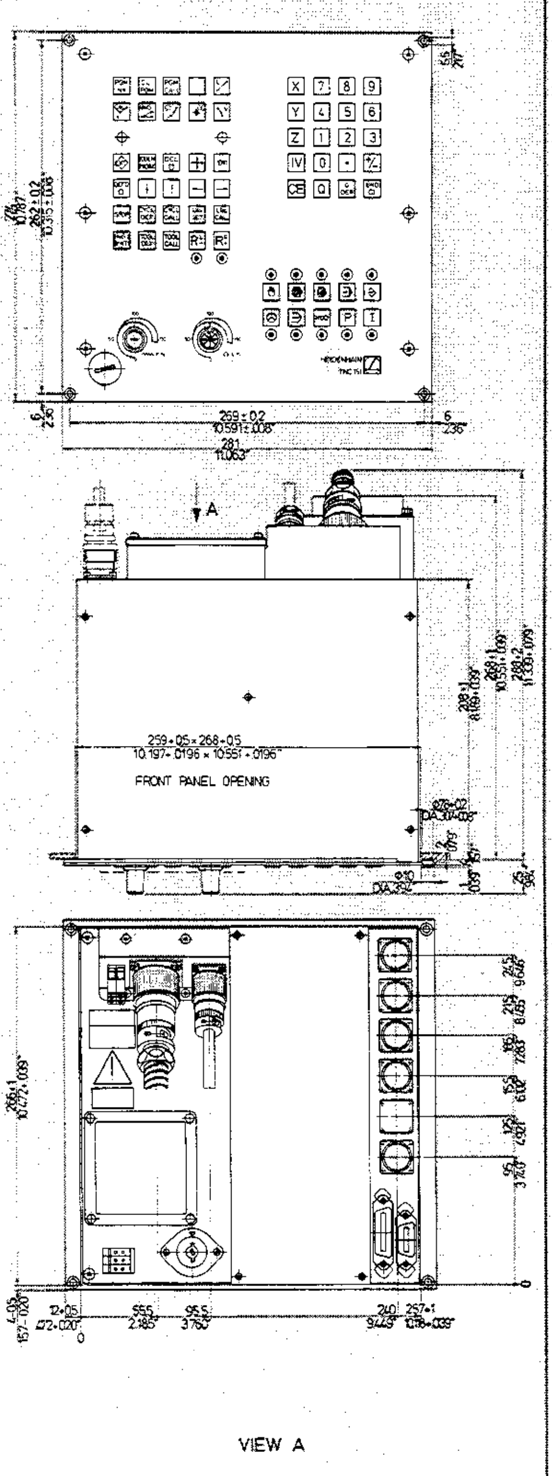
Dimensions

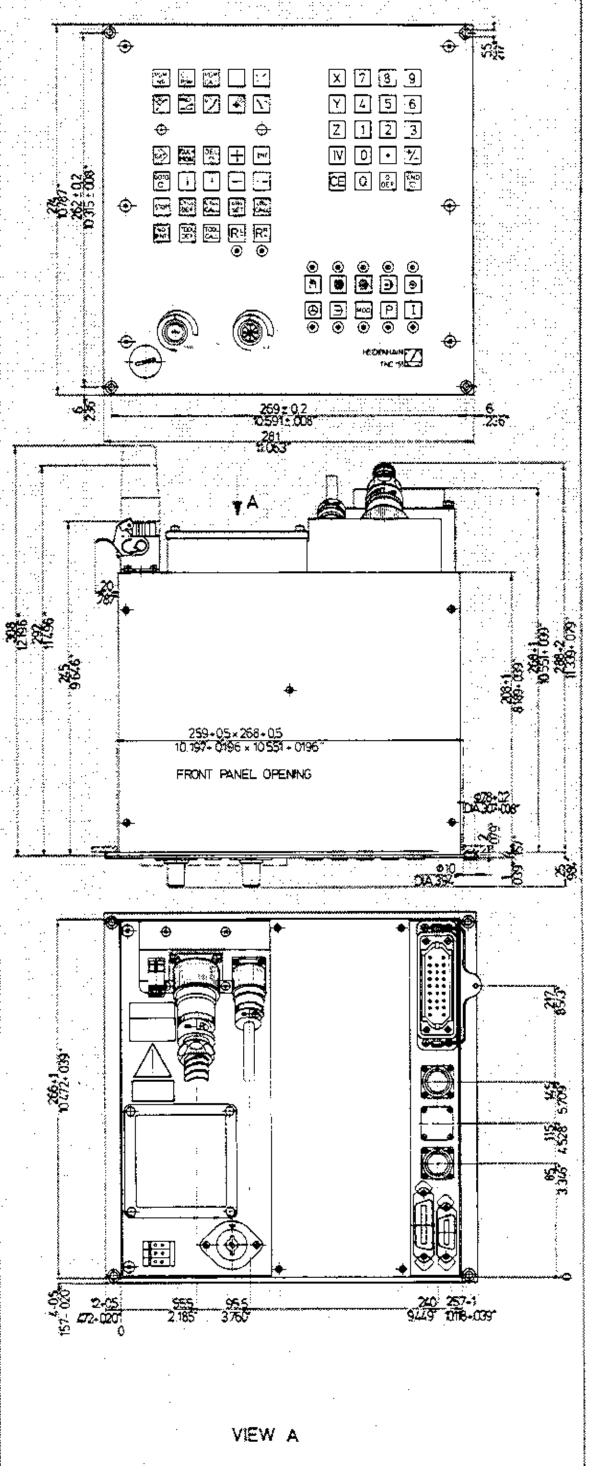
Logic and control block

TNC 151 B/Q TNC 151 BR/QR

TNC 151 F/W TNC 151 FR/WR

Dimensions mm/inch



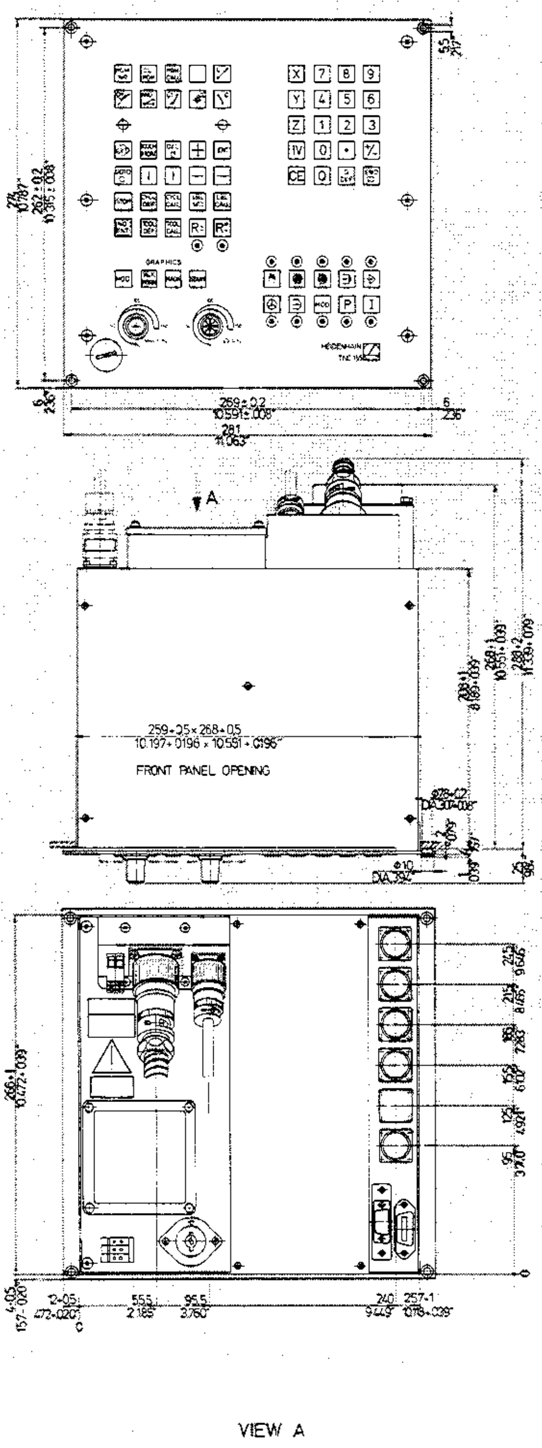


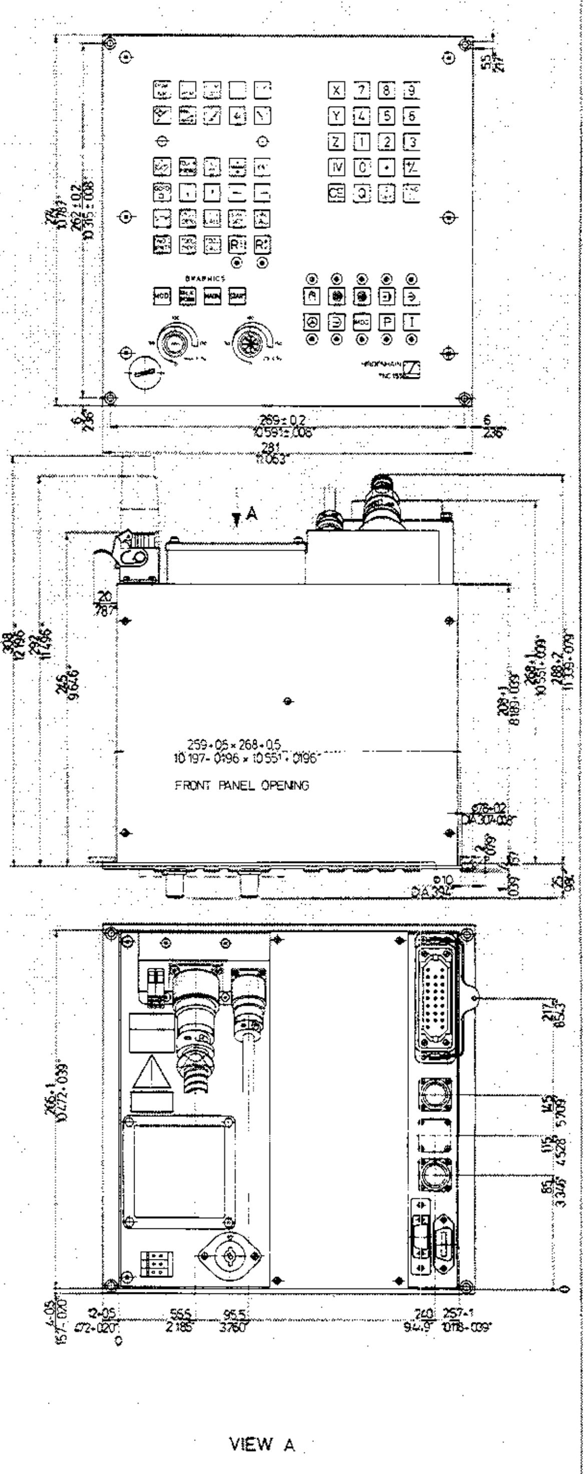
Dimensions

Logic and control block

TNC 155 B/Q TNC 155 BR/QR
TNC 155 F/W TNC 155 FR/WR

