Programming Instruction
EMCOTRONIC TM 02
Turning

Edition 91-4

Ref. No. EN4 247

Programming Instr.
EMCOTRONIC T2
91-4 EN4 247

Foreword

1. EMCOTRONIC TM 02 literature

The following literature is available on the EMCOTRONIC TM 02 specification:

*	Programming instructions	Ref. No.	..4247
*	Operating manual	Ref. No.	..4246

2. Literature for machines with EMCOTRONIC TM 02 control

This consists of the above specified pamphlets as well as the machine specific operating manuals, spare parts lists and circuit diagrams.

3. Structure of the EMCOTRONIC TM 02 literature:

The operating and programming manuals are suitable for self-tuition.

The programming manual contains numerous examples, which in addition to the overall descriptions, clearly and comprehensively specify the control features.

All routines are easy to copy from the operating manual.

The brief specification contains programming summaries and the most important operating routines.

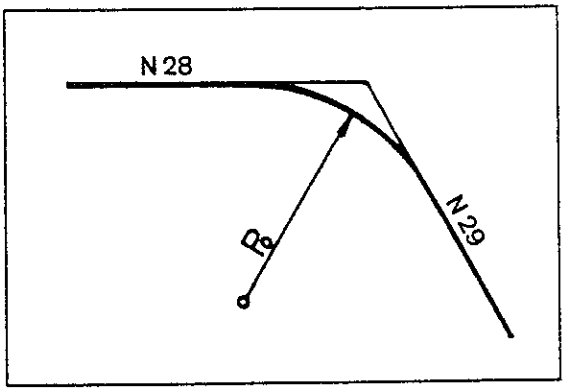
If you should have any improvement suggestions, please let us know.

Yours faithfully

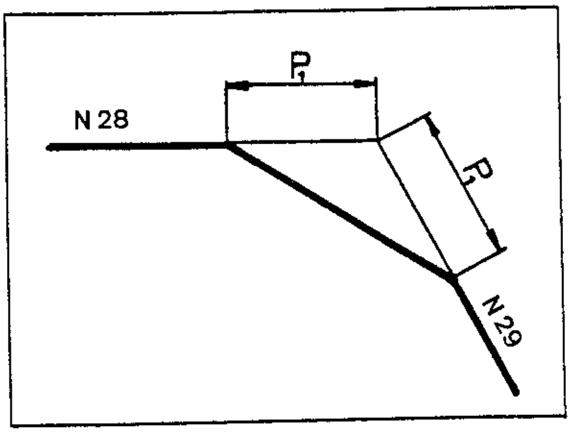
EMCO, Maier & Co., Hallein
TECHNICAL DOCUMENTATION

Software Extension
Emcotronic TM 02
DC 5.10

Inserting of chamfers and radii



N28 G01 X.. Z.. F.. P_0..
N29 G01 X.. Z.. F..



N28 G01 X.. Z.. F.. P_1..
N29 G01 X.. Z.. F..

MILLING

N 4	G01	X..	Y..	Z..	P_0 P_1 43

TURNING

N 4	G01	X..	Z..	P_0 P_1 43

- A radius or a chamfer can be inserted between two straight lines (e.g. block N28 and N29)

- A radius is defined with parameter P_0 [mm].

- A chamfer is defined with parameter P_1 [mm].
  The chamfer is laid symmetrically into the edge, i.e. length P_1 is identical on both enclosing straight lines.

- P_0 and/or P_1 is attached at the first of the two enclosing blocks (N28).

Conditions

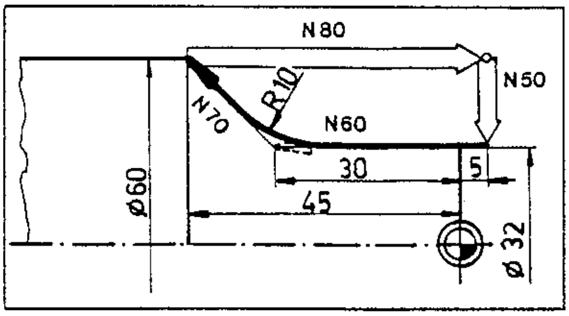
- The length P1 of an inserted chamfer must not be longer than the shorter of the enclosing straight lines, otherwise it would not result in an intersecting point.
- For the calculation of the chamfer and/or the radius the block in which the chamfer and/or the radius is programmed as well as the subsequent block is necessary.

  In these blocks no PSO change, no tool change and no scaling change must be carried out.

  In the execute mode the subsequent block is not available, therefore chamfers and radii cannot be programmed with P0 and/or P1 .
- The block in which the chamfer and/or the radius is programmed must contain exactly two position parameters (X+Y,X+Z,Y+Z).

Examples:

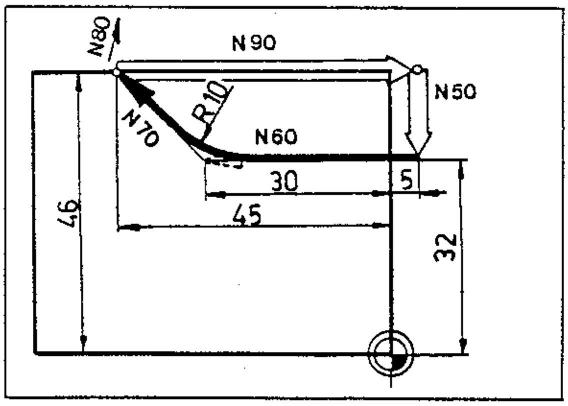
TURNING



.
.
.

```
N0050/G00/X32.000
N0060/G01 /X32.000/Z-30.000/F.../P0 = 10.000
N0070/G01/X60.000/Z-45.000/F...
N0080/G00/Z5.000
```

MILLING



.
.
.

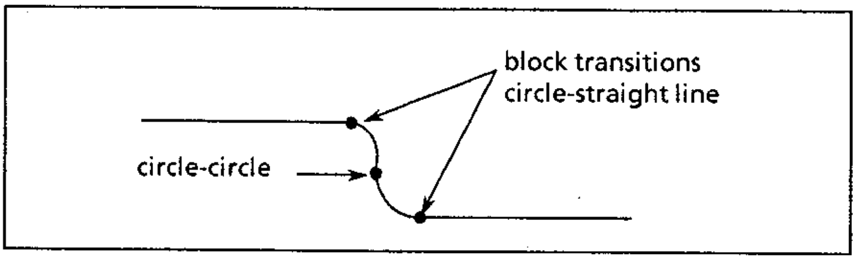
```
N0050/G00/Y32.000/Z-15.000
N0060/G01/X30.000/Y32.000/F.../P0 = 10.000
N0070/G01/X-45.000/Y46.000
N0080/G01/Z1.000
N0090/G00/X5.000/Y46.000
```

Software extension
EMCOTRONIC TM02
DC 5.00

Index

Block transitions without hold

From software 5.00 tangential block transitions to circles can be traversed without tool hold.

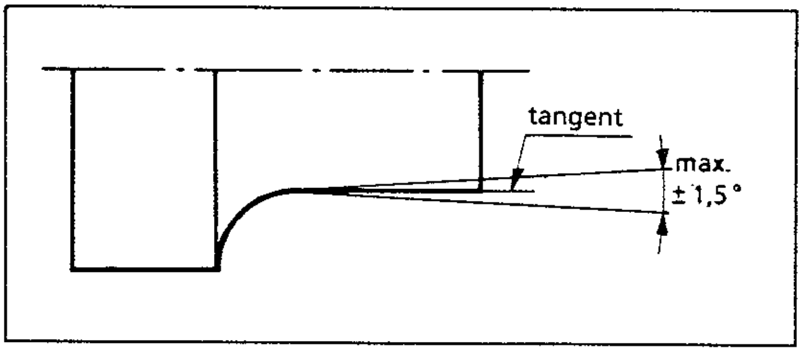


Advantage:

- Time saving
- There is no "free-cutting of the tool" at the block transition.

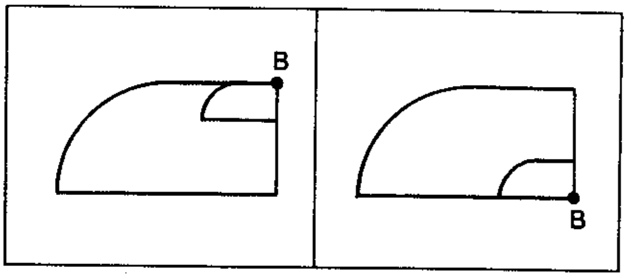
Basic conditions:

- The M39 "precise stop off" has to be active (is deselected with M38 "precise stop on").
- The block transition has to be tangential. A maximum deviation of ± 1,5° is permitted.



- The block after the transition (subsequent block) has to be programmed in the same feed. This limitation will be eliminated in the next software version.
- Both contour elements have to be in the same main plane (X-Y, X-Z or Y-Z plane).
- The feed override switch must not be actuated prior to the block transition.
- The contour elements must not be moved in rapid motion.
- If a contour element is too short or the feed is too large the control has too little time for calculating the following block transition and an exact hold is executed.

Scaling factor



Selection of the scaling factor

N 4	G51	X U ±43	Y V ±43	Z W ±43	P_7 ±43
		[mm]	[mm]	[mm]	[ ]

Deselection of the scaling factor

N 4	G50

A tool path can be scaled up or scaled down in a linear way from a reference point (B).

Data required:

1. Tool path:
   The tool path to be scaled up or scaled down is described in the program between G51 and G50. It can be opened or closed.

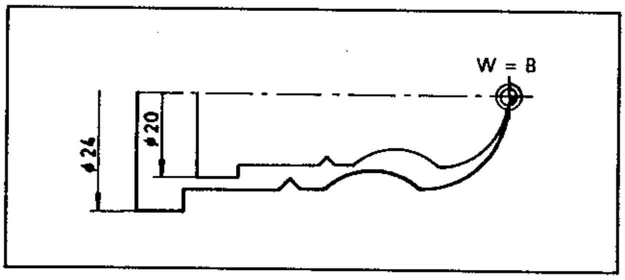
2. Reference point (B):
   The reference point is described with X,Y,Z (absolute) and U,V,W (incremental). It can be situated anywhere on or beside the contour or anywhere in the space.

3. Scaling factor (P_7):
   With P_7 the scale for scaling up or scaling down the tool path is determined.
   It can range from 0 up to ± 9999,999.
   e.g. scale1:2 ..... P_7 = 0,5
   scale1,38:1 ....P_7 = 1,38

Mind!
Thread pitches are scaled up or scaled down correspondingly.

Example 1:

You have written a program for ⌀ 24 and only get blanks with ⌀ 20.

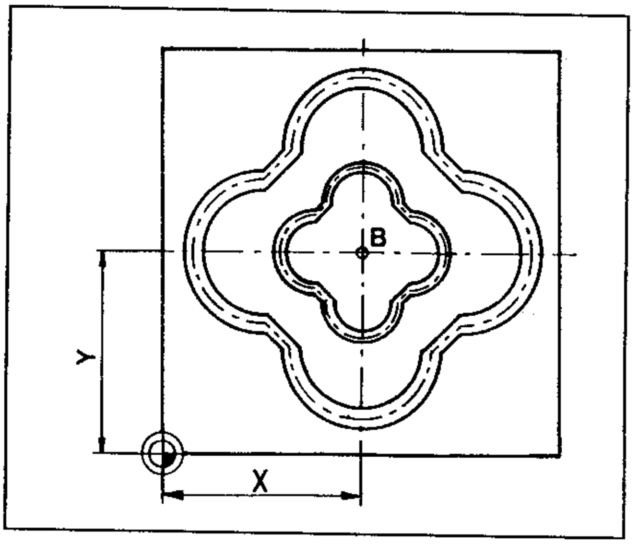


N .... G51, X = 0, Z = 0, $P_7 = \frac{20}{24}$

N ..... usual program for Ø 24

N G50

Example 2:



Memory extension

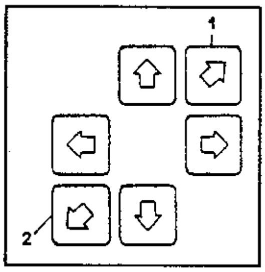
From software version DC 5.00 the Emcotronic control is provided with 64 kB RAM memory (previously 32 kB).

Reason: With the software version 5.00 additional capacity of the RAM memory was occupied (occupied RAM capacity increased from approx. 7 to approx. 12 kB).
Thus, there would be less space available than before for storing the programs.

Additional memory extension (option)
When retrofitting software version 5.00 you are therefore advised to install a memory extension.
The memory extension (ord.no. 276 110) consists of 3 memory modules for extension to 128 kB.

Listing memorized programs (key L)

Due to the memory extension more programs can be in the memory han can be listed on one screen page.



1 .. Display next page
2 .. Display previous page

For this reason the possibility was provided to turn forward and backward in the pages using the jog keys Y + and Y-.
The display can consist of a maximum of 3 pages. 258 programs can be displayed at a maximum. If there are more programs in the memory ALARM 675 "TOO MANY PROGRAMS IN THE MEMORY" is emitted.
Remedy:
Cancel programs that are not required any more.

Loading all stored programs from the machine memory onto cassette

After selecting "OUTPUT ALL" the inserted cassette is registered.

If the first cassette is full the message "INSERT NEXT TAPE" is displayed on the screen.
On the newly inserted cassette that program is started which did not have enough space on the previous cassette. If all programs are stored there is an automatic return into the EDIT mode.

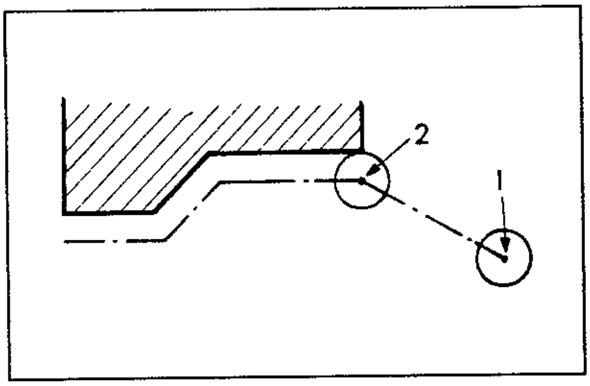
Note:
- Cassettes have to be formatted on the Emcotronic, otherwise there is an alarm and OUTPUT ALL is interrupted.
- The inserted cassette need not be empty; the programs already stored on it will be preserved.
- With the RESET key OUTPUT ALL can be interrupted, all the other keys are not active during OUTPUT ALL.

Changes in the operator monitor

From software DC 5.00 the R-parameters (machine specific data) cannot be changed by anyone in the operator monitor but only by a service engineer.

Reason:
By inexpert changes damage at the machine can be caused.

Note regarding radius compensation (G41/G42)



With G42 a "traversing command to the starting point" should be programmed,
e.g the tool position (1) during selection of G42 has to be different from the starting point (2) of the radius compensation.

If this traversing command is not indicated alarm 520 may occur (error during selection/deselection of the compensation).

N .... G00 X_1 Y_1 Z_1
N .... G42 $\underbrace{\text{G00 } X_2 \, Y_2 \, Z_2}$

traversing command
to the starting point

The dividing attachment (M27)

From software DC 5.00 the dividing attachment TARM HW 125 (Messrs. Walter) can be controlled.

Function:

- Adjust divisions to be carried out (min. 15°) at the control of the dividing attachment.
- The starting command for the division is carried out by M27 in the NC program.
- Subsequently there is a response from the dividing attachment to the machine control and the NC program continues to be executed.

An exact description, installation and connection instructions are to be found in the operating instructions (ord.no. F5Z 140 030) enclosed to the dividing attachment.

Bar loading magazine (M65)

From software DC 5.00 the bar loading magazine LM 1000 of Messrs. KUPA can be controlled.
With M65 the execution of the NC program is stopped until the response is emitted by the bar loading magazine.

Selection required in the operator monitor

Activate M65:

L39 Bit 1 = HIGH	......	M65 selected
L39 Bit 1 = LOW	......	M65 deselected

Activate bar loading magazine:

L25 Bit 2 = HIGH	......	bar loading magazine selected
L25 Bit 2 = LOW	......	bar loading magazine deselected

Notes:

- In your NC program you must not use skip blocks (SKIP) since these are used for bar change.

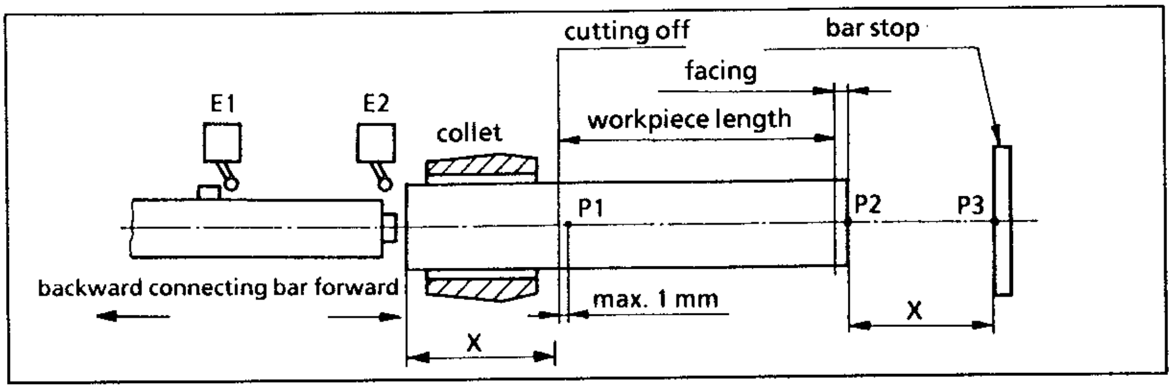
- M65 must not be programmed with active cutter radius compensation.

- The bar loading magazine starts pushing forward the bar if the clamping device is opened when the program is operative (CYCLE START active). You are therefore advised to start NC programs only with closed clamping device since otherwise immediately after pressing CYCLE START the bar is pushed out although it is not provided that the stop is swivelled in or that the tool turret is in the correct position.

- Alarm of the bar loading magazine:
  1. Red led blinks once/sec = material supply off
  2. Red led blinks twice/sec = irregular state

  Remedy: Eliminate state and switch the bar loading magazine on and off.

Functional description



P1 ... Pick-up position of the bar
P2 ... Machining position of the bar
P3 ... Ejecting position of the remaining bar
E1 ... Limit switch 1 for bar end
E2 ... Limit switch 2 for remaining bar ejection
} limit switches have to be adjusted correspondingly

Subprogram for "loading"

T.. Swivel in bar stop
Note: The bar stop is clamped in the tool turret.
It is absolutely necessary to use a spring-type bar stop.

G00 Move bar stop to P1.
Explanation:
If the bar stop immediately moves to P2 the bar would be provided with a too high impact speed due to the long approach travel.

M25 Open clamping device, with M25 the connecting rod is activated at the same time and pushes the bar forward.

G04 Dwell time (2 sec) until the bar has reached the bar stop.

G01 Move bar stop to P2 (G94, F3000). The bar is pushed by means of the connecting bar.

M65 Control waits for signal of the bar loading magazine. If the bar is at the bar stop (point P2) a limit switch provides a signal.

Explanation for the following six blocks:
If the bar is too short for a new workpiece the connecting bar overtravels limit switch 1, a signal is emitted to the Emcotronic thus inactivating "SKIP" until program end M30. (The skip blocks designated with "/" are executed.)

/G00 Move bar stop to P3
The connecting bar ejects the rest of the bar and overpasses limit switch 2. Thus, the connecting bar travels into the rear final position and initiates a bar change.

/M65 Waiting for signal of bar loading magazine: Bar is at the bar stop (point P3).

/G01 Move bar stop to P2 (G94/F3000)

M26 Close clamping device, connecting bar returns back.

M65 Waiting for signal of bar loading magazine: Connecting bar has reached rear final position.

G00 Move to tool change position

M17 Subprogram end

Explanation: Default Option
abbreviated: Def.

The abbreviation Def. (Default Option) is used in the manual under cycle parameters, and in the operator monitor.

Default Option is a standard presumption where a detail has been omitted.

Explanation:

In the control this implies:
Where you omit a word or parameter input marked with Def., the control presumes a specific value or condition as being applicable. This standard presumption is specificed by the control manufacturer.

Examples:

Thread cycle G85:
Where no D_5 (thread angle) is programmed, the control presumes that you intend to produce a thread by plunge-cut feed.

Longitudinal turning cycle G84:
Where no D_3 (cut division detail) is programmed, the control does not execute any cut division.

Operator monitor (MON):

Thread cycle G85:
Where no S_{01}(idle cuts) is programmed, the number of idle cuts specified in the operator monitor are executed. The factory setting is S_{01} = 1 in the operator monitor.

Total Index

Chapter 1

Summaries, Technical Data

Chapter 2

General notes on programming

Chapter 3

Tool programming
Zero point shift

Chapter 4

G-functions

Chapter 5

M-functions

Chapter 6

Operator Monitor

EMCOTRONIC TM 02

Chapter 7

The serial interface RS
232 C EMCOTRONIC

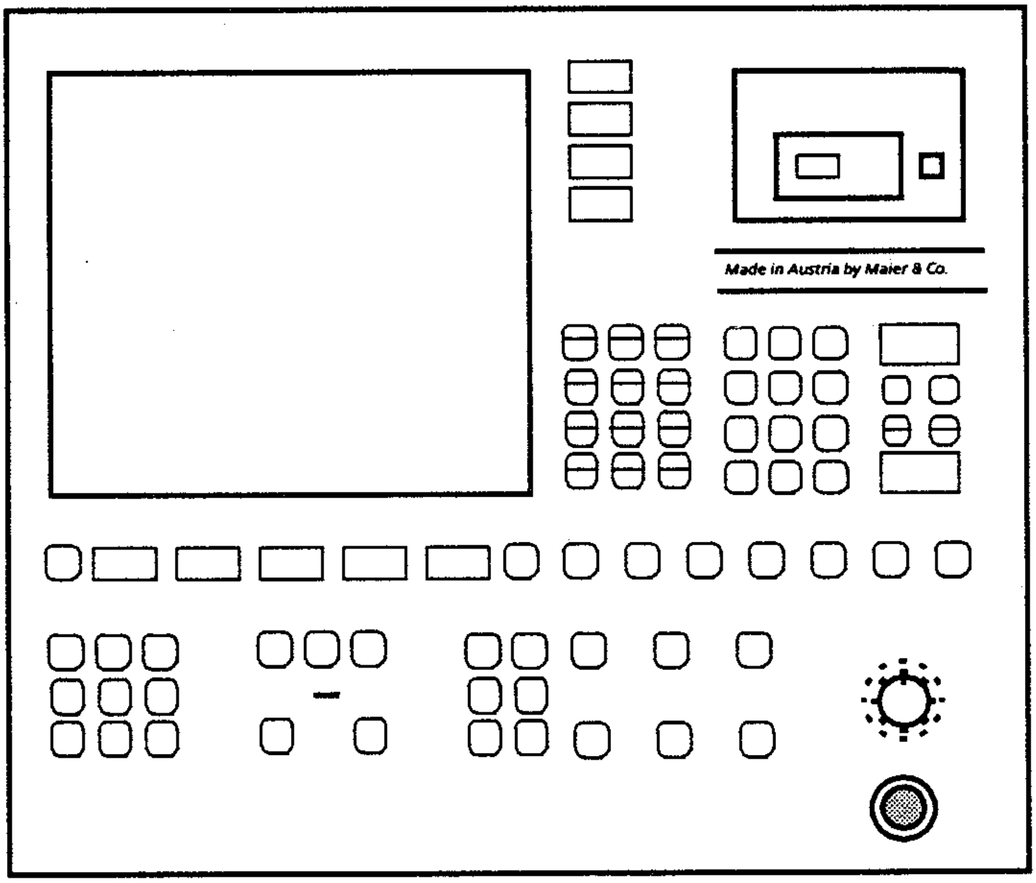
Chapter 8

Alarm list EMCOTRONIC TM 02

Chapter 1

Summaries, Technical Data

Front control panel EMCOTRONIC TM 02

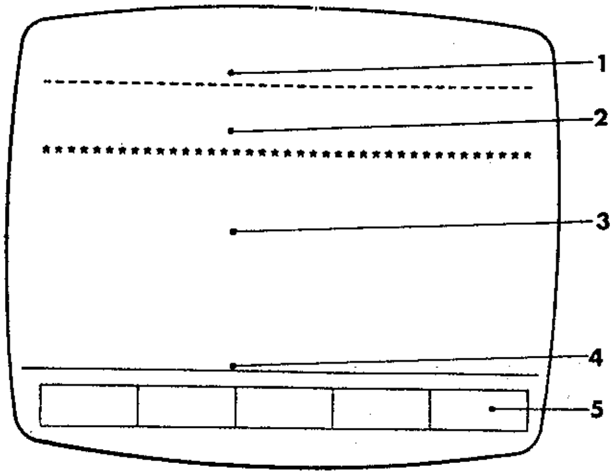


Division into 2 main groups

1. Screen
2. Control panel

1. The Screen

The screen is divided into 5 sections:



1) Information on

- main mode
- submodes
- display in mm or inch
- program number
- remarks
  COMPLETE
  NEW
  LOADING
  EXISTS
  SAVING
  DELETED

2) Alarm displays:

Complete list - see alarm messages.

3) Display and input field

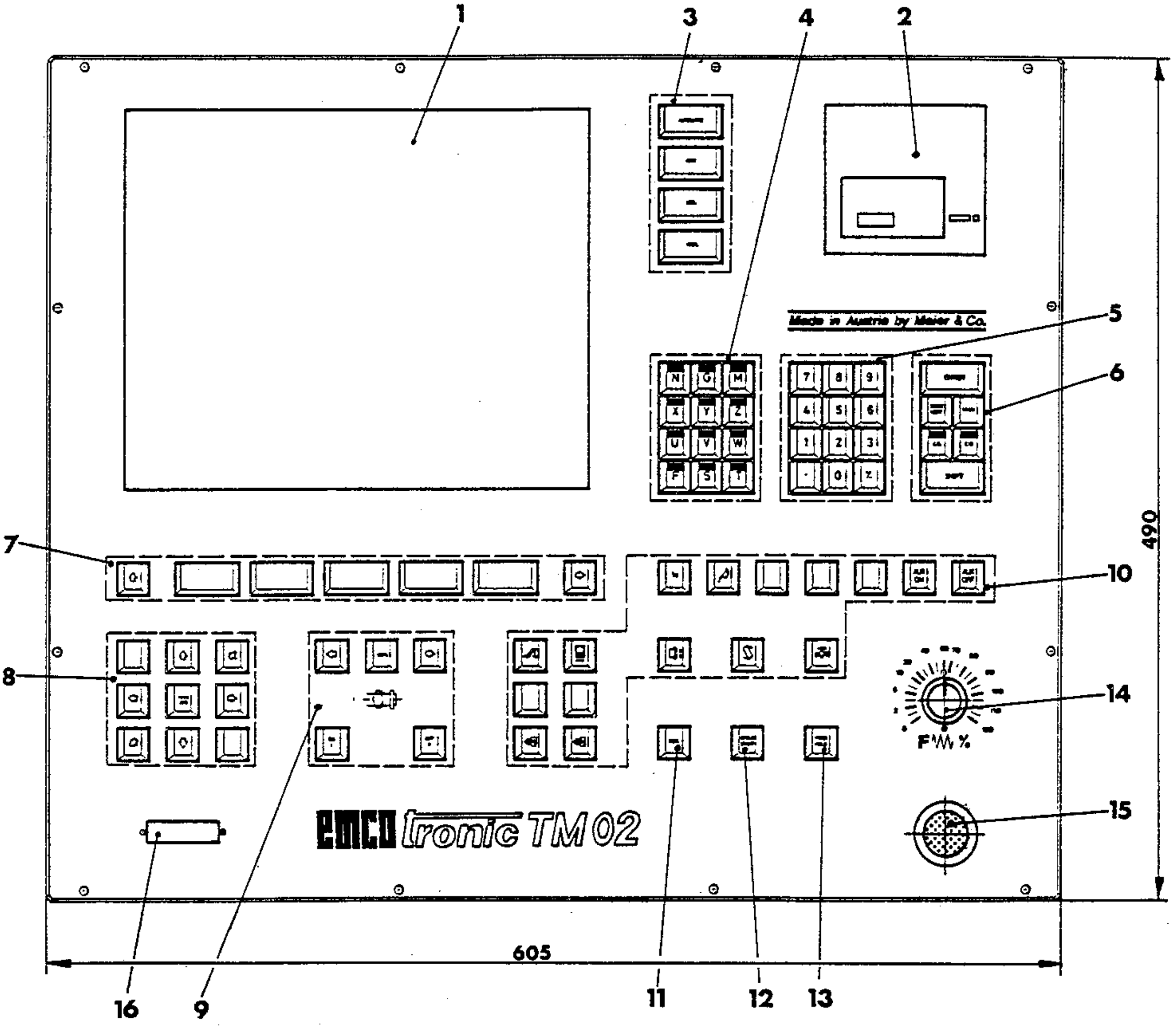
Contents are indicated in the modes.

4) - Buffer store in EDIT/EXC.

- Active block in AUTOMATIC mode

5) Softkey buttons - graphic simulation

2. Control Panel



1. Monitor
2. Cassette deck
3. Mode keyboard
4. Address keyboard
5. Digit keyboard
6. Function keyboard
7. Softkeys
8. MAN.JOG key
   (traversing in manual mode)
9. Speed override
10. Periphery keyboard
11. RESET key
12. CYCLE START key
13. FEEDHOLD key
14. FEED OVERRIDE switch
15. EMERGENCY-OFF key
16. Interface

Technical Data
EMCOTRONIC TM 02

Microprocessor 2-axis contour control
Linear and circular interpolation (2 1/2 D)
20 kB program memory (approx. 110 m tape)

Actual position
Distance left to traverse
Spindle speed
Tool compensation
Feed
Further parameters
12" monochrome monitor

Input accuracy	0.001 mm (0.0001 inch)
Output accuracy	≥ = 0.001 mm (0.0001 inch) (set conform to step resolution of the concerning machine, see "Technical Data of the Machine")

Thread pitch	0.01 - 10 mm
Feed override	0 - 120 %
Spindle speed override	50 - 120 %
Interpolation range	± 9999.999 mm
Tool memory	99 tools

Operating modes

Manual mode (manual slide movement)
Execute (processing the input memory)
Edit (program input via keyboard, interfaces, tool data and position shift register, operator monitor)
Automatic (execution of the NC programs)

Submodes

Single block, block skip, dry run, reference point, status, graphic

Program format

Structure according to DIN 66025
Decimal point input

Permanent program memory for machine data, tool data register and part programs, position-shift register

Data input/output

RS232C interface (V24 and 20 mA), 150 - 2400 baud
Tape recorder (Philips MDCR) 600 characters/sec (corresponding to 6 kbaud)

Technical data subject to changes and amendments!

Structure and Initial status of G-Codes

Group 0		G00: Rapid traverse
		G01: Linear-interpolation
		G02: G03: } Circular-interpolation
	*	G04: Dwell
		G33: Thread cutting in single block
	*	G84: Face and longitudinal turning cycle
	*	G85: Threading cycle
	*	G86: Grooving cycle
	*	G87: Drilling cycle with chip breaking
	*	G88: Drilling cycle with chip breaking and return to start point.
Group 1		G96: Constant cutting speed
	**	G97: Direct speed programming
Group 2		G94: Feed rate data in mm/min or 1/100 inch/min
	**	G95: Feed rate data in µm/rev. or 1/10.000 inch/rev.
Group 3	**	G53: Cancel workpiece zero point 1 and 2
		G54: Calling up workpiece zero point 1
		G55: Calling up workpiece zero point 2
Group 4		G92: 1. Speed limitation
	*	2. Changing of workpiece zero point coordinates in position shift offset 5 over NC-program.
Group 5	**	G56: Cancel workpiece zero point 3, 4 and 5
		G57: Calling up workpiece zero point 3
		G58: Calling up workpiece zero point 4
		G59: Calling up workpiece zero point 5
Group 6	*	G25: Subroutine call
	*	G26: Polygon call
	*	G27: Unconditional jump
Group 7	□	G70: Measurement data in inch
	□	G71: Measurement data in mm
Group 8	**	G40: Neutralization of the tool correction
		G41: Tool path correction left hand
		G42: Tool path correction right hand

* Effective block by block
** Initial status
□ Initial status in mode of operation MON can be determined

Structure and Initial status of M-Codes

Group		
Group 0	 ** 	M03 Spindle ON in clockwise rotation M04 Spindle ON in counterclockwise rotation M05 Spindle STOP M19 Precise spindle stop
Group 1	 **	M38 Precise stop ON M39 Precise stop OFF
Group 2	* * *	M00 Programmable intermediate stop M17 Subroutine end M30 Program end with return to program start
Group 3	 **	M08 Coolant ON M09 Coolant OFF
Group 5		M25 Open workholding tool M26 Tension workholding tool
Group 6		M20 Reverse tailstock sleeve M21 Forward tailstock sleeve
Group 7		M23 Reverse workpiece catcher M24 Forward workpiece catcher
Group 8	□ □	M50 Deselect direction logic of tool turret M51 Select direction logic of tool turret
Group 9	□ □	M52 Deselect of chip guard door automatic M53 Select of chip guard door automatic

* Blockwise effective
** Initial status
□ Initial status to determine in mode MON

Note:

If you have each M-code on your control depends on the hardware of your machine.

Addresses and their input dimensions

Addresses	metric	imperial
Path addresses absolute X, Z	± (mm)	± (inch)
Path addresses incremental U, W	± (mm)	± (inch)
Arc interpolation parameter I, K	± (mm)	± (inch)
1. F-thread pitch (G33, G85)	(µm)	(1/10000 inch)
2. F-minute feed (G94)	(mm/min)	(1/100 inch/min)
3. F-revolution feed (G95)	(µm/r)	(1/10000 inch/r)
1. S-direct speed programming (G97)	(rpm)	(rpm)
2. S-speed limitation (G92)	(rpm)	(rpm)
3. S-cutting speed (G96)	(m/min)	(inch/min)
4. S-exact spindle stop (M19)	[°]	[°]

The P-parameters in the program

Possible input: 0 - ± 10 000,000

Parameter		Default Option
P_0	G84: Taper dimension in X(U)(mm)	No taper dimension in X(U)
	G85: 1. Taper dimension on longitudinal threads $a < 45°$ (mm) 2. Thread angular runout on flat threads $a \leq 45°$ (mm)	No taper dimension
P_1	NOT USED	-----
P_2	G84: Taper dimension in Z(W) (mm)	No taper dimension in Z(W)
	G85: 1. Thread angular runout on longitudinal threads $a \leq 45°$ (mm) 2. Taper dimension on flat threads $a \leq 45°$ (mm)	Straight taper runout
P_3, P_4, P_5, P_6, P_7	NOT USED	-----

The D-parameters in the program

Possible input values: 0 - 32 767

Parameter		Default Option
D_0	G84: Allowance in X(U)(mm)	No allowance in X(U)
D_1	NOT USED	-----
D_2	G84: Allowance in Z(W)(µm)	No allowance in Z(W)
D_3	G84: Cutting division (µm)	No cutting division
	G85: Mode parameter (µm),()	------
	G86: Feed per cut (µm)	No feed per cut
	G87: Drill depth of 1st cut (µm)	No cut division
	G88: Drill depth of 1st cut (µm)	No cut division
D_4	G04: Dwell time (1/10 s)	No dwell time
	G85: Number of empty cuts ()	Number of empty steps specified in the operator monitor
	G86: Dwell time (1/10 s)	No dwell time
	G87: Dwell time (1/10 s)	No dwell time
	G88: Dwell time (1/10 s)	No dwell time
D_5	G85: Angle of thread (°)	Recessing feed
	G86: Tool width (µm)	-----
	G87: Percentage of cutting depth reduction (%)	No cutting depth reduction
	G88: Percentage of cutting depth reduction (%)	No cutting depth reduction
D_6	G85: Thread depth (µm)	---
	G86: Minimum drill depth (µm)	No minimum drill depth
	G87: Minimum drill depth (µm)	No minimum drill depth
D_7	G85: Mode parameter ()	See G85

Note the D-parameters in the operator monitor (MON).

Chapter 2

General notes on programming

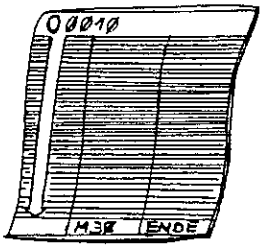
Program structure of the EMCOTRONIC TM 02

Program structure of the EMCOTRONIC TM 02 according to DIN 66025 and ISO 1056

Program:
A CNC-program contains all instructions and information required for the production of a workpiece.
A program consists of:
* Program start
* Program content
* Program end
On the EMCOTRONIC T1, we differentiate between main, subroutine and polygon programs.

MAIN PROGRAM

O10



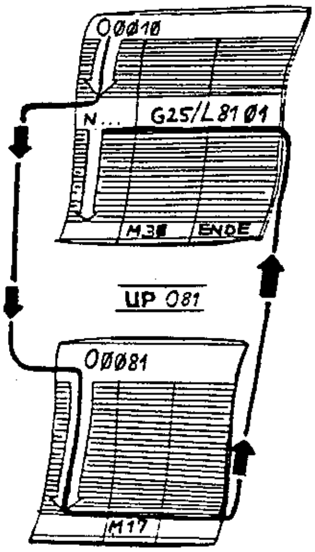
Structure main program:

1. Program start.
   The start of the program is the program number. The program number is specified under the address 0 (letter O).
2. Program content.
   NC-blocks.
3. Program end.
   M30

Specifications for program numbers for main programs:
Possible program numbers 0 0000 to 0 6999; Numbers can be specified in the operator monitor (MON), which are only permitted for subroutines. Program numbers 0 0080 to 0 0255 have been specified as subroutine numbers in the factory. For details see the operator monitor (MON).

MAIN PROGRAM

O10



Structure subroutine:

1. Subroutine start.
   The start of the program is the program number. Possible details from L_3 to O 0255. The lowest possible subroutine number is specified in the operator monitor with O_{22}. ALARM 630 is raised where a program number is not between O_{22} and O 0255.
2. Program content.
3. Program end.
   M17

Polygon programs:
The program numbers O 7000 to O 9999 are specified for polygon program graphic simulation. For details see graphic simulation.

The program blocks, NC-blocks

Address: N
Possible block numbers N 0000 to N 9999.
A block consists of the block number and the words. The words form the content of a block.
It is useful to number the program blocks in tens. Blocks can be subsequently inserted, without affecting the other program.
During input, grading of the program blocks in tens is automatically proposed by the control.

O 00 15

N 00 00
N 00 10
N 00 20

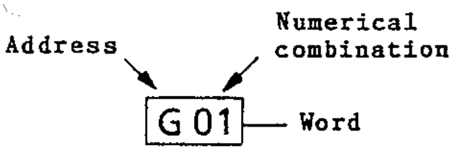
The words:

A block usually consists of several words.

N0010/G01/X40,000/Z5,000/F 120

The word:

A word consists of one letter (address) and a numerical combination. Each address has a specific meaning, according to which the associated numerical value complies. The addresses and their meaning are specified in the programming instruction.



Syntax specifications

Block length:

The maximum block length can vary between 3 and 4 lines according to the programmed words. ALARM 650 is raised where the block length is exceeded.
To achieve clarity in the program structure, a logical structure is recommended.

- Specifications on the sequence of words:
  Apart from the X(U), Z(W) sequence in the cycles G84, G85, G86, there is no absolute rule on the word sequence. However, so as to obtain a clear program structure, you should observe the following sequence:

  * Each block starts with the block number.
  * The G-function should be programmed after the block number.
  * Words for the co-ordinates X(U), Z(W). Observe the reversal of the X(U), Z(W) sequence in the cycles G84, G85 and G86.
  * Where G02, G03 is programmed, the interpolation parameter I, K should be programmed after X(U), Z(W).
  * Where cycles are programmed, the parameters should be programmed after the X(U), Z(W) addresses.
  * The F-word (feed thread pitch).
  * The S-word (spindle speed, cutting speed).
  * The T-word (tool address).
  * The M-word (additional functions).

- Several G- and M-functions of the same group:
  Where two or more G- or M-functions of the same group are in one block (not sensible), the last programmed function is effective.

- Same words in one block; apart from G- and M-words
  The last input word applies.

- Same G- and M-words of the same group in one block:
  With G- and M-words of the same group, the last input word applies.

- Decimal point programming:
  X, Z, U, W, P_0, P_2, I, K values must be programmed with the decimal point.
  Without a decimal point, the values would be considered as µm (on G71) or as 1/10000 inch (G70).
  Leading and following zeroes do not have to be programmed.

  Additional notes:

- Specifications upon tool call and when calling position shift registers:
  The first traverse command after tool call, and when calling a position shift register, must be a G00 command.
  Further information is given in the programming remarks and in the specification of the individual G-commands.

  The individual addresses and their meaning, the associated possible input dimensions, possible plus-minus inputs of numerical values can be found in the address summary. Detailed specifications are given in the following chapters.

Brief specification of the addresses:

X and Z path addresses:

The target points in the absolute co-ordinate system are specified with X and Z. The origin of the co-ordinate system is M (machine zero point) or a point W (workpiece zero point) specified by you.

The X-dimension is given as a diameter (factory setting). With parameter L_0, Bit 0, X-programming, you can also set the radius in the operator monitor.

U and W path addresses

Paths are incrementally specified with U and W..

I and K addresses:

I and K are interpolation parameters for arc programming. For a precise specification see G02, G03.

F-feed:

1) F in conjunction with G94.
Under the F-address, the feed is programmed as feed speed in inch/min (mm/min). For the input dimension see summary of addresses and their input dimensions.

2) F in conjunction with G95.
The feed is specified in inch/rev. or mm/rev.
For the input dimensions see summary of addresses and their input dimensions.

3) F in conjunction with G33 and G85.
The thread pitch in inch or mm is programmed under F.
For the input dimensions see summary of addresses and their input dimensions.

S-address:

1) S in conjunction with G96.
The cutting speed in inch/min (mm/min) is programmed. Input dimensions see summary.

2) S in conjunction with G97.
The spindle speed in rev./min is programmed.

3) S in a block with G92.
The upper speed limit is programmed.

4) S in a block with M19
The position stop of main spindle is programmed.

T-address:

The tool (tool changer position) and the tool data is called with the T-word.
See T-address for a precise specification.

M-function:

Switch and additional functions are called with M.
See M-function for a precise specification.

L-address:

1) With L, subroutines are called, repeats programmed and skip targets are specified.
See G25/M17, G27, G26 polygon programs for a precise specification.

2) L in tool register.
The cutter positions of the tools are recorded under the L-address.
See G40/G41/G42 for a precise specification.

R-address:

The nose radii of the tools are specified under R.
See G40/G41/G42 for a precise specification.

The P-parameters and the D-parameters:

Special types of execution are programmed on cycles with the P- and D-parameters.
Details are given under the specification of the particular cycles.

G-function:

Path conditions are called with G.
See G-functions for a precise specification.

O-address:

Numbers for NC-programs are specified with the O-address. These program numbers are used for recognition, e.g. of programs stored on cassette, and for marking the start of the program.

Division of the O-adress:

Main program numbers: O 000 to O 6999

Subrotine numbers: O_{22} to O 0255

The lowest valid subroutine number can be specified with O_{22} = 80, which is the factory setting.

Polygon program numbers: O 7000 to O 9999.

Tool data memory:

The tool data is input incrementally in the tool data memory with the X, Z-addresses (see tool programming).

Position shift register (PSO):

1. Direct incremental input in the position shift register with X (= radius dimension) and Z.

2. Details for position shift register 5 in G92 block: The shift dimensions are incrementally specified with X and Z (X = radius dimension). Upon activation, the X and Z values erase the old values in position shift register 5. The new X and Z values are stored.
   Where shift dimensions are given with U and W in the G92 block, upon activation these U and W values are added resp. subtracted (see zero point shift) to the old values in position shift register 5.

Skip-Blocks

For some cases (trial cut, serial production) it is quite useful that blocks can be skipped.

Skipped blocks are marked with a diagonal stroke (slash). It has to be put in after the block number.

```
N  90 G00 X20, Z30,
N100  /   M00 → skip block
```

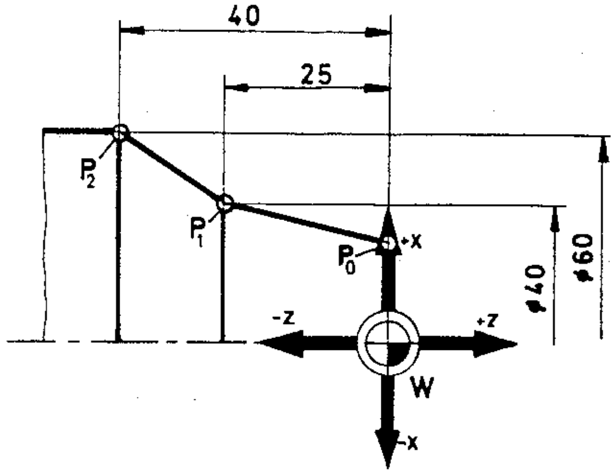
Key /

Sequence in program:

SKIP Key pressed:
Skip blocks will be not executed.

SKIP Key not pressed:
The skip blocks will be executed.

Absolute and Incremental Value programming



Absolute Value Programming

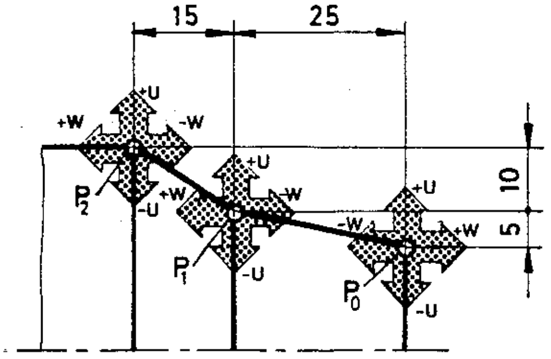
The description runs under the addresses

X, Z

The X and Z data always relate to the actual origin of the coordinates system.

Example:

$\rightarrow P_0$ N..../....
$P_0 \rightarrow P_1$ N..../G01/X40,000/Z-25,000/F....
$P_1 \rightarrow P_2$ N..../G01/X60,000/Z-40,000/F....
$P_2 \rightarrow$ N..../....



Incremental Value Programming

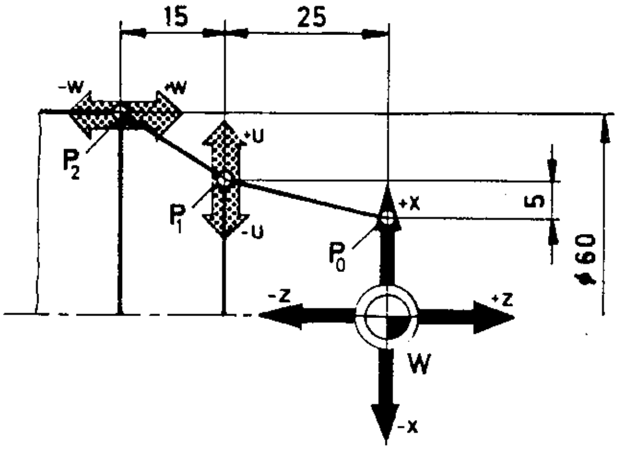
The description runs under the addresses

U, W .

The U and W addresses refer to the starting point of each block.

Example:

$\rightarrow P_0$ N..../
$P_0 \rightarrow P_1$ N..../G01/U5,000/W-25,000/F....
$P_1 \rightarrow P_2$ N..../G01/U10,000/W-15,000/F....
$P_2 \rightarrow$ N..../



Mixed Programming

The programming can also be mixed.

Example:

$\rightarrow P_0$ N..../
$P_0 \rightarrow P_1$ N..../G01/U5,000/Z-25,000/F....
$P_1 \rightarrow P_2$ N..../G01/X60,000/W-15,000/F....
$P_2 \rightarrow$ N..../

G-Codes, their Formats and Description of Formats

Specific addresses are asigned to most G-Codes.

Example:

G00/X ± ....,.../Z ± ....,...

oder

G01/X ± ....,.../Z ± ....,.../F....

For a short and easy to understand description of pertaining addresses (format description) the data are encoded.

Code:

1) Instead of giving the possible inputs, the number of decades is given.

   Example:

   Instead: N from 0 to 9999
   or N .... we write N4.

   $\underset{4}{N....} \rightarrow \underline{N4}$

2) The specification of the possible decades before or after a decimal point is coded with two figures.

   $X\underset{4}{....},\underset{3}{...} \rightarrow X43$

   The first figure: Decade before decimal point
   The second figure: Decade after decimal point

3) If the values could be negative or positive a + sign is written between address and number.

   X + 43

   Remark:
   For better determination quite often a ± sign is written (X±43).

Example:

N4	G01	X U ± 43	Z W ± 43	F4
		[mm]	[mm]	[µm/U] [mm/min]

N4: Four digits without decimalpoint and sign.

X, Z ± . . . . , . . .

U, W ± . . . . , . . .

± sign possible	Four digits before decimal point	Three digits after decimal point

F4: Four digits without decimal point and sign.

Example:

N4	G02 G03	X U ± 43	Z W ± 43	I ± 43	K ± 43	F4
		[mm]	[mm]	[mm]	[mm]	[µm/U] [mm/min]

Example:

N4	G04	D4 5
		[1/10 s]

D45 Five digits without decimal point and sign.

Self-holding functions, Words

The majority of the G- and M-functions and other words are self-holding. That is they remain active until they are overwritten or deselected.
This implies a simplification and reduction for the program.

1. Self-holding functions, words, word contents within a program

G- and M-functions

The G- and M-functions are divided into groups.

The self-holding G- and M-functions remain active until they are overwritten by another G- or M-function from the same group (see group division G/M-functions).

Some G- and M-functions can also be directly deselected.