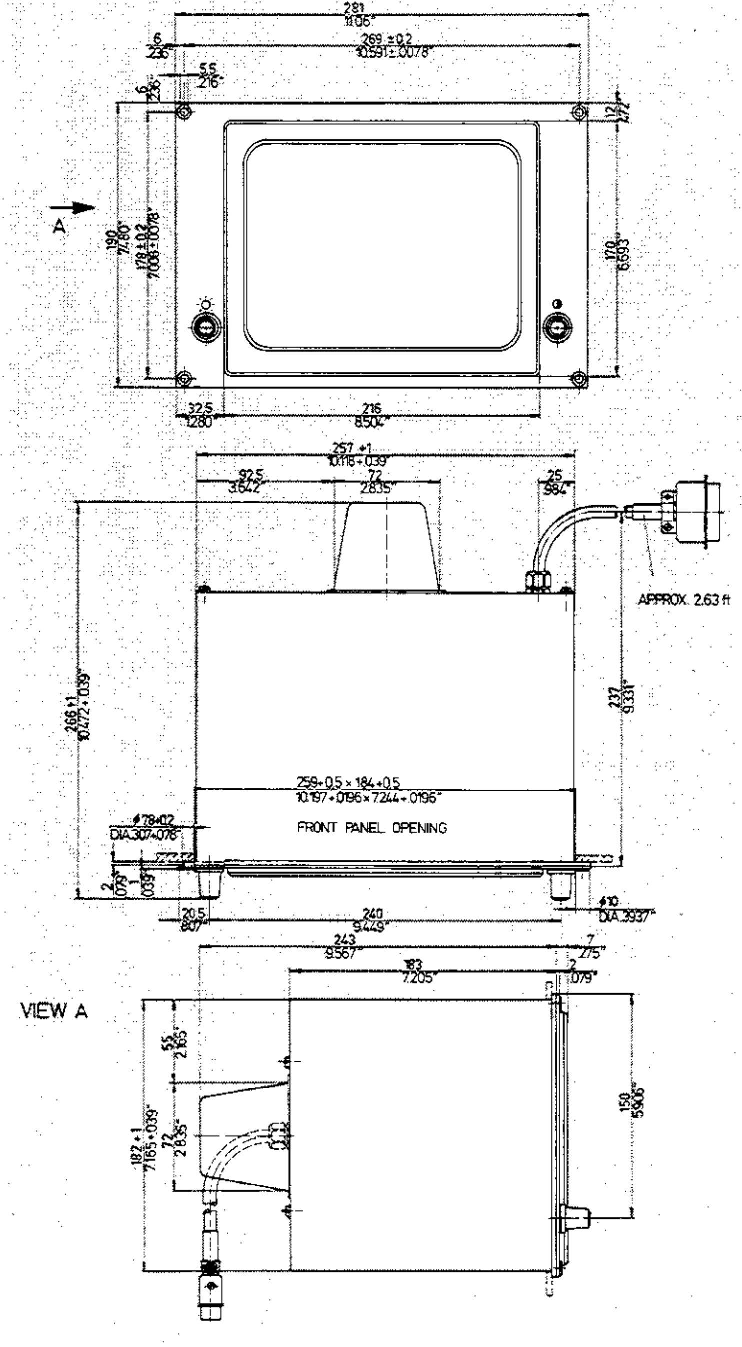
Dimensions mm/inch



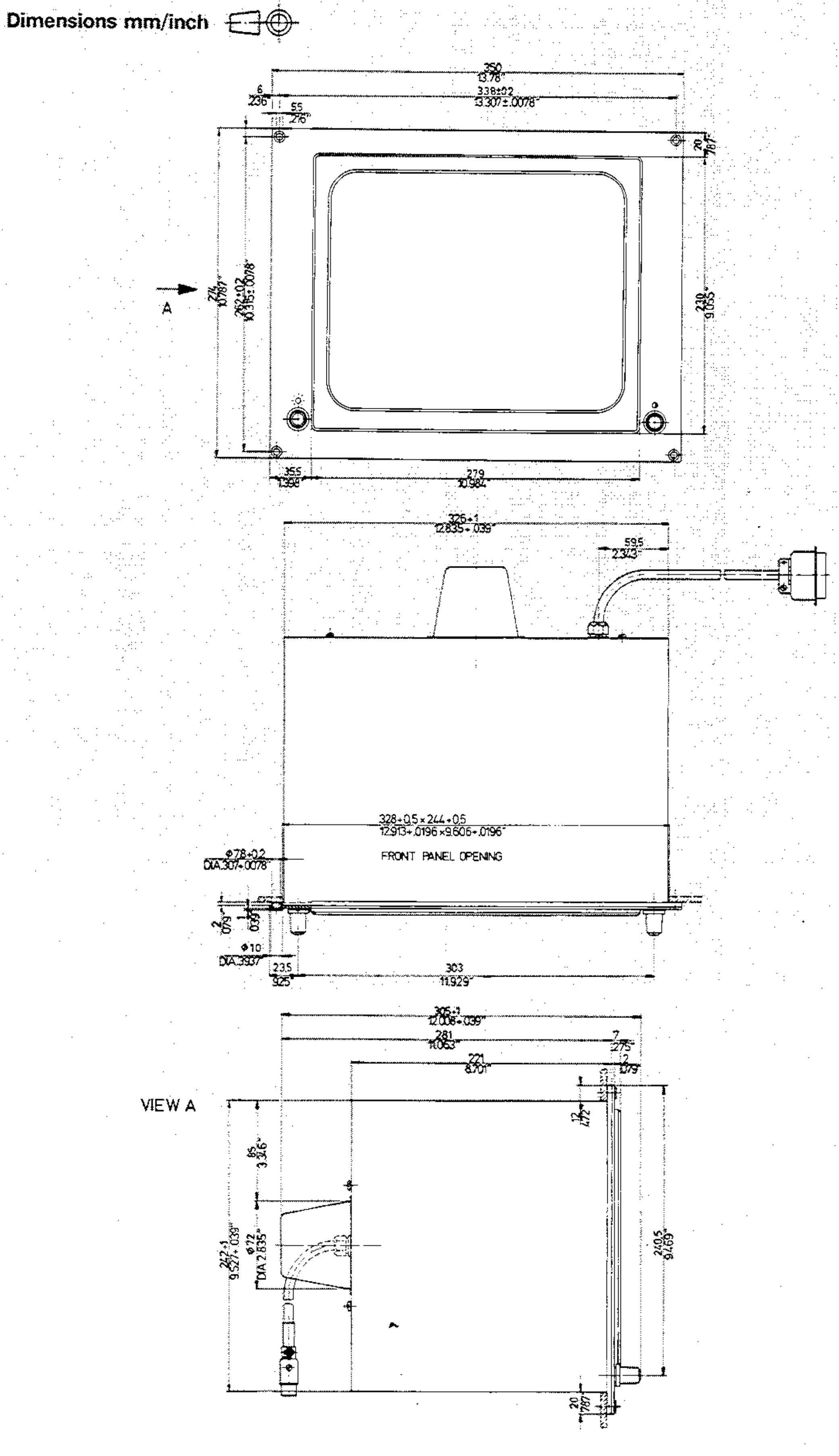


Dimensions
BE 111 9" video display unit

Dimensions mm/inch

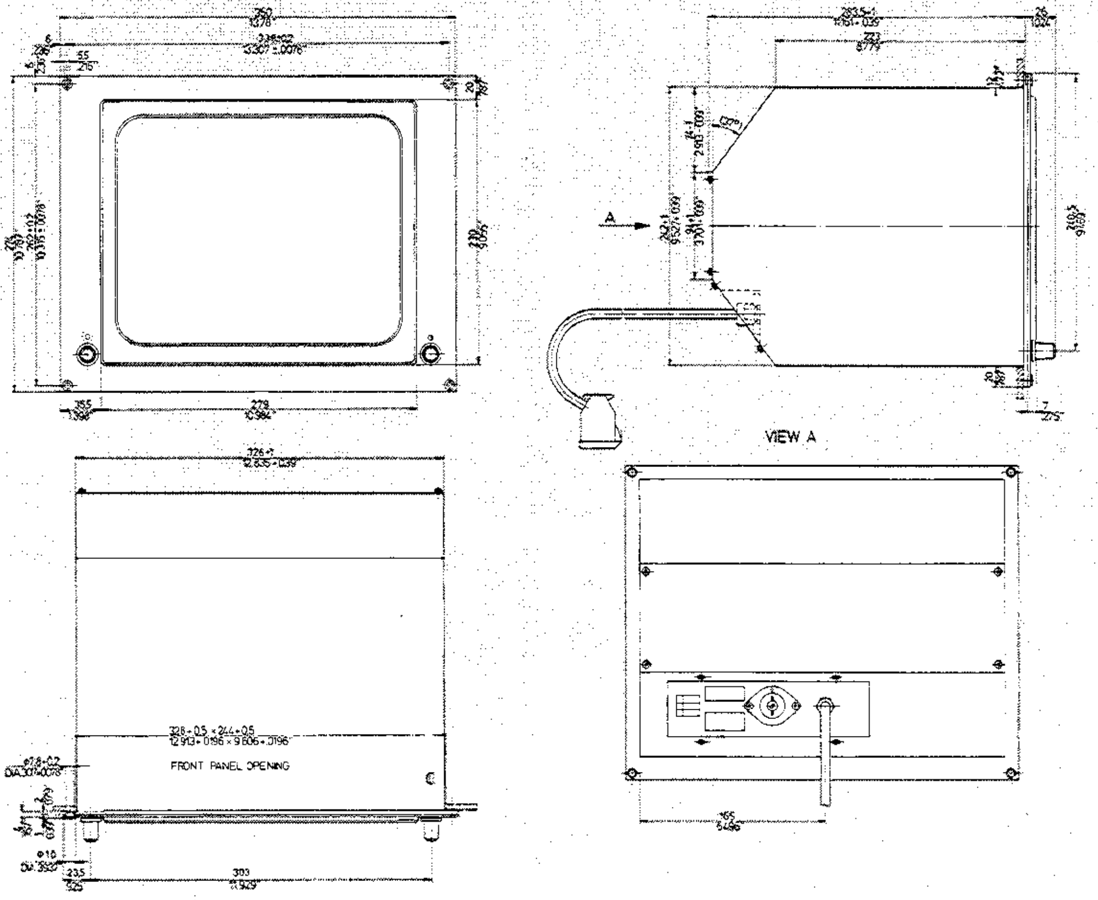


Dimensions
BE 211 12" video display unit



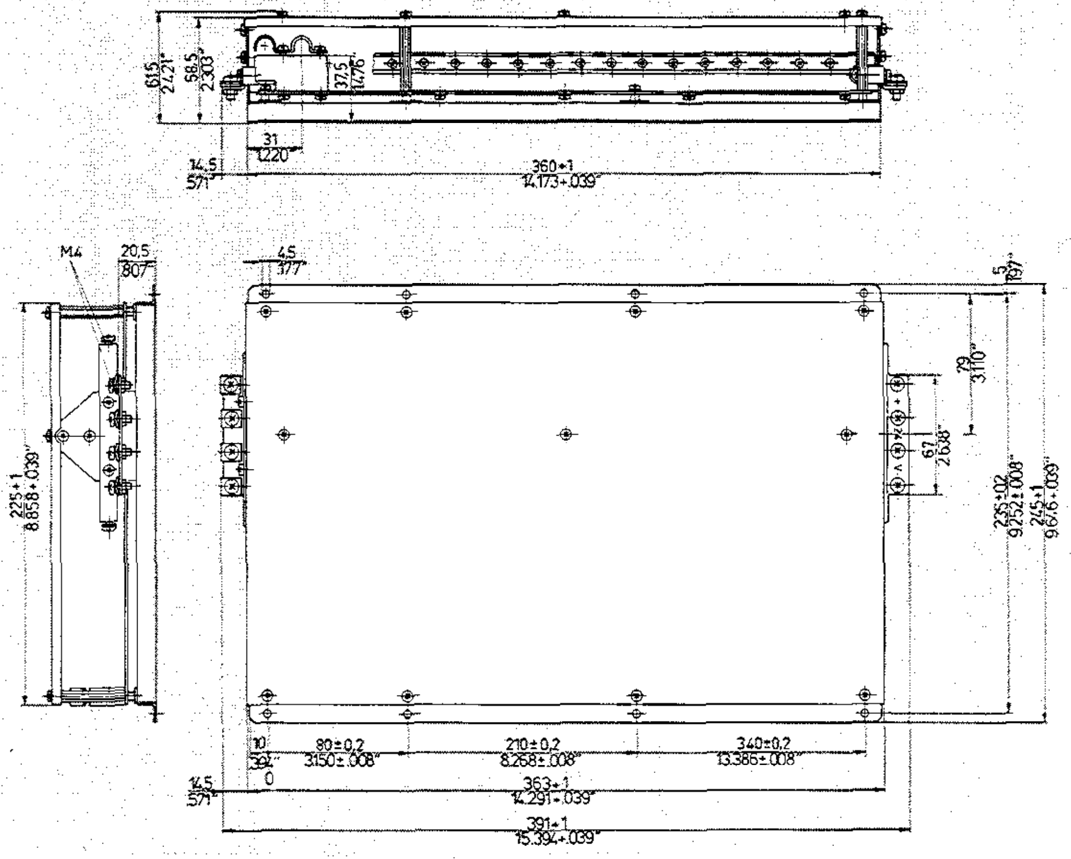
Dimensions
BE 411 video display unit

Dimensions mm/inch



Dimensions
PL 100/PL 110 PLC input/output board