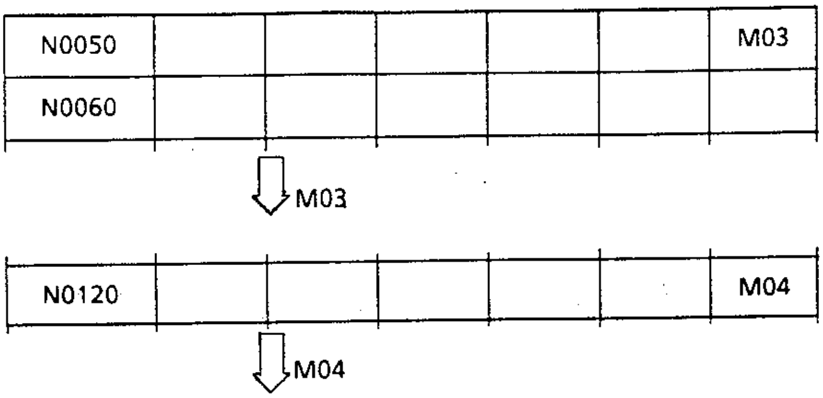
Deselection:
G54, G55 are deselected with G53.
G57, G58, G59 are deselected with G56.
G41, G42 must be deselected with G40.
M30 automatically causes end of program and
M05 spindle STOP
M09 coolant OFF
M23 collection tray BACKWARD

Example:
Acceptance of G00 in block N0110.
In block N0120, G00 is deselected with G01. G01 is active.

N0100	G00	X50,000	Z + 10,000	
N0110	$\boxed{\text{G00}}$	X36,000	Z + 2,000	
N0120	G01	X40,000	Z-10,000	F...

Example:

M03 is activated in block N0050. M03 is active in blocks N0050 to N0120. M03 is deselected by M04 in block N0120. M04 is active from block N0120.



Acceptance of words and word contents

X(U), Z(W), F, S, T word contents are accepted in the following blocks.
The contents are overwritten by programming another word.

Example:

In blocks N0050, N0060 and N0070, X, Z, F, S and T words with the same content are accepted.

N0040	G01	X40,000	Z10,000	F 120	S1500	T0303
N0050	G01	X35,000	$\boxed{\text{Z10,000}}$	$\boxed{\text{F 120}}$	$\boxed{\text{S1500}}$	$\boxed{\text{T0303}}$
N0060	G01	$\boxed{\text{X35,000}}$	Z-18,000	$\boxed{\text{F 120}}$	$\boxed{\text{S1500}}$	$\boxed{\text{T0303}}$
N0070	G00	X48,000	$\boxed{\text{Z-18,000}}$	—	$\boxed{\text{S1500}}$	$\boxed{\text{T0303}}$

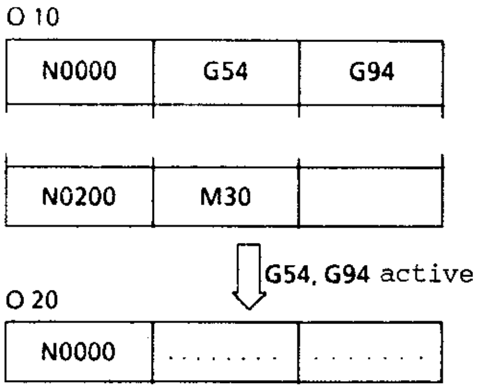
2. Acceptance of G-, M-functions and words in the following programs

G-, M-functions

All self-holding G-functions, apart from those of group 0, are also accepted in the next program.

Example: Acceptance of G-functions

The active G-functions G54, G94 in program 0 10, are accepted in program 0 20.



O 10

N0000	G54	G94

N0200	M30	

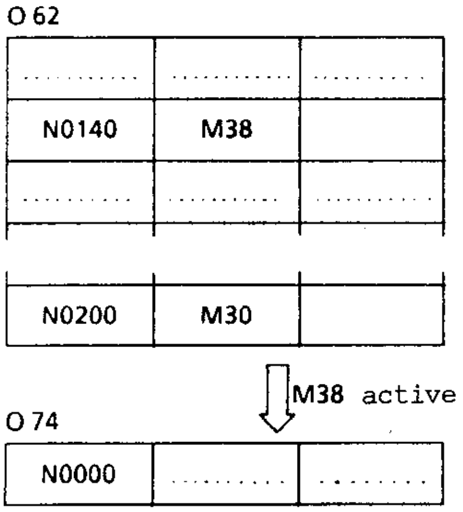
G54, G94 active

O 20

N0000	....	....

Example:

The M-function M38 active in program 0 62, is accepted in program 0 74.



O 62

....	....	....
N0140	M38	
....	....	....

N0200	M30	

M38 active

O 74

N0000	....	....

Words

F, S, T words are accepted in the following program.

Example:

The F, T, S words and their contents contained in blocks N0120, N0130 are activated in program O 20. At the end of the program, they are still effective and are accepted in program O 30.

Other words and parameters are not accepted in another program.

O 20

N0000				
N0120				F 150
N0130	T0303			
N0140				S1500
N0230	M30	-----	-----	-----

⇩ F 150, T0303, S1500 active

O 30

N0000				

Note:

When the STATUS submode is selected, you can see the active G, M, F, S, T conditions.

3. Acceptance from the EXECUTE operating mode:

All self-holding G- and M-functions, apart from the G-function of group 0, which were activated in the EXECUTE mode, remain active in the subsequent programs.

The actuation condition of the EMCOTRONIC TM 02

The control manufacturer specifies the actuation condition. Some conditions can be changed by the operator. The specification criterium are based on practical and technical safety requirements.

Example:
M05: When the control is switched on, the spindle may not accelerate.

The actuation condition is active when the control is switched on, not with EMERGENCY OFF or RESET.
The actuate condition in the STATUS mode is shown.

Actuation condition of the EMCOTRONIC TM 02

GROUP	0	1	2	3	4	5	6	7	8
G		97	95	53		56		71	40
M	05	39		09		25*	20*	23*	51*

ACT. F.: 00000
ACT. S.: 0000
ACT. T.: 0000

- On USA machines, G70 is set instead of G71.
- The M-functions marked (*) are only active on the appropriate hardware variants.

* M-functions - Group 5/6
Group 5: Chuck commands M25, M26
Group 6: Tailstock commands M20, M21

The following applies for these M-functions of groups 5 and 6:
The commands active prior to disconnection, are active after the control is switched on.

Actuation conditions which can be defined on the operator monitor:

1. By means of parameter O_{11} bit 0, it can be specified whether G70 or G71 is active after switching on the system.
   O_{11} bit 0 = 0 (low) ..... G71 - programming in mm
   O_{11} bit 0 = 1 (high) .... G70 - programming in inch

2. By means of parameter O_{11} bit 3, it can be specified whether M50 or M51 is active upon switching on.
   O_{11} bit 3 = 0 (low) ..... M50 - deselect the direction logic of the tool changer
   O_{11} bit 3 = 1 (high) .... M51 - select the direction logic of the tool changer

Programming notes
(in preperation)

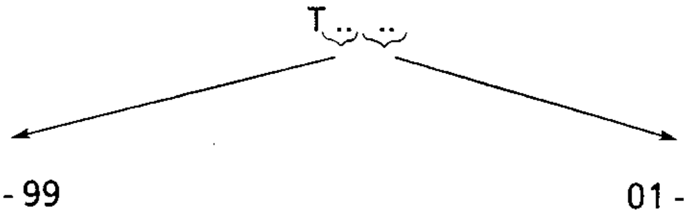
Chapter 3

Tool programming
Zero point shift

Tool programming

Tool are programmed under the T-address with 4-digit numbers.

Key



1. Tool number

Number of the tool space on the tool changer, tool only with quick-change toolholder.

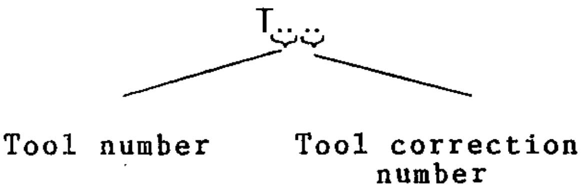
2. Tool correction number

The tool correction numbers are listed in the tool data memory.

Explanations of the T-address:

1. Tool number

The first two digits of the T-address are tool numbers. The tool number indicates the position of the tool in the tool changer. When the T-address is called, the tool changer moves to the position called.



2. Tool correction number

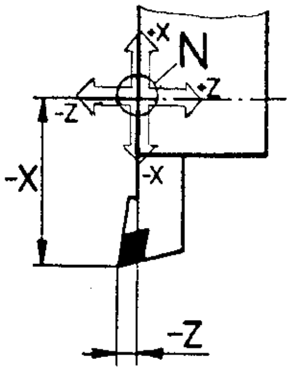
Tool correction values are stored under a correction number in the tool memory.

Example:

1 in tool data memory → correction T..01

20 in tool data memory → correction T..20

The tool compensation values



Tool Data:

The tool data X,Z are measured from point N. Imagine the coordinates system in point N. The tool lengths are gauged from point N. These measures are entered into the tool.

The cutting edge radius R

In addition to the tool data X,Z, the radius must also be input in the tool file.

The cutting edge position L

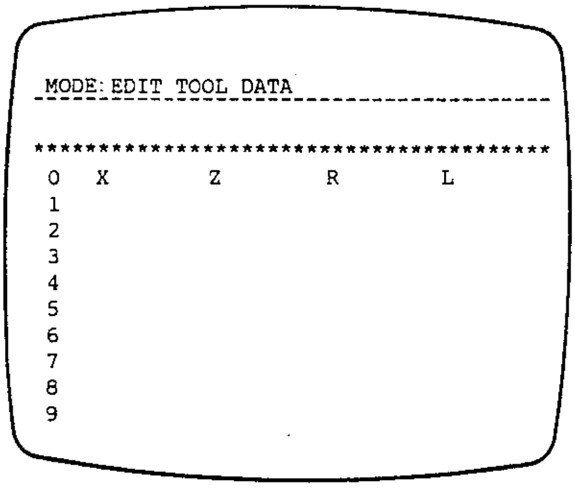
Input of the cutting edge length in the tool data memory under address L.

Acquisition of the tool data

Several methods, such as acquisition of the tool data with a caliper, marking a known diameter and a front face, as well as working methods with the optical presetting device, are comprehensively specified in the operating manual of the particular machine.

Input of the R- and L-address values

When working with G40, G41, G42 (cutter radius compensation), R- and L-address values must be programmed. A comprehensive specification is given in the chapter on cutting edge radius compensation G40, G41, G42.



```
MODE: EDIT TOOL DATA
--------------------------------------------
********************************************
0  X          Z          R          L
1
2
3
4
5
6
7
8
9
```

Tool data memory

The tool correction values are input in the EDIT operating mode in the tool data memory.

 1 complies with correction T .. 01
20 complies with correction T .. 20

Tool data: under X-, Z-address
Cutting edge radius: under R
Cutting edge length: under L

Accounting:
Where a tool is called with tool correction in the program, the control obtains the data X, Z (R, L) that has been input under the code number.

Calling the T-address

1. Syntax specifications

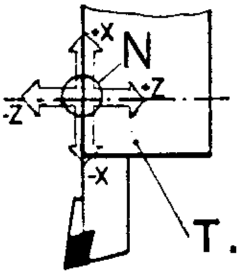
Every new T-address has to be called-up with a G00 block (otherwise Alarm sign).

Example: Call-up in same block with G00

```
N0090/M00
N0100/G00/X.../Z.../T0202
```

Example: After the T-call-up a G00 traverse instruction follows.

```
N0100/T0202
N0110/G94/F130
N0120/G54
N0130/G00/X.../Z.../
```



2. Tool correction not called up

T .. 00

When the key number T .. 00 is programmed, the dimensional system refers to the tool mounting reference point N. The tool changer moves into the position of the called tool, the tool correction is not taken into account by the control, a possibly, previously active correction is deselected.

Programming notes

1. Specification of the correction numbers

The correction number and tool number need not be identical, e.g. T05 01. For clarity, it is useful for the correction numbers to be identical with the tool numbers. For example: T03 03.

2. Deselection of the tool correction prior to traversing to the tool change point:

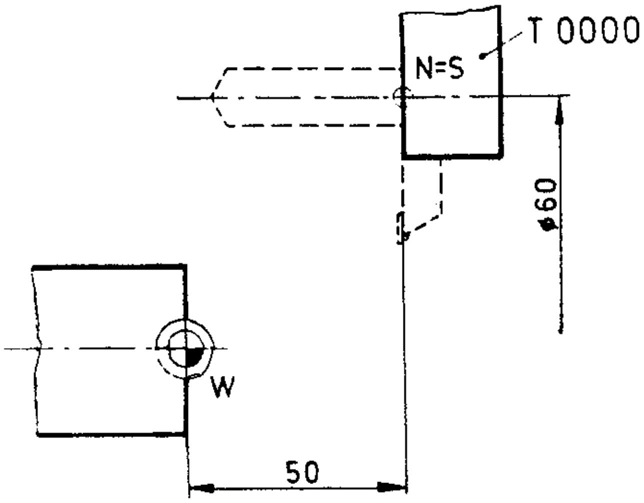
The tool correction of the active tool should be deselected prior to the return movement to the tool change point. The traverse path of the tool becomes shorter, since point N (tool mount reference point) is approached and not the cutting tip of the tool as with the active tool correction. In this way, it is possible to prevent the maximum traverse paths from being exceeded (see the examples).

Note:

The tool changer should be swivelled through in the MANUAL mode, so as to prevent collisions.

Example 1

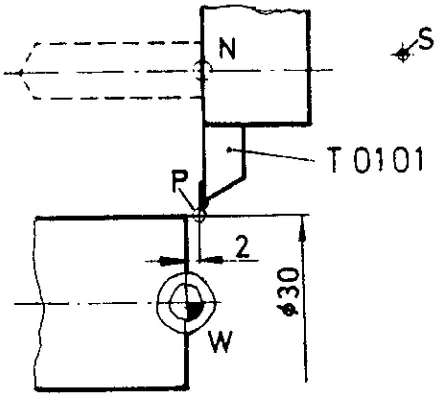
Approaching the tool change point with active tool correction



1. No tool active

When no tool is active, the coordinates of point N (tool mount reference point) are taken into account.

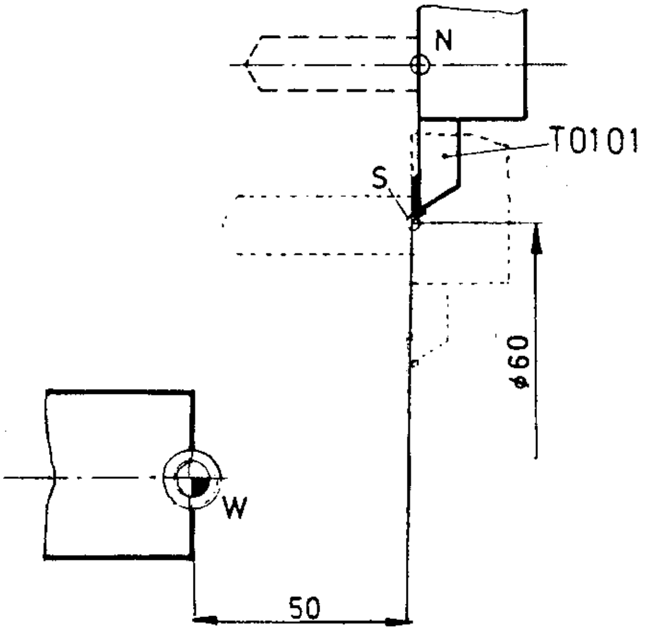
N..../T0000/G00/X60,000/Z50,000



2. Traversing with active tool correction

T0101 is active, the cutting tip of the tool traverses to the programmed point D.

N..../T0101/G00/X30,000/Z2,000



3. Traversing to the tool change point with active tool correction

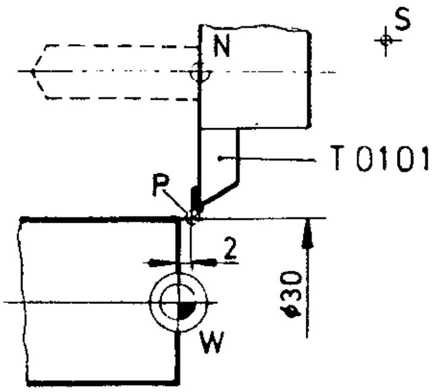
T0101 is active. Point S is approached by the tool tip of the tool.

N..../T0101/G00/X60,000/Z50,000

Note:
To keep the traverse path shorter, you should deselect the tool correction prior to approaching the tool change point (see example 2).

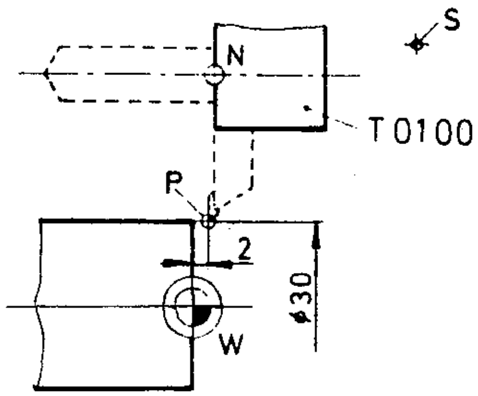
Example 2

Approaching the tool change point with deselected tool correction



1. Traversing with active tools

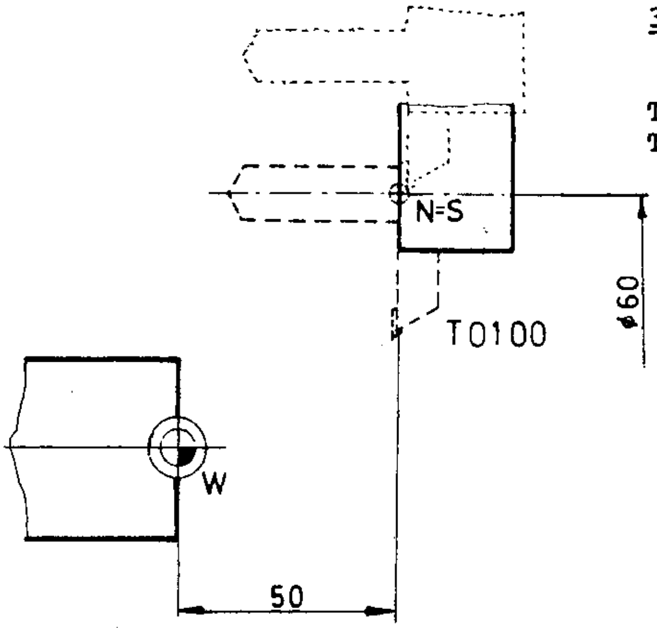
T0101 is active. Point P is approached by the tool tip.



2. Deselection of the tool correction

The tool correction is deselected = T0100

N..../T0100



3. Traversing to the tool change point with deselected tool correction

The tool correction is deselected = T0100, point S is approached.

N..../T0100/
N..../G00/X60,000/Z50,000

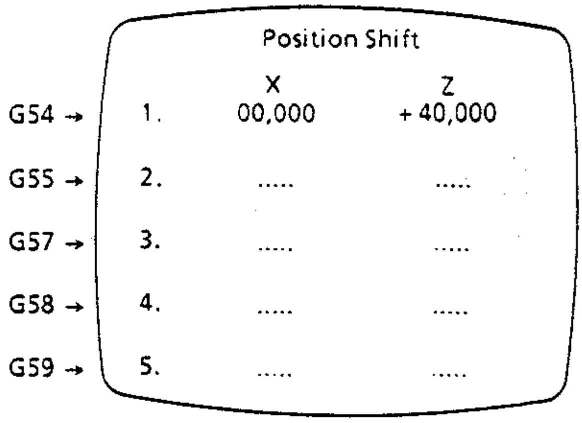
Zero point shifts

The origin of the co-ordinate system can be shifted to a position selected by you.
Through a call command, shift values that have been previously input in the position shift register are activated.

Position shift register

The dimensions for zero point displacement with correct prefix are input in the position shift registers 1 - 5.

Call commands Shift register 1 - 5



Position Shift

		X	Z
G54 →	1.	00,000	+ 40,000
G55 →	2.	.....	.....
G57 →	3.	.....	.....
G58 →	4.	.....	.....
G59 →	5.	.....	.....

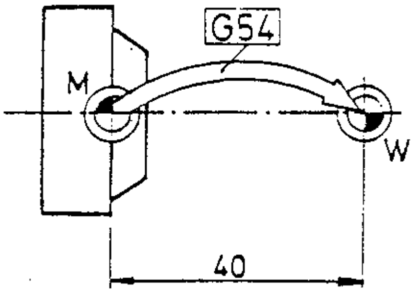
The position shift registers 1 - 5, also called **PSO (Position Shift Offset)**, are specified on the following pages.

Call commands

Where a call command is programmed in a CNC-program, the coordinate system is offset by the amount in the shift register.

Example

N..../G54/....



Input possibilities

1. Manual input in the position shift registers 1-5
   See EMCOTRONIC TM 02 operating manual, chapter 4-EDIT

2. Reading in position shift register data from cassette in the machine
   See EMCOTRONIC TM 02 operating manual, chapter 4-EDIT INTERFACE

3. Special event:
   G92 - Details for position shift register 5 in NC-program
   G59 - Activation of the shift values of position shift register 5

Call and deselection of the position shift register

Group division of the commands

Group 3	G53	Cancellation of G54, G55

	G54 = 1
	G55 = 2
		Call position shift register (PSO) 1, 2

Group 5	G53	Cancellation of G57, G58, G59

	G57 = 3
	G58 = 4
	G59 = 5
		Call position shift register (PSO) 3, 4, 5

Group division and shift/cancellation of a shift

Several commands of the same group in one program:

The last programmed command always applies.
The previous command is cancelled by the next (see examples).

Two commands from various groups:
Commands from various groups are added vectorially. (They do not cancel each other out! See examples).

Shift deselection
G53 cancels G54 and G55
G56 cancels G57, G58 and G59.

Syntax specifications:
The shift commands must be programmed in conjunction with a G00 block.

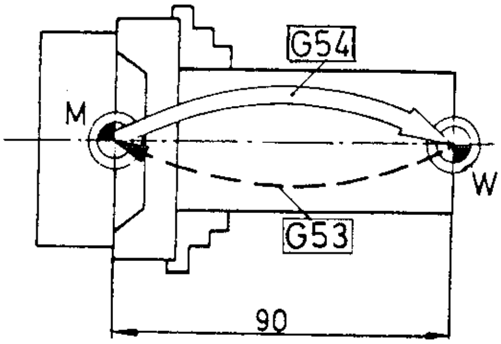
Possibility 1
In the same block as G00

```
N0100/G00/X.../Z.../G54
```

Possibility 2
The following traverse command is a G00 block.

```
N0100/G54
N0110/G94//F 120
N0120/G00/X.../Z...
```

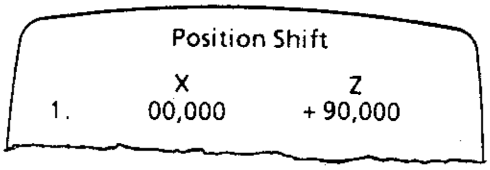
Examples G53 - G59



Example 1:
Call a zero point shift with G54 in Z-direction.
Deselect the zero point shift with G53.

N..../G54/.... call PSO 1 - shift from M to W

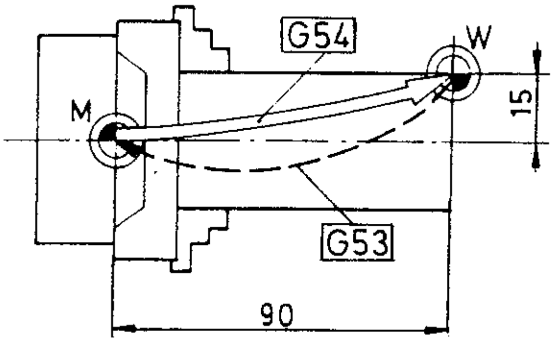
Activate ⇩



Position Shift

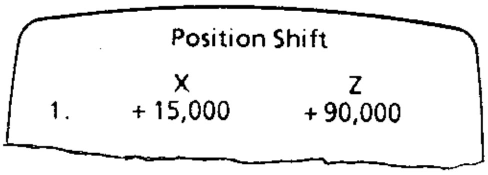
	X	Z
1.	00,000	+90,000

N..../G53/.... deselect PSO 1



Example 2:
Call a zero point shift with G54 in X- and Z-direction (X = radius dimension)
Deselect the zero point shift with G53.
N..../G54/....call of PSO 1 - shift from M to W

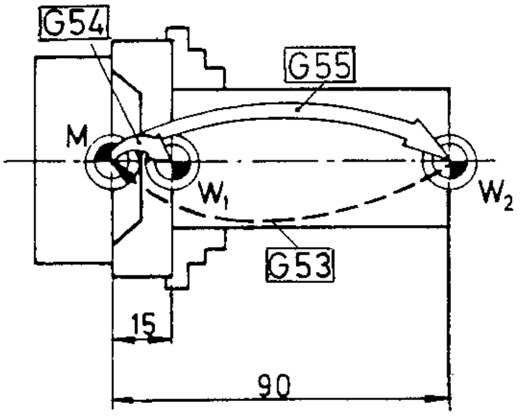
Activate ⇩



Position Shift

	X	Z
1.	+15,000	+90,000

N..../G53/.... deselect PSO 1

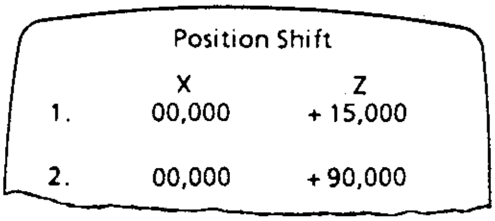


Example 3:
Call two zero point shifts of the same group with G54 and G55. The last shift called from the same group is effective.
Reset the zero point shift with G53.

N..../G54/.... call PSO 1 - shift from M to W1

N..../G55/.... deselect PSO 1 - shift from W1 to M
call PSO 2 - shift from M to W2

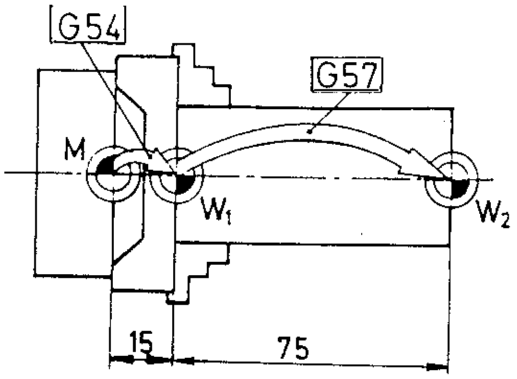
Activate ⇩



Position Shift

	X	Z
1.	00,000	+15,000
2.	00,000	+90,000

N..../G53/.... deselect PSO 2



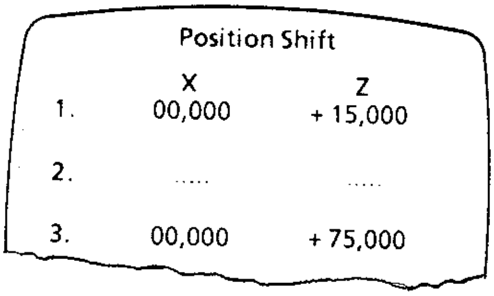
Example 4:

Call two zero point shifts of different groups with G54 and G57.

N..../G54/.... call POS 1 - shift from M to W1

N..../G57/.... call PSO 3 - shift from W1 to W2

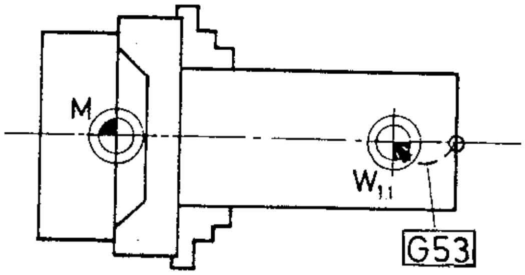
Activate



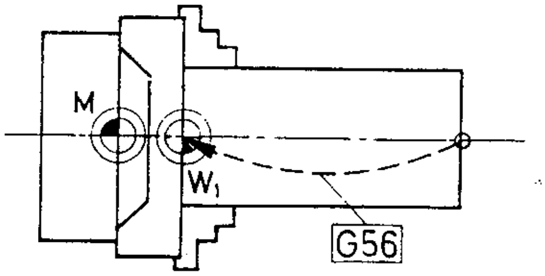
Position Shift

	X	Z
1.	00,000	+ 15,000
2.	.....	.....
3.	00,000	+ 75,000

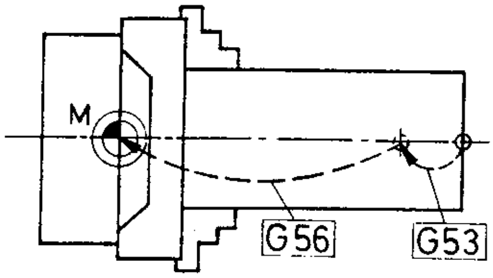
Note on deselection of zero point shifts of several groups:
Note the group relationship of the reset commands G53 and G56, and ensure that all shifted groups are deselected.



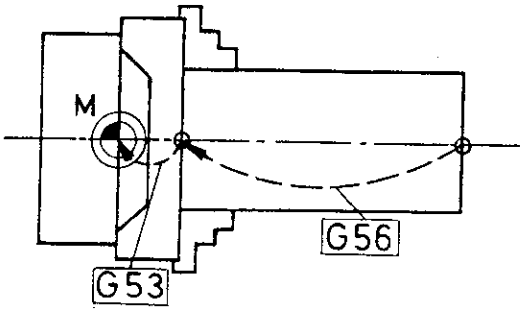
Deselection of PSO 1 (and PSO 2) with G53



Deselection of PSO 3 (PSO 4 and PSO 5) with G56



Deseleciton of PSO 1 and PSO 3 with G53 and G56



Deselection of PSO 3 and PSO 1 with G56 and G53

Special event

G92 - Details for position shift register 5 in NC-program

G59 - Activating the shift values of position shift register 5

Specifications:

Programming the shift values

The shift dimensions are recorded under G92 in the parts program.

Example:

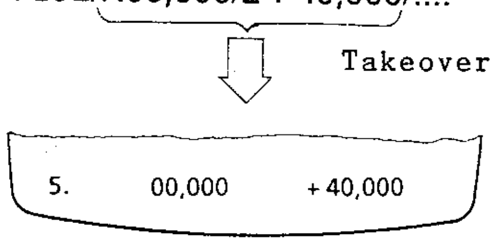
N..../G92/X00,000/Z+40,000

Activating the shift

During the program sequence, the shift dimensions are taken over in position shift register 5 (PSO 5).
The shift is carried out with G59.

Example:

N..../G92/X00,000/Z+40,000/....



Where a G59 occurs in one of the additional NC-blocks, the shift is carried out.

N..../G59

Syntax:

G59 cannot be programmed in the same block with G92, but must be programmed in the following blocks.

When G59 is active when a G92 block is reached, Alarm 700 is raised.

Where G59 is programmed together with G54 or G55, both shifts are added.

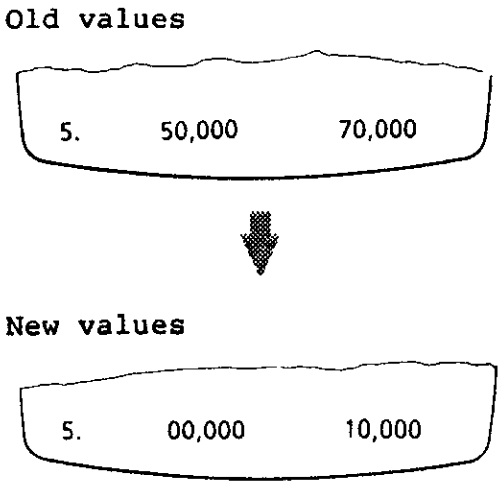
Deselection:

Deselection in the program takes place with G56.

Types of dimension input G92

1. Absolute values

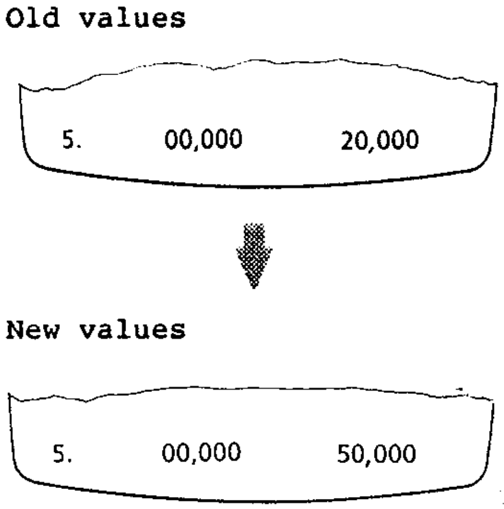
Where the shift dimensions are specified with X, Z under G92, the old values are erased on position shift register 5, and the G92 values are active.



N0100/G92/X00,000/Z10,000

2. Incremental values

Where the shift dimensions are specified under G92 with U, W, the U, W dimensions are added to the dimensions of position shift register 5.

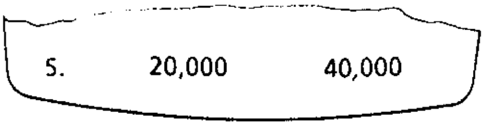


N0100/G92/U00,000/W30,000

Note:

Where shift values are input incrementally, these values are added during the repeat sequence of the program to the existing values in the position shift register 5.

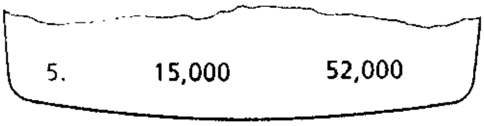
Old values



5. 20,000 40,000

↓

New values



5. 15,000 52,000

3. Mixed values

Where the dimensions are mixed under G92, that is are absolutely given with X, Z and incrementally with U, W then

- * the absolute G92 dimensions are taken over in the register,
- * the incremental G92 dimensions are added to the values of the position shift register 5.

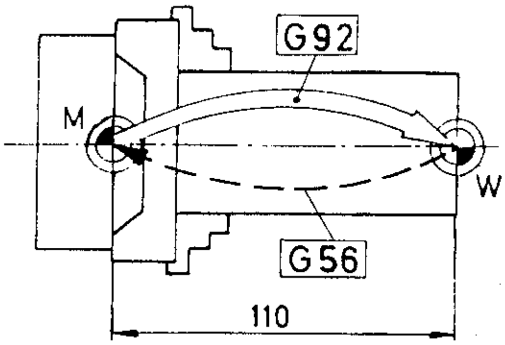
N..../G92/X30,000/W + 12.000/

Note:

Diameter values from the NC-program appear as a radius value in position shift register 5.

Exception:
Radius programming active.

Examples G92



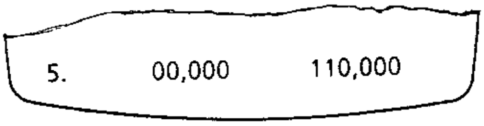
Example 1:

Input of a zero point shift with G92 in Z-direction.
Activate with G59.
Deselect with G50.

Call

N..../G92/X00,000/Z110,000

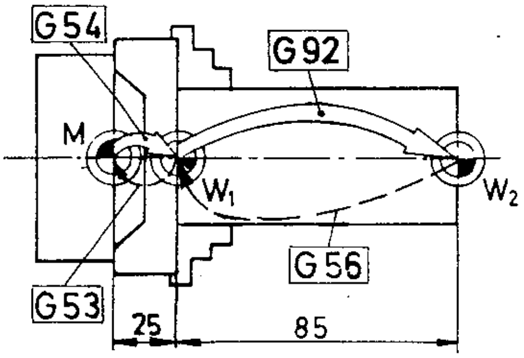
Accept ⇩



5. 00,000 110,000

N..../G59/.... Call from PSO 5 - shift from M to W

N..../G56/.... Deselect from PSO 5

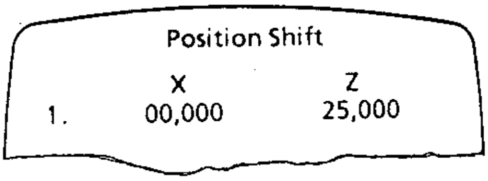


Example 2:

Call several zero point shifts.

N..../G54/... Call from PSO 1 - shift from M to W1

activate ⇩



Position Shift

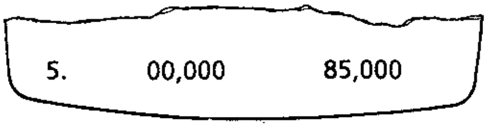
X Z

1. 00,000 25,000

Call W2

N..../G92/X00,000/Z85,000

Accept ⇩



5. 00,000 85,000

N..../G59/... Call from PSO 5 - shift from W1 to W2

N..../G56/G53/.. Deselect from PSO 5 and PSO 1

Chapter 4

G-functions

Structure and Initial status of G-Codes

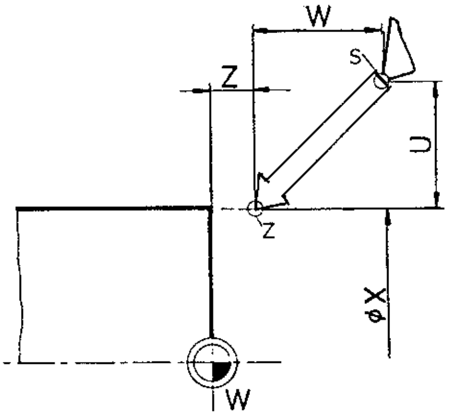
Group 0		G00: Rapid traverse
		G01: Linear-interpolation
		G02: G03: } Circular-interpolation
	*	G04: Dwell
		G33: Thread cutting in single block
		G84: Face and longitudinal turning cycle
		G85: Threading cycle
		G86: Grooving cycle
		G87: Drilling cycle with chip breaking
		G88: Drilling cycle with chip breaking and return to start point.
Group 1		G96: Constant cutting speed
	**	G97: Direct speed programming
Group 2		G94: Feed rate data in mm/min or 1/100 inch/min
	**	G95: Feed rate data in μm/rev. or 1/10 000 inch/rev.
Group 3	**	G53: Cancel workpiece zero point 1 and 2
		G54: Calling up workpiece zero point 1
		G55: Calling up workpiece zero point 2
Group 4	*	G92: 1. Speed limitation 2. Changing of workpiece zero point coordinates in position shift offset 5 over NC-program.
Group 5	**	G56: Cancel workpiece zero point 3, 4 and 5
		G57: Calling up workpiece zero point 3
		G58: Calling up workpiece zero point 4
		G59: Calling up workpiece zero point 5
Group 6		G25: Subroutine call
		G26: Polygon call
		G27: Unconditional jump
Group 7	□	G70: Measurement data in inch
	□	G71: Measurement data in mm
Group 8	**	G40: Neutralization of the tool correction
		G41: Tool path correction left hand
		G42: Tool path correction right hand

* Effective block by block
** Initial status
□ Initial status in mode of operation MON can be determined

G00 - Rapid traverse

N4	G00	X U ±43	Z W ±43
		[mm]	[mm]

G00 (rapid traverse) is a pure traversing movement - not a working movement!
The speed of rapid traverse is specified by the factory for the particular machine type. The feed override switch > 100 % is not effective.



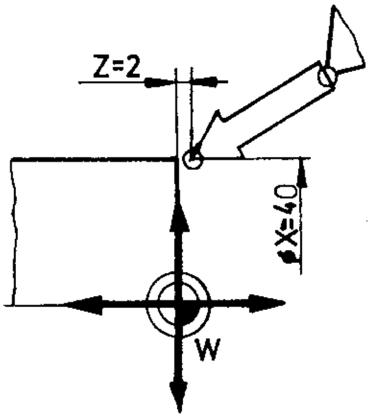
S ... Start point
Z ... Target point

Programming:

N .... Block number
G00 .. Rapid traverse
X, U } Absolute, incremental coordi-
Z, W } nates of the target point Z

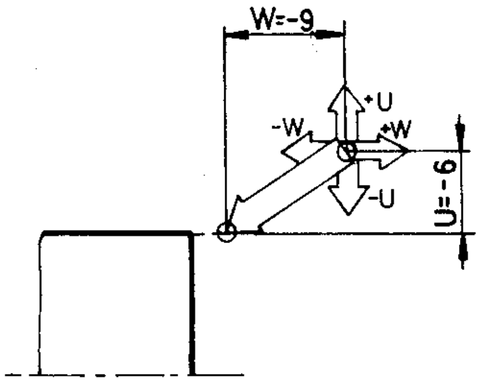
Notes:
The sequence of X(U), Z(W) is immaterial. These can also be programmed in a mixed (absolute and incremental) block, e.g. G00/X44.000/W-9.000

Programming absolute:



```
N100/.....
N110/G00/X40,000/Z2,000
N120/.....
```

Programming incremental:

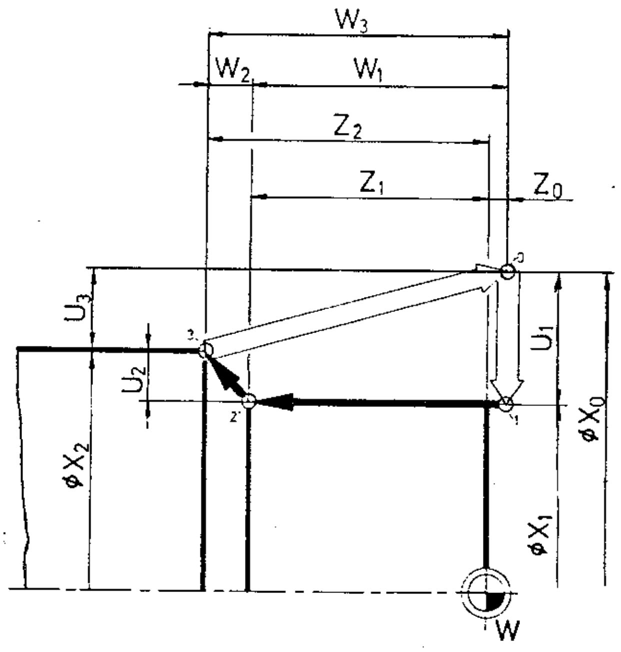


```
N100/.....
N110/G00/U-6,000/W -9,000
N120/.....
```

G01 - Straight line interpolation

N4	G01	X U ±43	Z W ±43	F4
		[mm]	[mm]	[µm/rev.] [mm/min]

G01 is a linear operating movement. The feed must be programmed. It can be input in [mm/min] (G94) or in [µm/rev.] (G95).
The feed (F) is self-holding.

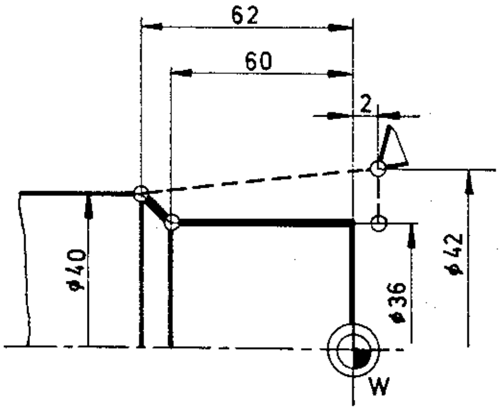


Programming:

N ..... Block number
G01 ... Function straight line interpolation
X, U }
Z, W } Absolute, incremental co-ordinates of the target point Z

N ...../G00/X_1(-U_1)
N ...../G01/-Z_1(-W_1)/F ...
N ...../G01/X_2(+U_2)/-Z_2(-W_2)/F ...
N ...../G00/X_0(+U_3)/+Z_0(+W_3)

Example for G01



Programming absolute:

```
N100/ .....
N110/G00/X42,000/Z2,000
N120/G00/X36,000
N130/G01/Z-60,000/F ...
N140/G01/X40,000/Z-62,000/F ...
N150/G00/X42,000/Z2,000
N160/ .....
```

Programming incremental:

```
N100/.....
N110/G00/.....
N120/G00/U-3,000
N130/G01/W-62,000/F.....
140/G01/U2,000/W-2,000/F.....
150/G00/U1,000/W64,000/
160/.....
```

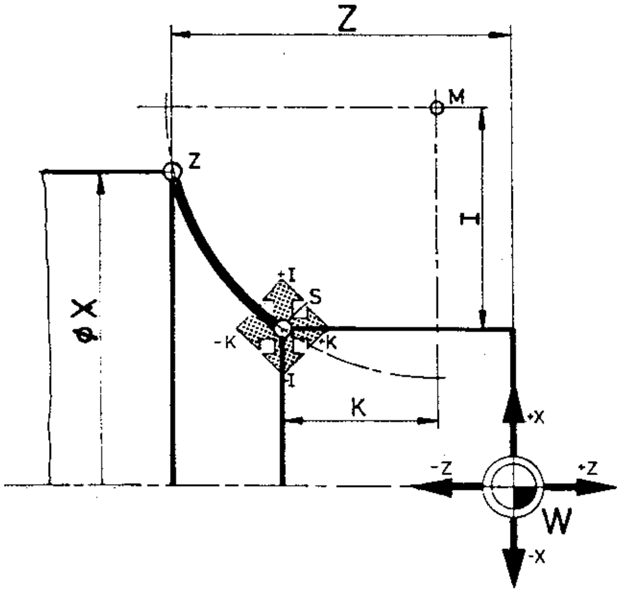
Arc interpolation
G02 - clockwise
G03 - counter-clockwise

N4	G02 G03	X U ±43	Z W ±43	I ±43	K ±43	F4
		[mm]	[mm]	[mm]	[mm]	[µm/rev.] [mm/min]

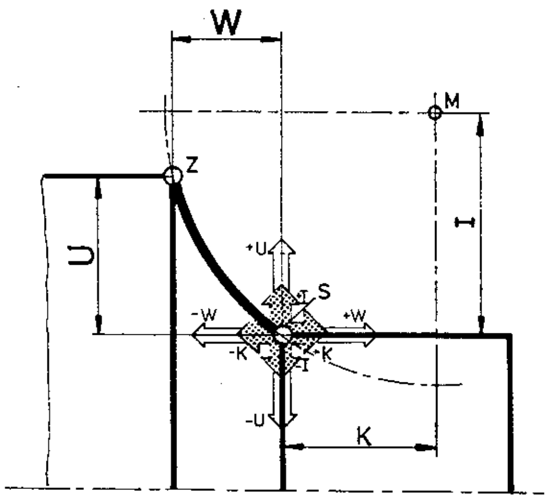
Programming:

N .... Block number
G02 .. Arc interpolation clockwise
G03 .. Arc interpolation counter-clockwise
X, U) Absolute, incremental co-ordinates
Z, W) of the target point
I, K . Arc centre point co-ordinates (incremental
from the arc starting point)
F .... Feed

Programming absolute:



Programming incremental:



- * The target point of the arc from zero point is specified with X, Z.
- * I, K is used to specify the arc centre point from the arc starting point.
N...../G02/X/-Z/I/K/F...

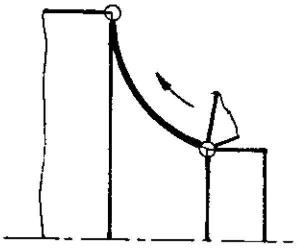
- * U, W specify the target point from the arc starting point.
- * I, K is used to specify the arc centre point from the arc starting point.
N...../G02/U/-W/I/K/F...

Notes on G02/G03

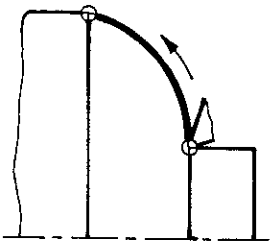
1. General:

Programming the arc interpolation with arc centre point co-ordination is carried out in accordance with DIN 66025.

2. Direction of rotation:



Rotation G02 clockwise



Rotation G03 counter-clockwis

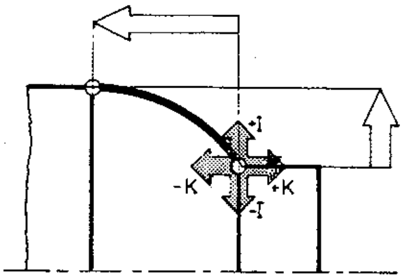
3. On specifying the centre point co-ordinates:

An arc is determined by specifying the start and end point and by specifying a centre point co-ordinate (I or K).

Specifying both centre point co-ordinates is an overspecification.

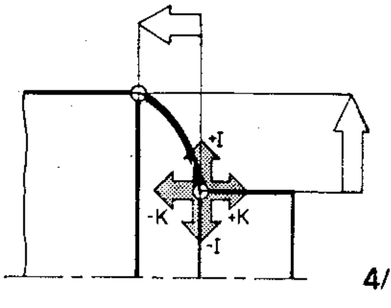
Applies for control TM 02:

Both of centre point co-ordinate have to be precisely programmed.



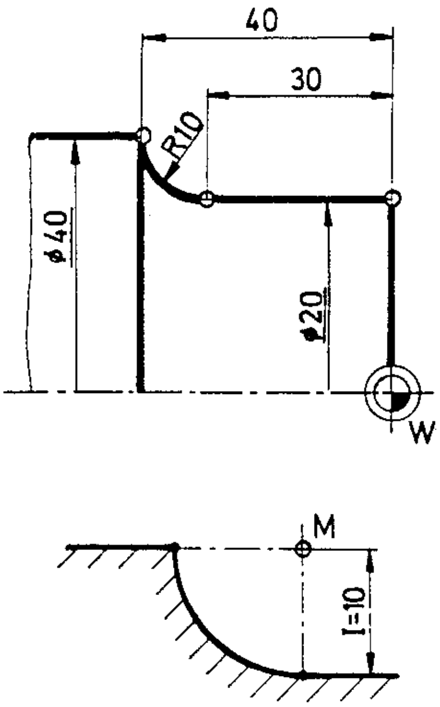
Example 1:

N...../G03/X(U)/Z(W)/I/-K/F...



Example 2:

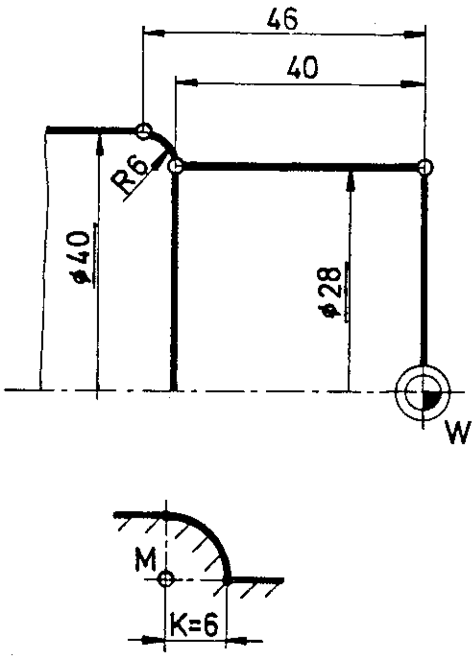
N...../G03/X(U)/Z(W)/I/-K/F...



Example 1:

Programming absolute:
N..../G01/X20,000/Z-30,000/F....
N..../G02/X40,000/Z-40,000/I10,000
K = 00,000/F....

Programming incremental:
N..../G01/.....
N..../G01/U10,000/W-10,000/I10,000
K00,000/F....



Example 2:

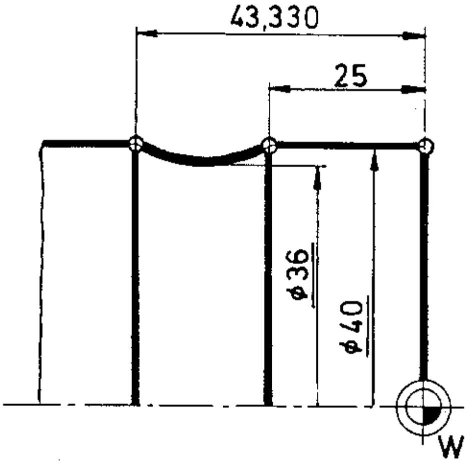
Programming absolute:
N..../G01/X28,000/Z-40,000/F....
N..../G03/X40,000/Z-46,000/I00,000
K-6,000/F....

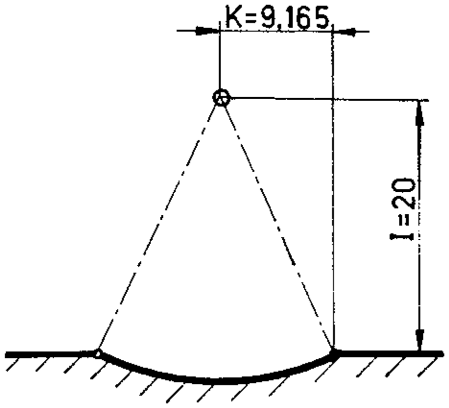
Programming incremental:
N..../G01/.....
N..../G03/U6,000/W-6,000/I00,000
K-6,000/F....

Example 3:

Programming absolute:
N..../G00/X40,000/Z-25,000
N..../G02/

Programming incremental:
N..../G00/.....
N..../G02/W-18,330/I20,000
K-9,165/F....





G04 - Dwell time

N4	G04	D$_4$ 5
		[1/10 s]

A dwell time is programmed under parameter D$_4$ with G04.

Input range:
1 - 10,000 (0.1 s - 1000 s)

Note:
G04 acts only blockwise and is only active at the end of the block. Irrespective of whether the dwell time is written before or after other words in the block.

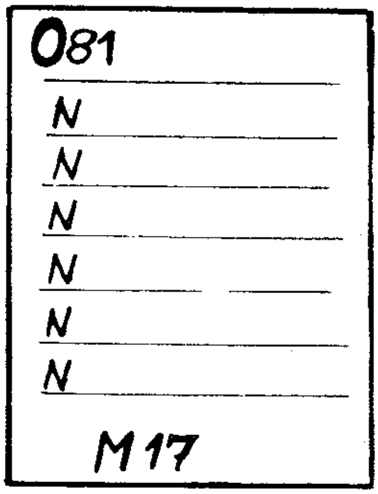
Example:
N0100/G04/D$_4$20/M03
N0110/G00/X40,000/Z-10,000

Block 100:
The main spindle is switched on (turning clockwise = M03). Prior to block N110 being processed, the programmed dwell time of 2 seconds is executed by the control.

G25 Subroutine call
M17 Return command7

Subroutine numbers: 0 80 - 0 99

Nesting limit: 10



A subroutine is called by the main program or a subroutine. In principle, the subroutine, as such, has the same structure as a main program.

It consists of:

+ Program number:

  Possible program numbers
  0 80 - 0 99 (see also remark)

+ Blocks

+ M17:
  Program end with return command.

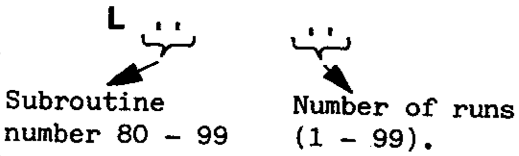
Subroutine call G25

A subroutine is called by the main program or a subroutine.

Subroutine call

+ G25 Subroutine call

+ L.... Address for subroutine number and number of runs.



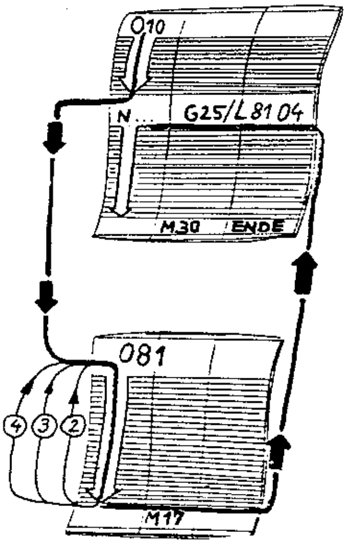
Example O 81:

Subroutine with 4 runs

MAIN PROGRAM

O 1Ø

Subroutine O 81



Example:

Nesting of subroutines

From subroutines, additional subroutines can be called. (Nesting of subroutines)
The EMCOTRONIC permits a ten-fold nesting.

MAIN PROGRAM
O 1Ø

Sub 1 O 8Ø

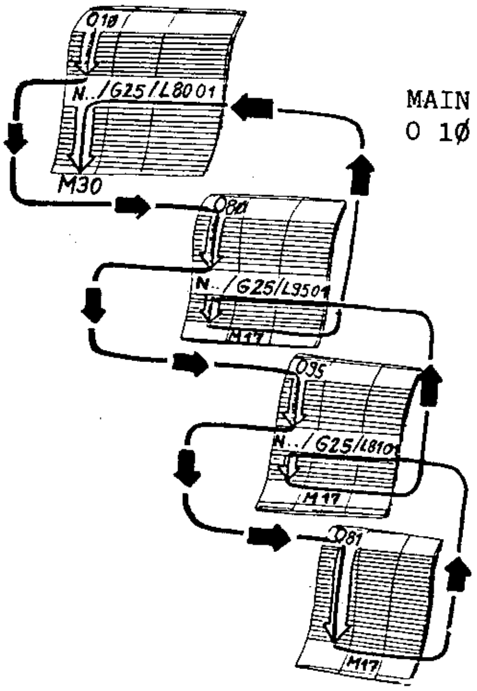
Nesting limit 1

Sub 2 O 95

Nesting limit 2

Sub 3 O 81

Nesting limit 3



Program numbers for subroutines

For easier identification, main programs and subroutines should be numbered so as to keep them apart.

For this reason, the following is specified by the manufacturer:

Possible main program numbers	O 0000 - O 6999
Possible subroutine numbers	O 80 - O 255

The numbers O 0000 - O 6999 can be used for the main program (sensibly, the numbers O 0080 - O 0255 are not used for main programs, where subroutines are also used by you).

Only the numbers O 0080 - O 0255 can be used as subroutine numbers, otherwise alarm A630 is actuated.

Remark:

The numerical range for subroutines can be changed by you in the MONITOR operating mode.

Example:

You wish to put in subroutines from program number O 0060.

O_{22} 80:	Erase number 80, and put in
O_{22} 60:	the number 60 under O_{22}.

G26 - Polygon program recall

N4	G26	L4

Polygon programs for graphic simulation in NC-programs are called with G26. The program number to be called is input under parameter L.

Polygon programs:
The program numbers 0 7000 to 0 9999 are specified for polygon program graphic simulation. For details see graphic simulation.

G27 - Unconditional Jump

N4	G27	L4

The G27 instruction causes a jump within the program sequence.
The block number to be jumped on is programmed under the L address.

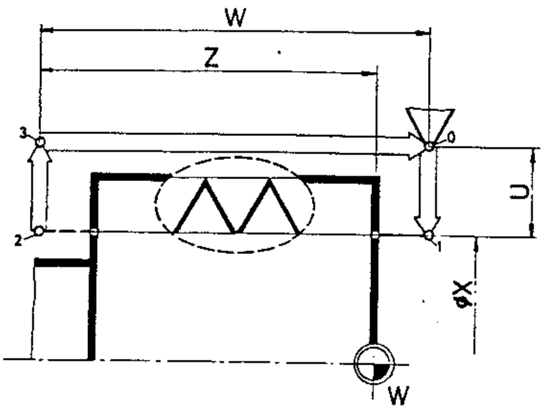
Example:
N100/G27/L250

The program jumps from block N 100 to block N250

G33 - Single block thread cutting

N4	G33	X U ± 43	Z W ± 43	F4
		[mm]	[mm]	[µm]

A thread can be cut in individual steps with G33. The feed and return movements must be programmed in separate blocks. The notes and explanations given on the pages for G85-thread cycle, on thread start, thread runout and thread pitch, also apply for G33.



Programming:

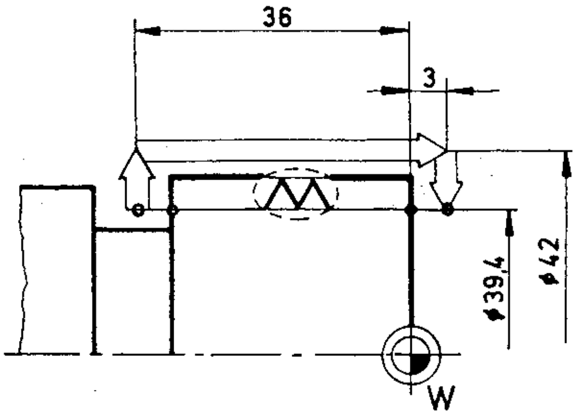
N ..	Block number
G33	Single block thread cutting
X, U }	Absolute incremental co-
Z, W }	ordinates of the target
F ...	Thread pitch

N...../G00/X_1(-U)

N...../G33/-Z(-W)/F.....

N...../G00/X_2(U)

N...../G00/Z(W)



Programming absolute:

```
N100/.....
N110/G00/X42,000/Z3,000
N120/G00/X39,400
N130/G33/Z-36,000/F.....
N140/G00/X42,000
N150/G00/Z3,000
```

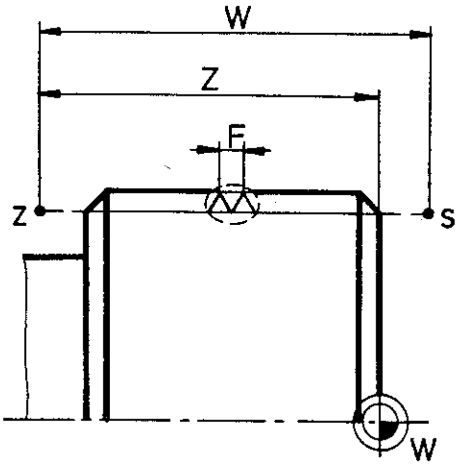
Programming incremental:

```
N100/.....
N110/G00/.....
N120/G00/U-1,300
N130/G33/W-39,000/F.....
N140/G00/U1,300
N150/G00/W39,000
```

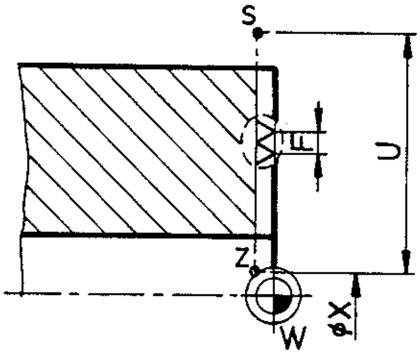
Notes on G33:

In the notes and explanations given on G85-thread cycle, you will find comprehensive explanations on the thread start, thread runout and thread pitch.

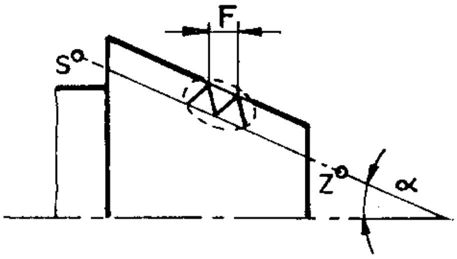
Thread pitch details:



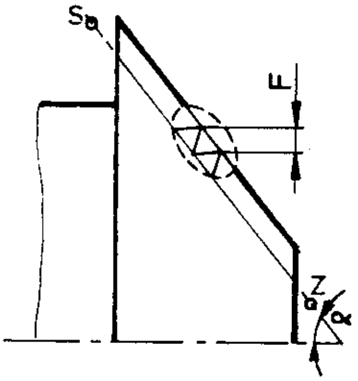
Longitudinal thread



Facing thread



Longitudinal thread (a < 45x):
When programming a longitudinal taper thread (a < 45x), F must be specified in the Z-direction.



Facing taper thread (a > 45x):
When programming a facing taper thread (a > 45x), F must be specified in the X-direction.

RADIUS COMPENSATION
EMCOTRONIC TM 02

Tool Path Correction

G40 Cancellation of the Tool (Path) Correction
G41 Tool Path Correction, left
G42 Tool Path Correction, right

Note:

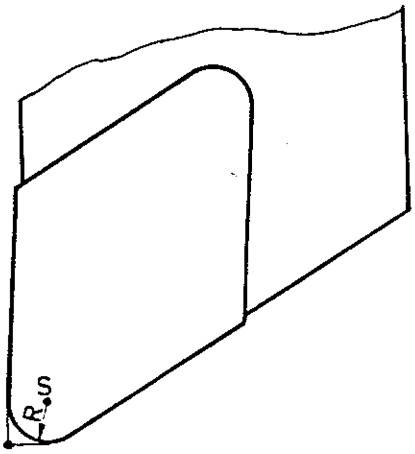
The tool path correction during turning should only be used, where it is necessary.
You will thus avoid unnecessary program errors and alarms.

Cutter Radius Compensation

G40 Cancellation of the Tool (Path) Correction

G41 Tool Path Correction, left

G42 Tool Path Correction, right



1. Purpose

The theoretical cutter point is measured and input in the tool data file.

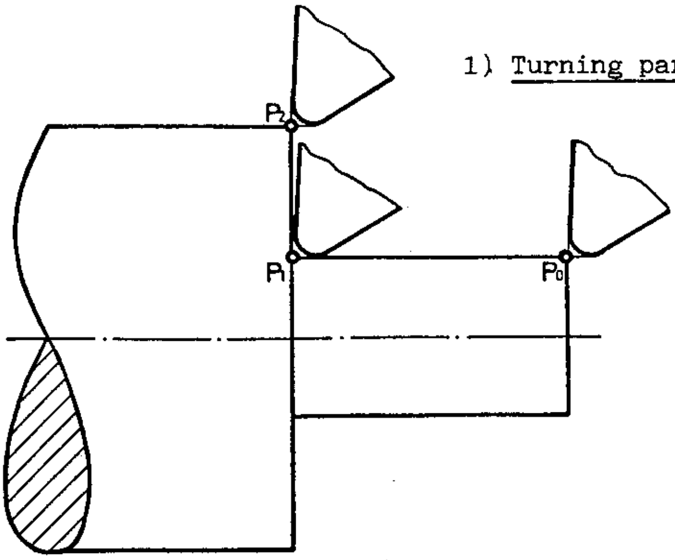
For technological cutting reasons, the cutter point is always equipped with a radius. The contour generating points during turning are not the theoretical cutter points, but the circumferential points of the point radius.

With angles and circles, deviations to the programmed contour are thus produced.

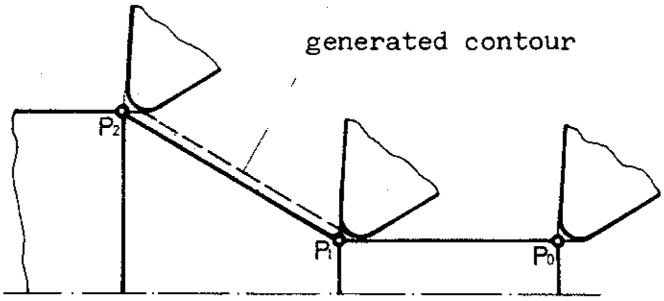
Examples for the Effect of the Cutter Radius

1) Turning parallel to the axes X, Z:

The programmed contour is in accordance with the produced contour.

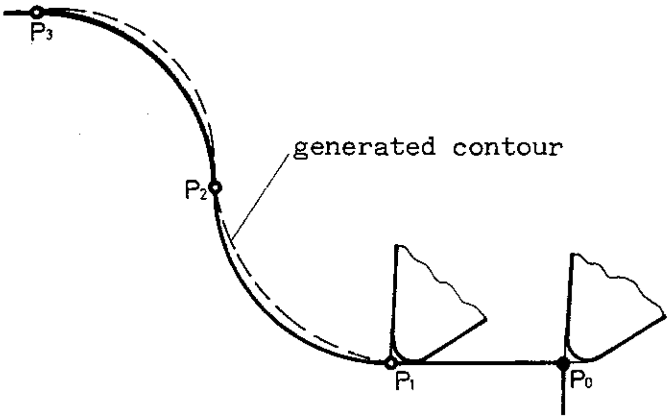


2) Turning of angles:



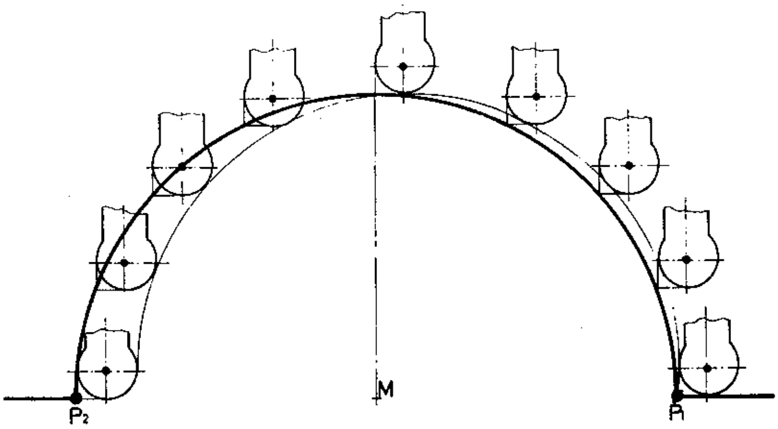
One spot of the point radius, and not the theoretical cutter point, generates the contour (programmed contour and produced contour do not agree).

3) Circular arcs



The programmed contour and the produced contour do not agree.

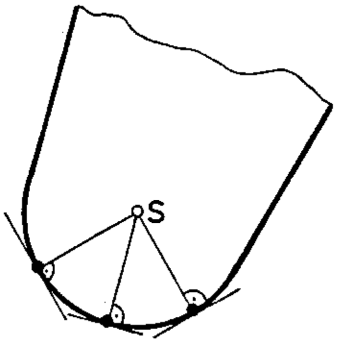
Shift of the circular arc



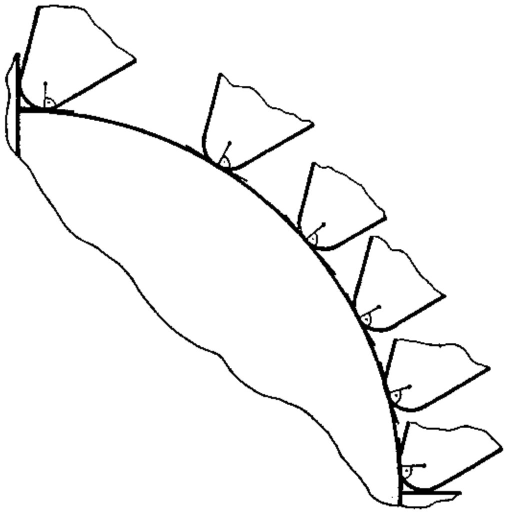
Circular arc centre point is displaced; P_2 is displaced.

Please note the Difference

With radius compensation

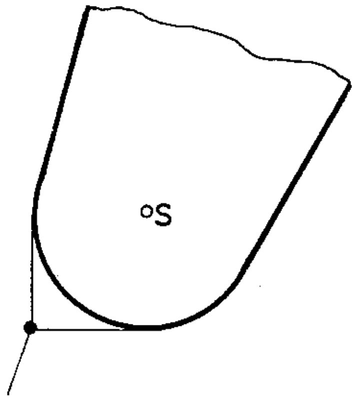


The control knows the particular contour generating points on the turning tool.

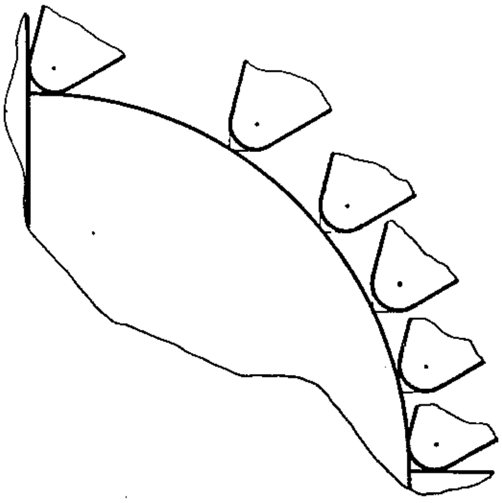


Tangent of the contour generating point is always normal on the path cutter radius centre point (S) and contour generating point

Without radius compensation



Theoretical cutter point



The theoretical cutter point is in each case the contour generating point.

2. Details required for Tool Path Correction

The control compensates for the effect of the point radius. It calculates these traverse paths, which then generate the actually programmed contour.

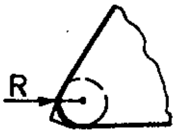
For this, the control must be given the necessary information.

These are: 1) Cutter radius (R)

2) Cutter length (L)

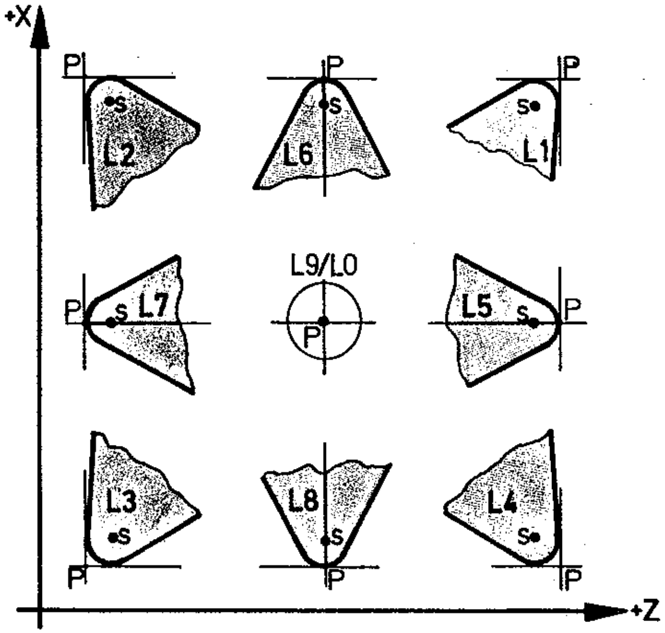
3) Tool path correction, left or right (G41 or G42)

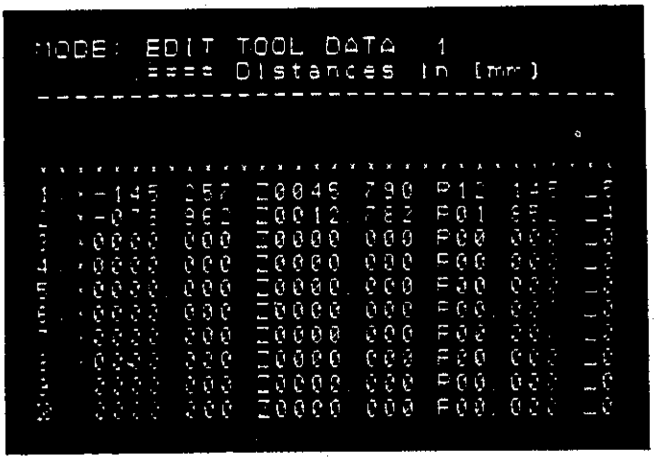
Fror this information, the control calculates the p[illegible] points for producing the programmed contour.



2.1 Cutter radius R

The cutter radius: In addition to the tool data X, Z, the radius must also be input in the tool data file. The dimension for the correction is dependent on the cutter radius. Input in mm (inch) with decimal point under the R address in the tool data memory.





```
MODE:  EDIT TOOL DATA   1
       ==== Distances in [mm]
----------------------------------

1  X-145.257 Z0045.790 R12.145  L6
2  X-07[illegible].962 Z0012.782 R01.[illegible]  L4
3  X0000.000 Z0000.000 R00.[illegible]  L3
4  X0000.000 Z0000.000 R00.[illegible]  L[illegible]
5  X0000.000 Z0000.000 R00.[illegible]  L[illegible]
6  X0000.000 Z0000.000 R00.[illegible]  L[illegible]
7  X0000.000 Z0000.000 R00.[illegible]  L[illegible]
8  X0000.000 Z0000.000 R00.[illegible]  L0
9  X0000.000 Z0000.000 R00.0[illegible]  L0
10 X0000.000 Z0000.000 R00.0[illegible]  L0
```

2.2 Cutter length

The computer must know the position of the theoretical cutter points (P) in relation to the radius centre point (S), to calculate the correct type of contour. Theoretically, a tool can be measured in nine positions.

Example: Left-hand side tool is inserted: Cutter position L3

Example: Internal copying tool is inserted: Cutter position L2

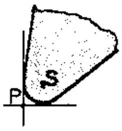
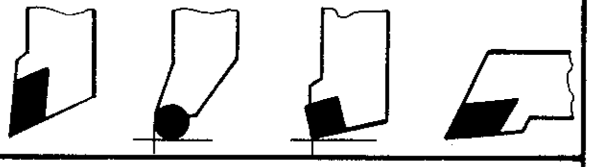
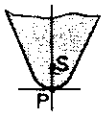
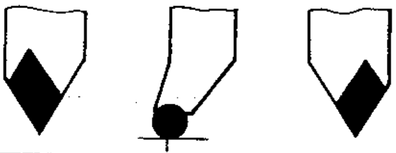
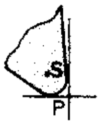
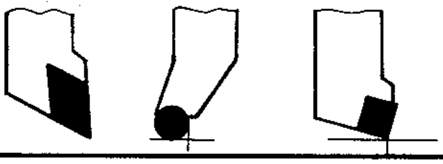
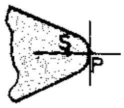
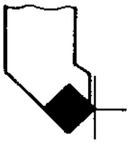
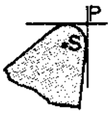
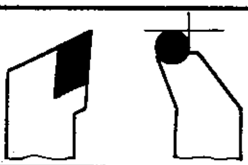
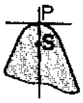
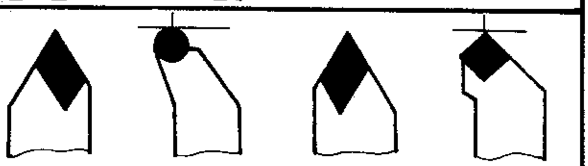
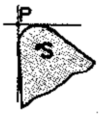
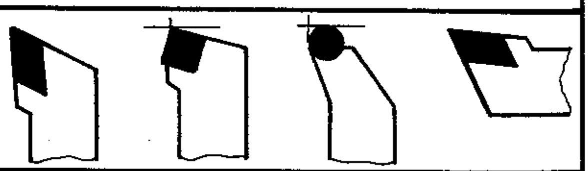
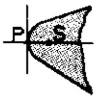
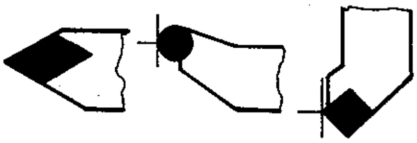
- Point P is the measured point for tool data X, Z.

Example for tool data in tool data memory

- R Point radius

- Input of the cutter position in the tool data memory under the address L.

Examples: Turning tool shapes
Cutter position L

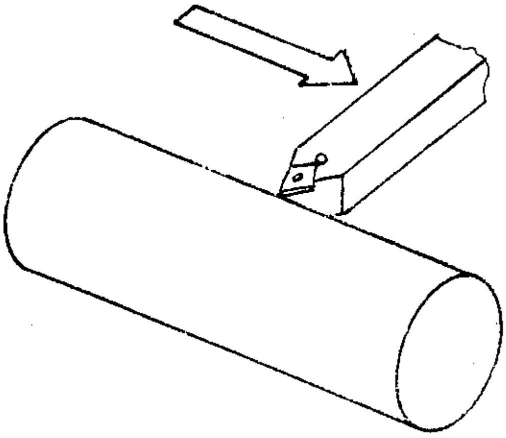
L Position of cutter radius		Turning tool shapes
L3		
L8		
L4		
L5		
L1		
L6		
L2		
L7		

The cutter positions are determined by the position of point P (measured and cutter point P input in the tool data memory) to point S (centre point of the radius).

2.3 Tool path correction, left or right

With the programming of G41, G42, the control is given the information on the tool movement in relation to the workpiece.

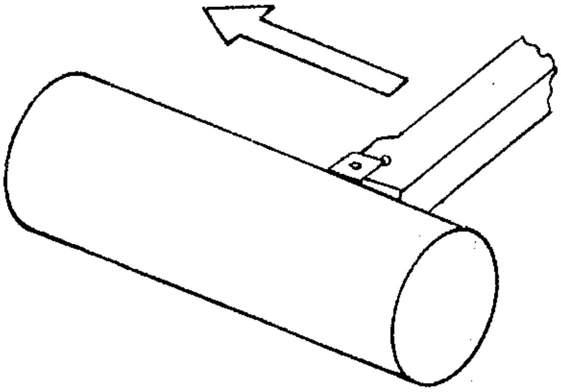
G41 Tool path correction, left



The tool is located to the left of the workpiece, when viewed in the direction of the relative tool movement.

Rule:
Place yourself on the workpiece, and look in the direction of feed. Where the tool is left - G41.

G42 Tool path correction, right



The tool is located to the right of the workpiece, when viewed in the direction of the relative tool movement.

Rule:

Look toward the turning tool, the turning tool is to the right of the workpiece - G42.

G40 Cancellation of the tool correction

The programmed path agrees with the traverse path of the theoretical cutter tip.

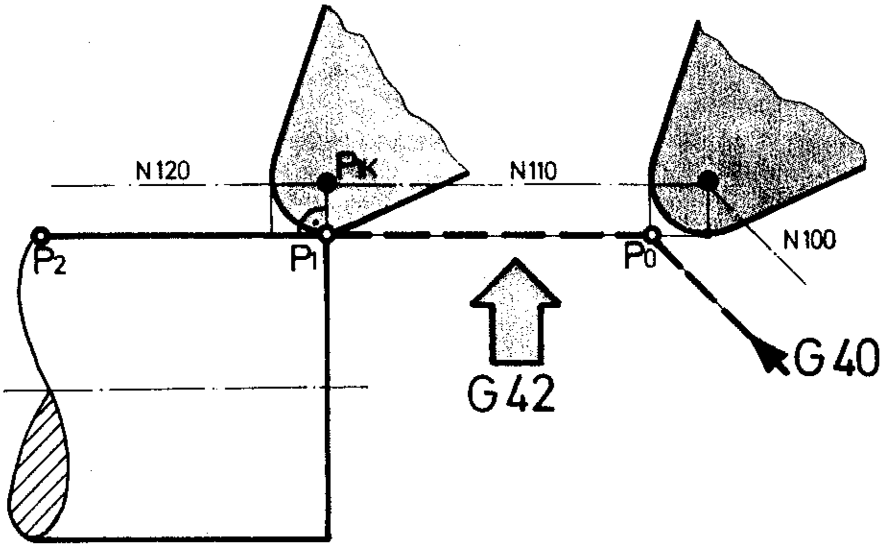
3. Programming

So that you can decide, when and how to select or deselect the radius compensation, you must know the types of starting and following.

3.1 Selecting the tool path correction

3.1.1 Neutral starting

Angles between programmed path $\overline{P_0P_1}$ (selection path) and path $\overline{P_1P_2}$ is 180°.



Tool moves with centre point to point P_{1K}.

$\overline{P_1\ P_{1K}}$ is a normal on the following path $\overline{P_1P_2}$.

Programming:

N .../G40

N 100/G00/X_{PO}/Z_{PO}

N 110/G01/X_{P1}/Z_{P1}/G42

N 120/G01/X_{P2}/Z_{P2}

Set N 100: Tool traverses with theoretical cutter point to Point P_0 (G40 active).

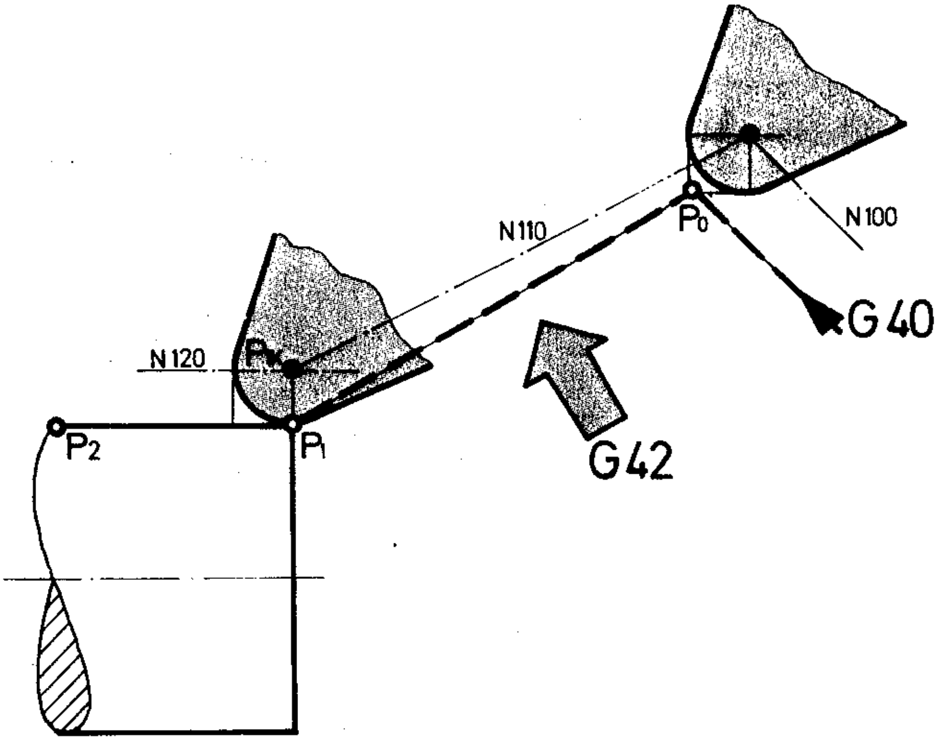
Set N 110: The contour generating point of the turning tool touches point P_1.

Note:

With G41/G42 call and deselection, the control knows the content of the prior and following set. It can thus calculate Point P_{1K}.

3.1.2 Approach angle less than 180°

Angle between programmed path P_0P_1 (selection path) and path P_1P_2 is less than 180°.



Set N 100

Tool traverses with theoretical cutter tip to P_0

Set N 110

Tool traverses with centre point to P_{1K}.

P_{1K} is a normal on the following path $\overline{P_1P_2}$.

The contour generating point of the turning tool is positioned at point P_1.

Programming:

N ../G40

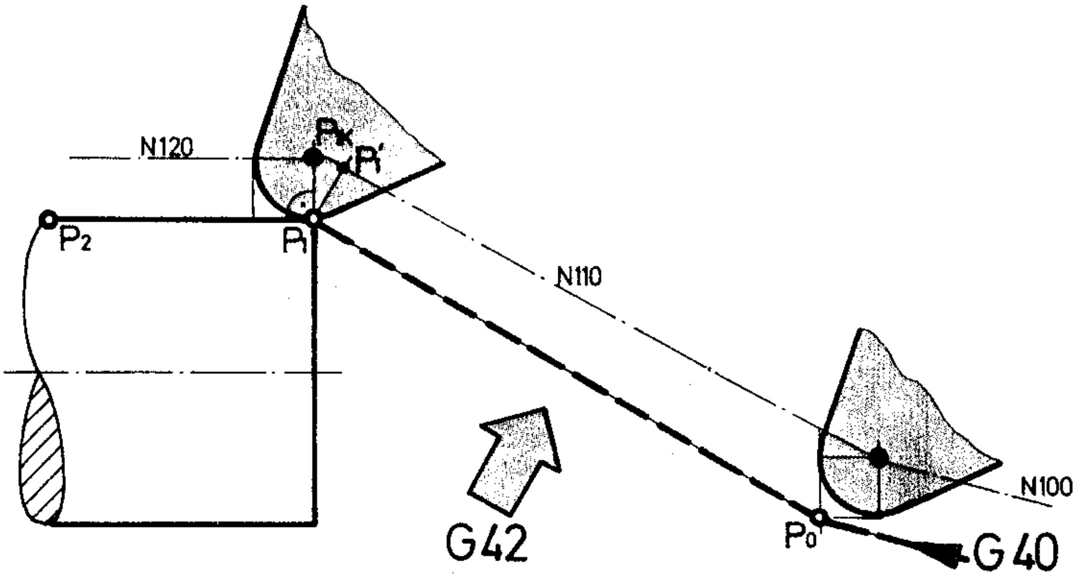
N 100/G00/X_{P0}/Z_{P0}

N 110/G01/X_{P1}/Z_{P1}/G42

N 120/G01/X_{P2}/Z_{P2}

3.1.3 Approach angle greater than 180°

Angle between programmed path $\overline{P_0P_1}$ (selection path) and path $\overline{P_1P_2}$ is greater than 180°.



Set N 100

Tool traverses with theoretical cutter tip to point P_0.

Set N 110

The tool traverses with centre point to P_1',

then in a circular arc to point P_{1K}.

The circular arc radius is equal to the radius of the tool.

$\overline{P_1P_1}$' is a normal on $\overline{P_0P_1}$ in point P_1.

$\overline{P_1P_{1K}}$ is a normal on $\overline{P_1P_2}$.

The contour generating point of the tool touches P_1.

Programming

N .../G40

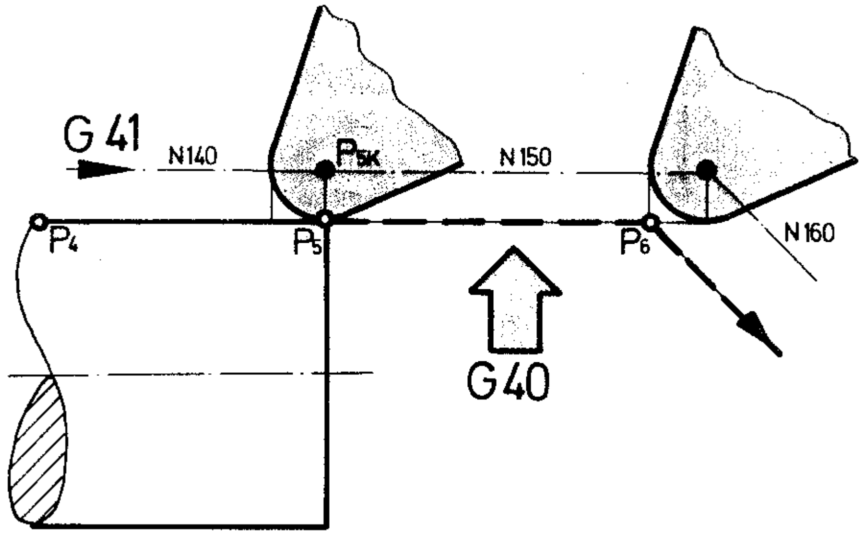
N 100/G00/X_{PO}/Z_{PO}

N 110/G01/X_{P1}/Z_{P1}/G42

N 120/G01/X_{P2}/Z_{P2}

3.2 Deselection of tool path correction

3.2.1 Neutral deselection



Set N 150

Tool traverses with theoretical cutter tip to point P_6.

Programming:

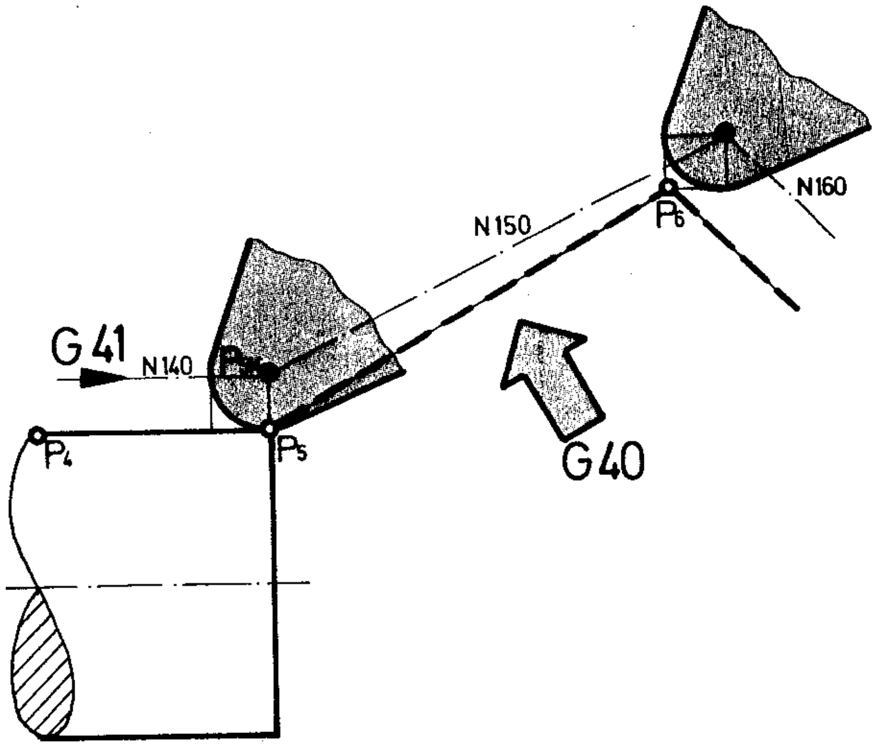
N .../G41

N 140/G01/X_{P5}/Z_{P5}/

N 150/G00/X_{P6}/Z_{P6}/G40

N 160/G00/X_{P7}/Z_{P7}

3.2.2 Deselection angle smaller than 180°



Set N 150 (G40 becomes active)

Tool traverses with theoretical cutter tip to point P_6.

Programming:

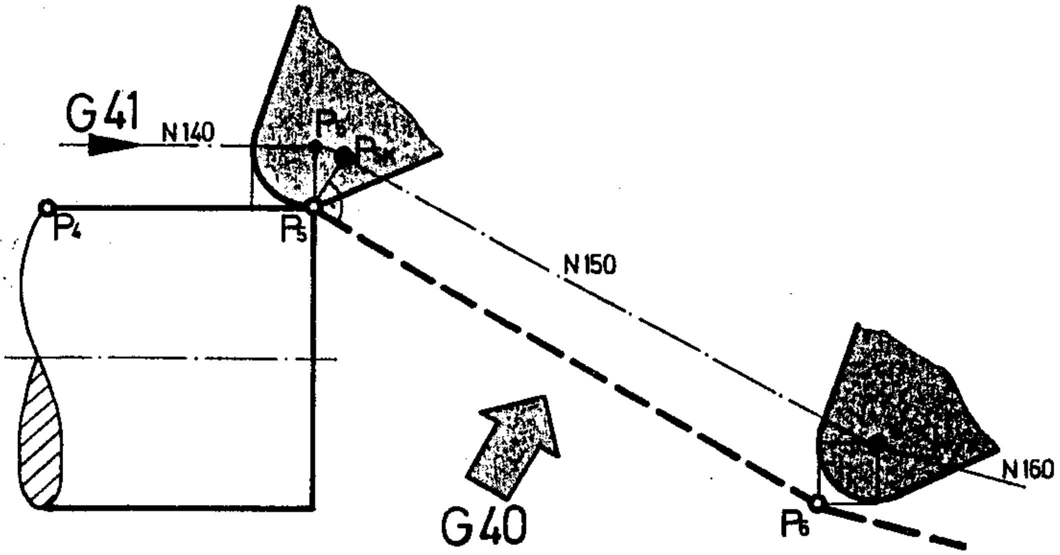
N .../G41

N 140/G01/X_{P5},Z_{P5}/

N 150/G00/X_{P6},Z_{P6}/G40

N 160/G00/X_{P7},Z_{P7}

3.2.3 Deselection angle greater than 180°



Set N 140

Point S of the tool traverses from point P5' to point P_{5K}.

P_{5K} is located on the contour normal of path $P_5 - P_6$.

Set N 150

Tool traverses with theoretical cutter tip to point P_6.

Programming:

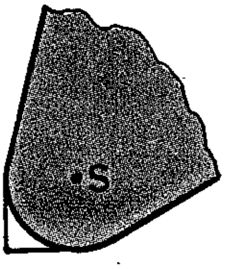
N .../G41

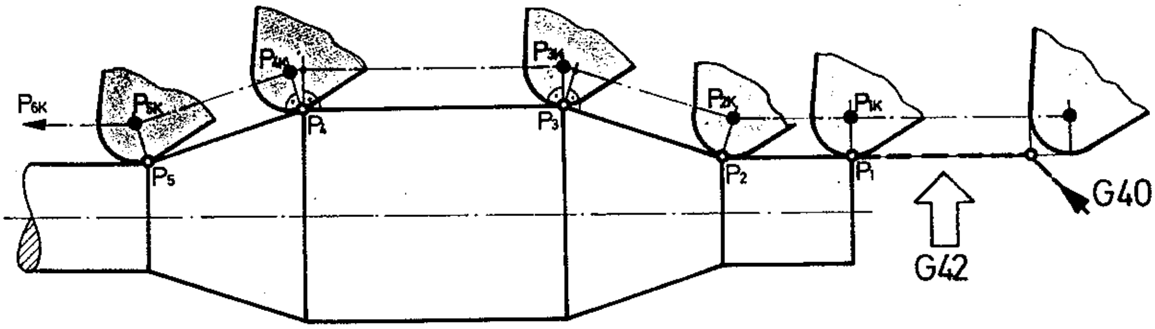
N 140/G01/X_{P5}/Z_{P5}/

N 150/G00/X_{P6}/Z_{P6}/G40

N 160/G00/X_{P7}/Z_{P7}/

3.3 Tool paths with active tool path correction

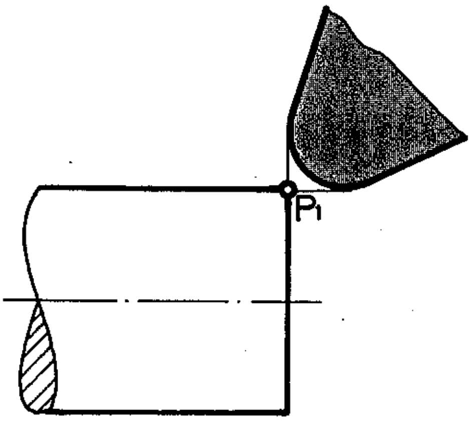




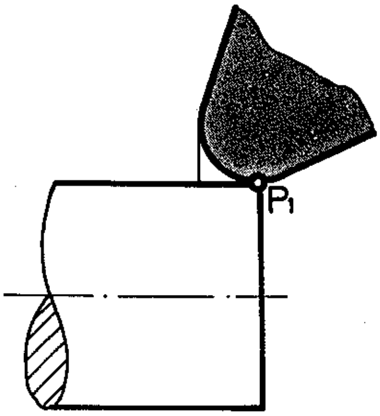
- With internal corners, the cutter centre point S traverses to the equidistant intersections.
- With outer corners, the cutter centre point S traverses in a circular arc around the programmed point up to the contour normal of the next set.
  See diagram Point P3.
  Set point end of S is P_{3K}.

3.4 Some notes on programming

1. Different turning tool position with G40 and G41/G42 active

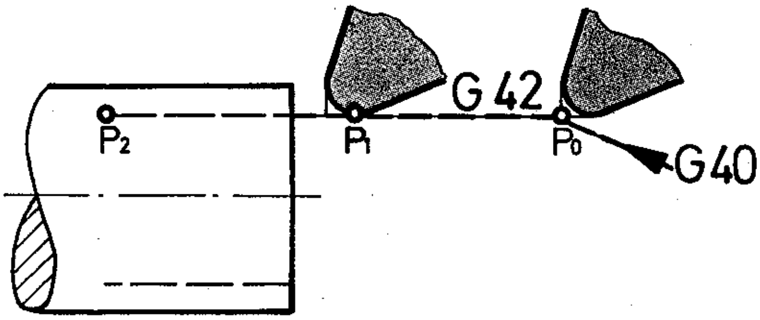


With G40 active:
The theoretical cutting point is at the programmed point.



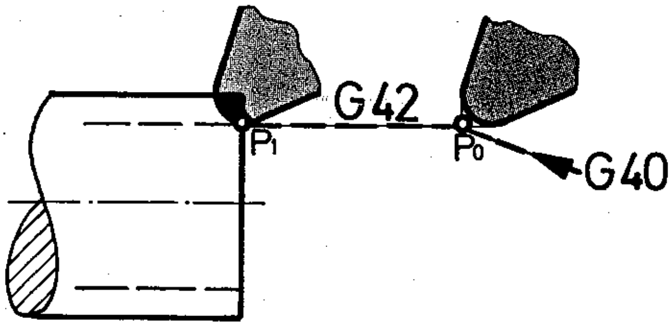
With G41/G42
The contour generating point of the tool radius is at the programmed point.

2. Cancelling the contour



For technological reasons, approach contour neutral.

Take into account appropriate distance to workpiece.