Dimensions mm/inch



Index

Index

Index

Index

Index

Index

Index

Index

Index

Index

Error messages

Error messages

Auxiliary functions M

M	Function	Active at block		Remarks page
		beginning	end	
M00	Stop program run/Spindle STOP/Coolant OFF		●	
M02	Stop program run/Spindle STOP/Coolant OFF/if required: clearing the status display (indepedent of machine parameters)/Return to block 1		●	
M03	Spindle ON: clockwise	●		
M04	Spindle ON: counterclockwise	●		
M05	Spindle STOP, Coolant off (with standard PLC-program)		●	
M06	Tool change/Stop program run (if req'd., depends on specified machine parameters)/Spindle STOP		●	
M08	Coolant ON	●		
M09	Coolant OFF		●	
M13	Spindle ON: clockwise/Coolant ON	●		
M14	Spindle ON: counterclockwise/Coolant ON	●		
M30	same as M02		●	
M89	Variable auxiliary function	●		
	– or –			
M89	Cycle call, modal (depends on machine parameters)		●	P94
M90	Constant tool path feed rate at external and internal corners	●		P29
M91	**within positioning block:** coordinates refer to the reference point (Reference point substituted for workpiece datum)	●		
M92	**within positioning block:** coordinates refer to a position defined by machine manufacturer via machine parameter, e.g. tool change position (workpiece zero is replaced)	●		
M93	M-function assignment reserved by HEIDENHAIN	●		
M94	Reduction of displayed value for rotary table axis to below 360° (programmed setting of actual value)	●		
M95	Changed approach behavior for start in internal corners: no calculation of point of intersection		●	P69
M96	Changed approach behavior for start at external corners: inserting a tangential circle		●	P68
M97	Contour compensation on external corners: point of intersection instead of tangential circle		●	P28
M98	End of contour compensation active blockwise: radius compensation RL/RR is cancelled only for the next positioning block		●	P30, P68
M99	Cycle call active blockwise		●	P94

Address codes (ISO)