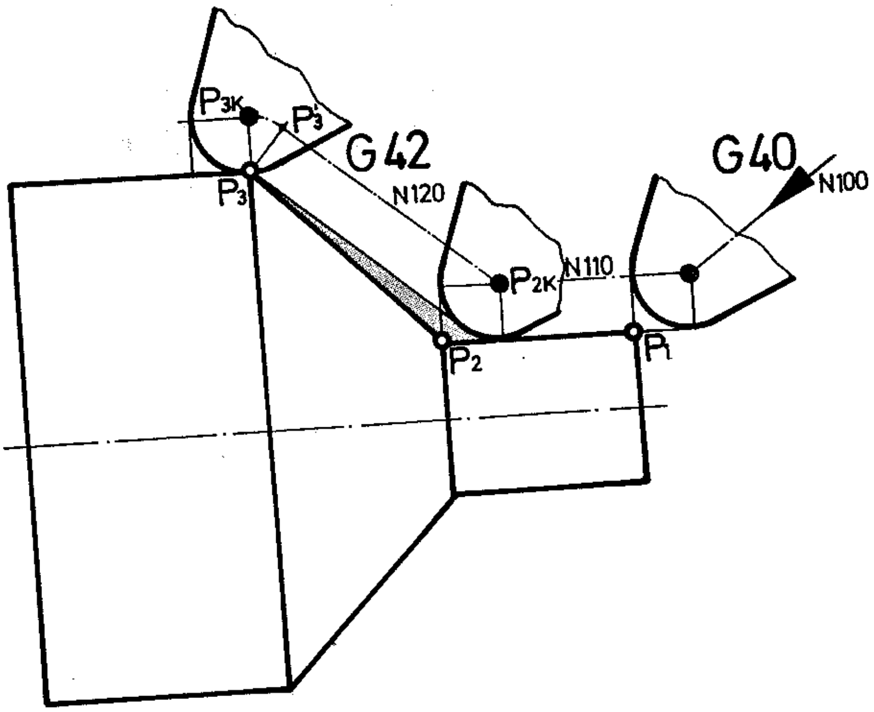


With this type of approach, cutting would already take place.

3. <u>Incorrect selection in one contour</u>

The contour $\overline{P_2P_3}$ was not produced.

The selection of G42 would have to take place in Set N 110.



Programming:

N .../G40

N 100/G01/X_{P1}/Z_{P1}

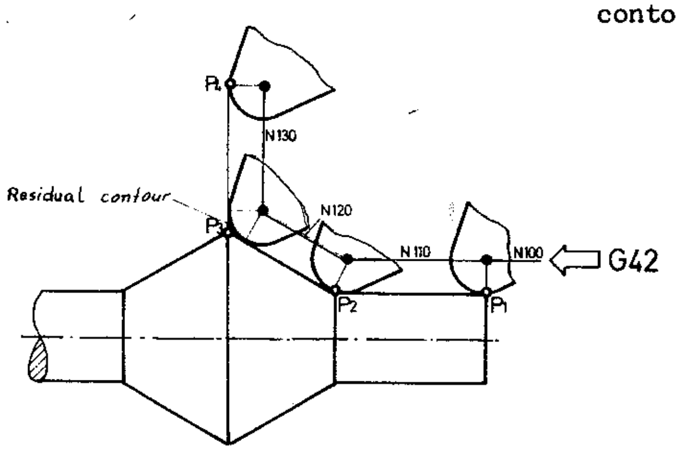
N 110/ X_{P2}/Z_{P2}

N 120 X_{P3}/Z_{P3}/G42

4. Pay attention to residual contour: Two typical examples

Example 1

With this programming, the residual contour would be retained.



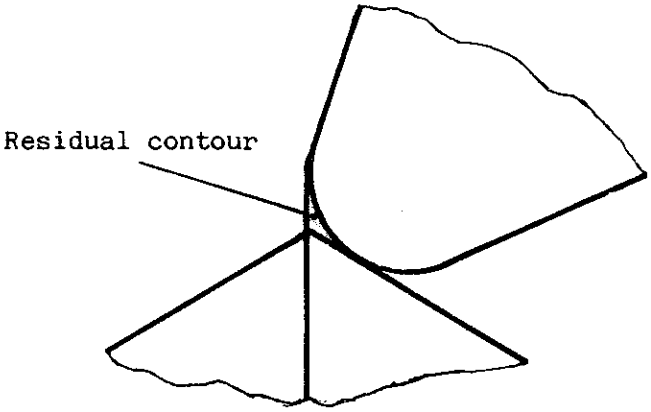
N ...   G42

N 100/G01/X_{P1}/Z_{P1}

N 110/ X_{P2}/Z_{P2}

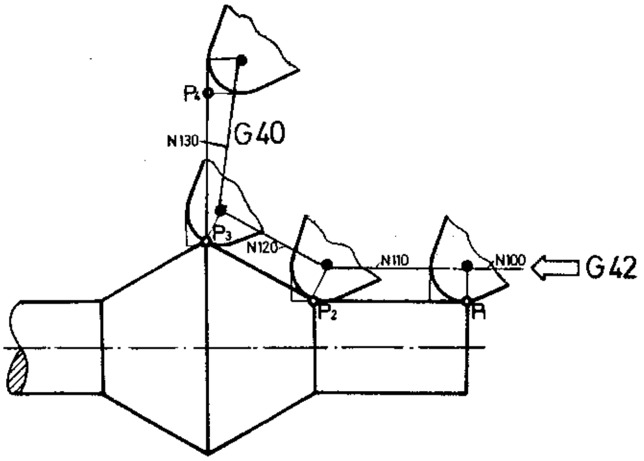
N 120/ X_{P3}/Z_{P3}

N 130/G00/X_{P4}/Z_{P4}



Possibilities for avoiding a residual contour:

1. Traverse further in the incline over Point P_3.
2. Deselect in Set N 130



N ...   G 42

N 100/G01/X_{P1}/Z_{P1}

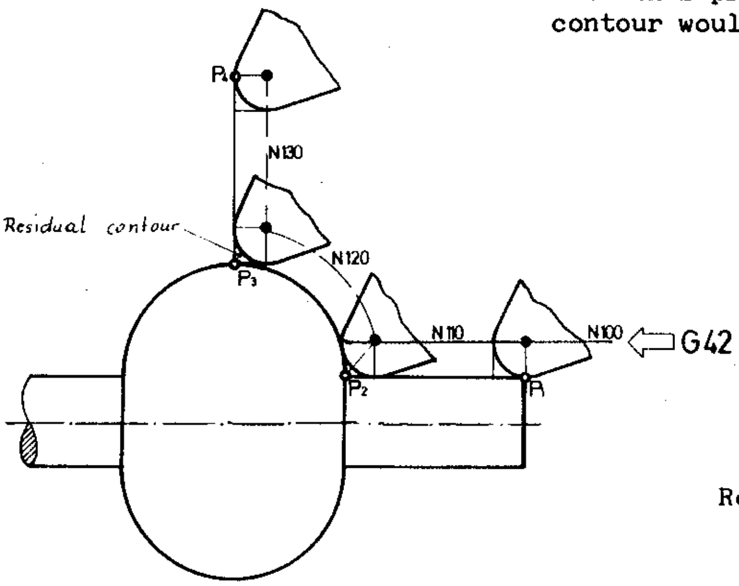
N 110/ X_{P2}/Z_{P2}

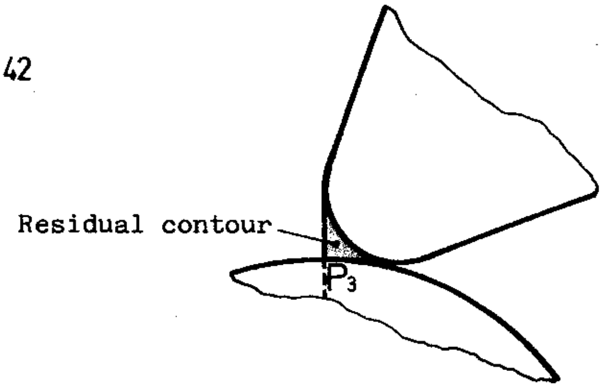
N 120/ X_{P3}/Z_{P3}

N 130/G00/X_{P4}/Z_{P4}/G40

Example 2

With this programming, the residual contour would be retained.





N .../G42

N 100/G01/X_{P1}/Z_{P1}

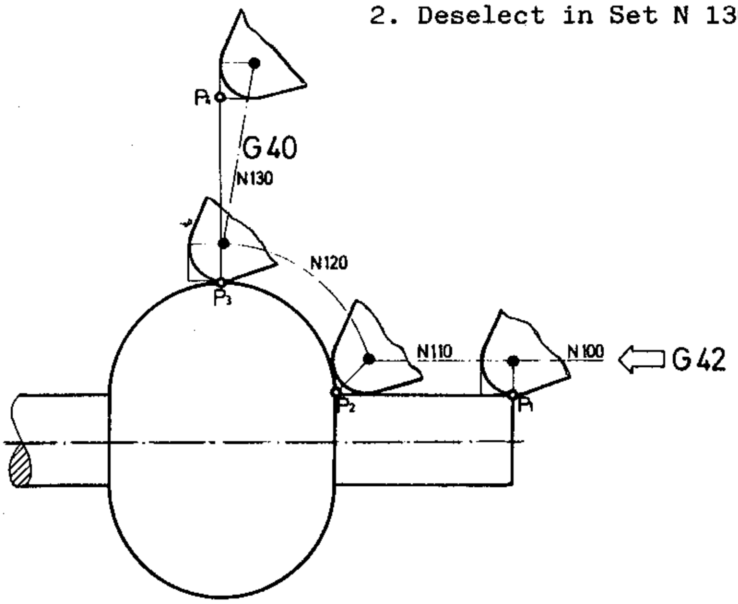
N 110/ X_{P2}/Z_{P2}

N 120/ X_{P3}/Z_{P3}

N 130/G00/X_{P4}/Z_{P4}

Possibilities to avoid residual contour:

1. Continue to traverse in circular arc beyond point P_3.
2. Deselect in Set N 130.



N .../G42

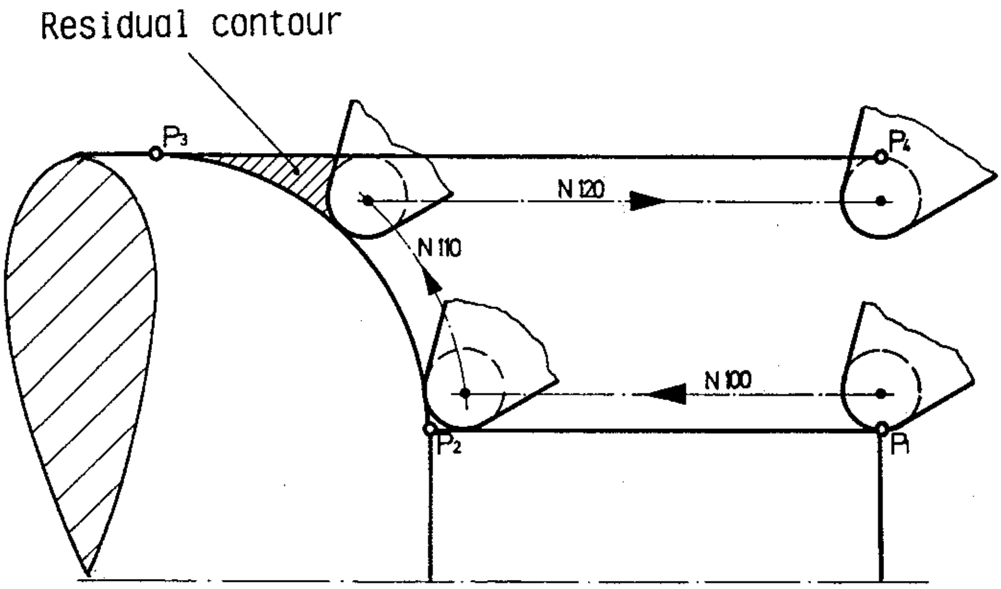
N 100/G01/X_{P1}/Z_{P1}

N 110 X_{P2}/Z_{P2}

N 120 X_{P3}/Z_{P3}

N 130/G00/X_{P4}/Z_{P4}/G40

Example 3:



You program P_1, P_2, P_3, P_4 with radius compensation.
The residual contour would remain, since the computer would naturally regard the path P_3P_4 as a contour.

4. Syntax Specifications and Alarms

4.1 Selection and deselection of cutter radius compensation

4.1.1 G40/G41/G42 may only be selected or deselected in conjunction with a G00, G01 set.

This can be:

* G00/G01 in the same set with G40 or G41/G42
* The set following G40 or G41/G42 of Group 0 must be a G00/G01 set. Between the G40 or G41/G42 and the G00/G01 set, not more than 5 sets from other groups than Group 0 may be programmed.

4.1.2 In G00/G01 set, a change in the X or Z value, or in both values (X, Z), must be programmed.

4.1.3 Deselection of cutter radius compensation with M30

Where the cutter radius compensation is deselected with M30, the M30 set must contain

1) programmed G00 or G01

2) a change in the XZ values must be programmed in G00 / G01 set.
   Otherwise Alarm 51

4.2 Number of sets active with G41/G42

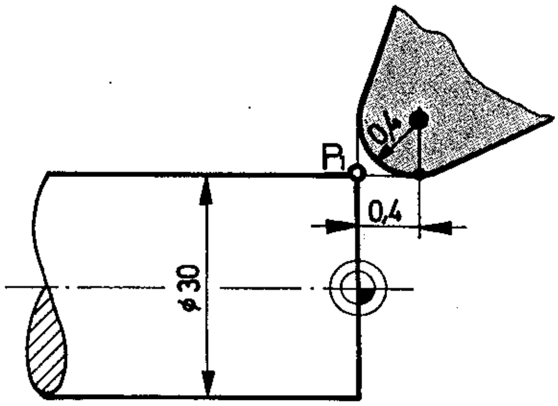
At least two sets with XZ value change must be programmed.
Otherwise Alarm 51

4.3 Tool change active with G41/G42

With tool call-up (call-up of new T address), the radius compensation must be deselected.
Otherwise Alarm 36

Change X, Z value(s) examples

This type of preselection will hardly occur in practice. For threshold cases, an understanding of the syntax specification is essential.



Set N 100:

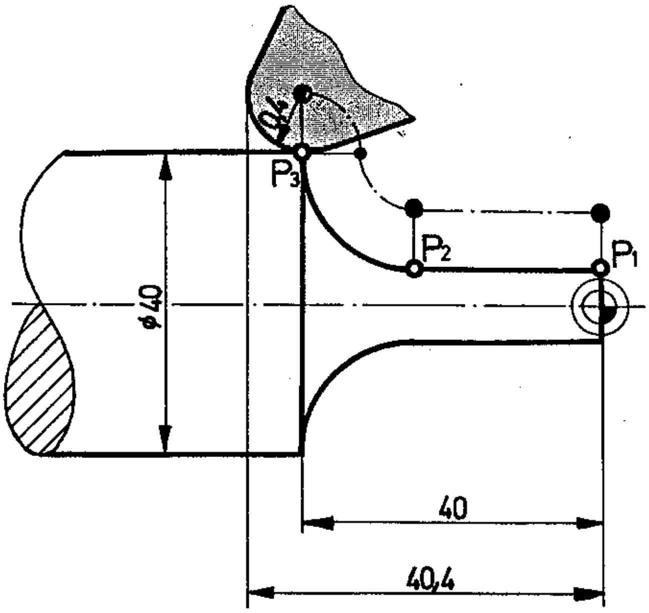
You program the tool to point P_1 (X = 30./Z = 0) G40 is active.
The theoretical cutter tip is in point P_1.

Set N 110:

The Z value is changed;
The tool does not carry out any traverse command.

```
N .../G40
N 100/G01/X 30./Z 0./
N 110/G01/(X 30.)/Z 0.4/G42
```

The same specification applies for deselection:



Set N 100:

The circular arc is actively produced with G42. The theoretical turning tool peak is at the end of the set at Z = - 40.4 mm.

Set N 110:

The Z value is changed;
The turning tool does not carry out any traverse command in this set.

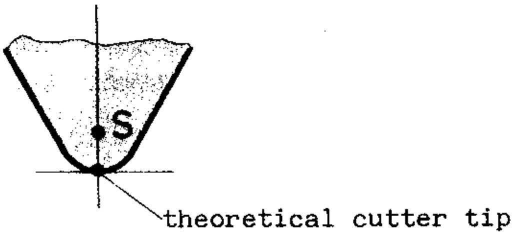
```
N .../G42
N 100/G02/X 40./Z -40./I..../
N 110/G01/(X 40.)/Z -40.4/G40
```

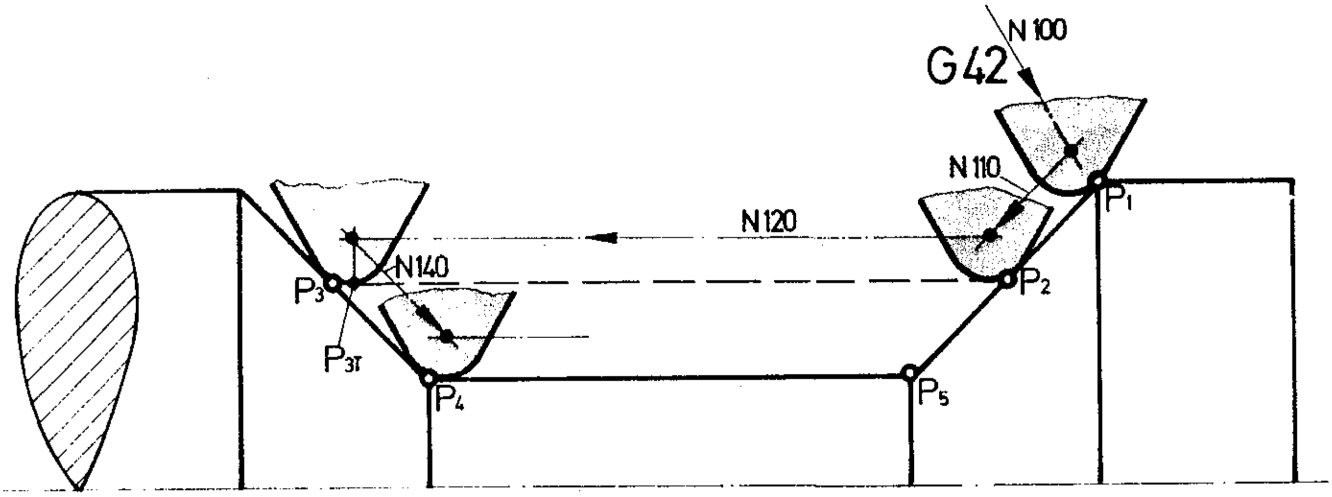
4.4 Direct change from G41 to G42

--> Alarm 53

With the radius compensation, and with reversal from G41 to G42 or vice versa, prior deselection of G40 is essential.

Example change of correction direction:





Programming:

N ..../G42

N 100/G01/X_{P1}/Z_{P1}

N 110/G01/X_{P2}/Z_{P2}

N 120/G01/X_{P3}/Z_{P3}

N 130/G01/X_{P3T}/Z_{P3T}/G40

N 140/G01/X_{P4}/Z_{P4}/G41

Set N 120:

Tool moves to P_3 as shown.

Set N 130:

No traversing movement or change in the Z-value; thus no alarm.

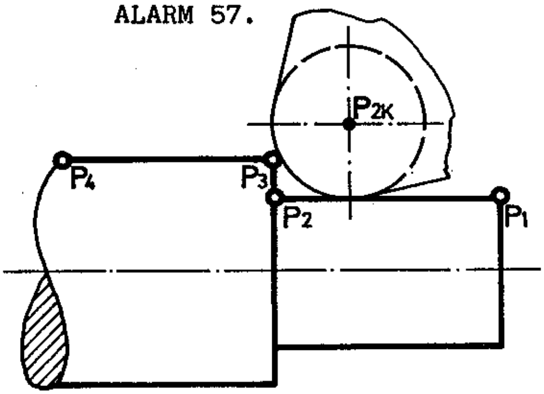
5. Geometry Alarms

The computer knows the content of the previous and the next set. It can therefore recognise contour infringements, which are caused in one previous set or one set ahead, and raise the alarm.

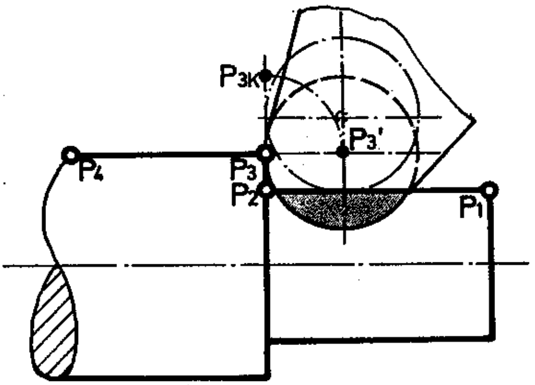
5.1 Stage smaller than cutter radius

Example 1:

Approach of the contour normal at target point would produce contour infringement in previous set --> ALARM 57.

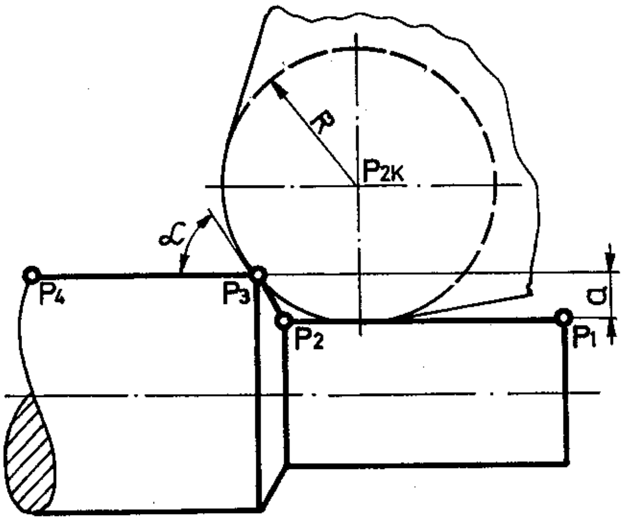


1: Tool approaches P_{2K}.

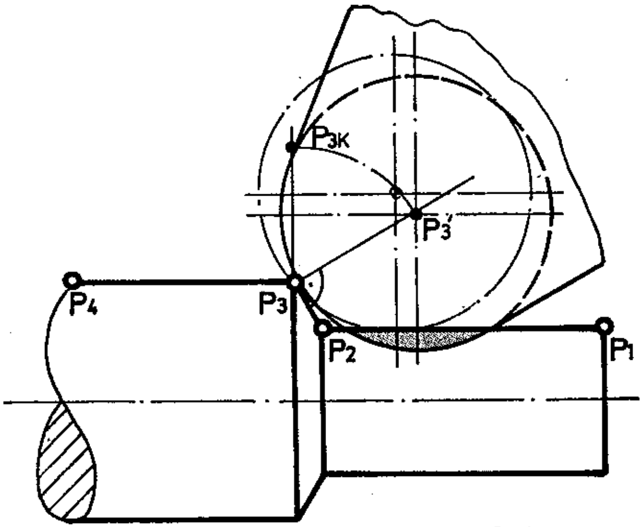


2: Tool would approach point P_3' (contour normal to $\overline{P_2P_3}$ is $\overline{P_3P_3'}$) and thereby infringe contour $\overline{P_1P_2}$. Alarm 57

Example 2:



1: Tool approaches P_{2K}.



2: Tool would approach normal to $\overline{P_2P_3}$.

Contour infringement in $\overline{P_1P_2}$ --> Alarm 57.

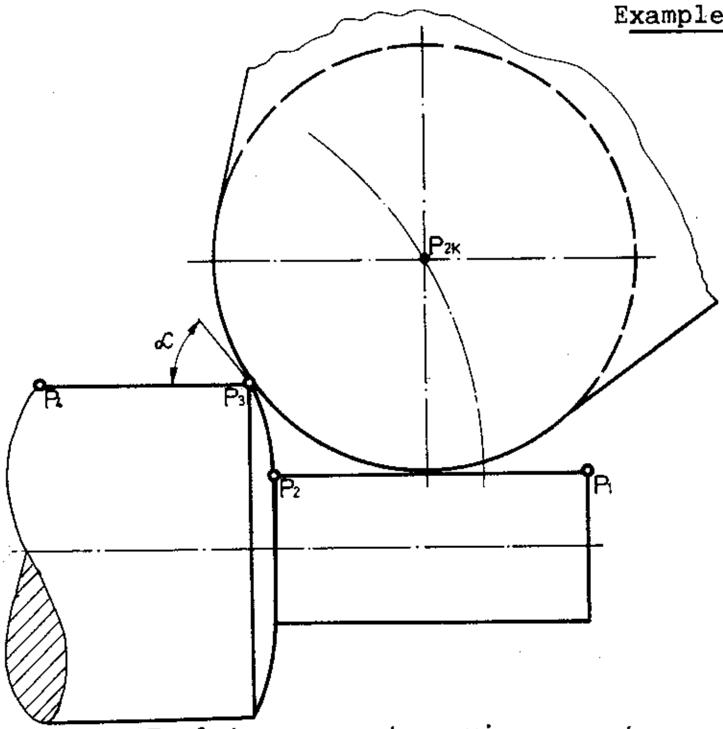
Rule:

a must be greater than or equal to R (1-cos Alpha), otherwise ALARM 57.

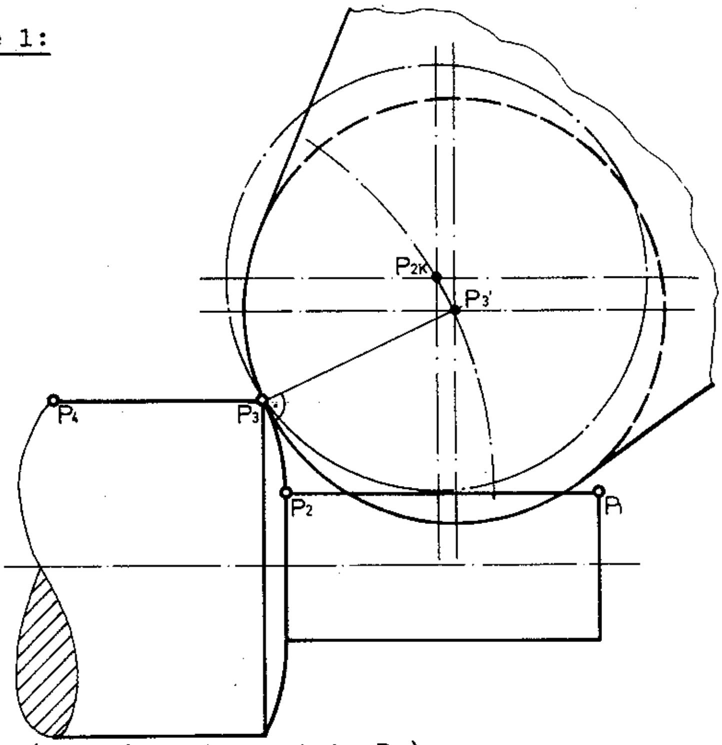
$$a \geqq R\ (1 - \cos \text{Alpha})$$

5.2 Small circle arc when compared with cutter radius

Example 1:

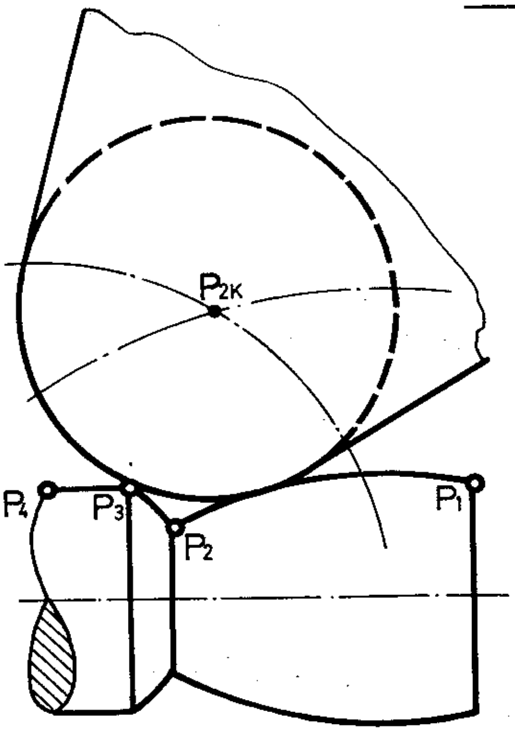


Tool traverses to equidistant intersection (P_{2K})

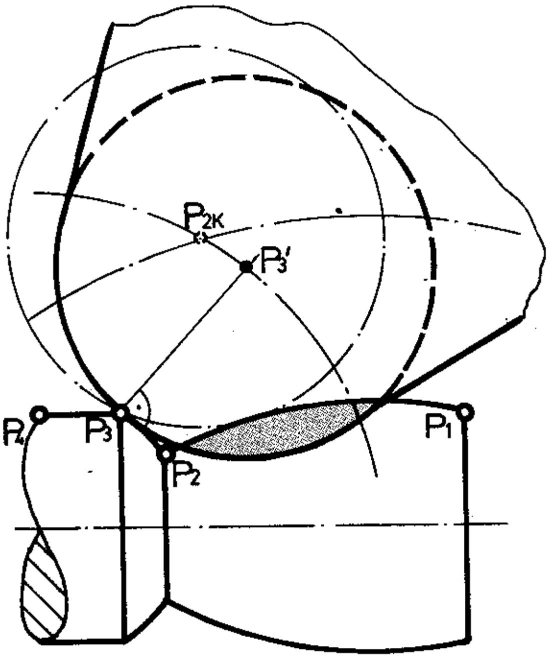


(normal at tangent to P_3)
Tool would traverse to Point P_3, and infringe contour $\overline{P_1P_2}$.

Example 2:



Tool traverses to equidistant intersection (P_{2K}).

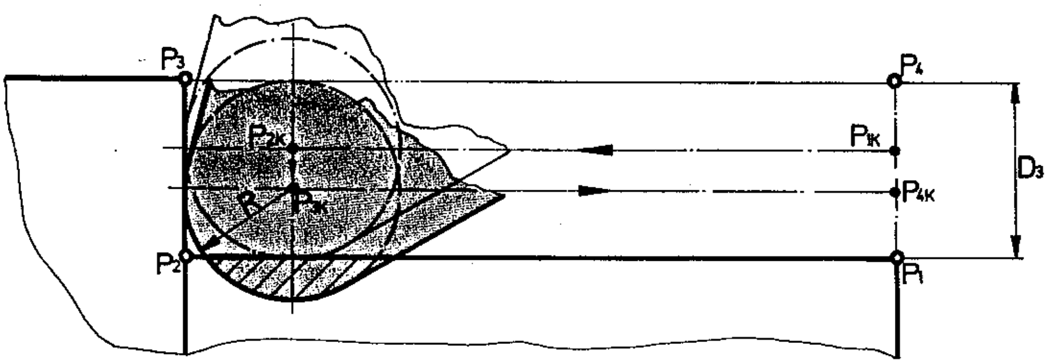


(normal at tangent P_3)
Tool would traverse to Point P_3, and infringe contour $\overline{P_1P_2}$.

5.3 Contour infringement G84

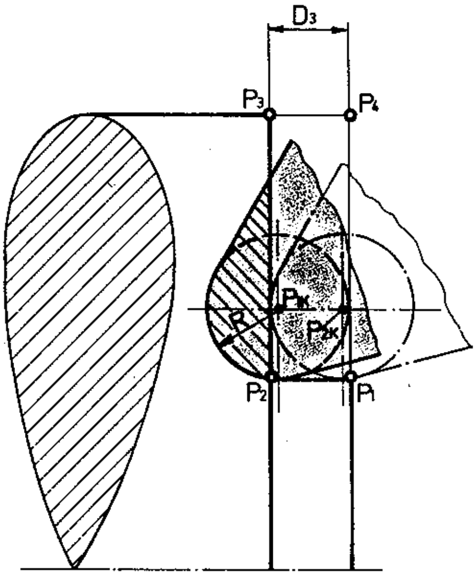
Feed depth $D_3 <$ 2x radius cutter tip → Alarm.

Longitudinal turning cycle



- Tool approaches equidistant intersection P_{2K}.
- Computer also views path $\overline{P_3 - P_4}$ as contour and would approach P_{3K} → Alarm.

Face turning cycle



- Tool approaches equidistant intersection P_{1K}.
- Computer also views path $\overline{P_2 - P_3}$ as contour and would approach P_{2K} → Alarm.

Rule:

$D_3 \geq 2 \times R$

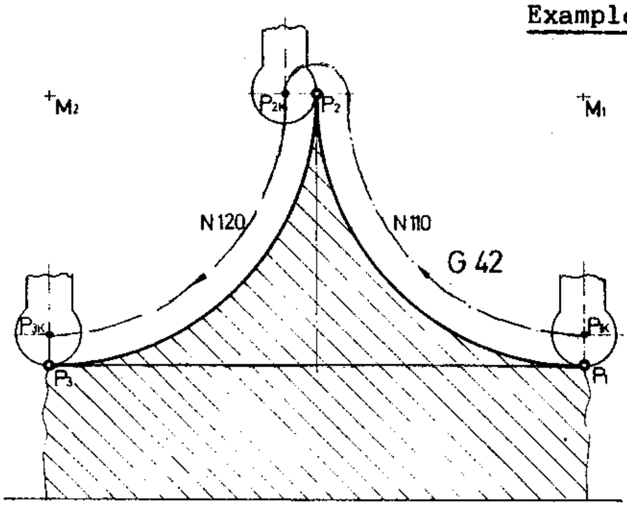
5.4 Contour Infringements with Circle Arcs

With unfavourable geometric configurations, minimum inaccuracies in centre point details lead to contour infringements.

For this reason:

The determining centre point coordinates must input precisely (in threshold cases larger).

Example:



N ... G42

N 100/G01/X_{P1} /Z_{P1}

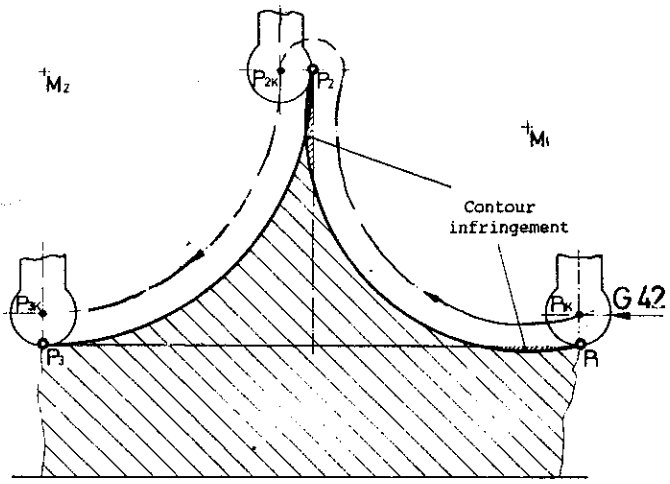
N 110/G02/X_{P2} /Z_{P2} / I_{M1} /K_{M1}

N 120/G02/X_{P3} /Z_{P3} / I_{M2} /K_{M2}

N 130

The contour is produced exactly; centre point coordinates are precisely specified.

Inaccurate centre point data:



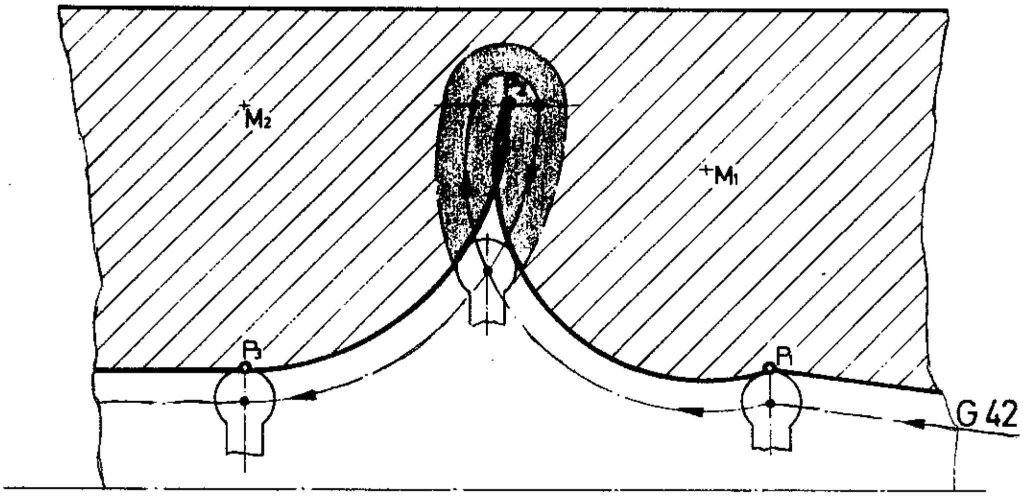
Through the specification of the centre point, the radius size is determined.
Where the centre point coordinates are specified too small, a contour infringement occurs.
Since the control knows the content of the next set, Alarm 57 is raised.

Theoretical example for easier understanding:

Internal turning

Where the same contour during internal turning with G42 and wrong centre point details are programmed, the internal corner changes into an outward corner.

Consequence: Serious contour infringement



For this reason:

The determining centre point coordinates must be input precisely (in threshold cases larger).

G53 - G59 Zero point shift with position shift register

The shift values are written with X and Z in the position shift register.

The shift values of the position shift register can be recalled with G54, G55, G57, G58 and G59.

G54, G55 are cancelled with G53.
G57, G58 and G59 are cancelled with G56.

Group division:

Group 3	G53	Cancellation of G54, G55
	G54 ≙ 1 G55 ≙ 2	Call position shift register (PSO) 1, 2
Group 5	G53	Cancellation of G57, G58, G59
	G57 ≙ 3 G58 ≙ 4 G59 ≙ 5	Call position shift register (PSO) 3, 4, 5

For details see section zero point shift.

G70 - Programming in inch

N4	G70

Where G70 is written at the beginning of a program, all dimensions are computed according to the imperial system of measurement.

G71 - Programming in mm

N4	G71

Where G71 is written at the beginning of a program, all dimensions are computed according to the metric system.

Notes on G70/G71:

* In the subroutine mode operator monitor (MON), the actuation condition G70 or G71 can be specified with parameter O_{11} Bit 0 /see section EMCOTRONIC monitor).

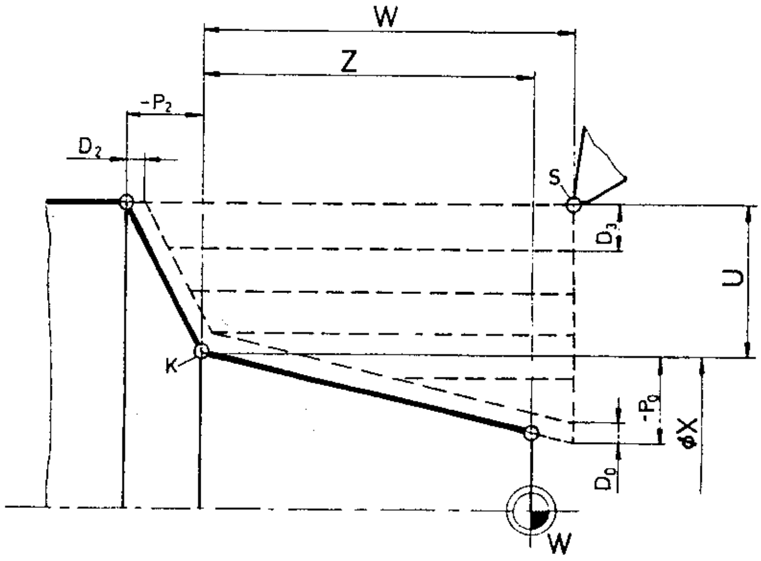
* G70/G71 are self-holding functions from the same group.

Actuation condition:

From the factory:	Europe: G71
	USA: G70

G84 - Longitudinal turning cycle

N4	G84	X U ±43	Z W ±43	P_0 P_2 ±43	D_0 D_2 5	D_3 5	F 4
		[mm]	[mm]	[mm]	[µm]	[µm]	[µm/rev.] [mm/min]



Programming:

N	Block number
G84 ..	Longitudinal turning cycle
X, U Z, W	Absolute, incremental co-ordinates of the contour joint (K)
P_0 ...	Taper dimension in X(U) (Def.)
P_2 ...	Taper dimension in Z(W) (Def.)
D_0 ...	Allowance X(U) (Def.)
D_2 ...	Allowance Z(W) (Def.)
D_3 ...	Cut division (Def.)
F	Feed

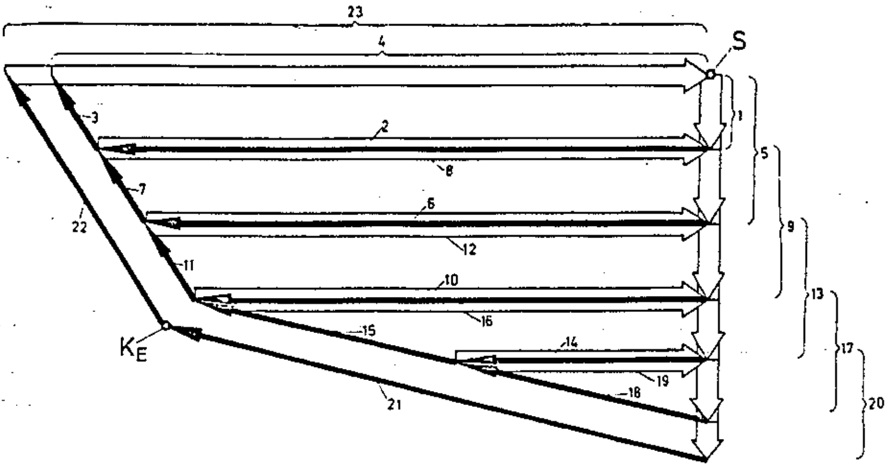
Notes:

1. In the longitudinal turning cycle, X(U) must be programmed prior to Z(W), otherwise this cycle will be considered as a face turning cycle by the control.

2. Longitudinal and face turning cycles are geometrically equal. However, the movement sequence differs. Please take this into account so as to avoid collisions.

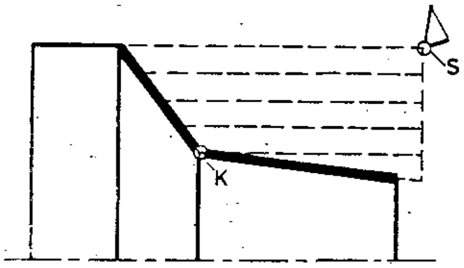
3. The parameters P_0, P_2, D_0, D_2, D_3 are marked with Def. (Default Option).
Default parameters can be programmed. The function of these parameters is explained in the examples G84 - longitudinal turning cycle.

Movement sequence:



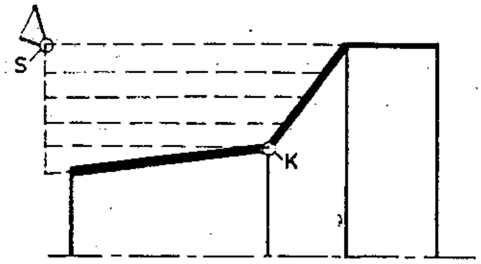
Types of cycles

Depending on the position prior to the start point (S) and the contour corner point (K), 4 types of cycle can be programmed.



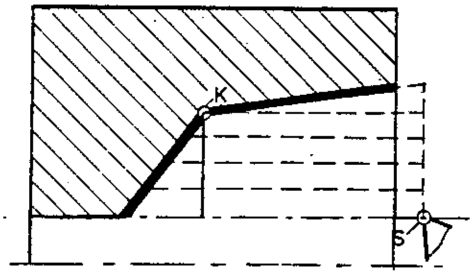
1st possibility:

External turning from right to left.
N...../G84/X(-U)/-Z(-W)/-P_0/-P_2/D_3/F.....



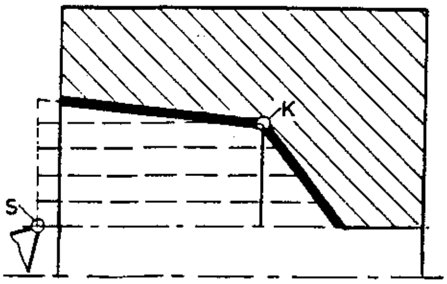
2nd possibility:

External turning from left to right.
N...../G84/X(-U)/Z(W)/-P_0/ P_2/D_3/F.....



3rd possibility:

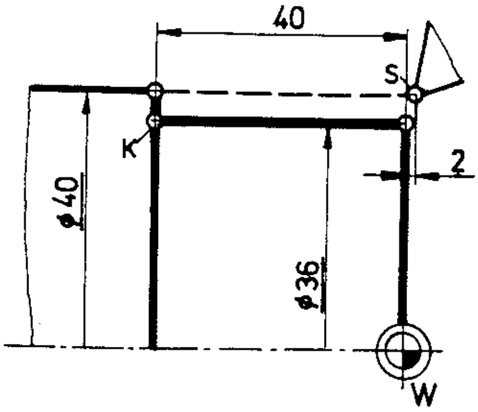
Internal turning from right to left.
N...../G84/X(U)/-Z(-W)/ P_0/-P_2/D_3/F.....



4th possibility:

Is hardly used when turning!

Examples G84 - Longitudinal turning cycle

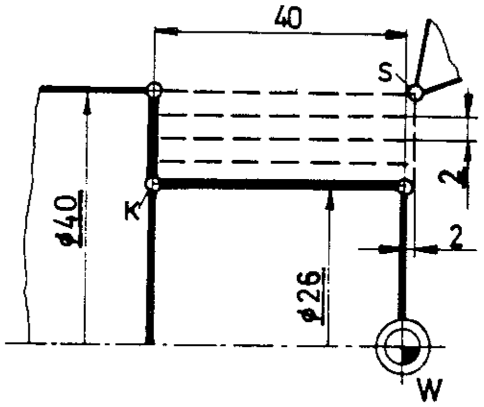


1st Example:
Longitudinal turning cycle without cut division D3.

Programming absolute:
N...../G00/X40.000/Z2.000
N...../G84/X36.000/Z-40.000/F.....

Programming incremental:
N...../G00/.....
N...../G84/U-2.000/W-42.000/F.....

No P0, P2 programmed → no taper dimension in X(U), Z(W)
No D0, D2 programmed → no finishing allowances in X(U), Z(W)
No D3 programmed → no cut division



2nd Example:
Longitudinal turning cycle with cut division D3.
Input of D3 in 1/1000 mm.

Programming absolute:
N...../G00/X40.000/Z2.000
N...../G84/X26.000/Z-40.000/
D3 = 2.000/F.....

Programming incremental:
N...../G00/.....
N...../G84/U-7.000/W-42.000/
D3 = 2.000/F.....

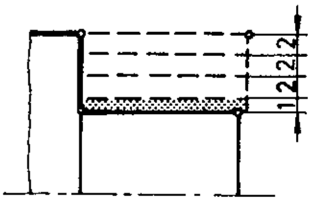
D3 programmed → cut division
No P0, P2 programmed → no taper dimension in X(U), Z(W)
No D0, D2 programmed → no finishing allowances in X(U), Z(W)

Note:
The values programmed under parameter D3 (= cut division), are divided into equal sized cuts < D3 by the control.

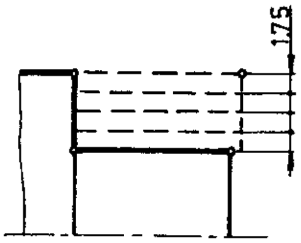
Input D3 = 2 mm
Feed U = 7 mm

Theoretic feed:
3 cuts 2 mm = 6 mm
Remainder = 1 mm
7 mm

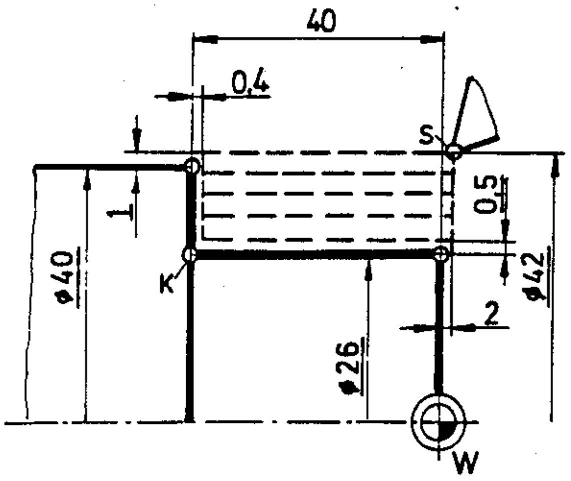
Effective feed:
4 cuts 1.75 mm = 7 mm



Theoretic feed



Effective feed



D0, D2 programmed → finishing allowances
D3 programmed → cut division
No P0, P2 programmed → no taper dimension in X(U), Z(W)

3rd Example:

Longitudinal turning cycle with cut division D3 and finishing allowance D0, D2.
D0 ..... Finishing allowance in X-direction
D2 ..... Finishing allowance in Z-direction
Input of D0, D2 in 1/1000 mm.

Programming absolute:

N...../G00/X42,000/Z2,000
N...../G84/X26,000/Z-40,000/
$D_0 = 500/D_2 = 400/D_3 = 2000$/F...

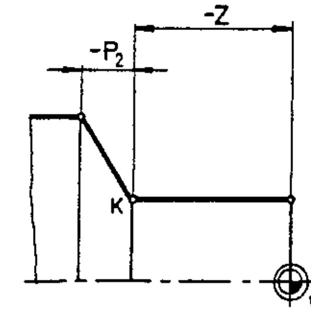
Programming incremental:

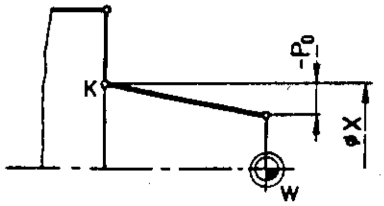
N...../G00/.....
N...../G84/U-7,000/W-42,000/
$D_0 = 500/D_2 = 400/D_3 = 2000$/F.....

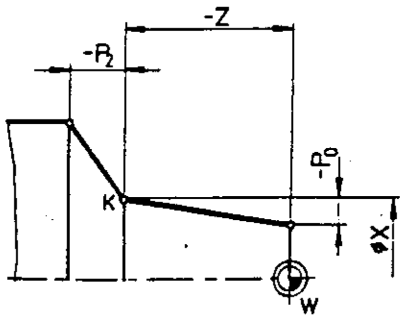
Possibility of programming the taper dimensions P0 and P2:

Attention:
An input against the feed direction made for P0 or P2, causes --> ALARM 210.

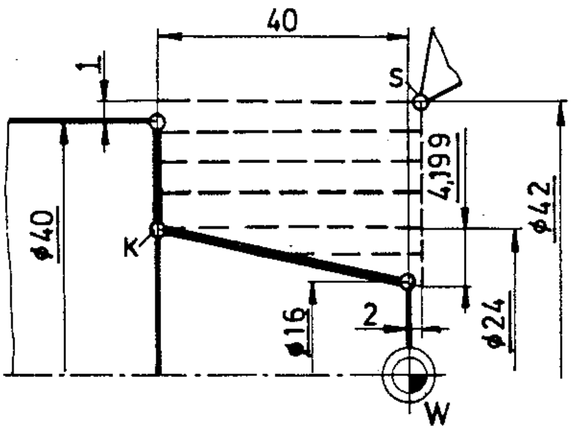
Examples for P0 and P2:







Note:
The start point in Z(W) direction is 2 mm prior to the workpiece edge. The dimension for parameter P0 must be taken into account when calculating the start point.



P0 programmed → taper dimension in X(U)
D3 programmed → cut division
No D0, D2 programmed → no finishing allowance in X(U) Z(W)
No P2 programmed → no taper dimension in Z(W)

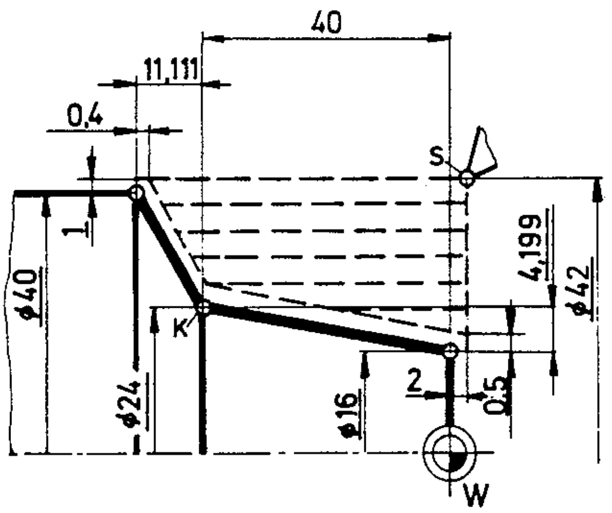
4th Example:
Taper turning: Longitudinal turning cycle with cut division D3 and taper dimension P0.
P0 ..... Taper dimension in X(U)
Input of P0 in mm.

Programming absolute:

```
N...../G00/X42.000/Z2.000
N...../G84/X24.000/Z-40.000/P0-4,199/
       D3=2000/F.....
```

Programming incremental:

```
N...../G00/.....
N...../G84/U-9.000/W-42.000/P0-4,199/
       D3=2000/F.....
```



P0, P2 programmed → taper dimensions in X(U) Z(W)
D0, D2 programmed → finishing allowances
D3 programmed → cut division

5th Example:
Taper turning: Longitudinal turning cycle with cut division D3, taper dimensions P0, P2 and finishing allowances D0, D2.

Programming absolute:

```
N...../G00/X42,000/Z2,000
N...../G84/X24,000/Z-40,000/P0-4,199/
       P2-11,111/D0=500/D2=400/
       D3=2000/F.....
```

Programming incremental:

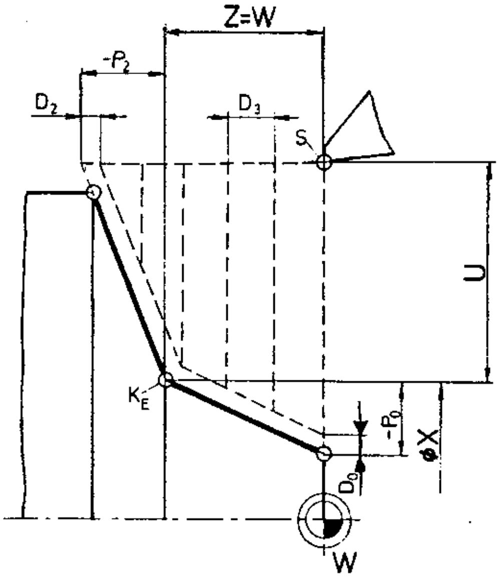
```
N...../G00/.....
N...../G84/U-9,000/W-42,000/P0-4,199/
       P2-11,111/D0=500/D2=400/
       D3=2000/F.....
```

G84 - Face turning cycle

N4	G84	Z W ±43	X U ±43	P_0 P_2 ±43	D_0 D_2 5	D_3 5	F 4
		[mm]	[mm]	[mm]	[µm]	[µm]	[µm/rev.] [mm/min]

Where co-ordinates Z(W) are programmed prior to X(U) with G84, the control carries out a face turning cycle. Longitudinal and face turning cycles are geometrically equal. However, the movement sequence differs.

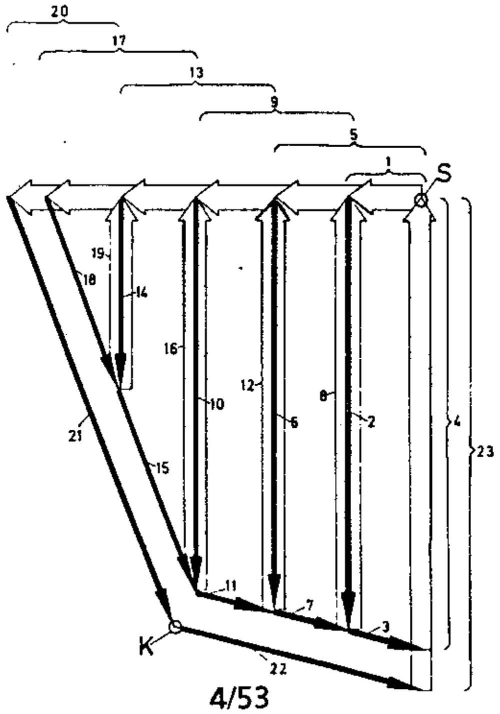
The parameters P0, P2, D0, D2, D3 are marked with Def. (Default Option). Default parameters can be programmed. The function of these parameters is explained in the examples G84 - Face turning cycle.



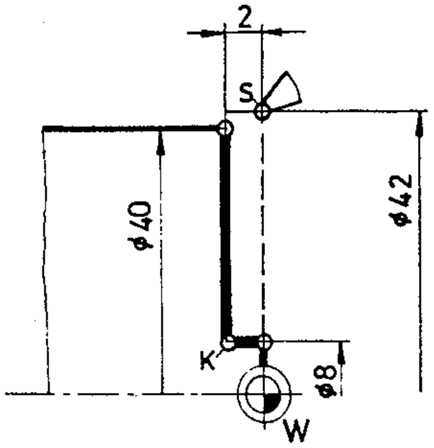
Programming:

N	Block number
G84 ..	Face turning cycle
Z, W X, U	Absolute, incremental coordinates of the contour joint (K)
P0 ...	Taper dimension in X(U) (Def.)
P2 ...	Taper dimension in Z(W) (Def.)
D0 ...	Allowance in X(U) (Def.)
D2 ...	Allowance in Z(W) (Def.)
D3 ...	Cut division (Def.)
F	Feed

Movement sequence:



Example G84 - Face turning cycle

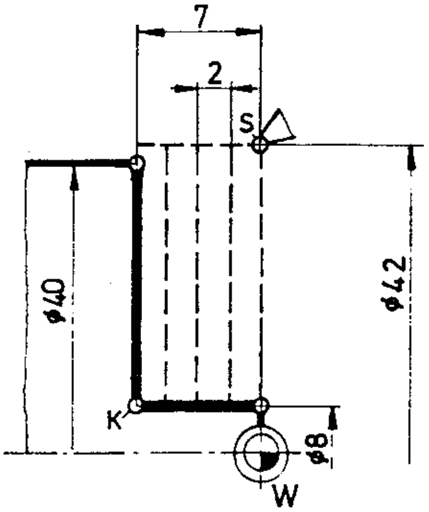


1st Example:
Face turning cycle without cut division D3.

Programming absolute:
N...../G00/X42.000/Z00.000
N...../G84/Z-2.000/X8.000/F.....

Programming incremental:
N...../G00/.....
N...../G84/W-2.000/U-17.000/F.....

No P0, P2 programmed → no taper dimension in X(U), Z(W)
No D0, D2 programmed → no finishing allowances in X(U), Z(W)
No D3 programmed → no cut division

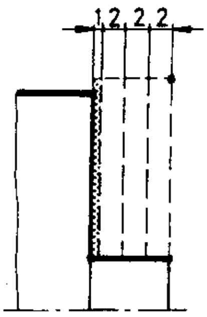


2nd Example:
Face turning cycle with cut division D
Input of D3 in 1/1000 mm.

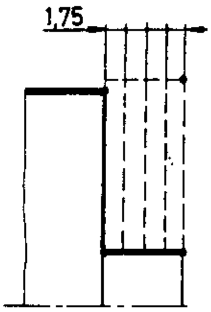
Programming absolute:
N...../G00/X42.000/Z00.000
N...../G84/Z-7.000/X8.000/
D_3 = 2.000/F.....

Programming incremental:
N...../G00/.....
N...../G84/W-7.000/U-17.000/
D_3 = 2.000/F.....

D3 programmed → cut division
No P0, P2 programmed → no taper dimension in X(U), Z(W)
No D0, D2 programmed → no finishing allowances in X(U), Z(W)



Theoretic feed



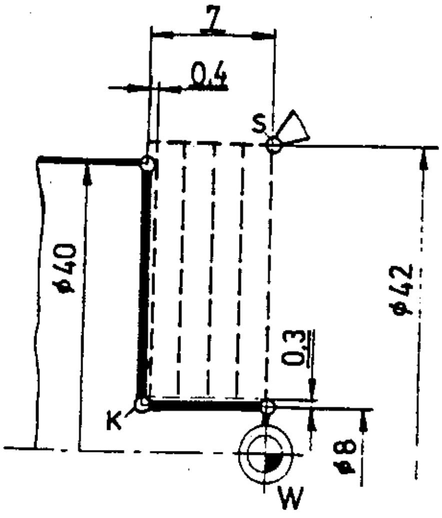
Effective feed

Note:
The values programmed under parameter D3 (= cut division), are divided into equal sized cuts < D3 by the control.

Input D3 = 2 mm
Feed U = 7 mm

Theoretic feed:
3 cuts 2 mm = 6 mm
Remainder = 1 mm
7 mm

Effective feed:
4 cuts 1.75 mm = 7 mm



D0, D2 programmed --> finishing allowances
D3 programmed --> cut division
No P0, P2 programmed --> no taper dimension
in X(U), Z(W)

3rd Example:

Face turning cycle with finishing allowances D0, D2 and cut division.
D0 ..... Finishing allowance in Z-direction
D2 ..... Finishing allowance in X-direction
Input of D0, D2 in 1/1000 mm.

Programming absolute:

N...../G00/X42,000/Z00,000
N...../G84/Z-7,000/X8,000/
D_0 = 300/D_2 = 400/D_3 = 2000/F...

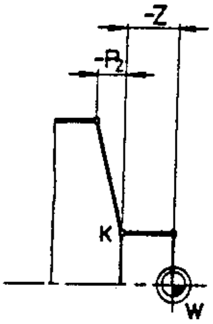
Programming incremental:

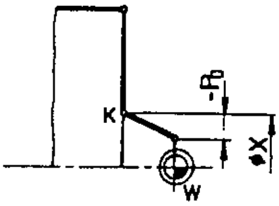
N...../G00/.....
N...../G84/W-7,000/U-17,000/
D_0 = 300/D_2 = 400/D_3 = 2000/F.....

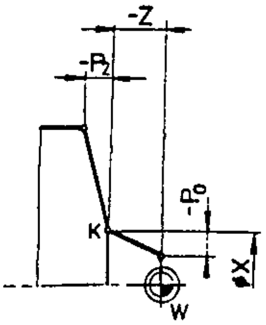
Possibility of programming the taper dimensions P0 and P2:

Attention:
An input against the feed direction made for P0 or P2, causes
--> ALARM 210.

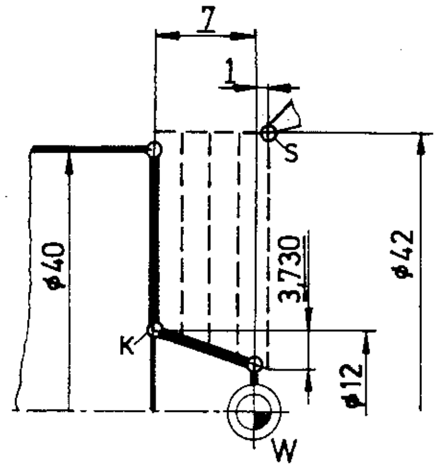
Examples for P0 and P2:







Note:
The start point in Z(W) direction is 1 mm prior to the workpiece edge. The dimension for parameter P0 must be taken into account when calculating the start point.



P0 programmed → taper dimension in X(U)
D3 programmed → cut division
No D0, D2 programmed → no finishing allowance in X(U) Z(W)
No P2 programmed → no taper dimension in Z(W)

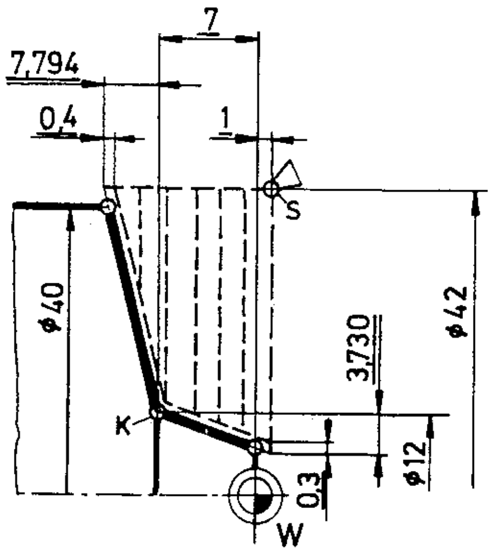
4th Example:
Taper turning: Flat turning cycle with cut division D3 and taper dimension P0.
P0 ..... Taper dimension in X(U)
Input of P0 in + mm.

Programming absolute:

N...../G00/X42,000/Z1,000
N...../G84/Z-7,000/X12,000/P_0-3,730/
D_3 = 2000/F.....

Programming incremental·

N...../G00/.....
N...../G84/W-8,000/U-15,000/P_0-3,730/
D_3 = 2000/F.....



P0, P2 programmed → taper dimensions in X(U) Z(W)
D0, D2 programmed → finishing allowances
D3 programmed → cut division·

5th Example:
Taper turning: Face turning cycle with cut division D3, taper dimensions P0, P2 and finishing allowances D0, D2.

Programming absolute:

N...../G00/X42,000/Z1,000
N...../G84/Z-7,000/X12,000/P_0-3,730/
P_2-7,794/D_0 = 300/D_2 = 400/
D_3 = 2000/F.....

Programming incremental:

N...../G00/.....
N...../G84/W-8,000/U-15,000/P_0-3,730/
P_2-7,794 /D_0 = 300/D_2 = 400/
D_3 = 2000/F.....

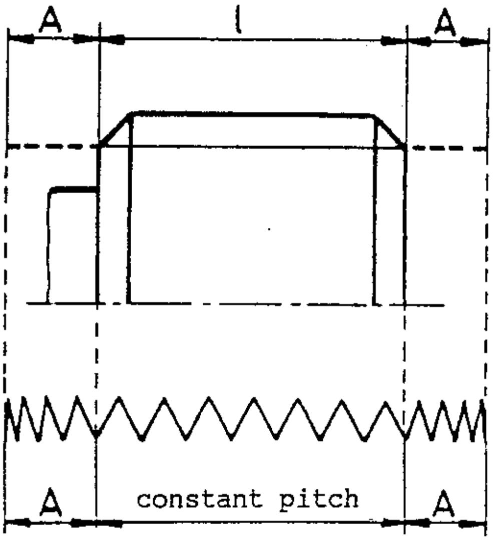
G85 - thread-cutting

INDEX

Thread-cutting cycle

1. Preliminary explanations

With the EMCOTRONIC T1 control you can program thread with a taper angle 0° to 90°. You can program very variable cycles with parameters.
On the next page you will find some fundamental explanations. They are intended to give you a better understanding of the possibilities of thread-cutting programs.



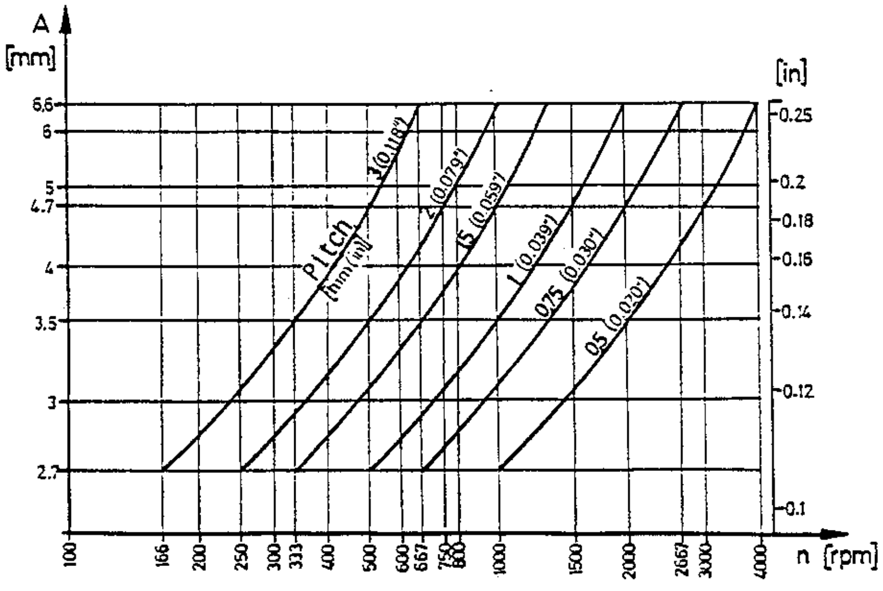
General remarks on thread approach and thread run-out

- You must accelerate the slides on "thread start".
- You must slow down the slides before the "thread end".

The pitch is not constant during the acceleration and deceleration phases. This must be taken into account during programming.
This means that the mechanical cutting process must lie within the constant pitch phase.

Example:

The table shows the relationships between pitch, speed and minimum value for approach and run-out during thread-cutting operations.

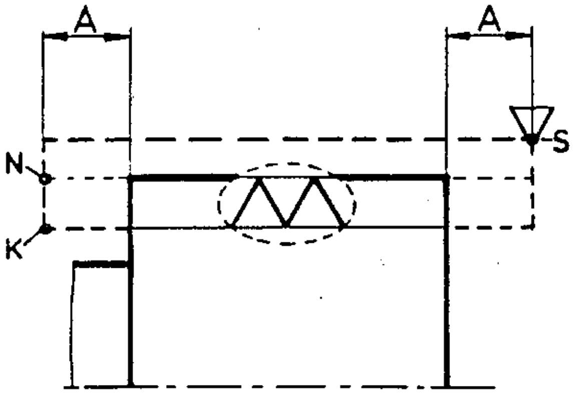


2. Geometrical definitions

Programming of starting point and thread end point

Possibility 1

(Example lengh-thread a < 45°)

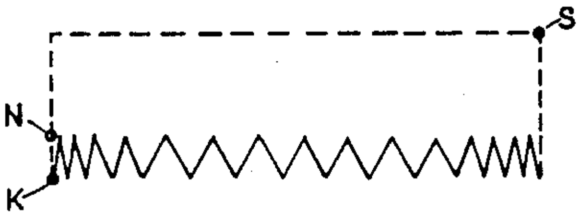


1. Establishing starting point

The starting point (S) is approached in the block before the cycle.

Z-direction:
Minimum distance A must be observed.

X-direction:
Provide distance from surface so that surface is not touched during return.

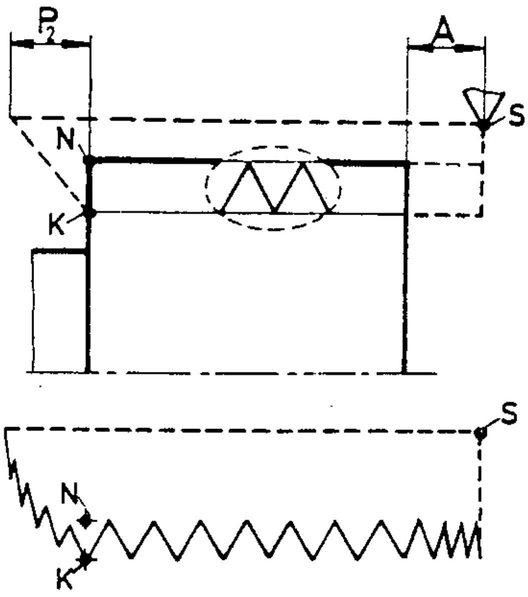


2. Establishing the thread end point

In the G85 cycle the thread end point K or N is programmed with X(U), Z(W) (for K and N, refer to parameter D_7).
The distance A must also be observed when the thread end point is being established.

Possibility 2

(Example lengh-thread a < 45°)



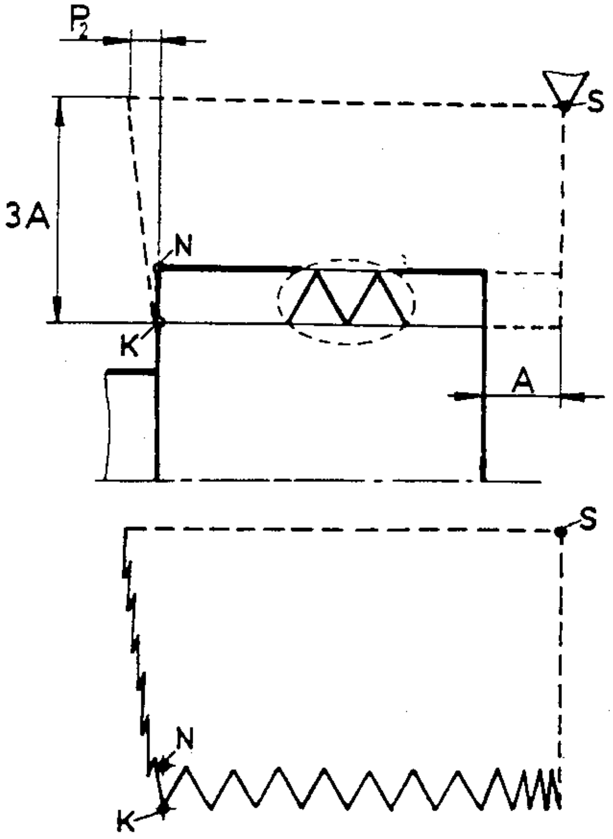
The thread approach is programmed with P_2

1. Establishing starting point

Same as possibility 1.

2. Programming K (or N) and P_2

- The effective thread end point (K or N) is programmed with X(U) and Z(W). This has the advantage that the distances from K and N can generally be taken directly from the drawing.
- The thread run-out is programmed with P_2.
  The lathe tool moves at a slant from S into the X-position.
  Advantage:
  Collision-free return is possible with small relief groove.



Conditions for the size of P_2:

* When P_2 is equal to or greater than A, no condition on the return value in X-direction.

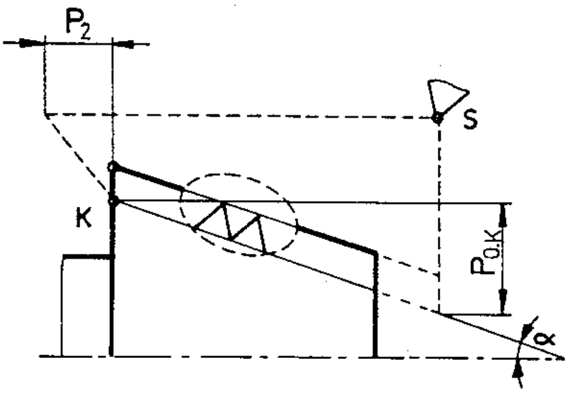
* When P_2 is smaller than A, 3 times the A path must be provided for the return motion in the X-direction. This means that you must set the starting point at an appropriate distance.

Notes:

If the value of P_2 falls short of the admissible value, an exact stop occurs at the programmed thread end point (K or N). The thread pitch cannot be switched on during the deceleration process. With default programming (optional programming) of P_2 (P_2 has not been programmed or P_2 = 0) no oblique thread run-out occurs.

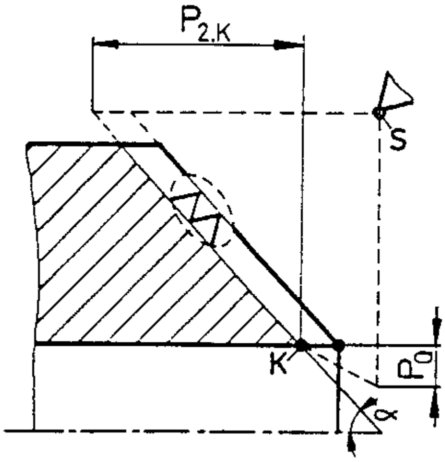
Notes to parameter P_0/P_2

By programming P_0/P_2 take care of the changin of the parameters by lengh - eg. traverse thread.



1. Length thread

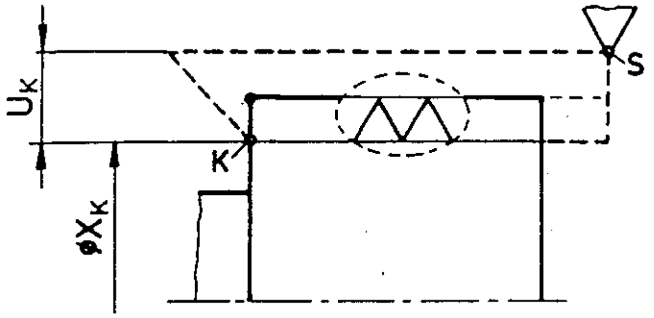
Parameter P_0 ... taper in ± mm
Parameter P_2 ... thread run out in ± mm



2. Transverse thread

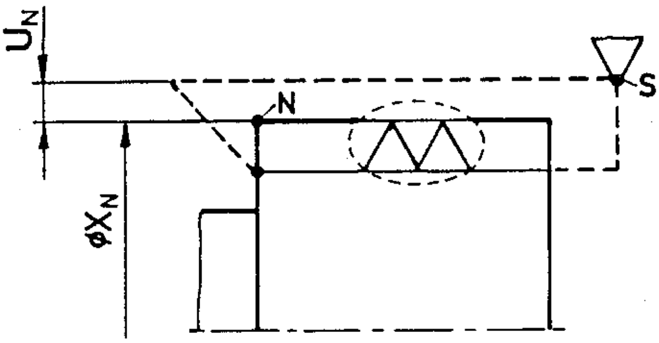
Parameter P_0 ... thread run out in ± mm
Parameter P_2 ... taper in ± mm

Programming of the minor diameter K or the nominal diameter N - Parameter D_7



On the Emcotronic you can program either
K... (thread end point of the minor diameter) or
N... (thread end point of the nominal diameter).

This is reported to the control with the parameter D_7.

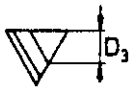
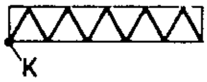
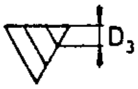
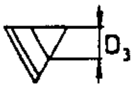
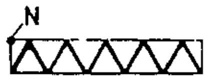
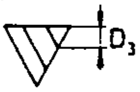
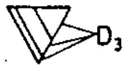
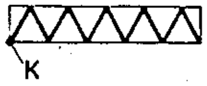
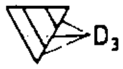
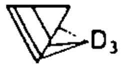
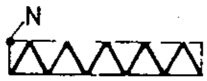
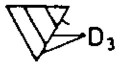


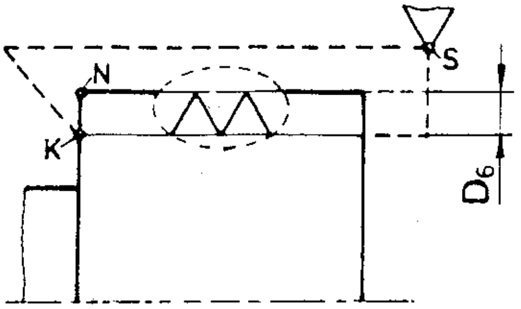
Programming of K

- No D_7 programmed (default = optional)
- D_7 = 0, 1, 4, 5

Programming of N

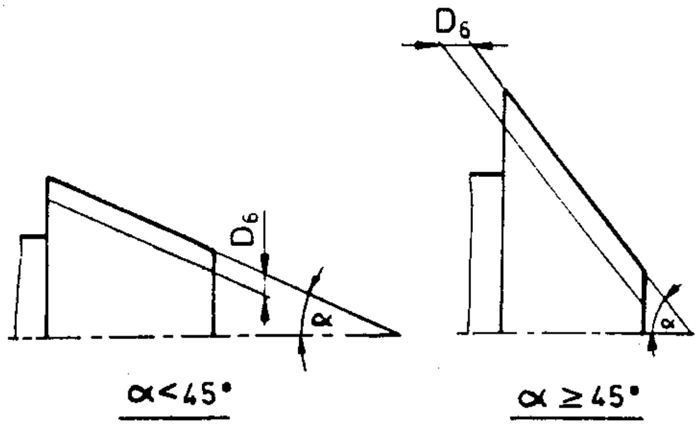
D_7 = 2, 3, 6, 7

No D_7 D_7=0	Dec.: 	
D_7=1	Konst.: 	
D_7=2	Dec.: 	
D_7=3	Konst.: 	
D_7=4	Dec.: 	
D_7=5	Konst.: 	
D_7=6	Dec.: 	
D_7=7	Konst.: 	



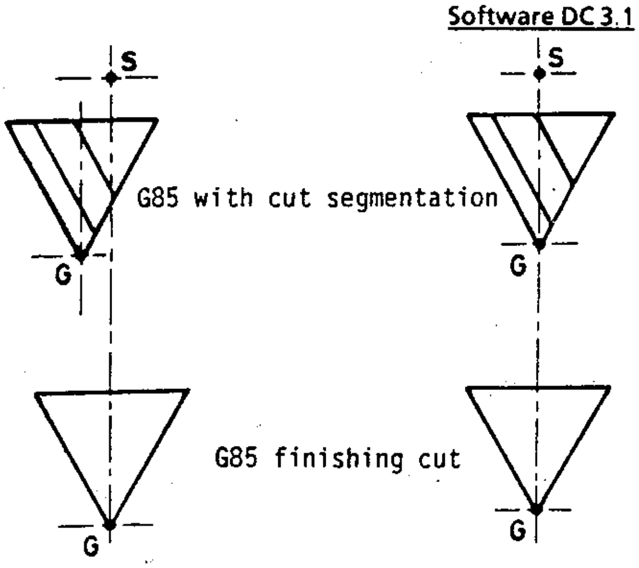
Programming of the depth of thread D_6

The depth of thread is programmed wit the parameter D_6.
Entry of D_6: (μm)



D_6 entry with tapered threads.

Innovation from software DC 3.1



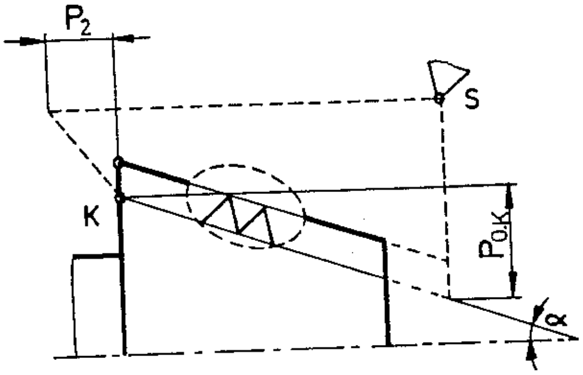
Position of the thread root:
With software under DC 3.1 the positio of the thread root G depended on the initial infeed. If a thread was manufactured with several G85 blocks, the same initial infeed D_3 always had to be programmed.
With software DC 3.1 the position of the thread root G remains constant. You can produce a thread in several G85 blocks without having to program the same initial infeed.

Application:
A thread manufactured with G85 is further processed in a second G85 block with a finishing tool; S (starting point) is now the same in both G85 blocks.

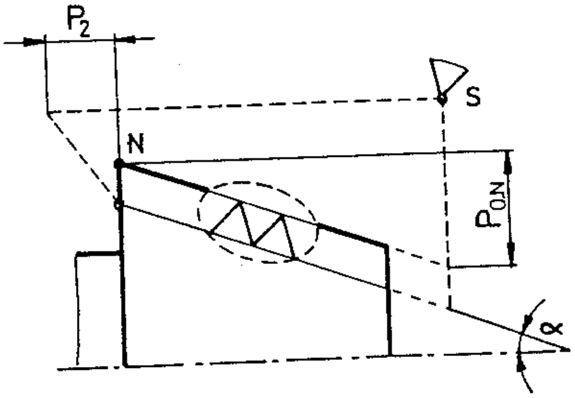
Taper by lengh-thread α < 45°

Programming of the taper P_0

The control calculates the taper angle from the value which is entered under P_0.
Entry: (+/- mm)



P_0 for programming minor diameter (K)



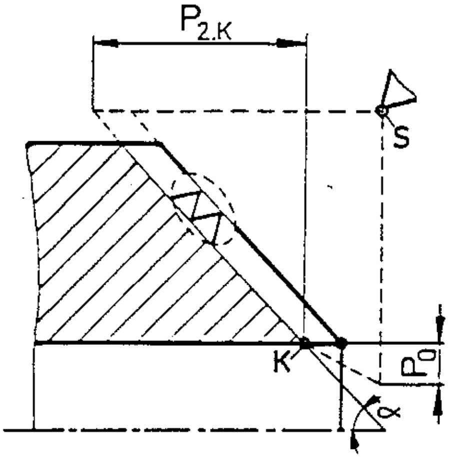
P_0 for programming nominal diameter (N)

Notes:
By programming P_0/P_2 take care at the change of the parameters by length e.g transverse thread.

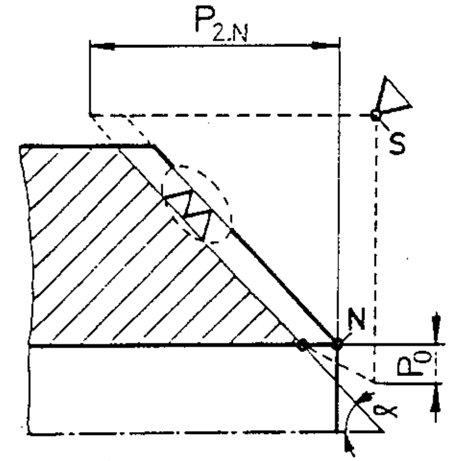
Taper by transverse thread α > 45°

Programming P_2

The control calculates the taper angle which is entered under P_2
Entry: (± mm)

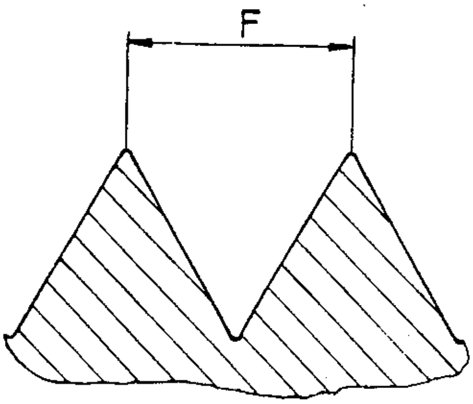


P_2 for programming minor diameter K



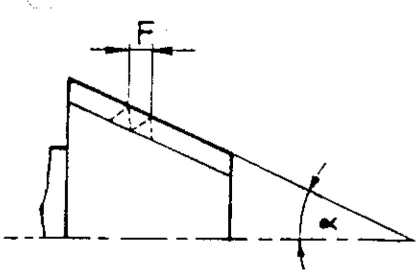
P_2 for programming nominal diameter N

Notes:
By programming P_0/P_2 take care at the change of the parameter ba lengh - e.g. transverse thread.

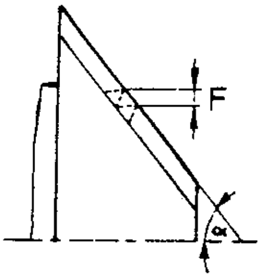


Programming of the thread pitch F

The thread pitch is programmed with F. The last effective feed is cancelled as long as G85 is active. Feed corrections with the override switch or an interruption with the Feedhold key are not executed until after the thread cuts. During thread--cutting the feed override is set internally to 100%. Entry: (µm).



∝<45°



∝ ≥45°

Entry of the pitch

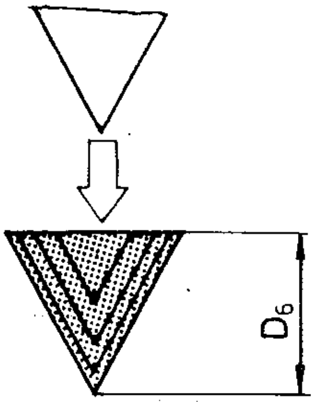
∝<45° longitudinal tapered thread, F is specified parallel to the Z-axis.

∝≥45° transverse tapered thread, F is specified parallel to the Z-axis.

3. Technological definitions

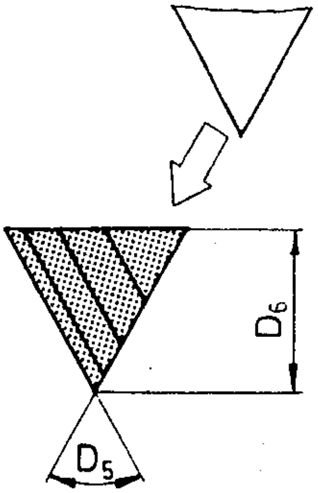
Infeed angle D_5:

The infeed of the threading tool can be programmed as plunge threading or compound threading.



Plunge threading:

The plunge threading is aktive with the default programming (optinonal programming) of D_5 (= D_5 not programmed) or if D_5 = 0 has been entered.



Compound threading:

Compound threading is aktive when a thread angle, see table, is programmed under D_5.
The infeed angle is less then half the angle of thread.

Thread angle (D5)	Infeed angle
40°	19°
55°	26°
60°	29°
80°	39°

If a value other than 0, 40, 55, 60 or 80 is programmed under D_5, ALARM 20 is given.

Estabilshing the no-load cuts D_4:

The number of no-load cuts required to clean and deburr a thread can be established with D_4.

Entry range: 0 to 20

If D_4 is not programmed (default = optional programming), the number of no-load cuts established on the user-monitor is executed.
D_4 = 1 has been set on the user-monitor at the works but this figure can be changed on the monitor.

The cut segmentation (decremental, constant)
The depth oft cut or number of cuts
D_3/D_7

D_3 and D_7 are combined parameters. With D_7 you establish
* whether D_3 is the number of cuts or depth of cut
  and
* whether the infeed is constant or incremental.

D_3/D_7 combinations

D_3 = depth of cut

If no D_7 or D_7 = 0, 1, 2, 3 is programmed

D_7 = 0,2 decremental infeed

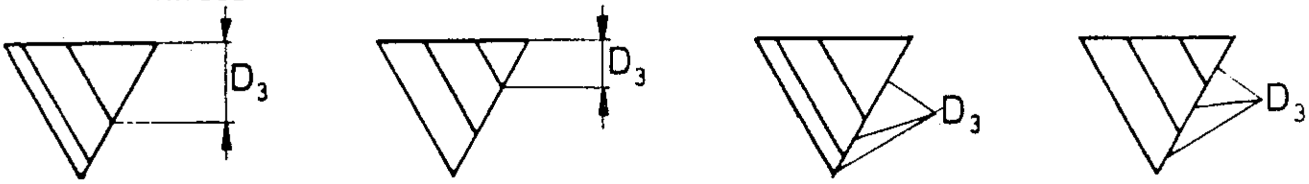
D_7 = 1,3 constant infeed

D_3 = number of cuts

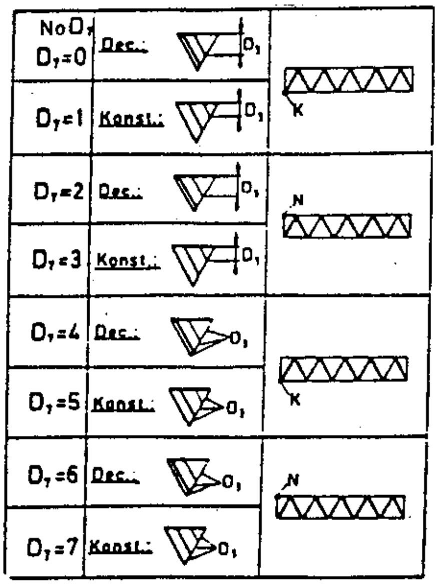
If D_7 = 4, 5, 6, 7 is programmed

D_7 = 4,6 decremental infeed

D_7 = 5,7 constant infeed



Parameter D_3/D_7

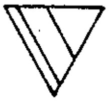


No D_7 D_7=0	Dec.:	D_3	K
D_7=1	Konst.:	D_3	
D_7=2	Dec.:	D_3	N
D_7=3	Konst.:	D_3	
D_7=4	Dec.:	D_3	K
D_7=5	Konst.:	D_3	
D_7=6	Dec.:	D_3	N
D_7=7	Konst.:	D_3	

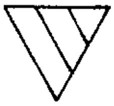
Note:

The plunge theading or compound threading can be established with parameter D_5.
The compound threading was shown in the outline table. All D_3 + D_7 combinations are also possible with plunge threading.

Relevant explanations

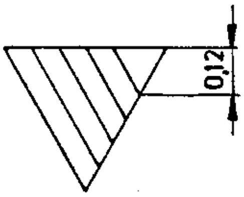


- Decremental infeed
  With decremental infeed the depth of cut decreases so that the chip section remains constant.



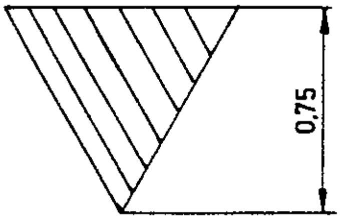
- Constant infeed
  The infeed takes place in individual passes with the depth of cut D_3.

- Establishment on the user monitor
  The minimum infeed is establisned on the user monitor with D_3.The value D_3 = 100 µm has been set at the works.



Examples:

1. You have selected decremental infeed and entered the depth of cut D_3 at 120. With the second infeed this value (120) would be fallen short of. As the minimum infeed depth is established on the user monitor at 100, the subsequent infeeds are executed at 100.



2. The depth of cut D_6 is 0.75 mm. You program a cut number of D_3 = 15. The thread is cut in 8 infeeds because D is established on the user monitor at 100 µm. Therefore, if you want smaller infeeds, you must change D_3 on the user monitor.

Note:

The plunge theading or compound threading can be established with parameter D_5.
The compound threading was shown in the outline table. All D_3 + D_7 combinations are also possible with plunge threading.

4. Outline of cycles

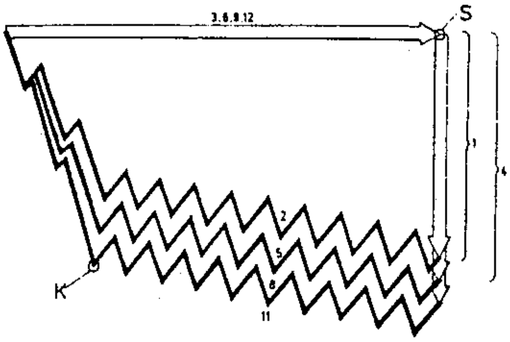
Difference between longitudinal and transverse thread-cutting cycles

Thread <45 °
(longitudinal thread-cutting cycle)

The coordinate X (U) must be programmed before Z (W).

Movement sequence

The first movement is an X-movement (infeed).

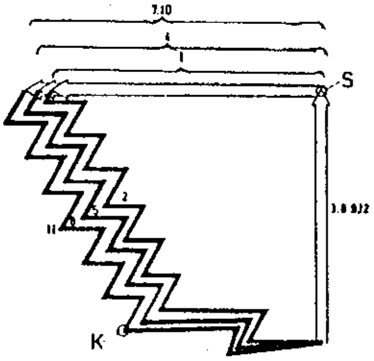


Thread ≥45 º
(transverse thread-cutting cycles)

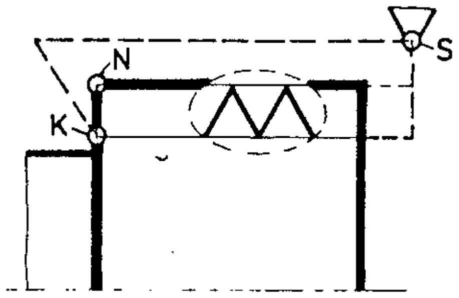
The coordinate Z (W) must be programmed before X (U).

Movement sequence

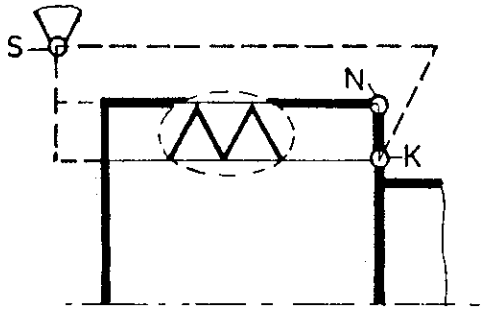
The first movement is a Z-movement (infeed).