Address code	Function	Input range	
		Numbers	Parameter
%	Program start or call	0 – 99999999	–
A-axis	(rotation about X-axis)	± 30000.000	Q0 – Q99
B-axis	(rotation about Y-axis)	± 30000.000	Q0 – Q99
C-axis	(rotation about Z-axis)	± 30000.000	Q0 – Q99
D	Parameter definition (Program parameter Q)	0 – 14	–
F	Feed rate	0 – 15999	Q0 – Q99
F	Dwell with G04	0 – 19999.999	Q0 – Q99
F	Scaling factor with G72	0 – 99.999	–
G	G-code	0 – 99	–
H	Polar coordinate angle		
	in incremental dimensions	± 5400.000	Q0 – Q99
	in absolute dimensions	± 360.000	Q0 – Q99
H	Angle of rotation with G73	± 360.000	Q0 – Q99
I	X-coordinate of circle centre/pole	± 30000.000	Q0 – Q99
J	Y-coordinate of circle centre/pole	± 30000.000	Q0 – Q99
K	Z-coordinate of circle centre/pole	± 30000.000	Q0 – Q99
L	Set label number with G98	0 – 254	–
L	Jump to label number	1 – 254.65535	–
L	Tool length with G99	± 30000.000	Q0 – Q99
M	Auxiliary functions	0 – 99	–
N	Block number	1 – 9999	–
	in “Transfer blockwise” mode	1 – 65534	–
P	Cycle parameter in machining cycles	01 – 12	–
P	Parameter in parameter definitions	01 – 03	–
Q	Program parameter “Q”	0 – 99	–
R	Polar coordinate radius	± 30000.000	Q0 – Q99
R	Circle radius with G02/G03/G05	± 30000.000	Q0 – Q99
R	Rounding-off radius with G25/G26/G27	0 – 19999.999	Q0 – Q99
R	Chamfer length with G24	0 – 19999.999	Q0 – Q99
R	Tool radius with G99	± 30000.000	Q0 – Q99
S	Spindle speed	0 – 30000.000	–
S	Angular spindle position with G36	0 – 360.000	–
T	Tool definition with G99	0 – 254	–
T	Tool call	0 – 254	–
U-axis	(linear movement parallel to X-axis)	± 30000.000	Q0 – Q99
V-axis	(linear movement parallel to Y-axis)	± 30000.000	Q0 – Q99
W-axis	(linear movement parallel to Z-axis)	± 30000.000	Q0 – Q99
X	X-axis	± 30000.000	Q0 – Q99
Y	Y-axis	± 30000.000	Q0 – Q99
Z	Z-axis	± 30000.000	Q0 – Q99
*	End of block	–	–

Auxiliary functions M

M	Function	Active at block		Remarks page
		beginning	end	
M00	Stop program run/Spindle STOP/Coolant OFF		●	
M02	Stop program run/Spindle STOP/Coolant OFF/if required: clearing the status display (indepedent of machine parameters)/Return to block 1		●	
M03	Spindle ON: clockwise	●		
M04	Spindle ON: counterclockwise	●		
M05	Spindle STOP, Coolant off (with standard PLC-program)		●	
M06	Tool change/Stop program run (if req'd., depends on specified machine parameters)/Spindle STOP		●	
M08	Coolant ON	●		
M09	Coolant OFF		●	
M13	Spindle ON: clockwise/Coolant ON	●		
M14	Spindle ON: counterclockwise/Coolant ON	●		
M30	same as M02		●	
M89	Variable auxiliary function	●		
	– or –			
M89	Cycle call, modal (depends on machine parameters)		●	P94
M90	Constant tool path feed rate at external and internal corners	●		P29
M91	**within positioning block:** coordinates refer to the reference point (Reference point substituted for workpiece datum)	●		
M92	**within positioning block:** coordinates refer to a position defined by machine manufacturer via machine parameter, e.g. tool change position (workpiece zero is replaced)	●		
M93	M-function assignment reserved by HEIDENHAIN	●		
M94	Reduction of displayed value for rotary table axis to below 360° (programmed setting of actual value)	●		
M95	Changed approach behavior for start in internal corners: no calculation of point of intersection		●	P69
M96	Changed approach behavior for start at external corners: inserting a tangential circle		●	P68
M97	Contour compensation on external corners: point of intersection instead of tangential circle		●	P28
M98	End of contour compensation active blockwise: radius compensation RL/RR is cancelled only for the next positioning block		●	P30, P68
M99	Cycle call active blockwise		●	P94

HEIDENHAIN

225 041 28 · 10 · 1/89 · H · Printed in the Federal Republic of Germany · Subject to change without notice