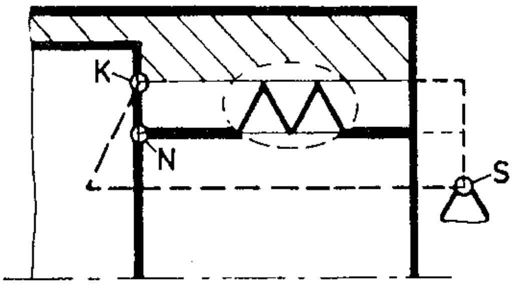
Application possibilities G85



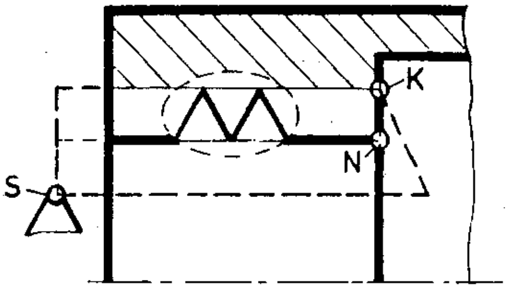
Right-hand thread, external
N...../G85/X(-U)/-Z(-W)/.....
(M04)



Left-hand thread, external
N...../G85/X(-U)/Z(W)/.....
(M04)



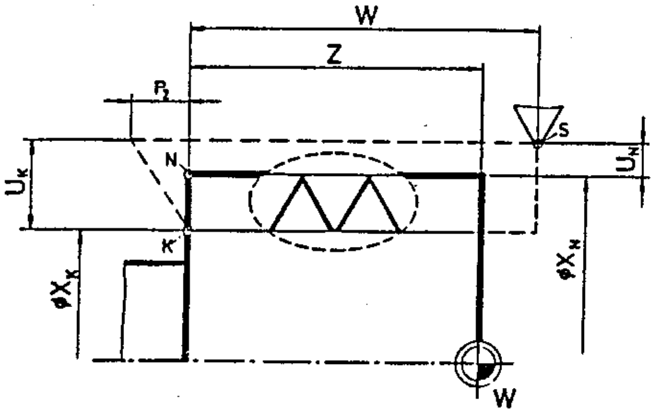
Right-hand thread, internal
N...../G85/X(U)/-Z(-W)/.....
(M03)

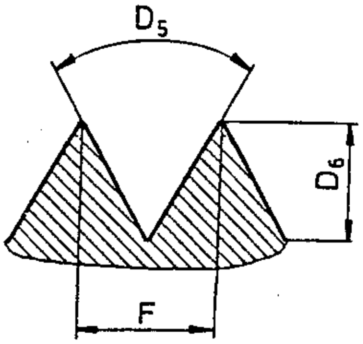


Left-hand thread, internal
N...../G85/X(U)/Z(W).....
(M03)

Longitudinal thread-cutting cycle (cylindrical)

N4	G85	X U ±43	Z W ±43	P_2 ±43	$D_3$5	$D_4$2	$D_5$2	$D_6$5	$D_7$1	F4
		[mm]	[mm]	[mm]	[µm] []	[]	[°]	[µm]	[ ]	[µm]

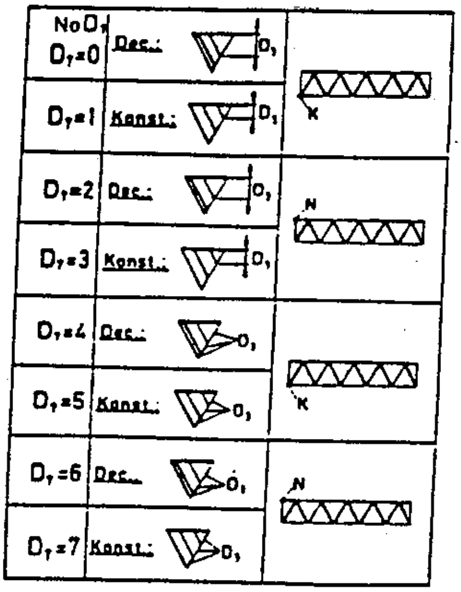




Programming:

N......block number
G 85...thread-cutting cycle
X, U } absolute, incremental
Z, W } coordinates of the thread end point K or N
P_2.....thread run-out (def.)
D_3.....see table
D_4.....number of no-load cuts (def.)
D_5.....angle of thread (def.)
D_6.....depth of thread
D_7.....see table (def.)
F......thread pitch

Parameter D_3/D_7



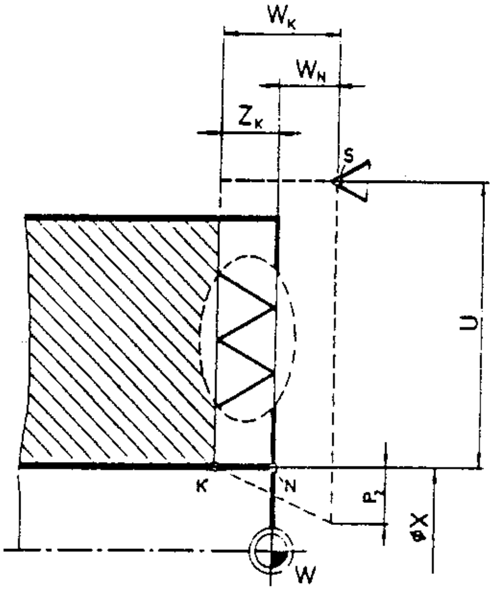
No D_7 D_7=0	Dec.: D_3	K
D_7=1	Konst.: D_3	
D_7=2	Dec.: D_3	N
D_7=3	Konst.: D_3	
D_7=4	Dec.: D_3	K
D_7=5	Konst.: D_3	
D_7=6	Dec.: D_3	N
D_7=7	Konst.: D_3	

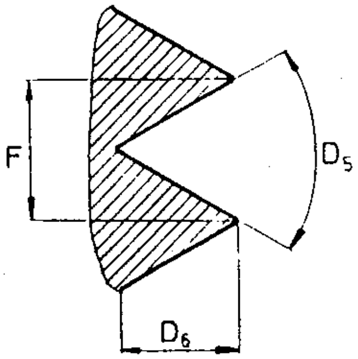
Parameter D_5

Thread angle (D5)	Infeed angle
40°	19°
55°	26°
60°	29°
80°	39°

Transverse thread-cutting cycle (cylindrical)

N4	G85	Z W ±43	X U ±43	P_2 ±43	$D_3$5	$D_4$2	$D_5$2	$D_6$5	$D_7$1	F4
		[mm]	[mm]	[mm]	[µm] []	[]	[°]	[µm]	[ ]	[µm]

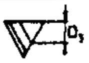
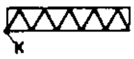
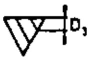
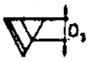
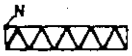
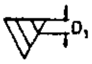
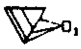
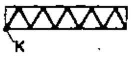
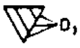
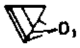
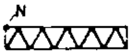
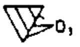




Programming:

N......block number
G 85...thread-cutting cycle
Z, W } absolute, incremental
X, U } coordinates of the thread end point K or N
P_2.....thread run-out (def.)
D_3.....see table
D_4.....number of no-load cuts (def.)
D_5.....angle of thread (def.)
D_6.....depth of thread
D_7.....see table (def.)
F......thread pitch

Parameter D_3/D_7

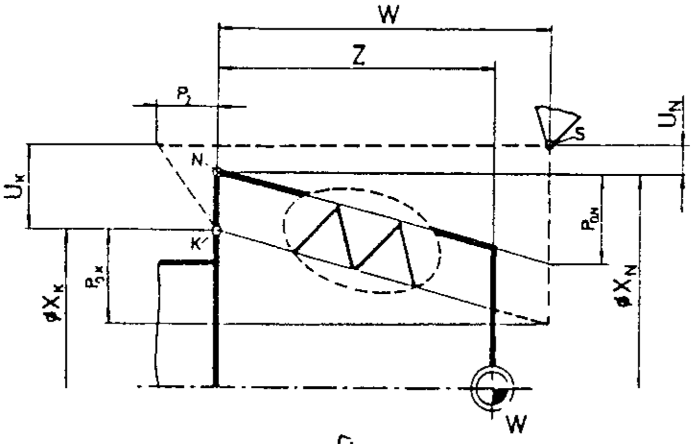
No D_7			
D_7=0	Dec.:		
D_7=1	Konst.:		
D_7=2	Dec.:		
D_7=3	Konst.:		
D_7=4	Dec.:		
D_7=5	Konst.:		
D_7=6	Dec.:		
D_7=7	Konst.:		

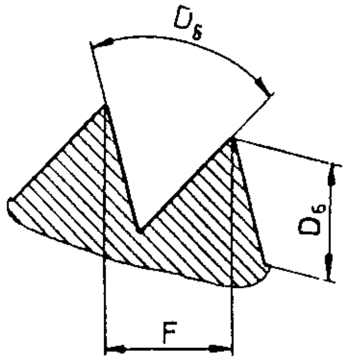
Parameter D_5

Thread angle (D5)	Infeed angle
40°	19°
55°	26°
60°	29°
80°	39°

Longitudinal thread-cutting cycle (tapered)

N4	G85	X U ±43	Z W ±43	P_0 ±43	P_2 ±43	$D_3$5	$D_4$2	$D_5$2	$D_6$5	$D_7$1	F4
		[mm]	[mm]	[mm]	[mm]	[µm] []	[]	[°]	[µm]	[ ]	[µm]

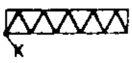
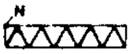
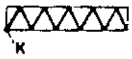
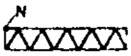




Programming:

N......block number
G 85...thread-cutting cycle
X, U } absolute, incremental
Z, W } coordinates of the threac end point K or N
P_0.....taper (def.)
P_2.....thread run-out (def.)
D_3.....see table
D_4.....number of no-load cuts (def.)
D_5.....angle of thread (def.)
D_6.....depth of thread
D_7.....see table (def.)
F......thread pitch

Parameter D_3/D_7

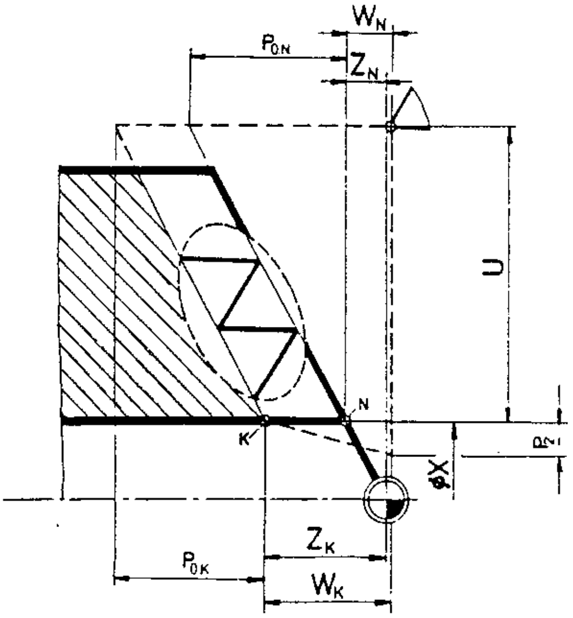
No D7		
D_7=0	Dec.. D_3	
D_7=1	Konst.. D_3	
D_7=2	Dec.. D_3	
D_7=3	Konst.. D_3	
D_7=4	Dec.. D_3	
D_7=5	Konst.. D_3	
D_7=6	Dec.. D_3	
D_7=7	Konst.. D_3	

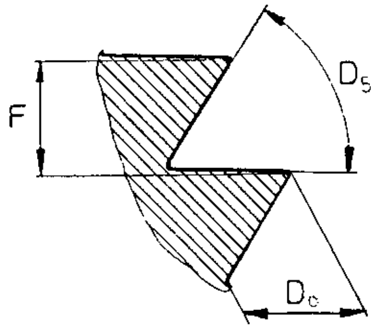
Parameter D_5

Thread angle (D5)	Infeed angle
40°	19°
55°	26°
60°	29°
80°	39°

Transverse thread-cutting cycle tapered

N4	G85	Z W ± 43	X U ± 43	P_0 ± 43	P_2 ± 43	$D_3$5	$D_4$2	$D_5$2	$D_6$5	$D_7$1	F4
		[mm]	[mm]	[mm]	[mm]	[µm] []	[]	[°]	[µm]	[ ]	[µm]

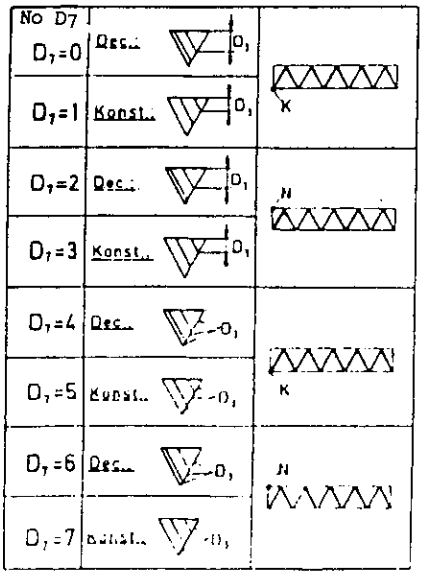




Programming:

N......block number
G 85...thread-cutting cycle
Z, W } absolute, incremental
X, U } coordinates of the thread end point K or N
P_0.....taper (def.)
P_2.....thread run-out (def.)
D_3.....see table
D_4.....number of no-load cuts (def.)
D_5.....angle of thread (def.)
D_6.....depth of thread
D_7.....see table (def.)
F......thread pitch

Parameter D_3/D_7

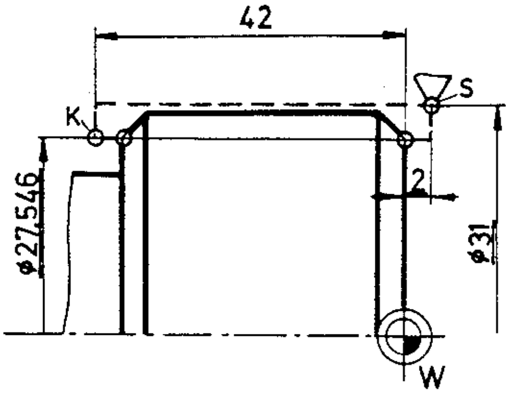


No D7		
$D_7=0$	Dec. D_3	K
$D_7=1$	Konst. D_3	
$D_7=2$	Dec. D_3	N
$D_7=3$	Konst. D_3	
$D_7=4$	Dec. D_3	K
$D_7=5$	Konst. D_3	
$D_7=6$	Dec. D_3	N
$D_7=7$	Konst. D_3	

Parameter D_5

Thread angle (D5)	Infeed angle
40°	19°
55°	26°
60°	29°
80°	39°

Examples G85 - thread cycle



D3 programmed	→	Cut division
D6 programmed	→	Thread depth
No P2 programmed	→	No thread runout
No D4 programmed	→	Allowance cut according to operator monitor
No D5 programmed	→	No flank feed
No D7 programmed	→	No constant feed, No programming of the nominal diameter

1st Example: Longitudinal thread M30 x 2

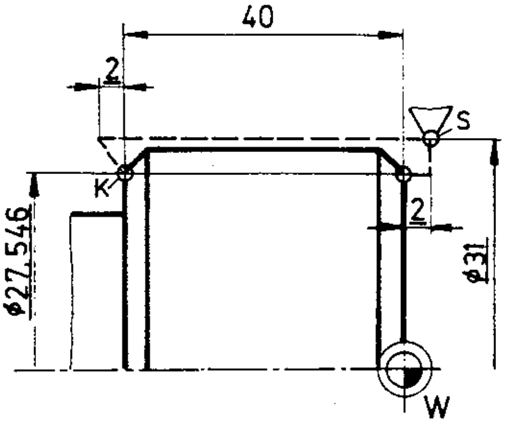
Longitudinal thread cycle with programming of the core diameter K, feed D3, thread depth D6 and pitch F (F is set parallel to the Z-axis).

Programming absolute:

```
N...../G00/X31.000/Z2.000
N...../G85/X27.546/Z-42.000/D3 =
         600/D6 = 1277/F2000
```

Programming incremental:

```
N...../G00/....
N...../G85/U-1.727/W-44.000/D3 =
        600/D6 = 1227/F2000
```



P2 programmed	→	Thread runout
D3 programmed	→	Cut division
D5 programmed	→	Flank feed
D6 programmed	→	Thread depth
No D4 programmed	→	Allowance cut according to operator monitor
No D7 programmed	→	No constant feed, No programming of the nominal diameter

2nd Example: Longitudinal thread M30 x 2

Longitudinal thread cycle with programming of the core diameter K, thread runout P2, feed D3, flank feed D5, thread depth D6 and pitch F (F is set parallel to the Z-axis).

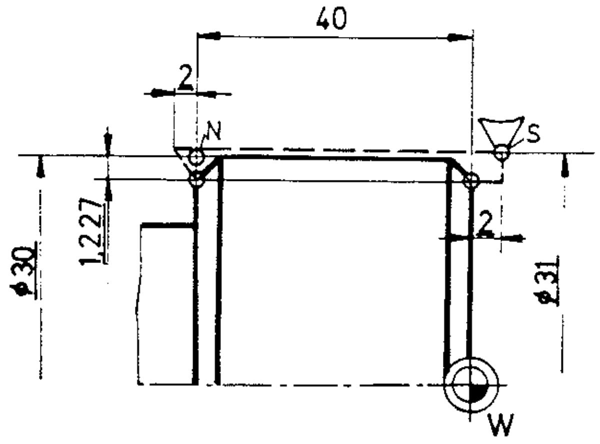
Programming absolute:

```
N...../G00/X31.000/Z2.000
N...../G85/X27.546/Z-40.000/P2 =
       - 2.000/D3 = 600/D5 =
         60/D6 = 1277/F2000
```

Programming incremental:

```
N...../G00/....
N...../G85/U-1.727/W-42.000/P2 =
       - 2.000/D3 = 600/D5 = 60/D6
       = 1227/F....
```

3rd Example - Longitudinal thread M30 x



P2 programmed	-→	Thread runout
D3 programmed	-→	Cut number
D4 programmed	-→	Finishing cut
D5 programmed	-→	Flank feed
D6 programmed	-→	Thread depth
D7 programmed	-→	Constant feed, Nominal diameter programming

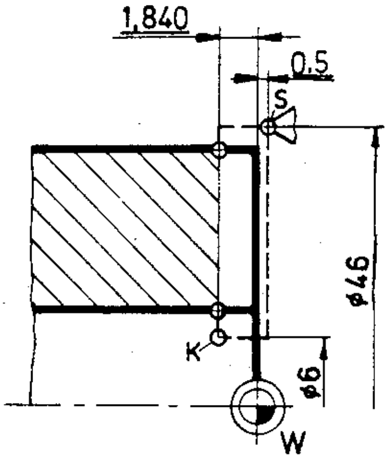
Longitudinal thread cycle with programming of the nominal diameter N, the thread runout P2, the cut number D3, the number of idle cuts D4, flank feed D5, thread depth D6, mode parameter D7 and pitch F (F is input parallel to the Z-axis).

Programming absolute:

```
N...../G00/X31.000/Z2.000
N...../G85/X30.000/Z-40.000/P2 =
       2.000/D3 = 6/D4 = 3/D5 =
       60/D6 = 1227/D7 = 7/F2000
```

Programming incremental:

```
N...../G00/.....
N...../G85/U-0.500/W-42.000/P2 =
      - 2.000/D3 = 6/D4 = 3/D5 =
      60  D6 = 1227/D7 = 7/F2000
```



4th Example: Facing thread pitch 3 mm

Facing thread cycle with programming of the core diameter K, feed D3, thread depth D6 and pitch F (F is set parallel to the X-axis).

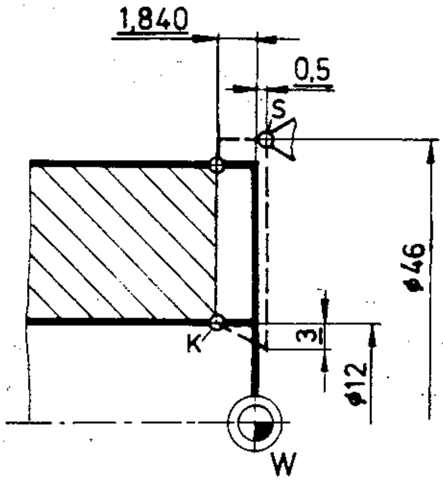
Programming absolute:

```
N...../G00/X46.000/Z0.500
N...../G85/Z-1.840/X6.000/D3 =
          600/D6 = 1840/F3000
```

Programming incremental:

```
N...../G00/.....
N...../G85/W-2.340/U-20.000/D3 =
          600/D6 = 1840/F3000
```

D3 programmed →	Cut division
D6 programmed →	Thread depth
No P2 programmed →	No thread runout
No D4 programmed →	Allowance cut according to operator monitor
No D5 programmed →	No flank feed
No D7 programmed →	No constant feed, No programming of the nominal diameter



5th Example: Facing thread pitch 3 mm

Facing thread cycle with programming of the core diameter K, thread runout P0, feed D3, flank feed D5, thread depth D6 and pitch F (F is set parallel to the X-axis).

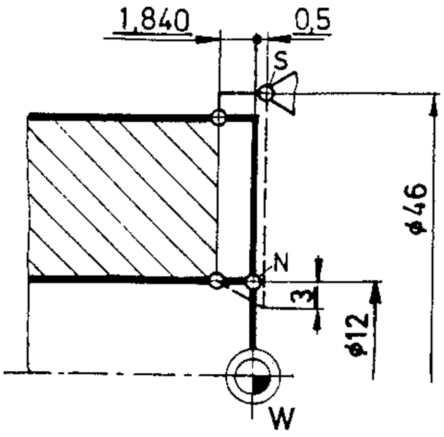
Programming absolute:

```
N...../G00/X46.000/Z0.500
N...../G85/Z-1.840/X12.000/P0 = - 3.000
        /D3 = 600/D5 = 60/D6 = 1840/F3000
```

Programming incremental:

```
N...../G00/.....
N...../G85/W-2.340/U-17.000/P0 = - 3.000
        /D3 = 600/D5 = 60/D6 = 1840/F.....
```

P0 programmed →	Thread runout
D3 programmed →	Cut division
D5 programmed →	Flank feed
D6 programmed →	Thread depth
No D4 programmed →	Allowance cut according to operator monitor
No D7 programmed →	No constant feed, No programming of the nominal diameter



P2 programmed	→	Thread runout
D3 programmed	→	Cut number
D4 programmed	→	Finishing cut
D5 programmed	→	Flank feed
D6 programmed	→	Thread depth
D7 programmed	→	Constant feed, Nominal diameter programming

6th Example: Facing thread pitch 3 mm

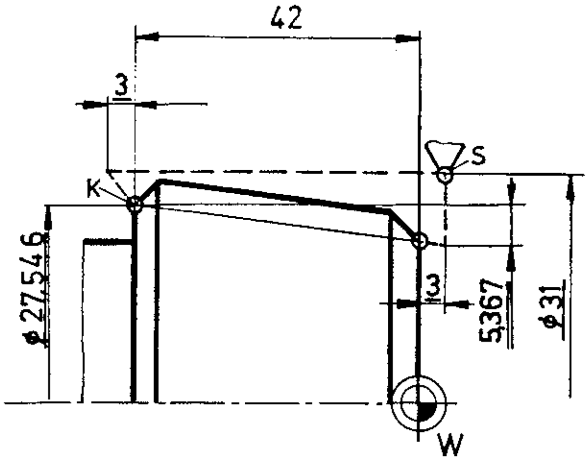
Facing thread cycle with programming of the nominal diameter N, the thread runout P0, the cut number D3, the number of idle cuts D4, flank feed D5, thread depth D6, mode parameter D7 and pitch F (F is input parallel to the Z-axis).

Programming absolute:

```
N...../G00/X46.000/Z0.500
N...../G85/Z-1.840/X12.000/
       P0 = 3.000/D3 = 7/D4 = 3/D5 = 60/
       D6 = 1840/D7 = 7/F3000
```

Programming incremental:

```
N...../G00/.....
N...../G85/W-0.5/U-17.000/
       P0 = -3.000/D3 = 7/D4 = 3/D5 = 60
       D6 = 1840/D7 = 7/F3000
```



P0 programmed	→	Taper angle
P2 programmed	→	Thread runout
D3 programmed	→	Cut number
D4 programmed	→	Finishing cut
D5 programmed	→	Flank feed
D6 programmed	→	Thread depth
D7 programmed	→	Constant feed, Core diameter programming

7th Example: Taper thread a < 45x pitch 3 mm

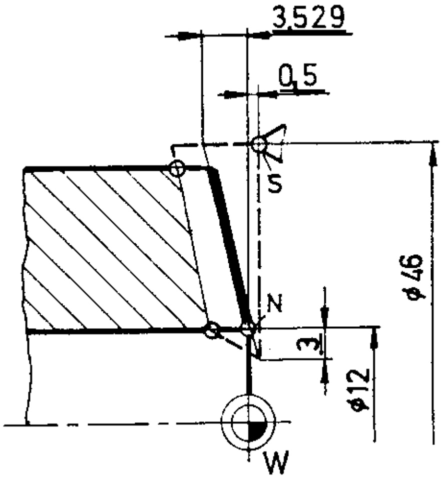
Taper thread cycle with programming of the core diameter K, the taper angle P0, the thread runout P2, the cut number D3, the number of idle cuts D4, flank feed D5, thread depth D6, mode parameter D7 and pitch F.

Programming absolute:

N...../G00/X31,000/Z3,000
N...../G85/X27,546/Z-42,000/
P_0-5,367/P_2-3,000/D_3 = 600/
D_4 = 3/D_5 = 60/D_6 = 1840/D7 = 1/
F....

Programming incremental:

N...../G00/.....
N...../G85/U-1,727/W-45,000/
P_0 = -5,367/P_2 = -3,000/D_3 = 600/
D_4 = 3/D_5 = 60/D_6 = 1840/D_7 = 1/
F....



P0 programmed	→	Thread runout
P2 programmed	→	Taper angle
D3 programmed	→	Cut number
D6 programmed	→	Thread depth
D7 programmed	→	Constant feed, Nominal diameter programming
No D4 programmed	→	Finishing cut according to operator monitor
No D5 programmed	→	No flank feed

8th Example: Taper thread a > 45x pitch 4 mm

Taper thread cycle with programming of the nominal diameter N, the taper angle P2, the thread runout P0, the feed D3, thread depth D6, mode parameter D7 and pitch F.

Programming absolute:

N...../G00/X46,000/Z0,500
N...../G85/Z00,000/X12,000/P_2 = 3,529/
P_0 = -3,000/D_3 = 700/D_6 = 2454/
D_7 = 2/F....

Programming incremental:

N...../G00/.....
N...../G85/W-0,500/U-17,000/
P_2 = 3,529/P_0 = -3,000/D_3 = 700/
D_6 = 2454/D_7 = 2/F.....

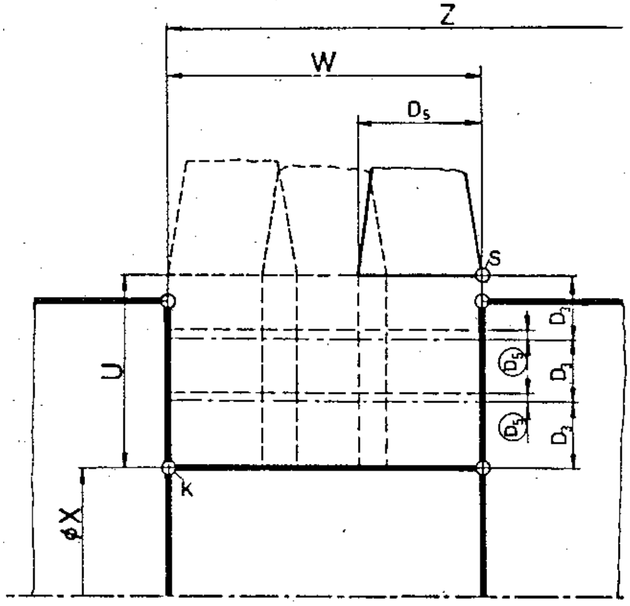
G86 - Plunge-cut cycle

(alongside)

N4	G86	X U ±43	Z W ±43	D35	D45	D55	F4
		[mm]	[mm]	[µm]	[1/10 s]	[µm]	[µm/rev.] [mm/min]

During programming G86, note the edge of the tool that was measured (see note G86).
The co-ordinates X(U) must be programmed prior to Z(W), otherwise the control views G86 as a facing plunge-cut cycle.

Parameters D3 and D4 are marked Def. (Default Option). Default parameters can be programmed. The function of these parameters is explained in the examples of G86 - Plunge-cut cycle.



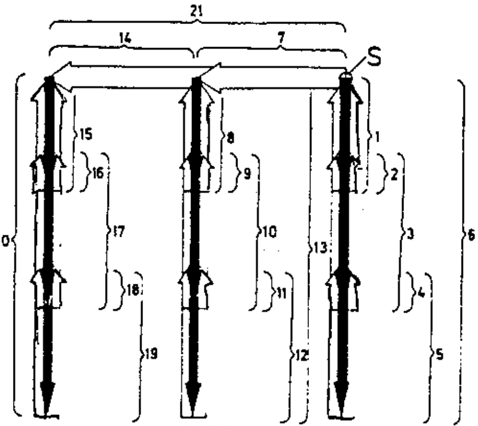
Programming:

N	Block number
G86 ..	Plunge-cut cycle
X, U	Absolute, incremental coordi-
Z, W	nates of the contour joint (E
D3 ...	Feed per cut (Def.)
D4 ...	Dwell time (Def.)
D5 ...	Tool width
F	Feed

Operator monitor:

(D5) ... Return motion per cut set in the factory (D5) = 500 fm

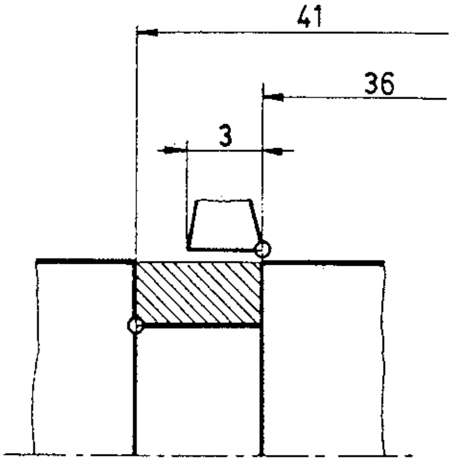
Movement sequence:



Notes on G86 - Plunge-cut cycle:

1. Measuring the recessing tool:

Note the tool edge that is measured, since the control presumes that the right-hand tool edge was measured.



Tool edge RIGHT measured
N..../G00/X42.000/Z-36.000
N..../G86/X30.000/Z-41.000/D5 = 3000/F...

2. Recess width larger than the tool width

Where the width of the programmed recess is larger than the tool width, the control divides the remaining recess width into partial recesses after the 1st recess, with equal sized width. The minimum overlap of the individual part recesses is 1/10 mm.

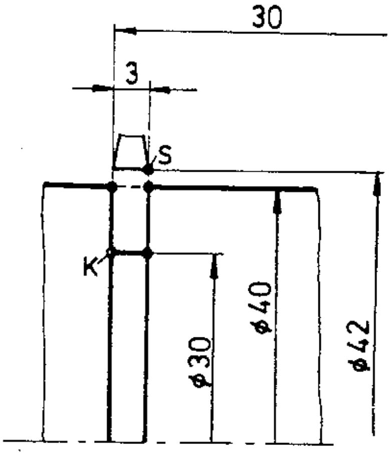
3. Programming the dwell time

To obtain an improved surface quality on the recess base, a dwell time can be programmed with D4.

4. Feed per cut

Where no D3 is programmed, the recess movement is carried out in a single movement, without cut division.

Examples G86 - Plunge-cut cycle (alongside)



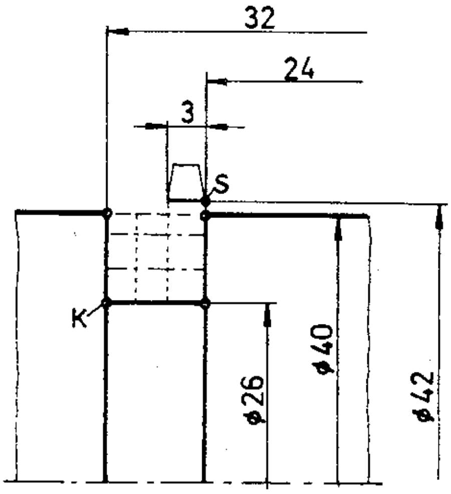
1st Example:

Plunge-cut cycle alongside without cut division, D5 tool width must be programmed.
Input of D5 in 1/1000 mm.

Programming absolute:
N...../G00/X42.000/Z-27.000
N...../G86/X30.000/Z-27.000/
D_5 = 3000/F.....

Programming incremental:
N...../G00/.....
N...../G86/U-6.000/W-3.000/
D_5 = 3000/F.....

D5 programmed → Tool width
No D3 programmed → No cut divisio
No D4 programmed → No dwell time



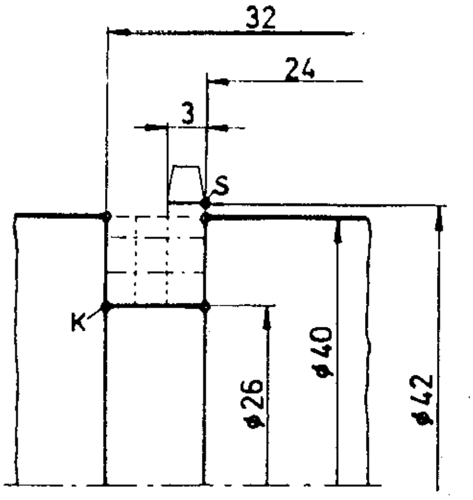
Example 2:

Plunge-cut cycle alongside, recess width greater than tool width D5 and feed per cut D3.
Input of D3 in 1/1000 mm.

Programming absolute:
N...../G00/X42.000/Z-24.000
N...../G86/X26.000/Z-32.000/
D_3 = 1500/D_5 = 3000/F.....

Programming incremental:
N...../G00/.....
N...../G86/U-8.000/W-8.000/
D_3 = 1500/D_5 = 3000/F.....

D3 programmed → Cut division
D5 programmed → Tool width
No D4 programmed → No dwell time



D3 programmed → Cut division
D4 programmed → Dwell time
D5 programmed → Tool width

Example 3:

Plunge-cut cycle alongside, recess width greater than tool width D5, feed per cut D3 and dwell time at recess base D4.
Input of D4 in 1/10 s.

Programming absolute:

N...../G00/X42,000/Z-24,000
N...../G86/X26,000/Z-32,000/
D_3 = 1500/D_4 = 50/D_5 = 3000/F.....

Programming incremental:

N...../G00/.....
N...../G86/U-8,000/W-8,000/
D_3 = 1500/D_4 = 50/D_5 = 3000/F.....

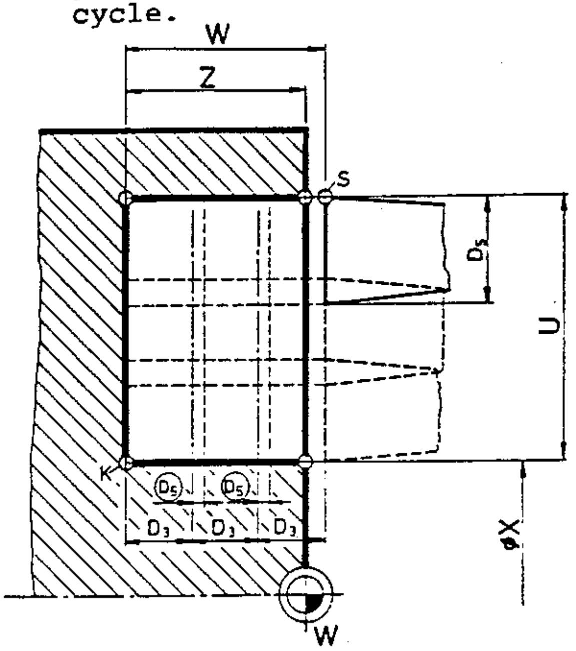
G86 - Plunge-cut cycle

(facing side)

N4	G86	Z W ±43	X U ±43	D35	D4	D55	F4
		[mm]	[mm]	[µm]	[1/10 s]	[µm]	[µm/rev.] [mm/min]

Where co-ordinate Z(W) is programmed prior to X(U), the control carries out a front facing plunge-cut cycle.

Parameters D3 and D4 are marked Def. (Default Option). Default parameters can be programmed. The function of these parameters is explained in the examples of G86 - Plunge-cut cycle.



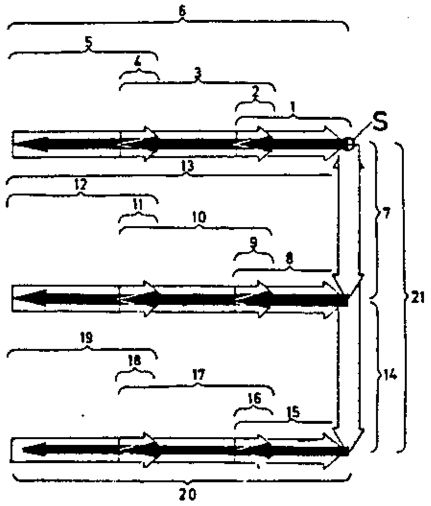
Programming:

N	Block number
G86 ..	Plunge-cut cycle
Z, W X, U	Absolute, incremental coordinates of the contour joint (K)
D3 ...	Feed per cut (Def.)
D4 ...	Dwell time (Def.)
D5 ...	Tool width
F	Feed

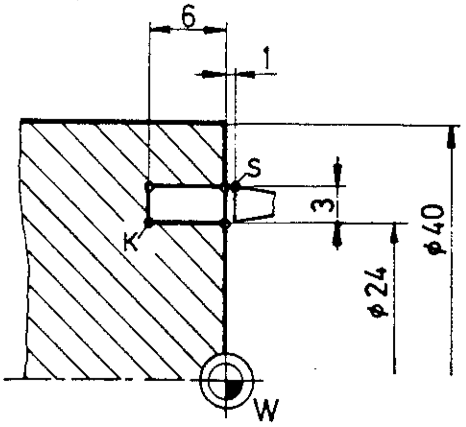
Operator monitor:

(D5) ...	Return motion per cut set in the factory (D5) = 500 fm

Movement sequence:



Examples G86 - Plunge-cut cycle (facing side)



1st Example:

Plunge-cut cycle facing side without cut division D3, D5 tool width must be programmed.
Input of D5 in 1/1000 mm.

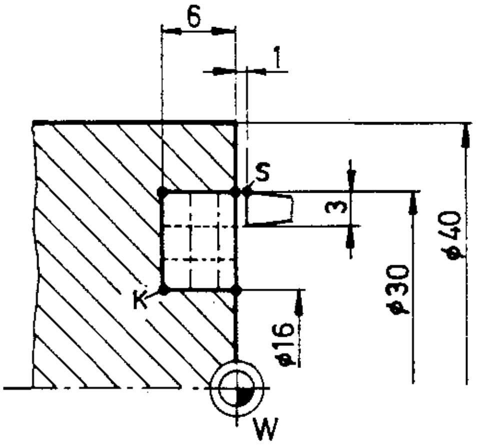
Programming absolute:

```
N...../G00/X30.000/Z-1.000
N...../G86/Z-6.000/X24.000/D5 = 3000/F..
```

Programming incremental:

```
N...../G00/.....
N...../G86/W-7,000/U-3,000/D5 = 3000/F..
```

D5 programmed -→ Tool width
No D3 programmed -→ No cut division
No D4 programmed -→ No dwell time



Example 2:

Plunge-cut cycle facing side, recess width greater than tool width D5 and feed per cut D3.
Input of D3 in 1/1000 mm.

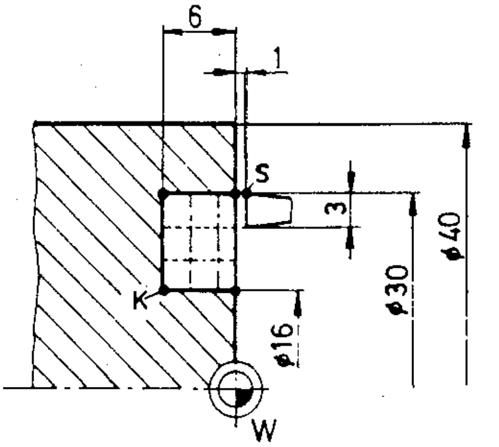
Programming absolute:

```
N...../G00/X30,000/Z1,000
N...../G86/Z-6.000/X16,000/
      D3 = 2000/D5 = 3000/F.....
```

Programming incremental:

```
N...../G00/.....
N...../G86/W-7.000/U-7.000/
      D3 = 2000/D5 = 3000/F.....
```

D3 programmed -→ Cut division
D5 programmed -→ Tool width
No D4 programmed -→ No dwell time



D3 programmed → Cut division
D4 programmed → Dwell time
D5 programmed → Tool width

Example 3:

Plunge-cut cycle facing side, recess width greater than tool width D5, feed per cut D3 and dwell time at recess base D4.

Programming absolute:

N...../G00/X30,000/Z1,000
N...../G86/Z-6,000/X16,000/
D3 = 2000/D4 = 50/D5 = 3000/F.....

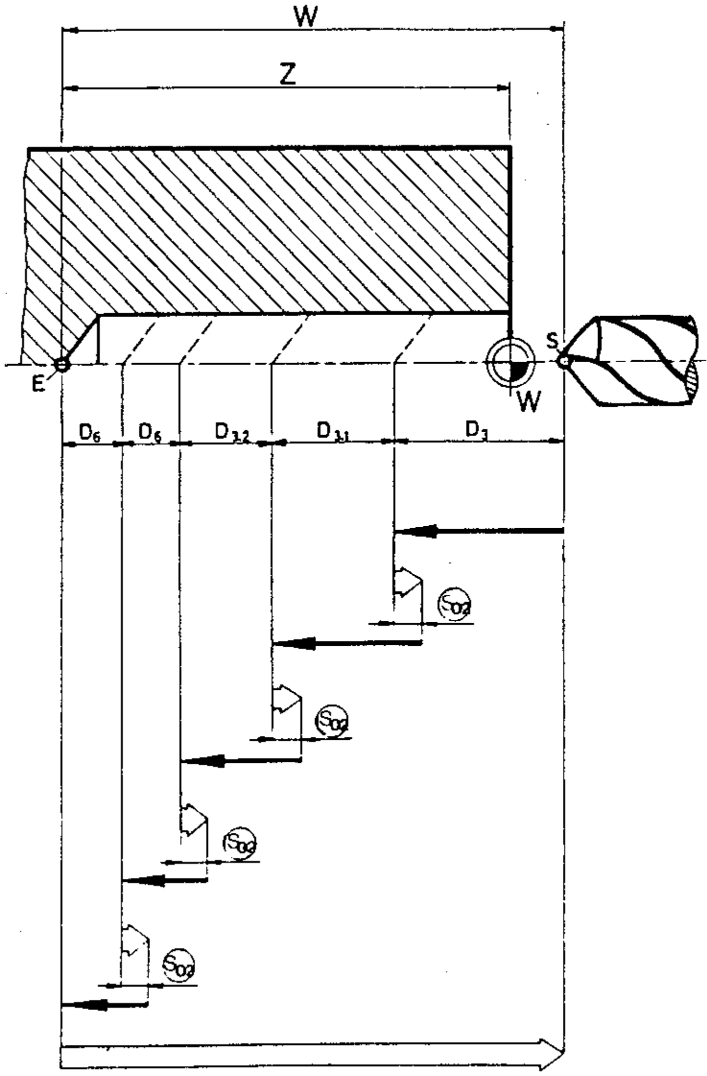
Programming incremental:

N...../G00/.....
N...../G86/W-7,000/U-7,000/
D3 = 2000/D4 = 50/D5 = 3000/F.....

G87 - Chip cutting cycle

N4	G87	Z W ±43	D35	D45	D55	D65	F4
		[mm]	[µm]	[1/10 s]	[%]	[µm]	[µm/U] [mm/min]

Parameters D3, D4, D5 and D6 are marked Def. (Default Option).
Default parameters can be programmed. The function of these parameters is explained in the examples of G87/G88 - Drilling cycles.



Programming:

N	Block number
G87 ..	Chip cutting cycle
Z, W	Absolute, incremental coordinates of the target point (Z)
D3 ...	Drill depth of 1st cut (Def.)
D4 ...	Dwell time at target point (Def.)
D5 ...	Percentage of cut depth reduction (Def.)
D6 ...	Minimum drill depth (Def.)
F	Feed

Operator monitor:

S00..... Minimum feed, only active when D6 is not programmed. S00 is set in factory with 100 fm

S02..... Return per cut. S02 is set in the factory at 500 fm.

Note:

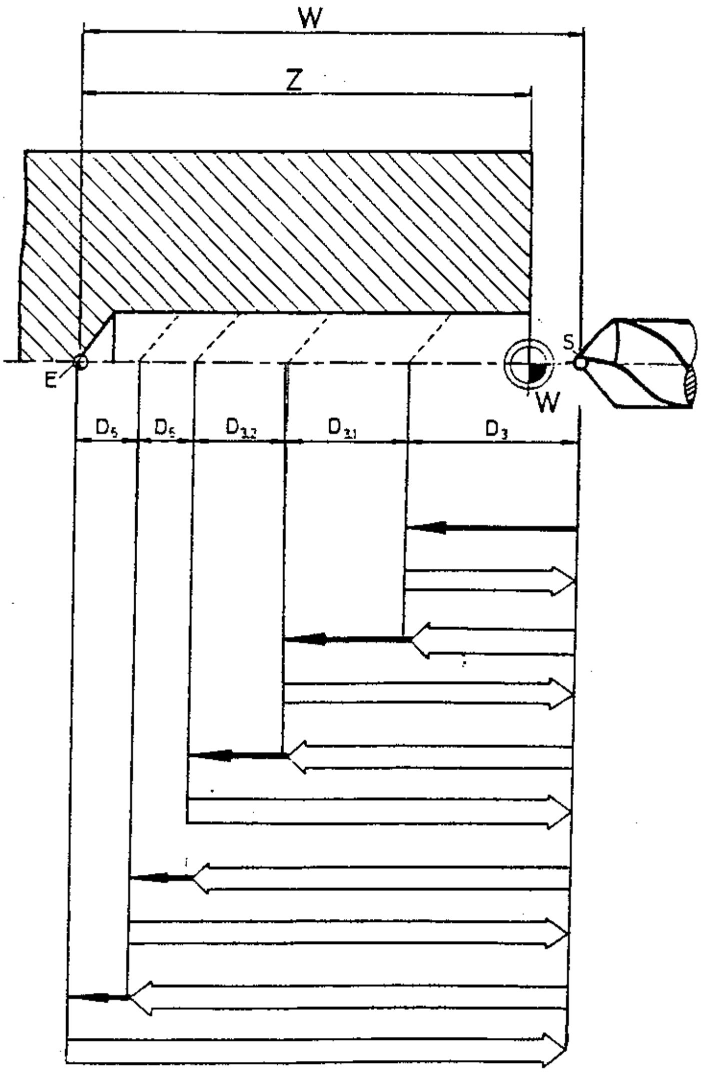
When programming D5, the feed is reduced by the input percentage. The control computes the cut depth reduction from the equation: D3xn = D3xn-1 x D5/100

G88 - Redraw drilling cycle

N4	G88	Z W ±43	D35	D45	D55	D65	F4
		[mm]	[µm]	[1/10 s]	[%]	[µm]	[µm/rev.] [mm/min]

G88 - Redraw drilling cycle

Parameters D3, .D4, D5 and D6 are marked Def. (Default Option).
Default parameters can be programmed. The function of these parameters is explained in the examples of G87/G88 - Drilling cycles.



Programming:

N	Block number
G88 ..	Redraw drilling cycle
Z, W	Absolute, incremental coordinates of the target point (Z)
D3 ...	Drill depth of 1st cut (Def.)
D4 ...	Dwell time (Def.)
D5 ...	Percentage of cut depth reduction (Def.)
D6 ...	Minimum drill depth (Def.)
F	Feed

Operator monitor:

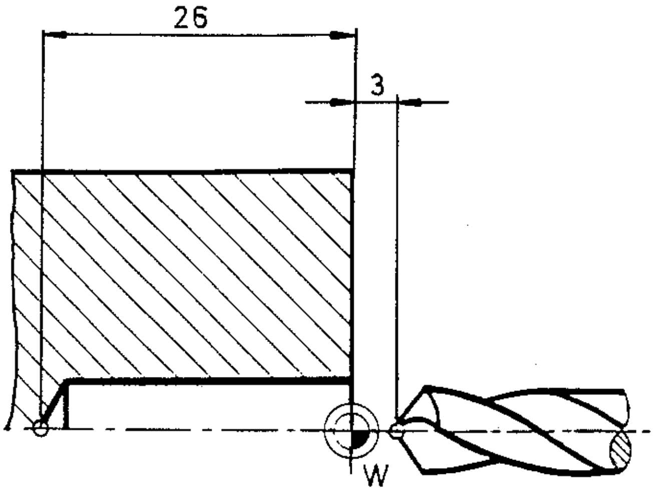
S00 ... Minimum feed, only active when D6 is not programmed. S00 is set in the factory with 100 fm.

Note:

When programming D5, the feed is reduced by the input percentage. The control computes the cut depth reduction from the equation:
D3xn = D3xn-1 x D5/100

Examples G87/G88

These examples apply for G87-chip cutting cycle and G88-redraw drilling cycle. Programming the parameters is the same on both cycles.



No D3 programmed → No cut division
No D4 programmed → No dwell time
No D5 programmed → No reduction percentage
No D6 programmed → Minimum drilling depth according to operator monitor

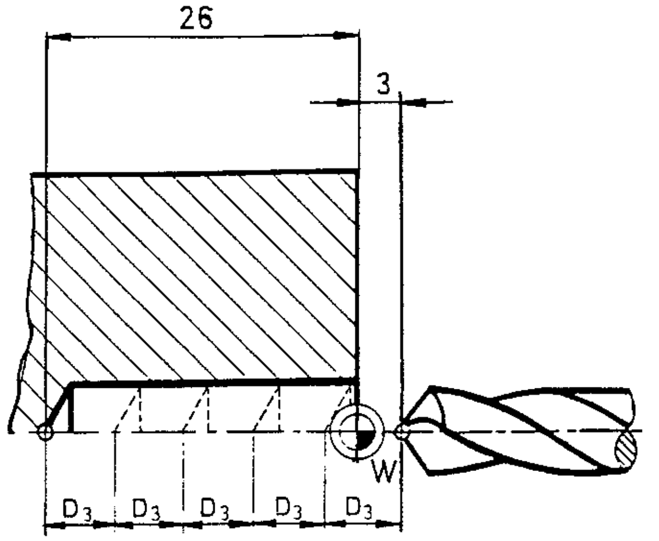
Example 1:
Drilling cycle drilled in one sequence.

Programming absolute:

```
N..../G00/X00.000/Z3.000
N..../G87/Z-26.000/F....
      G88
```

Programming incremental:

```
N..../G00/....
N..../G87/W-29.000/F....
      G88
```



No D3 programmed → No cut division
No D4 programmed → No dwell time
No D5 programmed → No reduction percentage
No D6 programmed → Minimum drilling depth according to operator monitor

Example 2:
Drilling cycle with D3 (drilling depth of the 1st cut). Where D3 is programmed without D5 and D6, the division is made in constant steps < D3.
Input of D3 in 1/1000 mm.

Programming absolute:

```
N..../G00/X00.000/Z3.000
N..../G87/Z-26.000/D3 = 6000/F....
      G88
```

Programming incremental:

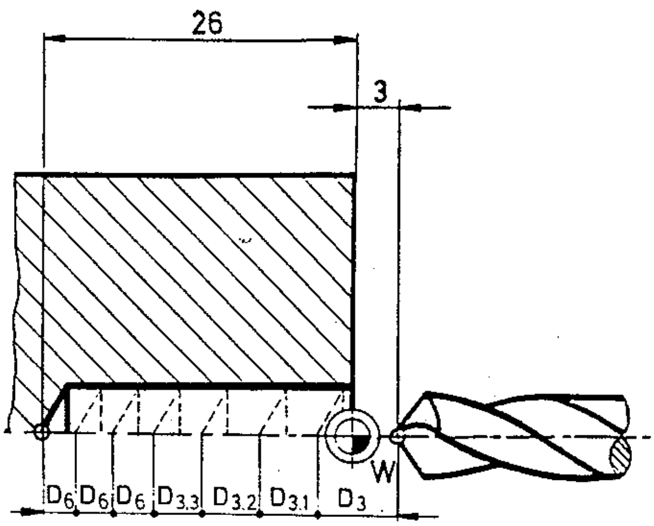
```
N..../G00/....
N..../G87/W-29.000/D3 = 6000/F....
      G88
```

Note:
Input D3 = 6 mm
Drilling depth 29 mm

Theoretic feed:

4 feeds	6 mm =	24 mm
Remainder	=	5 mm
		29 mm

Effective feed:
5 feeds 5.8 mm = 29 mm



D3 programmed -→ Cut division
D4 programmed -→ Dwell time
D5 programmed -→ Reduction percentage
D6 programmed -→ Minimum drilling depth

<u>3rd Example:</u>

Drilling cycle with D3 drilling depth of the 1st cut, D4 dwell time, D5 cut depth reduction and D6 minimum drilling depth.
Input of D4 in 1/10 s.
Input of D5 in %.
Input of D6 in 1/1000 mm.

Programming absolute:

```
N...../G00/X00,000/Z3,000
N...../G87/Z-26,000/D3=7000/D4=50/
    G88       D5=80/D6=3000/F.....
```

Programming incremental:

```
N...../G00/.....
N...../G87/W-29,000/D3=7000/D4=50/
    G88       D5=80/D6=3000/F.....
```

Note:

The feed is reduced with the percentage input under D5, until the minimum drilling depth specified by D6 undercut is reached.

```
1.  Feed: D3 ... 7 mm
2.  Feed: D3.1 = D3 x (D5/100) =
          7 x (80/100) = 5.6 mm
3.  Feed: D3.2 = D3.1 x (D5/100) =
          5.6 (80/100) = 4.48 mm
4.  Feed: D3.3 = D3.2 x (D5/100) =
          4.48 (80/100) = 3.584 mm

5.⎫
6.⎬ Residual traverse path
7.⎭
```

G92 - Speed limitation

N4	G92	S4
		[rev./min]

Function G92 is a double function.

Where G92 is programmed in conjunction with parameter S, the control views G92 as a speed limitation. The control views the value input under parameter S as a speed limitation. The control processes the value input under parameter S in r/min.

Task G92/S:
At high speeds, considerable centrifugal forces occur, these reduce the clamping force of the chuck.
Where G96 (= constant step speed) is programmed, G92 should also be programmed, otherwise the speed would considerably increase for small workpiece diameters.

G92 - Details for position shift register 5 in NC-program

N4	G92	X U ±43	Z W ±43
		[mm]	[mm]

Where G92 is programmed in conjunction with parameters X, (U) and Z, (W), shift details follow for the position shift register 5. Shift dimensions (X = radius dimension) are specified with X and Z. Upon processing a G92 block, the X- and Z-values erase the older values in position shift register 5.
Where U and W shift values are specified in the G92 block, these U- and W-values are added or subtracted with the old values in the position shift register.

Activating the shift:
With G59, the shift in the co-ordinate system is implemented.

Note:
G59 may not be programmed in the same NC block as G92.

For details see chapter zero point shift.

G94 - Feed specification in 1/100 inch/min (mm/min)

N4	G94

Where G94 is programmed, the input feed values are executed in 1/100 inch/min (mm/min).

Possible input values see Technical Data of the CNC-machines!

G95 - Feed specification in 1/10,000 inch/revolution (fm/revolution)

N4	G95

G95 is the actuation condition of the control. Where no G94 is programmed, all feed values are automatically executed in 1/10,000 inch/rev. (fm/rev.).

G96 - Constant cutting speed

N4	G96	S
		[m/min]

A constant cutting speed can be programmed with G96. The controls governs the speed in relation to the workpiece diameter.

$$V = \frac{D \times S \times \pi}{1000}$$

V ... Cutting speed (m/min)
D ... Workpiece diameter (mm)
S ... Speed (rev./min)

G97 - Direct speed programming

N4	G97	S
		[rev./min]

G97 is the actuation condition of the control.
With G97, one can switch back to direct speed programming, provided G96 has been previously programmed.

Chapter 5

M-functions

The M-functions

Programming

M-functions are actuation or additional functions. The M-commande can stand alone in a program block or togetzer with other instructions. Commands from the same group cancel each other, i.e. the last M-instruction programmed cancels the previous instruction from the same group.

Note:
The following pages contain a list of the M-functions which are standard for the EMCOTRONIC TM 02. Whether these M-functions are active on the actual machine, depends on the machine variant.

Structure and Initial status of M-Codes

Group		
Group 0		M03 Spindle ON in clockwise rotation
		M04 Spindle ON in counterclockwise rotation
	**	M05 Spindle STOP
		M19 Precise spindle stop
Group 1		M38 Precise stop ON
	**	M39 Precise stop OFF
Group 2	*	M00 Programmable intermediate stop
	*	M17 Subroutine end
	*	M30 Program end with return to program start
Group 3		M08 Coolant ON
	**	M09 Coolant OFF
Group 5		M25 Open workholding tool
		M26 Tension workholding tool
Group 6		M20 Reverse tailstock sleeve
		M21 Forward tailstock sleeve
Group 7		M23 Reverse workpiece catcher
		M24 Forward workpiece catcher
Group 8	□	M50 Deselect direction logic of tool turret
	□	M51 Select direction logic of tool turret
Group 9	□	M52 Deselect of chip guard door automatic
	□	M53 Select of chip guard door automatic

* Blockwise effective
** Initial status
□ Initial status to determine in mode MON

Note:

If you have each M-code on your control depends on the hardware of your machine.

M00 - Programmed intermediate stop

N4/M00

The slides are stopped, main spindle and coolant are switched off.

M03 - Main spindle ON clockwise

N4/M03

M04 - Main spindle ON counter-clockwise

N4/M04

M05 - Main spindle STOP

N4/M05

Through M30, M05 is activated as the end of the program.

M08 - Coolant ON

N4/M08

M09 - Coolant OFF

N4/M09

Through M30, M09 is activated as the end of the program.

M17 - End of subroutine

N4/M17

The subroutine is completed with M17. M17 causes a return to the next highest level of the part program. For details see subroutine technique G25/M17.

M19 - Exact spindle stop

N4/M19/S4

[°]

Through programming of M19, the main spindle can be positioned. The main spindle position is input in parameter S (in angular degrees °).
Input range: 0 - 360°

M20 - Tailstock retract

N4/M20

M21 - Tailstock advance

N4/M21

When executing M20/M21 in the part program, the main spindle must be stopped (the same applies to M25/M26).
⟶ALARM 940

Notes on the operation with automatic tailstock sleeve

* The main spindle can not be switched on and CYCLE START is not effective as long as the sleeve is in an undefined condition.

* All possible alarms and their causes which may occur during the tailstock operation are described in the list of the EMCOTRONIC alarm messages.

* Upon switching on, the control assumes the same tailstock state as before the last switch-off. If the control was switched off with the tailstock in an undefined state, the related symbol will flash on the symbol line after switching on the control.
Move the sleeve into the defined position manually.

* A tailstock movement via the keyboard is only possible if the main spindle is stopped and CYCLE START is inactive.

* When pressing one of the tailstock keys, the related symbol on the symbol line will flash until the tailstock sleeve has reached the respective travel limit (rear travel limit or clamped state).

* The tailstock sleeve will only move as long as the key is pressed, i.e. it can be positioned in jog mode.

* If M20 is immediately followed by M21, M20 will first be completed before the M21 operation is initiated (and vice versa, i.e. M21 ⟶ M20).

M23 - Collection tray retract

N4/M23

M30 activates M 23 at the program end.
Definitions for M23/M24:
See operator monitor MON parameter L_{11} bit 0.

M24 - Collection tray advance

N4/M24

Definitions for M23/M24:
See operator monitor MON parameter L_{11} bit 0.

Notes on the collection tray operation

It is not recommended to keep the collection tray ready for operation during the entire machining process as otherwise chips and other contaminations may cause malfunctions in the collection tray operation.

M25 - Open workholding tool

N4/M25

M26 - Tension workholding tool

N4/M26

A chuck or collet can be fitted.

1. Chuck

1.1 Actuate condition - chuck
N.../M26
Internal clamping of jaws.

1.2 Actuate condition - collet
N.../M26
External clamping of jaws.

2. Collet

2.1 Actuate condition - collet
N.../M26
The collet is clamped.

2.2 Actuate condition - chuck
N.../M26
The collet is opened.

3. Actuate condition chuck - collet:

The actuate condition "chuck" or "collet" can be specified on the operator monitor.

Actuate condition - chuck:
O_{11}: bit 2 LOW (value 0)

After a M26 command, the jaws will close with the chuck mounted.

With the collet mounted, the collets will open upon execution of the M26 command (mechanical reversal).

Actuate condition - collet:
O_{11} bit 2 HIGH (value 4)

With the chuck mounted:
With M26 (clamping command), the jaws will open (external clamping).

With the collet mounted:
With M26 (clamping command), the collet will close (mechanical reversal).

4. Specifications and notes on M25/M26

4.1 Programming M25 - opening workholding tools

The main spindle must be at a standstill (M05 or M00 must be programmed in advance). This also means that the stopping phase of the main spindle must be completed. (Programming of dwell, if required.)

4.2 Programming M26 - tension workholding tools

Main spindle must be at a standstill (no problem as spindle cannot be started with the workholding tools open).

4.3 Switching on the main spindle:

As long as the workholding tool is not clamped (respective symbol on the symbol line is not lit or flashes), the main spindle cannot be started.
A flashing symbol means that the workholding tool is in an undefined condition.

Conditions for a switchover chuck < - > collet

- The workholding tool must be "open".
- Main drive must be stopped.
- No CYCLE START may be active.

4.4 Actuate condition

After switching on, the control assumes the same workholding tool condition as before the last switch-off.

Additional information on workholding tools

Final position monitoring

* The monitoring of the final positions of the draw rod (draw bar) can be activated (deactivated) on the operator monitor with parameter L_{02} Bit 0.

* When switching on the "dryrun" function, the final position monitoring is automatically switched off.

Open and close workholding tool via keyboard

* The workholding tool can only be opened by means of the related key if the main spindle is stopped and no CYCLE START is active.

* If the workholding tool is actuated via the workholding tool open/close key, the operation can be interrupted by pressing the key again: when pressing the key during the closing process (the respective symbol is still flashing), the workholding tool will be opened again and vice versa.

M30 - End of program with return to start of program
N4/M30

N4/M30

Effect: End of block/program, return to start of program

M30 also causes:
- Coolant off
- Main spindle off
- Collection tray back
- G40

M38 Precise Stop ON

N4/M38

M39 Precise Stop OFF

N4/M39

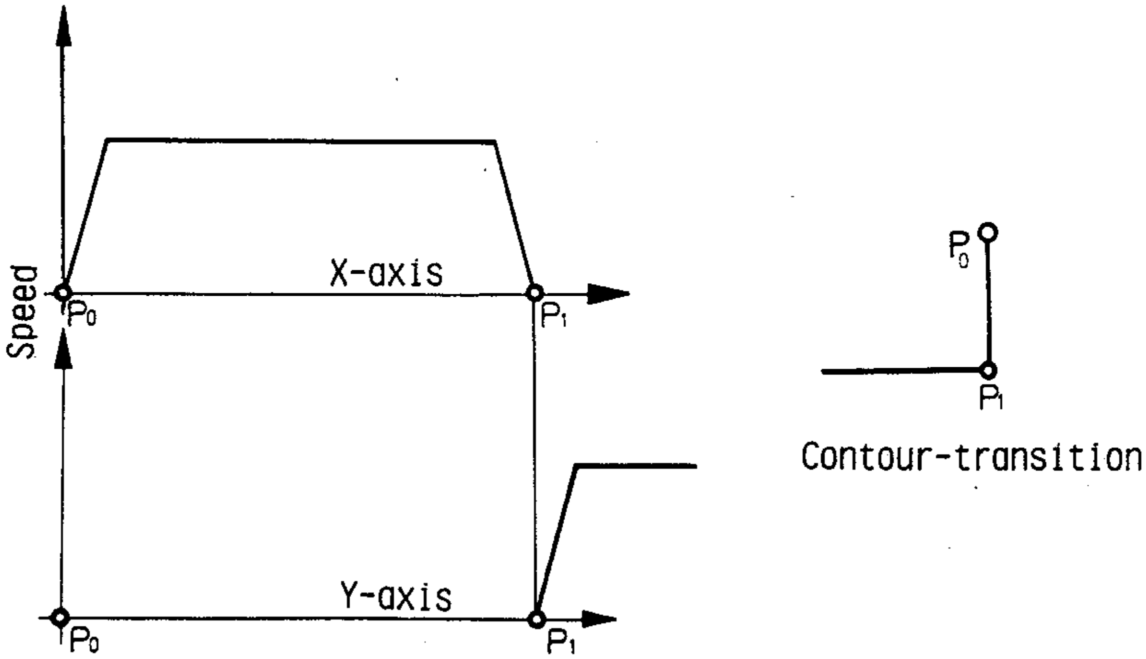
Comments on M38/39

If you want a sharp transition you have to program M38.
The axis movement in the programmed target point stops completely and only then the next block is traversed.

Remarks: * Note down the time difference when manufacturing a workpiece with and without precise stop.

* The control knows the contents of the following traverse instruction

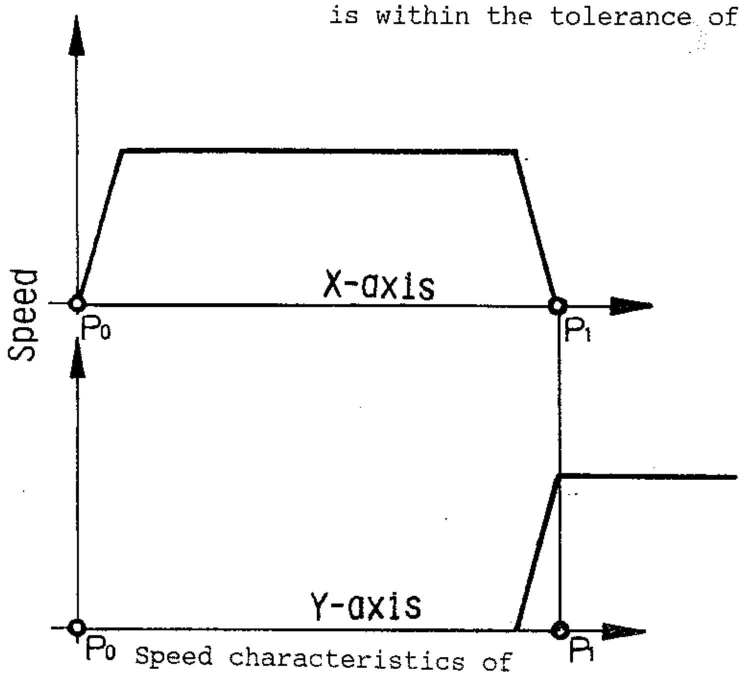
M38 Precise Stop ON



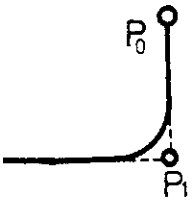
Dwell: 20 m/sec. at point P_1

M39 Precise Stop OFF

The EMCCTRONIC TM 02 control is laid out to accelerate in Y-axis already before reaching the target point in X-axis. Thus a continuous movement with contour transitions is achieved.
The contour transition is not acute-angled (Parabola, Hyperbola).
The size of contour transitions usually is within the tolerance of the drawings.



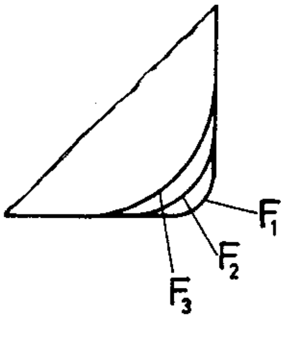
Speed characteristics of slides with M39



Contour-transition

Size of Transition Arcs in Relation to the Speed of Feed:

The larger the feed, the larger the arc of circle.



M50 Deselect direction logic

N4/M50

The tool changer is actuated by the control so that it only swivels in one direction of rotation.

M51 Select direction logic

N4/M51

N4/M51

The direction of rotation of the tool changer is selected by the control so that the programmed tool changer position is reached over the shortest path.

Defining the actuate condition

The actuate condition (M50 or M51) can be input in parameter L_1 on the operator monitor (MON).

M50: O_{11} bit 3 = 0 (low) → value 0

M51: O_{11} bit 3 = 1 (high) → value 8

(See also description of the operator monitor MON.)

M52 Deselect door automatic

N4/M52

M53 Activate door automatic

N4/M53

Actuating the automatic chip safety door:

1. Actuation via key board

   Through pressing the chip door key, the door can be opened/closed.

2. Activation in the program

   M52 Deselect door automatic
   M53 Select activate door automatic

   The activation of M53 and pressing the cycle start key initiate the closing of the automatic chip door. M00 and M30 in the program cause the opening of the automatic chip safety door.

   Exception:
   M30 with active bar feed.

   Note:
   The activate condition can be input in parameter O_{11} bit 4 on the operator monitor.

Chapter 6

Operator Monitor
EMCOTRONIC TM 02

Operator monitor EMCOTRONIC TM 02

On the operator monitor (MON), the operator can change the machine and control conditions.
These conditions are defined by the input of parameters.

Parameter groups on the operator monitor

Parameter	
D	General monitoring data
G	Tool changer data
L	Periphery data
M	Main drive data
O	General setting data
R	Machine specific position data
S	Machining cycle setting data
T	Data/graphics

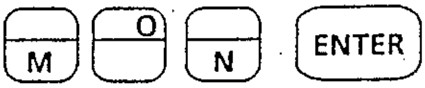
Table of parameters on the operator monitor

D	G	L	M	O	R	S	T
00 01 02 03	08	02 11 25 46 47	08	00 01 02 03 11 22 40	00 01 02 03 04 05 06 07 08 09	00 01 02	24

Calling the operator monitor

The calling of the operator monitor (MON) is carried out in the EDIT mode. Any active part program must be deselected before (RESET).

Data input



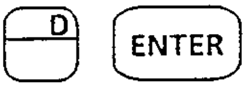
1. Switching to operator monitor mode

   Input M, O, N, ENTER; the control will then be set to operator monitor mode.

2. Selection of parameters:

   There are two ways of selecting parameters:

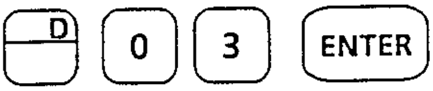
2.1 Selecting a parameter group



Example:
Through the input of D, the first parameter of this group (D_{00}) will be displayed.

With the ENTER key, you can select the parameters within this group one after the other.

2.2 Selecting individual parameters



Example:
Through the input of D_{03}, the respective parameter will be displayed.

With the ENTER key, you can select all parameters within this group one after the other.

3. Input and storage of a parameter

   - Correction of the displayed value through pressing the CLEAR ENTRY or CLEAR WORD key followed by the input of the correct value.

   - ENTER, transmission into the memory, the value with the next index is displayed.

4. Leaving the operator monitor

   The input is completed by pressing any operating mode key or RESET.
   When pressing RESET, the EDIT mode remains active.

   Note:
   Store the last input value by pressing ENTER!

D-parameters - general monitoring data

D_{00} Input of the baudrate for the serial interface

With D_{00}, the data transmission speed (baudrate), with which the data are read in or output via the V24/20 mA interface, is input.
Input range: 150 - 4800 baud

The baudrate to be set depends on the connected peripheral unit (see description of the peripheral unit).

D_{01} Priority or the door limit switch

In the normal machine condition (as after delivery), the main drive cannot be switched on with the chip safety door open.

When opening the safety chip door with the door limit switch active, main spindle, feed drives and coolant are switched off (exception: coolant with D_{01} = 7). The factory setting is D_{01} = 1.

* D_{01} = 1
Operating modes: MANUAL
MANUAL/REFERENCE
In these operating modes, the door limit switch is not effective.

* D_{01} = 3
Operating modes: MANUAL
MANUAL/REFERENCE
In these operating modes, the door limit switch is not effective.

Operating modes: AUTO/DRYRUN
Machine operation with the safety chip door open possible. Main spindle cannot be started.

* D_{01} = 7
Operation modes: MANUAL
MANUAL/REFERENCE
AUTO/DRYRUN
AUTO/SINGLE
EXECUTE

* D_{01} = 8
All definitions for D_{01} = 7 concerning the door limit switch also apply to D_{01} = 8. The only difference being that the coolant ON/OFF condition is different when opening the door during an active CYCLE START.

* Coolant ON/OFF condition:

- o With M08 (coolant ON) active, the coolant is switched off with every change of the operating mode. The LED of the coolant key flashes indicating that M08 is still active. Through pressing the coolant key, the coolant can be switched on again.
- o Independently of the door state, the coolant can be switched on and off in all operating modes with the exception of DRYRUN.
- o CYCLE START key not pressed:
  When opening the chip safety door, the coolant is switched off.
- o CYCLE START key pressed:
  The chip safety door is opened with the coolant active:
  D_{01} = 1 Alarm 400 and coolant OFF.
  D_{01} = 3 Alarm 400 and coolant OFF.
  D_{01} = 7 Coolant is not switched off.
  D_{01} = 8 Coolant is switched off.

D_{02}/D_{03} (O_{40}bit4,O_{40}bit5) workpiece counter and presetting

1. Part number display

Activation the display
Set O_{40}/bit 4 to high

Through setting the parameter O_{40}bit4 on the operator monitor MON, the workpiece counter can be activated. In automatic mode, the number of program runs (parts) will then be displayed (except in case of an alarm condition). After each M30 (program end), the piece number will be increased by 1.

Setting the workpiece counter D_{03}

Via parameter D_{03}, the value of the workpiece counter can be set (e.g. reset to 0 through input of D_{03} = 0).

Note:
The counter capacity reaches from 0 to 32,767; above this figure, the counter is automatically reset to 0.

2. Presetting the part number O_{40}/bit 5

You can enter a specific number. Through pressing the start key once, the program will automatically be repeated according to the set number as a CYCLE START is activated after each M30. Through pressing the key "1x" (single part), the automatic cycle start can be suppressed and the processing of the program will be stopped at every M30.

Activation:
Set parameter O_{40}/bit 5.

Input of the number of automatic runs D_{02} (rated number of pieces)
Input the number of pieces in parameter D_{02}.

Example:
16 automatic cycles

Input: D_{02} = 16
After 16 repetitive cycles, the program will stop.

G parameters - tool changer data

G_{08} bit 0 ....Addressing the tool changer

G_{08} bit 0 = 0 (low) .... all safety functions active

G_{08} bit 0 = 1 (high) ... reduced safety functions

G08 bit 0 set to high:

One-key operation in MANUAL mode when swivelling the tool changer (factory setting: G_{08} bit 0 = 0).

L parameters - periphery data

L_{02} bit 0 ..... end position monitoring

L_{02} bit 0 = 0 LOW (value 0)
End position monitoring deactivated.

L_{02} bit 0 = 1 HIGH (value 1)
End position monitoring activated.

L_{11} bit 0 ... Collection tray
(for EMCOTURN 240)

L_{11} bit 0 = 0 LOW (value 0)

No program interruption during the collection tray movements.

L_{11} bit 0 = 1 HIGH (value 2)

Program interruption as follows:
for M24: until the collection tray is in advanced position
for M23: until the collection tray is swung-out

L_{25} bit 0 ... bar feed

L_{25} bit 0 ... skip bar end signal
L_{25} bit 0 = 1 HIGH

The bar end signal is skipped.
The factory setting is L_{25} bit 0 = 0 LOW.

L_{25}bit 0 bar feed selection

With L_{25}bit 1 = 1 HIGH (value 2), the bar feeder is selected.

1) Activating the feed/bar feed

- The feed is activated through pressing the CYCLE START key in the AUTOMATIC mode.

  In the following cases, the feed cannot be activated:

  - The DRYRUN key is pressed in AUTOMATIC mode.
  - Workholding tool and chip safety door are open (ALARM 400).
  - Bar end signal is active (ALARM 960).

2) Deactivating the feed/bar feed

- When CYCLE START is extinguished.
- Pressing RESET at the bar feeder.
- Pressing RESET at the control.
- With the bar end signal active (ALARM 960).
- When EMERGENCY OFF was activated.
- When workholding tool and chip safety door are open (ALARM 400).

L_{46} ..... chip conveyor operating time

With L_{46}, the chip conveyor operating time is set in seconds [sec].

Possible input values: 0 - 255

If L_{46} = 0, the chip conveyor is switched off.

L_{47} ..... chip conveyor interruption time

With CYCLE START active, the chip conveyor is switched on and off in fixed intervals. These intervals (= interruption time) in seconds [sec] can be set with L_{47}.

Possible input values: 0 - 255

If L_{47} = 0, the chip conveyor will operate continuously.

Note:
After the chip conveyor has been switched on and off again after the operating time, the interruption time is reset to 0 (even if the chip conveyor has been switched on by pressing the chip conveyor key).

M parameter - main drive data

M_{08} ... Specifying the spindle position

When the NC program includes a block with M19 without an S-function being present, the control will move the spindle to the position input in M08 on the operator monitor when processing this block.

Input: [°]
The factory setting is M_{08} = 0.

O-parameters - general setting data

Parameter O_{00}

	Bit 0	Bit 1
	Interface data display	RS 232: Automatic output of a leader/trailer
Status with bit = 0 (low)	No display	No leader/trailer
Value	0	
Status with bit = 1 (high)	Display	Automatic output of a leader/trailer
Value	1	2

With parameter O_{00}bit 0, you can display the data while the program is read (MON).

O_{00} bit 0:

1) O_{00} bit 0 low:

   No display of the program data during the reading process.

2) O_{00} bit 0 high:

   This mode allows for the direct editing of the program via an external keyboard, e.g. Teletype, PC.
   The input data are displayed on the screen.
   There is no input check of O numbers (program numbers), i.e. whether a program with this number exists (no "exists" message displayed).
   In this way, programs in the memory and the actual offset values can be modified as well. For detailed information, see section about interface operation in the operating instructions EMCOTRONIC TM 02.

O_{00} bit 1:

When O_{00} bit 1 = 1 (high):

During the read-out, a leading and trailing part consisting of 50 ASCII "ZERO characters" each is automatically generated.

O_{01} parameter

Data format specification for the serial interface

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
	Data format	End part program	Length of the individual character		Parity check	Parity odd/even	Number of stop bits	
Status bit = 0 LOW	EMCO internal	No ctrl Z			No parity check (disable)	Odd parity		
Value	0	0	0	0	0	0	0	
Status bit = 1 high	ISO	ctrl Z			Parity check (enable)	Even parity		
Value	1	2	4	8	16	32	64	128

Bit 2	Bit 3	
0 (low) value = 0	0 (low) value = 0	→ invalid
1 (high) value = 4	1 (low) value = 0	→ invalid
0 (low) value = 0	1 (high) value = 8	→ 7 bits
1 (high) value = 4	1 (high) value = 8	→ 8 bits

Bit 6	Bit 7	
0 (low) value = 0	0 (low) value = 0	→ invalid
1 (high) value = 64	0 (low)	→ 1 stop bit
0 (low) value = 0	1 (high) value = 128	→ 1 1/2 stop bits
1 (high) value = 0	1 (hight) value = 128	→ 2 stop bits

O_{01} bit 0:

Bit 0 = low
EMCO internal data format, only for EMCO test purposes.
Bit 0 = high
ISO format (see also data format EMCOTRONIC)

O_{01} bit 1:

Bit 1 = low
No "ctrl Z" at the end of data transmission
Bit 1 = high
At the end of the data transmission, a "ctrl Z" control character is inserted.

O_{01} bit 2/bit 3:

Bits 2 and 3 are combined. By means of these bits, the character length can be defined. Usually a character length of 7 or 8 bits is used.

O_{01} bit 4:

Specification whether a parity check is to be carried out or not.

O_{01} bit 6/bit 7:

Defining the number of stop bits. The number depends on the connected peripheral devices (see description of peripheral devices).

O_{01} bit 5:

With bit 5, it can be specified whether a sum check for even or odd parity is to be carried out. This check is irrelevant, of course, when no parity check was specified in bit 4.

Example for parameter O_{01}

		Value
Bit 0	ISO FORMAT	1
Bit 1	No ctrl Z	0
Bit 2	Character	0
Bit 3	7 bits •	8
Bit 4	Parity check	16
Bit 5	Even parity	32
Bit 6	1 stop bit	64
Bit 7		0
	Input value O_{01} = 121	

O_{02} Specifying the number of tool data to be stored

With O_{02}, you can define the number of tool data which are to be stored on external storage media.

Input range: 0 - 99
The factory setting is O_{02} = 99.
The 5 PSO registers are always stored, even if O_{02} = 0.
See also interface operation PSO and TO data.

Data Formats EMCOTRONIC

Input in EMCOTRONIC:

The program input to the interface RS 232C is basically done as with an input via the control board.

The sequence of characters sent must be in accordance with the exact sequence of operating keys on the EMCOTRONIC. Therefore it is necessary to know the data input procedure on the EMCOTRONIC - compare operating manual EMCOTRONIC.

The translation of the EMCOTRONIC instructions (e.g. ENTER, PREVIOUS) in ASCII codes you can find in the translation chart.

There are devices with which you can edit directly to the machine. The entered values can be seen on the monitor of the EMCOTRONIC. For this the interface read out has to be activated. (Parameter L0 Bit 2 has to be set to High; value for Bit 2 High = 4, compare user monitor).

Further Remarks:

- Programs can be started instead of % also with the letter "0". All characters before the first % or "0" are ignored.
- Commantaries can be written between round brackets on external devices. These contents in round brackets are not taken-over to the EMCOTRONIC when transfering data.
- The read-in procedure will be automatically finished by the EMCOTRONIC if there is a M30 instruction at the end of the block. If there is no M30 at the end of the block the transfer procedure will not be interruppted.
  (Purpose: Various programs can be entered one after the other)
- Automatic Start of the Read-in Operation:

  With ○ Zi Zi INP or with
  ○ INP (○ flahes)
  The read-in procedure is automatically started.

Edit of EMCOTRONIC to External Devices:

Edit can be done in two formats.The edit mode can be determined in the user monitor.

ISO Format

User monitor:
Parameter L4: Bit Ø has to be set High (value = 1)

Program format:

```
% ZiZi ␣ crlf N ZiZiZiZi ␣ ␣ / ␣ GZiZi ␣
M ZiZi crlf N ZiZiZiZi ␣ ␣ PZi = ZiZi.ZiZiZi ␣
DZi = ZiZiZi crlf X ZiZi.ZiZiZi crlf
```

Note:
The EMCO format is for internal use.

Translating Chart

ASCII-character	Generation on external keyboard	Hex-Code	Interpretation by EMCOTRONIC	
			ISO-Format*	EMCO-Format*
NUL	ctrl Space Bar	ØØ	-	
SOH	ctrl A	Ø 1	-	
STX	ctrl B	Ø 2	C.B.	
ETX	ctrl C	Ø 3	-	
EOT	ctrl D	Ø 4	-	
ENQ	ctrl E	Ø 5	-	ENTER
ACK	ctrl F	Ø 6	-	
BEl	ctrl G	Ø 7	-	
BS	ctrl H/Backspace	Ø 8	SHIFT/ENTER	
HT	ctrl I/Tabulator	Ø 9	-	
LF	ctrl J/Line feed	Ø A	STORE/NEXT	-
VT	ctrl K	Ø B	-	
FF	ctrl L	Ø C	-	
CR	ctrl M/return	Ø D	ENTER	-
SO	ctrl N	Ø E	-	NEXT
SI	ctrl O	Ø F	-	
DLE	ctrl P	1 Ø	PREVIOUS	
DC1	ctrl Q	11	-	
DC2	ctrl R	12	-	
DC3	ctrl S	13	SHIFT	
DC4	ctrl T	14	-	
NAK	ctrl U	15	-	
SYN	ctrl V	16	-	
ETB	ctrl W	17	C.W.	
CAN	ctrl X	18	-	
EM	ctrl Y	19	-	
SUB	ctrl Z	1A	-	
ESC	ctrl [/ESC	1B	"Escape" getting out of the interface mode	
FS	ctrl \	1C	-	
GS	ctrl]	1D	-	
RS	ctrl ~	1E	-	
US	ctrl ?	1F	-	
SP	Space bar	2 Ø	ENTER	
!	!	21	-	
"	"	22	-	
#	#	23	-	
$	$	24	-	
%	%	25	0	
&	&	26	-	
\	\	27	-	
(	(	28	(	
)	)	29	)	
*	*	2A	-	
+	+	2B	-	
,	,	2C	-	
-	-	2D	change sign ±	
.	.	2E	decpoint	
/	/	2F	/	
Ø	Ø	3 Ø	Ø	
1	1	31	1	
2	2	32	2	
3	3	33	3	

* Can be set in user monitor under O_{01}:

Bit Ø = 1 ...ISO

Chart Continuation

ASCII-character	Generation on external keyboard	Hex-Code	Interpretation by EMCOTRONIC (both Formats)
4	Like ASCII-character	34	4
5		35	5
6		36	5
7		37	7
8		38	8
9		39	9
:		3A	-
;		3B	-
<		3C	-
=		3D	-
>		3E	-
?		3F	-
@		4Ø	-
A,a		41,61	-
B,b		42,62	-
C,c		43,63	-
D,d		44,64	D
E,e		45,65	-
F,f		46,66	F
G,g		47,67	G
H,h		48,68	-
I,i		49,69	I
J,j		4A,6A	J
K,k		4B,6B	K
L,l		4C,6C	L
M,m		4D,6D	M
N,n		4E,6E	N
O,o		4F,6F	O
P,p		50,70	P
Q,q		51,71	-
R,r		52,72	R
S,s		53,73	S
T,t		54,74	T
U,u		55,75	U
V,v		56,76	V
W,w		57,77	W
X,x		58,78	X
Y,y		59,79	Y
Z,z		5A,7A	Z
[		5B	-
\		5C	-
]		5D	-
∧		5E	-
-		5F	-
'		6Ø	-
{		7B	-
|		7C	-
}		7D	-
~		7E	-
DEL	delete	7F	CE

O_{03} Defining the interface operation

O_{03} bit 0:

If O_{03} bit 0 = 1 (high), the programs already stored in the control memory are overwritten with the softkey INPUT ALL active (no "ALREADY EXIST" message is displayed on the screen).

O_{03} bit 1:

If O_{03} bit 1 = 1 (high), the programs stored on cassette are overwritten.
(There is no "ALREADY EXIST" message on the screen.)

O_{11} parameter

	Bit 0 X value input	Bit 1 Actuate condition G70/G71	Bit 2 Actuate condition of workholding tools	Bit 3 Actuate condition tool changer	Bit 4 Actuate condition door automatic
Status with bit = 0 (LOW)	Radius values	Metric (G71)	Chuck	M50	M52
Value	0	0	0	0	0
Status with bit = 1 (HIGH)	Diameter values	Inch (G70)	Collet	M51	M53
Value	1	2	4	8	16.

O_{11} bit 2 .... Actuate condition chuck - collet

Actuate condition - chuck:

O_{11}: bit 2 LOW (value 0)

With the chuck mounted, the jaws will close after reading in M26.

With the collet mounted, the collets will open (mechanical reverse) when reading in M26.

Actuate condition - collet:

O_{11} bit 2 (value 4)

With the chuck mounted:
After reading in M26 (clamping command), the jaws will open (external clamping).

With the collet mounted:
After reading in M26 (clamping command), the collet will close (mechanical reverse).

O_{11} bit 3 ... Actuate condition of direction logic - tool changer

Tool changer with direction logic:
M50 - deselect direction logic
O_{11} bit 3 = 0 (low) → value 0

M51 - select direction logic
O_{11} bit 3 = 1 (high) → value 8

O_{11} bit 4 ... Actuate condition - door automatic

With automatic chip safety door:
O_{11} bit 4 = 0 (low) → value 0
M52 = deselect door automatic

O_{11} bit 4 = 1 (high) → value 16
M53 = select door automatic

- With M53 active and cycle start pressed, the automatic chip safety door is closed and the NC program started.

- The commands M00 and M30 will open the automatic chip safety door. Exception: M30 with automatic CYCLE START (part presetting).

O_{22} Specifying the lowest valid subroutine number

With O_{22} the lowest valid subroutine number is specified.
The highest valid subroutine number is O 0255.
The factory setting is O_{22} = 80.

Example:
O_{22} = 75
Subroutines can be input starting from subroutine number O 0075 up to O 0255.

Parameter O_{40}

	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
	Software limit switch MANUAL	Tool measurement with scratching method	Memory lock	Control lock	Part counter	Work presetting	Contour violation with G41/G42 active	FFS mode
Status with bit = 0 (low)	Effective	No tool measurement with scratching method	No lock	No lock	Not active	Not active	ALARM 570	Not active
Value	0	0	0	0	0	0	0	0
Status with bit = 1 (high)	Not effective	Tool measurement with scratching method	Lock	Lock	Active	Active	No ALARM 570	Active
Value	1	2	4	8	16	32	64	128

O_{40} bit 0: set to high (1)

The software limit switch is ignored in the manual mode. The bit will automatically be cancelled when RESET is pressed again (exception: RESET in the T/PSO mode or on the monitor).

O_{40} bit 2: memory lock

Bit 2 = 1 (high) ⟶ input value 4

In the edit mode, no program selection and thus no program modifications are possible, editting of PSO and tool data only incrementally with cursor keys.

Reason for memory lock:
The non-authorized access to program data by third persons is to be inhibited in order to avoid any errors due to the modification of offset data.

O_{40} bit 3: control lock

Bit 3 = 1 (high) ⟶ input value 8

The complete control is locked. The screen only displays EDIT. All functions are locked with the exception of the main switch and EMERGENCY OFF.

O_{40} bit 4 .... Activate workpiece counter

See workpiece counter parameter D_{02}/D_{03}

O_{40} bit 5 .... Work presetting mode

See workpiece counter parameter D_{02}/D_{03}.

O_{40} bit 6:

O_{40} bit 6 = 64 (high):

If the control detects a contour violation with the radius compensation active, no ALARM 570 is output.
The factory setting is O_{40} bit 6 = 0.

O_{40} bit 7 .... Selecting the FFS mode

The factory setting is O_{40} bit 7 = 0.

R parameters - machine specific position data

The numerical values of the R parameters depend on the machine version. (Dimensions see operating instructions for the respective machine.)

R_{00} Reference point in X

R_{01} Not assigned

R_{02} Reference point in Z

R_{03} Software limit switch in X+

R_{04} Not assigned

R_{05} Software limit switch in Z+

R_{06} Software limit switch in X-

R_{07} Not assigned

R_{08} Software limit switch in Z-

R_{09} Safety distance from the software limit switch in MANUAL mode

Within this range, the feedrate is automatically reduced (safety feed).

S parameter - Setting parameters for machining cycles

S_{00} Specifying the minimum feed for G83, G86 (input in µ m)

The factory setting is S_{00} = 100 µ m.

S_{01} Number of finishing cycles for G85

Factory set to S_{01} = 1.

S_{02} Specifying the retract movement for G86, G87 (input in µ m)

Factory set to S_{02} = 500 µ m.

T parameters - Setting parameters for graphic simulation

T_{24}bit 0 Suppression of sketch display

T_{24}bit 0 = 1 HIGH (value 1)
No auxiliary sketches are displayed.
The. factory setting is T_{24} bit 0 = 0 LOW (value 0).

T_{24}bit 0 Display of frames

T_{24}bit 1 = 1 HIGH (value 2)
The frames are displayed.
The factory setting is T_{24} bit 1 = 0 LOW (value 0).

Chapter 7

The serial interface RS
232 C EMCOTRONIC

The Serial Interface
RS 232C of the EMCOTRONIC

1. Technical data

Function:

Data can be read out and in via the interface. The most frequent peripheral equipment is paper tape reader, paper tape punch, printer, computer. Computer specific software is required for computer connections, whilst the other equipment can be connected directly.

Data format EMCOTRONIC:

* 1 start bit
* 7 data bits
* 1 parity bit: for even parity, i.e. even number of holes on the paper tape for monitoring transmission errors.
* 1 stop bit

Baud rate EMCOTRONIC:

150 - 2400 BAUD
The baud rate is programmable in the MON operating mode under DØ.
Note: The baud rate of the EMCOTRONIC - Peripheral Equipment, must agree.

Electrical data V24:

Voltage: + 12 V = logic 0
- 12 V = logic 1

Electrical data 20 mA:

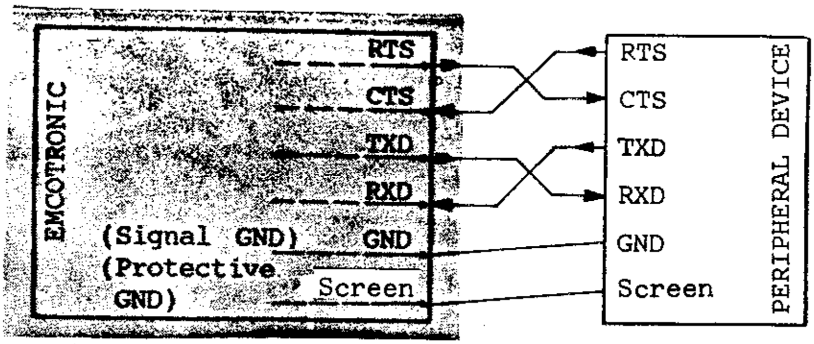
Current: 20 mA = logic 1
0 mA = logic 0

Plug type RS 232c:

EMCOTRONIC	Peripheral equipment
25 terminal RS 232c plug (male)	see equipment description

2. The general connection RS 232C of two devices via V24

2.1 No internal cross-bonding



Meaning of the individual pins:

RTS: request to send = Output : Device requests data

CTS: clear to send = Input = Device indicates readiness to send data

TXD : Transmit data = Data output

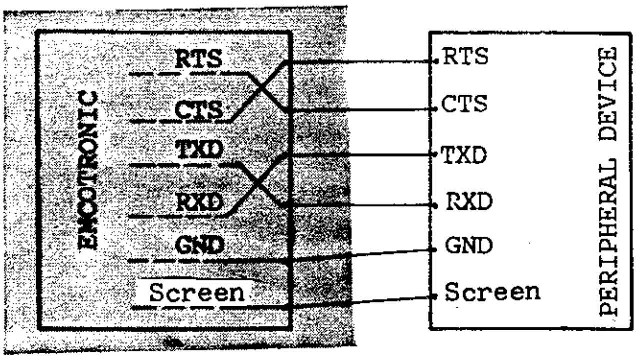
RXD: Receive data = Data input

GND: Ground = Earth

Screen = Cable screen

2.2 Internal cross-bonding 1 x

In the EMCOTRONIC V24 interface (not with 20 mA), the lines are already internally cross-bonded. For this reason, the following connection diagram must be observed



Precondition:

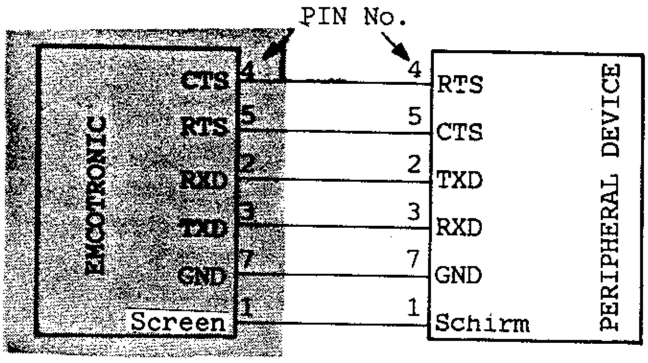
The manufacturer of the 2nd device has not already internally cross-bonded the lines (which is normally the case).

2.3 Internal cross-bonding 2 x

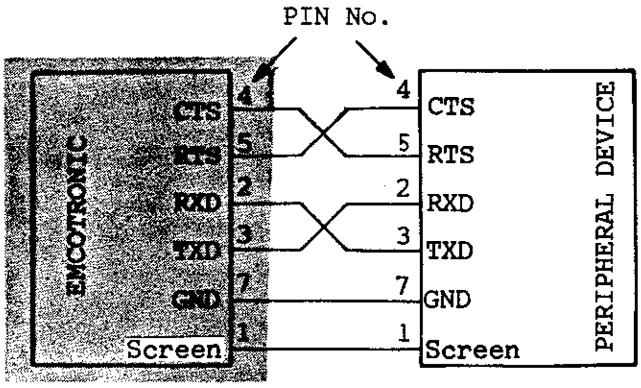
Where the manufacturer of the peripheral device has already made internal cross-bonding, the cable must be cross-bonded. Diagram, see above.

3. RS 232C-V24 connection EMCOTRONIC

3.1 EMCOTRONIC is cross-bonded
Peripheral device is not cross-bonded



3.2 EMCOTRONIC ıs cross-bonded
Peripheral device is cross-bonded



4. 20 mA - Connection EMCOTRONIC

4.1 General

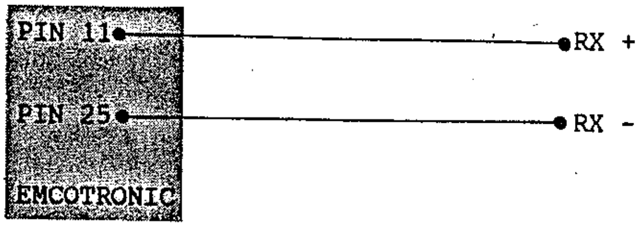
The 20 mA interface of the EMCOTRONIC is not cross-bonded internally.

Two loops must be drawn due to the FULL DUPLEX operation of the EMCOTRONIC 20 mA interface.

* One for transmission

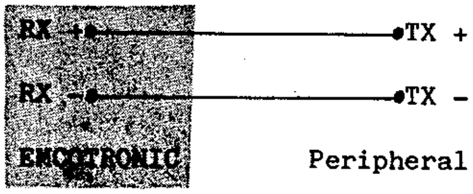
* One for reception

Transmission loop RS 232c

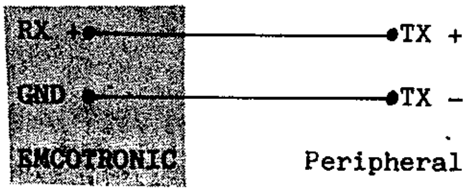


Reception loop RS 232c

Passive peripheral device



Active peripheral device



With the reception loop, there is a difference, depending on whether the peripheral device is active or passive in effect.

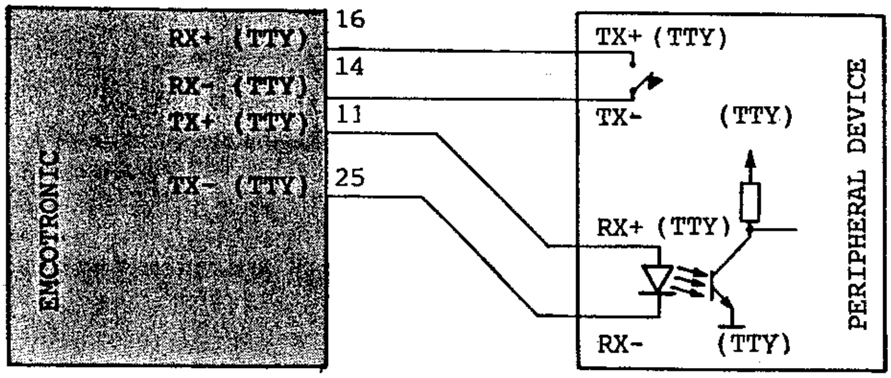
Active device: Itself supplies the signal current.

Passive device: Switches the signal current of the EMCOTRONIC On and Off (no own signal current present).

4.2 Circuit diagram 20mA RS 232c

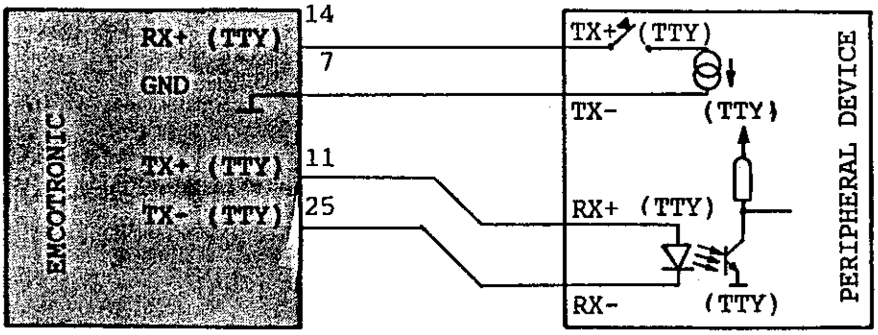
4.2.1 Passive transmitter device

(usual design of the peripheral equipment)



4.2.2 Active transmitter

(infrequent design of the peripheral equipment)

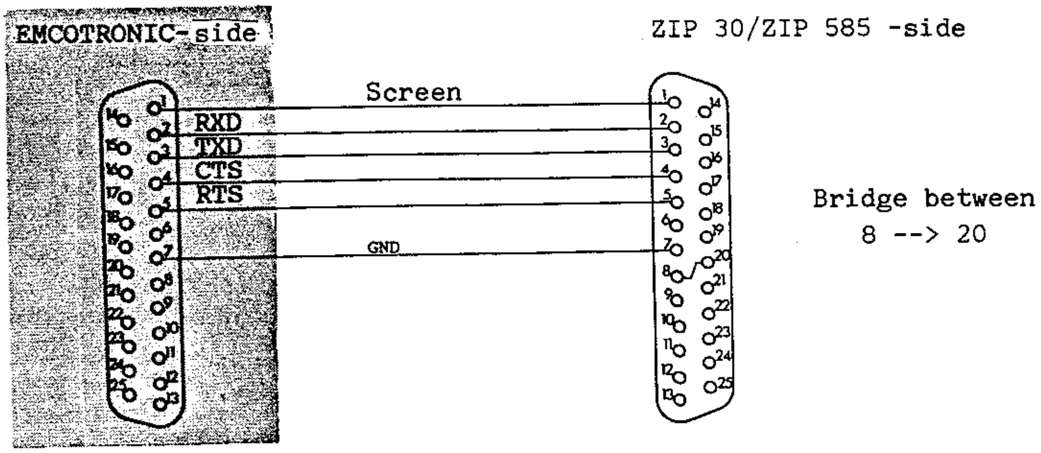


5. Connection occupancy of the equipment offered by EMCO

- * Matrix printer with attached paper tape reader, punch; RS 232c (V24 or 20 mA interface)
  Teledynamics ZIP 30 KSR, ZIP 30 ASR, ZIP 30 ASR/EDIT 1600
  Order No.: 573 470

- * Portable paper tape reader, punch with RS 232c (V24 or 20 mA interface)
  Teledynamics ZIP 585
  Order No.: 573 480

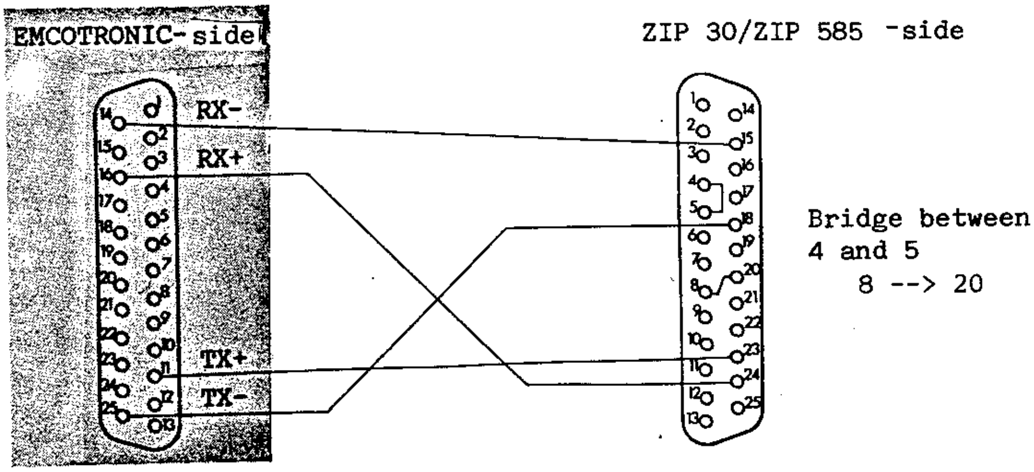
5.1 Connection occupancy when using the V24 interface:



Plug (male)

Socket (female)

5.2 Connection occupancy when using the 20 mA interface:



Plug (male)

Socket (female)

6. Circuit diagram RS 232C

Data Formats EMCOTRONIC

Input in EMCOTRONIC:

The program input to the interface RS 232C is basically done as with an input via the control board.

The sequence of characters sent must be in accordance with the exact sequence of operating keys on the EMCOTRONIC. Therefore it is necessary to know the data input procedure on the EMCOTRONIC - compare operating manual EMCOTRONIC.

The translation of the EMCOTRONIC instructions (e.g. ENTER, PREVIOUS) in ASCII codes you can find in the translation chart.

There are devices with which you can edit directly to the machine. The entered values can be seen on the monitor of the EMCOTRONIC. For this the interface read out has to be activated. (Parameter O_{0C} Bit 2 has to be set to High; value for Bit 2 High = 4, compare user monitor).

Further Remarks:

- Programs can be started instead of % also with the letter "O". All characters before the first % or "O" are ignored.
- Commantaries can be written between round brackets on external devices. These contents in round brackets are not taken-over to the EMCOTRONIC when transfering data.
- The read-in procedure will be automatically finished by the EMCOTRONIC if there is a M30 instruction at the end of the block. If there is no M30 at the end of the block the transfer procedure will not be interruppted.
  (Purpose: Various programs can be entered one after the other)
- Automatic Start of the Read-in Operation:

  With ○ Zi Zi INP or with
  ○ INP (○ flahes)
  The read-in procedure is automatically started.

Edit of EMCOTRONIC to External Devices:

Edit can be done in two formats.The edit mode can be determined in the user monitor.

ISO Format

User monitor:
Parameter O_{01}: Bit Ø has to be set High (value = 1)

Program format:

```
% ZiZi ␣ crlf N ZiZiZiZi ␣ ␣ / ␣ GZiZi ␣
M ZiZi crlf N ZiZiZiZi ␣ ␣ PZi = ZiZi.ZiZiZi ␣
DZi = ZiZiZi crlf X ZiZi.ZiZiZi crlf
```

Note:
The EMCO format is for internal use.

Translating Chart

<table>
<tr><th rowspan="2">ASCII-character</th><th rowspan="2">Generation on external keyboard</th><th rowspan="2">Hex-Code</th><th colspan="2">Interpretation by EMCOTRONIC</th></tr>
<tr><th>ISO-Format*</th><th>EMCO-Format*</th></tr>
<tr><td>NUL</td><td>ctrl Space Bar</td><td>ØØ</td><td colspan="2">-</td></tr>
<tr><td>SOH</td><td>ctrl A</td><td>Ø 1</td><td colspan="2">-</td></tr>
<tr><td>STX</td><td>ctrl B</td><td>Ø 2</td><td colspan="2">C.B.</td></tr>
<tr><td>ETX</td><td>ctrl C</td><td>Ø 3</td><td colspan="2">-</td></tr>
<tr><td>EOT</td><td>ctrl D</td><td>Ø 4</td><td colspan="2">-</td></tr>
<tr><td>ENQ</td><td>ctrl E</td><td>Ø 5</td><td>-</td><td>ENTER</td></tr>
<tr><td>ACK</td><td>ctrl F</td><td>Ø 6</td><td colspan="2">-</td></tr>
<tr><td>BEl</td><td>ctrl G</td><td>Ø 7</td><td colspan="2">-</td></tr>
<tr><td>BS</td><td>ctrl H/Backspace</td><td>Ø 8</td><td colspan="2">SHIFT/ENTER</td></tr>
<tr><td>HT</td><td>ctrl I/Tabulator</td><td>Ø 9</td><td colspan="2">-</td></tr>
<tr><td>LF</td><td>ctrl J/Line feed</td><td>Ø A</td><td>STORE/NEXT</td><td>-</td></tr>
<tr><td>VT</td><td>ctrl K</td><td>Ø B</td><td colspan="2">-</td></tr>
<tr><td>FF</td><td>ctrl L</td><td>Ø C</td><td colspan="2">-</td></tr>
<tr><td>CR</td><td>ctrl M/return</td><td>Ø D</td><td>ENTER</td><td>-</td></tr>
<tr><td>SO</td><td>ctrl N</td><td>Ø E</td><td>-</td><td>NEXT</td></tr>
<tr><td>SI</td><td>ctrl O</td><td>Ø F</td><td colspan="2">-</td></tr>
<tr><td>DLE</td><td>ctrl P</td><td>1 Ø</td><td colspan="2">PREVIOUS</td></tr>
<tr><td>DC1</td><td>ctrl Q</td><td>11</td><td colspan="2">-</td></tr>
<tr><td>DC2</td><td>ctrl R</td><td>12</td><td colspan="2">-</td></tr>
<tr><td>DC3</td><td>ctrl S</td><td>13</td><td colspan="2">SHIFT</td></tr>
<tr><td>DC4</td><td>ctrl T</td><td>14</td><td colspan="2">-</td></tr>
<tr><td>NAK</td><td>ctrl U</td><td>15</td><td colspan="2">-</td></tr>
<tr><td>SYN</td><td>ctrl V</td><td>16</td><td colspan="2">-</td></tr>
<tr><td>ETB</td><td>ctrl W</td><td>17</td><td colspan="2">C.W.</td></tr>
<tr><td>CAN</td><td>ctrl X</td><td>18</td><td colspan="2">-</td></tr>
<tr><td>EM</td><td>ctrl Y</td><td>19</td><td colspan="2">-</td></tr>
<tr><td>SUB</td><td>ctrl Z</td><td>1A</td><td colspan="2">-</td></tr>
<tr><td>ESC</td><td>ctrl [/ESC</td><td>1B</td><td colspan="2">"Escape" getting out of the interface mode</td></tr>
<tr><td>FS</td><td>ctrl \</td><td>1C</td><td colspan="2">-</td></tr>
<tr><td>GS</td><td>ctrl]</td><td>1D</td><td colspan="2">-</td></tr>
<tr><td>RS</td><td>ctrl ~</td><td>1E</td><td colspan="2">-</td></tr>
<tr><td>US</td><td>ctrl ?</td><td>1F</td><td colspan="2">-</td></tr>
<tr><td>SP</td><td>Space bar</td><td>2 Ø</td><td colspan="2">ENTER</td></tr>
<tr><td>!</td><td>!</td><td>21</td><td colspan="2">-</td></tr>
<tr><td>"</td><td>"</td><td>22</td><td colspan="2">-</td></tr>
<tr><td>#</td><td>#</td><td>23</td><td colspan="2">-</td></tr>
<tr><td>$</td><td>$</td><td>24</td><td colspan="2">-</td></tr>
<tr><td>%</td><td>%</td><td>25</td><td colspan="2">0</td></tr>
<tr><td>&</td><td>&</td><td>26</td><td colspan="2">-</td></tr>
<tr><td>\</td><td>\</td><td>27</td><td colspan="2">-</td></tr>
<tr><td>(</td><td>(</td><td>28</td><td colspan="2">(</td></tr>
<tr><td>)</td><td>)</td><td>29</td><td colspan="2">)</td></tr>
<tr><td>*</td><td>*</td><td>2A</td><td colspan="2">-</td></tr>
<tr><td>+</td><td>+</td><td>2B</td><td colspan="2">-</td></tr>
<tr><td>,</td><td>,</td><td>2C</td><td colspan="2">-</td></tr>
<tr><td>-</td><td>-</td><td>2D</td><td colspan="2">change sign ±</td></tr>
<tr><td>.</td><td>.</td><td>2E</td><td colspan="2">decpoint</td></tr>
<tr><td>/</td><td>/</td><td>2F</td><td colspan="2">/</td></tr>
<tr><td>Ø</td><td>Ø</td><td>3 Ø</td><td colspan="2">Ø</td></tr>
<tr><td>1</td><td>1</td><td>31</td><td colspan="2">1</td></tr>
<tr><td>2</td><td>2</td><td>32</td><td colspan="2">2</td></tr>
<tr><td>3</td><td>3</td><td>33</td><td colspan="2">3</td></tr>
</table>

* Can be set in user monitor under O_{01}:

Bit Ø = 1 ...ISO

Chart Continuation

ASCII-character	Generation on external keyboard	Hex-Code	Interpretation by EMCOTRONIC (both Formats)
4	Like ASCII-character	34	4
5		35	5
6		36	5
7		37	7
8		38	8
9		39	9
:		3A	-
;		3B	-
<		3C	-
=		3D	-
>		3E	-
?		3F	-
@		4Ø	-
A,a		41,61	-
B,b		42,62	-
C,c		43,63	-
D,d		44,64	D
E,e		45,65	-
F,f		46,66	F
G,g		47,67	G
H,h		48,68	-
I,i		49,69	I
J,j		4A,6A	J
K,k		4B,6B	K
L,l		4C,6C	L
M,m		4D,6D	M
N,n		4E,6E	N
O,o		4F,6F	O
P,p		50,70	P
Q,q		51,71	-
R,r		52,72	R
S,s		53,73	S
T,t		54,74	T
U,u		55,75	U
V,v		56,76	V
W,w		57,77	W
X,x		58,78	X
Y,y		59,79	Y
Z,z		5A,7A	Z
[		5B	-
\		5C	-
]		5D	-
^		5E	-
-		5F	-
'		6Ø	-
{		7B	-
|		7C	-
}		7D	-
~		7E	-
DEL	delete	7F	CE

Chapter 8

Alarm list EMCOTRONIC TM 02

Alarm messages EMCOTRONIC TM 02 - Turning

(Version 6.00, status 91-4)

Alarm messages 000 - 029: AXIS CONTROLLER

ALARM 000: AC NOT READY

When switching on the control and during the data transmission from the data controller to the axis controller, it is checked whether the axis control unit correctly reads in the commands from the data controller within a certain time limit. If this limit is exceeded, the above alarm will be displayed which must the normally be removed via the hardware (axis controller!).

ALARM 001: X-AXIS: SOFTWARE LIMIT SWITCH OVERTRAVELLED

EXECUTE/AUTOMATIC mode: The programmed path is monitored by software limit switches which, in case of an overtravel, will output ALARM 001, 002, 003 (due to false data in the position-shift register, due to false tool data or a false circular path end point lie within the valid travelling range).
MANUAL: After positioning the axis at the reference point, the software limit switches are valid and will output an alarm and stop the axes when these limits are overtravelled.

ALARM 003: Z-AXIS: SOFTWARE LIMIT SWITCH OVERTRAVELLED

See alarm 001.

ALARM 020: MAIN DRIVE NOT READY

This Alarm will be displayed due to an error message output by the main drive at the following point of time:
* If, after switching on the control, no ready signal is output by the main drive.
* If the operator tries to switch on the main drive without a ready signal being present.
* If an error occurs at the main drive during the operation.
* If there is an error in the power supply of the main drive and/or the machine.

The acknowledging of this alarm is only possible through switching the main drive off and the on again after removing the error cause.

ALARM MESSAGES 030 - 080: PERIPHERY

ALARM 030: LUBRICANT PRESSURE FAILURE

Depending on the overall travel of the slides, the lubrication pump is switched on for an adjustable amount of time. At the end of this time, the built-up pressure in the lubrication lines is checked. Possible error causes:
* Lubrication pump does not operate.
* Operating time of the lubrication pump too short (false adjustment).
* Air in the lubrication system.
* Pressure switch at lubricant pump defective
* Lubricant line leakage
* Not enough oil in lubricant tank

The error cause should be eliminated immediately to avoid mechanical damage on the machine.

ALARM 031: LUBRICANT PUMP OVERLOAD

The thermal monitoring unit of the lubrication pump has been triggered (e.g. pump operating time too long, overload).
Check thermal protective switch in switch cabinet; no switch must be in "0" position.

ALARM 040: FRONT DOOR NOT CLOSED

This error occurs in the following cases with the maximum priority of the door limit switch:

* When actuating the "CYCLE START" key with the chip safety door open.
* In the MANUAL mode with the chip safety door open, if
  - the axes are to be travelled manually (jog keys)
  - the tool changer is to be swivelled and
  - the main drive is to be switched on.
* Upon opening the door, if either main drive or CYCLE START are active.

Depending on the set priority for the door limit switch, ALARM 040 may also occur under the following condition:
* Automatic chip safety door, M53 active: if, after a CYCLE START in AUTOMATIC mode, the door is not closed after 10 sec..

Independently of the actual priority of the door limit switch, ALARM 040 will occur under the following operating conditions:
* If a CYCLE START is initiated with the bar feed active and with the chip safety door and workholding tool open.
* If the door is opened with the bar feed activated and open workholding tools as long as CYCLE START is active.
* If the operator tries to operate the collection tray with the door open.

On the operator monitor, different priority levels can be selected for the door limit switch depending on the hardware condition of the axis controller (see description of the operator monitor).

ALARM 050: TOOL TURRET HARDWARE FAILURE

All types:
* No modification of the position code within 5 sec. after swivel start (or no strobe for turrets with strobe signal)
* Actual position after swivelling does not correspond to the position calculated by the processor or is invalid.
* Check: No lock within 5 sec. after direction reversal (or after reaching the programmed position for Sauter turrets with direction logic)

Sauter turret without direction logic (type 1):
* Check: Pre-indexing when starting a swivel process not active and cannot be activated for 1 second through starting the tool changer in reverse direction (in doing so the control tries to release a possibly clamped pre-indexing pin)

Sauter turret without direction logic (type 1) and Duplomatic BSVN 160 (type 6):
* Check: Pre-indexing does not respond within 5 sec. after switching on or off

Hydraulic Sauter 4-faced turret (type 4):
* Invalid position code (more than one position bit active)
* After the advance swivelling time input in G1, a position signal is still present
* After the time limit input in G2 for the locking process, no position signal is present

ALARM 060: TOOL TURRET NOT READY

All types:
* Check: Lock not active (monitoring is only interrupted during the swivelling process)
* Upon cycle start after triggering a tool changer alarm, if the latter was not acknowledged through manual swivelling in MANUAL mode, or after aborting a swivelling process through RESET or EMERGENCY OFF
* When leaving the protected monitor

EMCO turret:
* Each time after switching on the control

ALARM 070: TOOL TURRET MOTOR FAILURE

All types except type 4:
A thermal overload of the tool changer motor occurs with the motor active.

ALARM 080: COOLANT PUMP OVERLOAD

This error occurs in case of high power consumption of the coolant motor either caused through mechanical overload (coolant viscosity, chips) or due to a phase error (phase protection defect or coolant pump not connected).
After error elimination the thermal protective switch in the switch cabinet has to be switched on.

ALARM 090: AUXILIARY DRIVES NOT READY

The auxiliary drives are not switched on thus inhibiting the start of main drive, axes and additional drives (tool changer, coolant / lubrication pump etc.).
After acknowledgement of the alarm message the auxiliary drives can be switched on with the "AUX ON" key.

***ALARM MESSAGES 100 - 190:* AXIS CONTROLLER**

ALARM 100: AC SYNTAX ERROR

Wrong format of a command to the axis control unit (AC). In normal operation, this error should not occur. After display of this alarm, the control must be re-initialized (switch off/on).

ALARM 101: X-AXIS: PROXIMITY DETECTOR ERROR

The inductive proximity switch for the stop check of the X axis is defect.

ALARM 103: Z-AXIS: PROXIMITY DETECTOR ERROR

The inductive proximity switch for the stop check of the Z axis is defect.

ALARM 104: X-AXIS: DEVICE NOT PRESENT

There is an error in the communication between processor and drive, i.e. the drive cannot be addressed by the processor.
Possible error causes:
* The corresponding drive board is not in the slot provided.
* The corresponding drive board is defective.
* If ALARMS 104-107 occur together there is an error in the 24V DC voltage supply unit. Check glass tube fuses on 24V supply unit (Y1A715000).

ALARM 106: Z-AXIS: DEVICE NOT PRESENT

see ALARM 104

ALARM 107: MAIN DRIVE: DEVICE NOT PRESENT

see ALARM 104

ALARM 110: AC OUTPUT BUFFER OVERLOW

The processing of the status messages of the axis controller (AC) is too slow. In normal operation, this error should not occur. After display of the alarm, the control must be reinitialized (switching off/on).

ALARM 111: X-AXIS: ENCODER SUPPLY ERROR

There is an error in the area of the position encoder of the corresponding drive.
Possible error causes:
* Position encoder defective
* Interruption of cable connection between position encoder and drive unit.

This error can be acknowledged by switching the control off and on.

ALARM 112: X-AXIS: SET SPEED NOT REACHED

There is an error within the driving unit which can only be acknowledeged through switching the control off and on again.
Possible error causes:
* Overload of axis drive
* Mechanical defect of axis drive
* Defect of control unit of axis drive unit

ALARM 113:X-AXIS: POWER SUPPLY NOT READY

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
* Defect of power supply unit on power board of drive unit
* Defect of control unit of drive unit

ALARM 114: X-AXIS: THERMAL OVERLOAD

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
* Thermal overload on power board of drive unit due to an overload of the drive
* Heavy contamination of filters of switch cabinet ventilator

ALARM 115: X-AXIS: MOTOR HIGHLOAD

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.

ALARM 116: X-AXIS: POSITION OVERFLOW

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
The drive cannot reach the position preset by the computer.
Possible error causes:
* The drive is blocked mechanically
* The drive cannot travel due to lacking supply voltage
* There is a defect in the drive unit
* Due on an error of another drive unit the voltage supply of this drive was also interrupted
* The drive was moved manually with auxiliary drives switched off

ALARM 117: X-AXIS: OVERCURRENT

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
* Carbon brushes of drive motor are worn out
* The drive motor is defective
* The power board of the drive unit is defective
* The control unit of the drive unit is defective

ALARM 118: X-AXIS: MOTOR OVERLOAD

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
* Excessive stress of drive during chip removal
* Mechanical defect on drive (sluggish operation)

ALARM 119: X-AXIS: LIMIT SWITCH OVERTRAVELLED

This alarm occurs if the slide is traversed over the limit of the working area. This is possible with existing ALARM 460.
To acknowledge this alarm the slide has to be removed to the working area with switched off auxiliary drives.

Subsequently the alarms that occur additionally have to be acknowledged by switching off and on.

ALARM 130: VALUE OUT OF RANGE OR INVALID INPUT DATA

This error occurs if the axis controller reads-in setting data which cannot be processed. The causes are false machine status data (MSD).
Remedy: New setting of the machine status data (reading-in of the MSD cassette).

ALARM 131: Z-AXIS: ENCODER SUPPLY ERROR

see ALARM 111.

ALARM 132: Z-AXIS: SET SPEED NOT REACHED

see ALARM 112.

ALARM 133: Z-AXIS: POWER SUPPLY NOT READY

see ALARM 113.

ALARM 134: Z-AXIS: THERMAL OVERLOAD

see ALARM 114.

ALARM 135: Z-AXIS: MOTOR HIGHLOAD

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.

ALARM 136: Z-AXIS: POSITION OVERFLOW

see ALARM 116.

ALARM 137: Z-AXIS: OVER CURRENT

see ALARM 117.

ALARM 138: Z-AXIS: MOTOR OVERLOAD

see ALARM 118.

ALARM 139: Z-AXIS: LIMIT SWITCH OVER-TRAVELLED

see ALARM 119.

ALARM 140: MAIN DRIVE SYNCHRONIZATION ERROR

The axis controller does not receive the correct signals in order to execute the rotation feed start command.
Error causes:

* A speed sensor of the main drive does not operate.
* Hardware error at the axis controller.
* Speed decrease due to main drive highload
* False, but plausible setting data otherwise ALARM 130).
* No synchronization pulse.

ALARM 141: MAINDRIVE: SET SPEED NOT REACHED

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:

* Overload of main drive
* Shaft encoder belt defective
* The control unit of the main drive is defective
* The armature fuse of the DC main drive motor is defective

ALARM 142: MAINDRIVE: ENCODER SUPPLY ERROR

see ALARM 111.

ALARM 143: MAINDRIVE: EXITATION CURRENT EXCEEDED

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:

* Bad zero conductor and/or earthing connection of the machine
* Voltage fluctuations in the power supply system
* Fuses on power unit of the main drive unit are defective
* Field winding in the main drive motor is interrupted

ALARM 144: MAINDRIVE: THERMAL OVERLOAD

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.

a) The thermal protective switch of the main drive ventilator motor has actuated:
   Possible error causes:
   * Fuses on power unit of main drive unit are defective

- * Main drive ventilator motor is defective
  After elimination of the error cause the thermal protective switch in the switch cabinet has to be switched on.

b) The thermal protective switch of the main drive ventilator motor has not actuated:
  - * Overload of main drive motor
  - * Heavily contaminated air filter in main drive ventilator

ALARM 145: MAINDRIVE: MAXIMUM SPEED EXCEEDED

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
- * Defect on main drive unit
- * Potentiometer $n>n_{max}$ on control unit of main drive unit is adjusted wrongly
- * Wrong machine status data
  Remedy: Read in MSD cassette

ALARM 146: MAINDRIVE: POSITION CONTROLLER ERROR

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
- * Defect on main drive unit
- * The main drive unit was overloaded for a short period
- * The main drive is blocked mechanically

ALARM 147: MAINDRIVE: POWER SUPPLY ERROR

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
- * Disturbances of power supply
- * Defect of power unit on main drive unit

ALARM 148: MAINDRIVE: MAIN POWER SUPPLY ERROR

There is an error within the driving unit which can only be acknowledged through switching the control off and on again.
Possible error causes:
- * Phase sequence error
  Remedy: Change phase sequence on terminals
- * Phase failure in power supply system
- * Fuse cartridges in switch cabinet are defective
- * Asymmetries in power supply system
- * Wrong order of supply voltage
- * Short-time failure influences of power supply system
- * Bad zero conductor and/or earthing connection of machine to power supply system

ALARM 150: AXIS OUT OF SYNCHRONIZATION. REFERENCE POSITION LOST

This alarm occurs at pulse motor operated axis drives.

a) If the slide cannot be traversed after acknowledgement of the alarm and after new switch-on of the auxiliary drives, the exact error cause is indicated by luminous diodes at the pulse motor board.
Possible error causes:
  - * Thermal overload of the pulse motor board
  - * Overload on-state current due to aa defective pulse motor
  - * Overvoltage or undervoltage due to bad electric connections

b) If the slide can be traversed without switching off and on the control the following error causes are possible:
  - * Too high load of axis drive (e.g. collision)
  - * Wrong machine setting data
    Remedy: Read in MSD cassette.
  - * The slide is sluggish as regards mechanics (lubrication)
  - * The inductive proximity switch is defective
  - * The distance of the inductive proximity switch is too large

ALARM 151: X-AXIS OUT OF SYNCHRONIZATION: REFERENCE POSITION LOST

The monitoring unit for the axis movements has detected an error in the X drive position caused by a feed motor overload.

ALARM 153: Z-AXIS OUT OF SYNCHRONIZATION.REFERENCE POSITION LOST

The monitoring unit for the axis movements has detected an error in the Z drive position caused by a feed motor overload.

ALARM 160: BAD PARAMETER FOR G02 OR G03

In a circular command, a false parameter or a parameter with a false value was input. This alarm occurs under the following conditions:
- * Centre coordinate input missing.
- * Centre coordinate outside the numerical range of the machine (the second centre coordinate, which is not input, may be generated in this way).

* The second centre coordinate does not fit into a circle.

ALARM 170: TRIED TO START WITH FEED = 0

This alarm occurs when the operator tries to execute an axis movement which is not possible due to the following reasons:

* Straight feed: F = 0 active (F code missing)

* Rotational feed:
a) F = 0 active (F code missing)
b) Main drive not switched on
c) S = 0 active (no speed was programmed)

Note: Resetting the feed rate override switch to zero will not trigger this alarm as long as the input of the feed command is correct.

ALARM 180: WRONG CENTER COORDINATE SPECIFIED

The centre coordinate of the axis with the shorter travelling distance between start point and target must be programmed.

ALARM 190: RADIUS TOO LARGE

ALARM MESSAGES 200 - 260: MACHINING CYCLES

ALARM 200: INVALID VALUE OF D OR P PARAMETER

* G04/86/87/88: The maximum value for D4 (10000, i.e. dwell time of 1000 seconds) was exceeded.

* G85: For D3 a value of 0 was input or D3 was not programmed; an invalid value was programmed for D5 (only 0, 40, 55, 60 and 80 degrees are permissble).
D6 is larger than the distance between start point and target in travelling direction.

* G86: D5 is larger than the overall width of the groove

* G87: The value of D5 is higher than 100

ALARM 210: INVALID TAPER PARAMETER

* G84/85 with division of cut depth: The sign of a taper parameter in the feed axis must correspond to the feed direction

* G84/85 without division of cut depth: The amount of a taper parameter in the feed axis is higher than the entire feed with the sign of this taper parameter being opposite to the feed direction

* G84/85: The amount of a taper parameter in the non-feed axis is higher than the related distance between start point and target where the sign of this taper parameter reduces the distance between start point and target

ALARM 220: INVALID REMINDER

G84: A reminder input in D0/D2 is higher than the whole feed in the respective axis.

ALARM 230: INVALID CYCLE TARGET

* G84: Start and target coordinate in one axis may onlly be identical if a taper parameter is programmed in this axis. In this case no taper parameter may be programmed for the other axis.

* The distance between start point and target is too high (a maximum of 1FFFF H -131071- steps is permissible)

* G85/86: Start point and target may not be identical in any axis

* G87/88: The machining path may not be 0

ALARM 240: NO OR INVALID STEP DEPTH

* G84: D3 = 0 was programmed

* A reminder parameter (D0/D2) but no cut depth division was programmed

* G85: D3 is larger than D6 or larger than the distance between start point and target

* G86: D3 is larger than the distance between start point an target

ALARM 250: D OR P PARAMETER FOR GIVEN CYCLE MISSING

* G85: D3 is not programmed

* G96: D5 is not programmed

ALARM 260: DRILL NOT CENTERED

G87/88: When starting a drilling cycle, the drill is not positioned in the Z-axis.

ALARM MESSAGES 290-340: PROGRAM SEQUENCE SUBROUTINES,G27

ALARM 290: NO SIMULATION OUT FROM A SUBROUTINE

It is not allowed to start a simulation from a subroutine.

ALARM 300: MORE THAN 10 SUBROUTINES NESTED

ALARM 310: SUBROUTINE NOT IN MEMORY

* A subroutine called by G25 could not be found in the part program memory of the control.
* The called subroutine does not include any blocks.

ALARM 320: G25/G27 NOT ALLOWED IN EXECUTE-MODE

These jump instructions are not useful when processing indicidual blocks/codes from the block memory in the EXECUTE mode and are thus not executed.

ALARM 330: M17 WITHOUT G25 OR M30 IN A SUBROUTINE

* M17 was found in a part program which was started as main program.
* M 30 in a part program which was called by G25

ALARM 340: G25/G27 NOT ALLOWED IN LAST BLOCK OF PROGRAM

ALARM MESSAGES 350 - 440: PART PROGRAM INTERPRETER

ALARM 350: INVALID CUTTER RADIUS

G41/42: The radius of the active tool is zero. No tool offset is active.

ALARM 360: NO CHANGE OF T-WORD WITH G41/G42 ACTIVE

With the tool radius compensation active, no other tool offset function can be called.

ALARM 361: NO M65 WITH G41/G42 ACTIVE

With active cutter radius compensation no M65 must be programmed.

ALARM 370: NO CHANGE OF SCALE WHEN CYCLE ACTIVE

This alarm occurs if a new scaling factor is selected but a cycle is active (concerning milling) or if G51 is set active with a cycle in the same block.

ALARM 371: NO CHANGE OF PSO WHEN SCALE ACTIVE

This alarm occurs if with active scaling fator a PSO register is selected or deselected. A PSO register can only be changed if the scaling factor is not active.

ALARM 372: NO RELATIVE MOVES AFTER G51

After G51 an absolute move must ensue to make the starting point of the zoomed contour independent of the slide starting position (see programming instructions).

ALARM 373: NO NEGATIVE SCALE ALLOWED

Only positive values are allowed for the P7 parameter when indicating a scaling factor.

ALARM 374: INVALID PARAMETER FOR G51

When indicating the reference point for the scaling factor an invalid D or P parameter was indicated.

ALARM 375: SCALE CALCULATION OVERFLOW

The size of the resulting contour is too great. Check the reference point for the scaling factor and thescaling factor.

ALARM 380: BAD OR MISSING PARAMETER IN G25/G27

* In a block with G25/G27 no L-code was programmed.
* It was tried to call the part program already active.
* An L-code belonging to G27 includes a block number not present in the active part program.

ALARM 381: AFTER CHAMFER/RADIUS ONLY G01 ALLOWED

After a block with programmed chamfer/radius no other traverse command than G01 (i.e. no cycle and no G00) is allowed.

ALARM 382: MISSING POSITIONPARAMETER FOR CHAMFER/RADIUS

The block after a programmed chamfer/radius must contain position parameters (absolute or incremental).

ALARM 383: THREE DIMENSIONAL CHAMFER/ RADIUS NOT ALLOWED

The programmed chamfer/radius must be in the same plane, inserting three-dimensional chamfers or radii is not possible.

ALARM 384: CHAMFER/RADIUS CALCULATION OVERFLOW

There was an overflow when computing the correction points for the chamferer/radius to be inserted.
Please check again the position data of the programmed chamfer/radius as well as the subsequent block.

ALARM 385: NO CHANGE OF PSO IF CHAMFER/ RADIUS ACTIVE

The PSO must not be changed in the block with the programmed chamfer/radius since otherwise the chamfer/radius that is to be inserted cannot be calculated correctly.

ALARM 386: NO CHANGE OF SCALE IF CHAMFER/RADIUS ACTIVE

The scaling factor must not be changed in the block with the programmed chamfer/radius, since otherwise the chamfer/radius that is to be inserted cannot be calculated correctly.

ALARM 387: NO CHANGE OF TOOL WHEN CHAMFER/RADIUS ACTIVE

Due to the calculation of the tool data during tool exchange no change of tool must be carried out in the block with the programmed chamfer/radius, since otherwise the chamfer/radius that is to be inserted cannot be calculated correctly.

ALARM 388: CHAMFER/RADIUS IN EXECUTE MODE NOT ALLOWED

Programmed chamfers/radii cannot be executed in the EXECUTE mode because the subsequent block is necessary for a chamfer/radius that is to be inserted.

ALARM 389: PROGRAMMED CHAMFER/RADIUS TOO GREAT

The indicated chamfer/radius is to large. It must not be larger than the shorter of the two straight lines between which the chamfer/radius should be inserted.

ALARM 390: CHANGE OF G-CODE GROUP 7/9 ONLY IN FIRST BLOCK

ALARM 391: NEGATIVE CHAMFER/RADIUS NOT ALLOWED

The indicated chamfer/radius must not be negative.

ALARM 400: NO G-CODE FOR GIVEN PARAMETER ACTIVE

A selected parameter can be assigned to a G-code:

* No G-Code from group 0 for a selected position parameter is active (is also displayed, if a position parameter is programmed in a block with G04).
* An L-code was programmed without G25/27.

ALARM 410: INVALID G-CODE

This alarm occurs, when a G-code which has been programmed is not by the control. The valid G-code block depends, inter alia, on the type of software the costumer desires for the machine (e.g. G41,42).

ALARM 416: BAD PARAMETER FOR G02 OR GO3

A false parameter or a parameter with a false value was input in a circular movement command. This alarm occurs under the following conditions:

* No centre coordinate
* Centre coordinate outside the numerical range of the machine (the second centre coordinate which was not input can be generated in this way).
* The second centre coordinate does not fit into a circle.

ALARM 418: WRONG CENTER COORDINATE SPECIFIED

The centre coordinate of the axis with the smaller travelling distance between start point and target must be programmed.

ALARM 419: RADIUS TOO LARGE

ALARM 420: INVALID M-CODE

This alarm occurs if an M-code which cannot be processed by the control is programmed. The set of valid M-codes depents among others on the customer-specific periphery of the machine (e.g. M20/21, M23/24, M25/26, M50/51).

ALARM 430: INVALID T-WORD

If a tool offset is selected, the tool number must be input as well.

ALARM 440: TARGET LIMITS EXCEEDED

EXECUTE/AUTOMATIC mode: The programmed targets are monitored by softwar limit switches which trigger ALARM 440 if required.(Possibly false data in the position-shift register or false tool data.)

ALARM MESSAGES 450 - 490: ***GENERAL OPERATING ERRORS***

ALARM 450: ENTERED CAUTION ZONE

Manual mode: overtravel of the safety distance to the software limit switches. The control will switch to a lower feedrate in order to be able to stop when reaching the software limit switches wihout a considerable brake path length.

ALARM 460: REFERENCE POSITION NOT ACTIVE

The reference coordinate system of the machine is not active but after positioning the axis at the reference point; only this allows for the absolute positioning and display of absolute positions.

ALARM 470: RESTART MAIN DRIVE

* When switching off FEEDHOLD: The main drive was switched off during FEEDHOLD but was not switched on again.
* When switching off DRYRUN: If, at this point of time, M03 or M04 are active, the main spindle must be switched on again when DRYRUN is deselected.

ALARM 480: NO OR INVALID PARAMETER FOR G-GROUP 0

* An arc centre parameter was programmed although neither G02 nor G03 is active.
* In a cycle of the G-code group 0, an invalid D- or P-parameter was programmed.
* G04: Parameter D4 (dwell time) was not programmed.
* G84/85/86: The target must be input for both axes.
* G87/88: The target must and may only be programmed in Z.

ALARM 490: OFFSET CHANGED, GO WITH G00

After the changing of tool offset or position shift registe, the control will only accept G00 as travelling command.
E.g. T505 G01 U10. F500 = > ALARM 490

ALARM MESSAGES 500 - 580: ***TOOL RADIUS COMPENSATION***

ALARM 500: TOO MANY BLOCKS WITHOUT SLIDE OPERATION

G41/42: More than five consecutive blocks without change of the XZ-value were programmed.

ALARM 510: TOO FEW POINTS PROGRAMMED

G41/41: Prior to deselecting the compensation with G40 or M30, at least two blocks with a change of the XZ-value must be programmed.This error will also occur when G41/42 are called in the EXECUTE mode.

ALARM 520: ERROR AT COMPENSATION START OR END

* The first movement after selecting or deselecting the tool compensation must be with G00 or G01.
* When selecting or deselecting the compensation, the XZ-values must have changed as against the subsequent or previous values. The change of only one value is also permissible.

ALARM 530: NO IMMEDIATE CHANGE OF G41/42

G41/42: In order to switch between G41 and G42, the compensation must first be deselected with G40 and a movement must be executed.This requires a change of the XZ-values.

ALARM 540: BAD CIRCLE PARAMETER

G41/42: A circular movement command includes a false parameter or a false numerical value for a parameter. Possible error causes see ALARM 160.

ALARM 560: RADIUS TOO LARGE

G41/42: The radius of a circular movement is too large (compare ALARM 190).

ALARM 570: RADIUS TOO LARGE FOR GIVEN CONTOUR

G41,42: The radius of the selected tool is too large for the programmed contour.
Possible causes of error:

* Programming of an arc with a smaller radius than the tool radius.
* Programming of an internal corner which is limited by two arcs if special geometrical conditions apply (especially when the tool radius is considerably larger than the smallest programmed radius - cf. section on cutter radius compensation).

Note: Contour violations in blocks more than one block after the presently processed block or blocks which will not be processed but after the violation cannot be detected.

ALARM 580: PART PROGRAM MUST END WITH G40 ACTIVE

G41/42: Part programs must be ended with the compensation function deselected (deslection with G40 or M30).

ALARM MESSAGES 600 - 710: EDITOR

ALARM 600: INCORRECT EDITING SEQUENCE

* Attempt to select a block although no part program is selected.
* Attempt to select a word although no block is selected (this is only possible in the EXECTUE mode).
* Input error for the functions "Erase program memory/erase offset register": the input sequence "PROGKILL/OFFSKILL" was not observed.
* Reading in of data from cassette/RS232: The data format input in MON parameter L4 for the reading process does not correspond to the data format used for writing.

ALARM 610: INVALID PARAMETER ENTERED

Another parameter than D, L or R was selected on the operator monitor.

ALARM 620: INPUT VALUE OUT OF NUMERICAL RANGE

* EDIT mode: Attempt to input a numerical value outside the limits fixed in the machine specifications.
* AUTOMATIC/EXECUTE: The limit for straight or rotational feed was exceeded.

Note: The numerical input limits are specific to the machine and thus to be obtain from the respective machine instructions.

ALARM 630: INVALID SUBROUTINE NUMBER

A part program can only be ended with M17 as subroutine if its 0 number lies within the numerical range valid for subroutines (this range is input in parameter L3 on the operator monitor (MON)).

ALARM 640: BLOCK NUMBER ALREADY EXISTS

Attempt to change the number of a block to the number of another block which is already present in the active part program.

ALARM 650: BLOCK MEMORY OVERFLOW

Attempt to input a too long block.

ALARM 660: USER MEMORY OVERFLOW

The memory capacity of the control for part programs is exhausted with the data already stored.

ALARM 670: CANNOT OPEN PROGRAM - TOO FEW MEMORY

To activate a program a certain freely available memory is necessary (corresponds to the program length of the program which is to be activated). If this memory is not available this alarm esues.
Remedy: Create space by deleting programs that are not used any more.

ALARM 675: TOO MANY PROGRAMS IN THE MEMORY

ALARM 690: INVALID INDEX ENTERED

* EDIT and EXECUTE Mode: Attempt to input a P- or D-parameter with index > 7.
* Tool data or shift register selection: attempt to input a tool index > 99 or a shift register index > 5.

ALARM 700: NO CHANGE OF ACTIVE TOOL DATA/PSO

EDIT: Attempt to change the active selected tool offset or the active shift register. A change is only possible after deselecting the respective tool or register. This deselection is simply executed through pressing the RESET key or, in the EXECUTE mode, through processing a block with deselection function (other tool or T0 or other register or G53/56).
AUTOMATIC/EXECUTE: Attempt to change the shift register 5 with G92, although G59 is active.

ALARM 710: PROGRAM NUMBER ALREADY EXISTS

Attempt to change a program number to the number of a program which is already stored in the memory.

Note:
The following alarm messages 730 to 779 only occur in connection with the graphic simulation.

<u>ALARM MESSAGES 730 - 760:</u> GRAPHICIPP - INTERPRETER

ALARM 730: PRINTER NOT READY , HARDCOPY TERMINATED

It was tried to print a screen content without the printer being ready for operation (e.g. printer turned off)

ALARM 731: PRINTER OFFLINE

It was tried to print a screen content without the printer being ready for reception.
Help: turn ONLINE at the printer

ALARM 732: OUT OF PAPER

ALARM 733: PRINTER NOT CONNECTED

Printing cable not connected properly or defective

ALARM 734: PRINTER ERROR

The printer sets his error line during printing. For error elimination the printing manual needs to be consulted.

ALARM 740: GRAPHIC LIMITS EXCEEDED

A position should be approached which exceeds the numerical limit of the graphics. (Check the actual tool shift data and the offset registers. It is also possible that too large a scale was chosen.)

ALARM 741: TOOL NOT DEFINED

If a tool is called in the polygon program of which no marked line is programmed, alaram is effected.

ALARM 742: TOOL TOO LARGE

Occurs if the machining part of a tool is larger than the screen. This alarm can be eliminated by chosing another scale.

ALARM 743: ONLY 1 MB AVAILABLE

For the 3d-display and the zoom function a storage retrofit kit from 1 MB to 2 MB is required.

ALARM 744: 3D GRAPHIC NOT ACTIVATED

It was tried to display a 3d-picture without activating the 3d- graphics with the softkey 'AKTIVATE 3D'. To achieve a 3d-display this softkey needs to be activated and the programme has to be executed once again.

ALARM 745: WRONG MACHINE

It was tried to activate the 3d-graphics on a lathe.

ALARM 750: POLYGON PROGRAM NOT AVAILABLE

* It was attempted to call a polygon program with G26, which is not stored in the program memory.
* In case of a tool change (manually or via T-code): the polygon program corresponding to the desired tool changer position is not stored in the memory.

ALARM 751: INVALID POLYGON PROGRAM NUMBER

* It was attempted to call a polygon program with G26 with the program number in the L parameter being outside the numerical range reversed for polygon programs (7000 - 9999 with the exception of the number 8001 to 8899 reserved for tool profile programs).

- * An invalid value was input in parameter L in program 8000 (for tool polygon programs only numbers 8001 to 8899 are valid).

ALARM 752: INVALID G-CODE IN POLYGON PROGRAM

- * A G-code which is generally not permissible in polygon programs was detected in such a program.
- * G61 is only valid in tool polygon programs (8001 - 8899).
- * G63, G64 and G68 are only valid in blank polygon programs (7000 - 7499).
- * G67 is only valid in blank (7000 - 7499), workholding tool (9000 - 9499) and sleeve polygon programs (9500 - 9999).

ALARM 753: INVALID PARAMETER IN POLYGON PROGRAM

- * A parameter not permissible in polygon program was found in such a program.
- * The parameters L and T are only valid in allocation programs (o 8000 and o 8900 - 8999).
- * In o 8000 and o 8900 - 8999, only parameters L and T are valid.

ALARM 754: NO G-CODE FOR GIVEN PARAMETER IN POLYGON PROGRAM

A parameter in a block of a polygon program cannot be assigned to a G-code. This alarm e.g. occurs when programming a position parameter without previously or simultaneously programming a G- code.

ALARM 755: INVALID PARAMETER FOR GIVEN G-CODE IN POLYGON PROGRAM

- * The parameter I, J, K are only valid for G02/02.
- * G63 only accepts Z.
- * G61, G63, G64: parameter X is not permissible.

ALARM 756: BAD PARAMETER FOR G02 OR G03 IN POLYGON PROGRAM

A false parameter or a parameter with a false numerical value was input in a circular movement command. This alarm occurs under the following conditions:

- * Centre coordinate missing
- * Too many target coordiantes (X, Y, Z - must all be programmed anew)
- * Centre coordinates outside the numberical range of the machine (the second centre coordinat which is not indicated can be generated in this way).
- * The second centre coordinate does not fit into a circle.

ALARM 757: PARAMETER FOR GIVEN G-CODE MISSING

- * G60, G68: Either both or none of the parameters must be programmed.
- * G67: At least one parameter must be programmed.

ALARM 758: WRONG CENTER COORDINATE SPECIFIED

The centre coordinate of the axis with the smaller travelling distance from start point to target must be programmed.

ALARM 759: RADIUS TOO LARGE IN POLYGON PROGRAM

ALARM 760: SYNTAX ERROR IN POLYGON PROGRAM

- * T and L in o 8000 may only be programmed together.
- * This error may also occur if the instructions for the graphic tool changer are missing or are insufficient or wrong.

ALARM MESSAGES 780 - 799:
GENERAL OPERATING ERRORS

ALARM 780: SAFETY LOCK ACTIVE

The control is in a locked state, where two priorities are to be distinguished: general lock of the control and memory lock. The lock can only be released by authorized operating personnel.

ALARM 785: WPC-PRESET = 0, CYCLE START IGNORED

If the number of pieces is set to zero (operator monitor) in the "workpiece presetting" mode, the cycle start key is ignored.

ALARM 790: OPERATING ERROR IN TOOL MEASURING MODE

This alarm may occur in measurements with reference part:

- * When starting with SHIFT - T, no tool offset may be selected.
- * With the T-LED flashing, only X and Z values may be input. The numerical value for X or Z was not input prior to pressing ENTER.
- * The confirmation of tool data with SHIFT-T with the T-LED flashing is only possible if the

dimensions of the reference part in X and/or Z were input immediately before.

ALARM MESSAGES 800 - 870: DATA INTERFACE CASSETTE, RS 232)

ALARM 800: CASSETTE DRIVE NOT READY

* No cassette was input.
* Hardware error of the cassette recorder

ALARM 810: WRITE PROTECTED CASSETTE IN USE

The black write protection button was removed from the cassette which is presently used.

ALARM 811: INTERFACE CONTROLLER NOT READY

The interface controller does not respond within a defined period of time.This alarm only occurs in case of a hardware error of the control. Initialize control through switching it off and on again.

ALARM 820: BLOCK STRUCTURE ERROR

* Use of a non-formatted cassette
* Serious damages to a cassette due to mechanical or electrical causes (remedy: new formatting)
* Hardware error of the cassette device
* Starting from software version DC V3.0, it was attempted to write data on a cassette which was formatted with an older software version. If this is not possible, it may be necessary to read-in the data from the cassette into the control, re-format the cassette and then store the data from the control onto the newly formatted cassette.

ALARM 830: BLOCK CHECKSUM ERROR

* Error in the data transmission between cassette and memory (countermeasure: new formatting)
* Hardware error of the cassette device
* Using an older software version, the user has tried to read-in a cassette onto which date were stored with software version 3.0 ff.

ALARM 840: INSUFFICIENT TAPE SPACE

The user has tried to store a program on a cassette; however, the program length exceeds the remaining memory capacity.

ALARM 850: PROGRAM NOT FOUND

* Attempt to read-in a program not yet stored.
* Attempt to read-out a non-existent program.
* Attempt to read machine data from other than MSD cassettes.
* False input sequence when reading-in the MSD cassette.ALARM 860: INTERFACE OPERATING ERROR

False operating sequence when reading-in part programs via the serial interface (see description of the INTERFACE mode).

ALARM 870: WRONG BAUDRATE SELECTED

On the monitor, the baudrate for the data transmission via the serial interface can be set and input in D0. Only values between 150 - 4800 are permissible.

ALARM 880: INTERFACE ERROR

Occurs if e.g. during RS232 read-in the baud rate or the configuration of the serial interface in control and PC do not correspond to each other.

ALARM MESSAGES 900 - 969: PERIPHERAL DEVICES

ALARM 900: CHUCK/COLLET NOT READY

* Triggering of a limit switch with the chuck collet closed, if the final position monitoring is active
* If, when operating the workholding tools, the respective pressure switch does not respond within a time specified by the machine data
* Front-end-chuck and identical controlled pneumatic clamping cylinders (e.g. EMCO-TURN 220) : if upon opening/closing the chuck the pressure switch does not respond within the time specified in the machine data or if the switch is not released after completion of the opening/closing process
* In case of an attempt to switch on the main spindle although the chuck/collet is not closed.

ALARM 905: DUST EXTRACTOR THERMAL OVERLOAD

The thermal control of the dust extractor has been actuated (e.g. too long operating time, overload).
Possible error causes:
* Overload
* Motor is defective
* Phase failure of power supply voltage of motor

ALARM 910: CHUCK/COLLET PRESSURE FAILURE

Failure of the system pressure within the chuck/collet circuit. The monitoring of the system pressure is only carried out if the workholding tools (and, where applicable also the automatic tailstock sleeve) are in a stationary condition (LED of the related key does not flash)

ALARM 911: HYDRAULIC THERMAL OVERLOAD

The thermal control of the hydraulic system has been actuated (e.g. too long operating time, overload).
Possible error causes:
* Overload
* Motor is defective
* Phase failure of power supply voltage of motor

ALARM 915: CONVEYOR THERMAL OVERLOAD

The thermal control of the conveyor has been actuated (e.g. too long operating time, overload).
Possible error causes:
* Overload
* Motor is defective
* Phase failure of power supply voltage of motor
* The chip conveyor is blocked mechanically

ALARM 925: M27 TIMEOUT ERROR

Ensues if M27 ready message does not appear within the time limit selected in the machine data.

ALARM 926: M65 TIMEOUT ERROR

Ensues if M65 ready message does not appear within the time limit selected in the machine data.

ALARM 930:TAILSTOCK NOT READY

* Sleeve has reached its front limit position
* If, with the sleeve clamped, either one of both limit switches is triggered (front or rear position of the sleeve) or the pressure message "sleeve clamped" is not output
* If, with the sleeeve at its right limit position, no respective limit message is output
* If, after having reached the front limit position, the key "sleeve advance" is operated without moving the sleeve to its rear limit position before

ALARM 940: NO M20/M21/M25 DURING SPINDLE ROTATION

If, with the main spindle switched on or coming to a standstill, the operator tries to open the workholding tool or to moove the tailstock sleeve

ALARM 950:WORKPIECE CATCHER NOT READY

* Failure of the signal "collection tray swung-out" with the output "swing-in collection tray"
* Failure of the signal "collection tray in front position" with the output "swing-in collection tray"
* M24: Time limit exceeded during the advance (the signal "collection tray in front position" must become inactive within 5 sec.)
* M24: Time limit exceeded during the swing-in process (the signal "collection tray swung-out" must become inactive within 2 sec.)
* M23: Time limit exceeded during the swing-out process (the signal "collection tray swung-out" must become active within 5 sec.)
* M23: Time limit exceeded during the retract (the signal "collection tray in front position" must become active within 2 sec.)

ALARM 960:BAR END REACHED

* When the program is processed, this alarm is only output if the bar end signal is output with the chuck/collet open (bar feed in MSD activated, bar end detection in MSD not deselected)
* A cycle start with the bar end signal active is only permissible in the SINGLE mode

ALARM 961: BAR FEED HARDWARE FAILURE

<u>ALARM MESSAGES 970 - 992:</u> OPERATING SYSTEM ERRORS

ALARM 970: FATAL ERROR, CONTACT EMCO!

ALARM 975: FATAL ERROR, CONTACT EMCO!

ALARM 976: FATAL ERROR, CONTACT EMCO!

ALARM 980: FATAL ERROR, CONTACT EMCO!

ALARM 981: FATAL ERROR, CONTACT EMCO!

ALARM 982: FATAL ERROR, CONTACT EMCO!

ALARM 983: FATAL ERROR, CONTACT EMCO!

ALARM 984: FATAL ERROR, CONTACT EMCO!

ALARM 985: FATAL ERROR, CONTACT EMCO!

ALARM 990: FATAL ERROR, CONTACT EMCO!

ALARM 991: FATAL ERROR, CONTACT EMCO!

ALARM 992: FATAL ERROR, CONTACT EMCO!

ALARM 993: FATAL ERROR, CONTACT EMCO!

ALARM 994: FATAL ERROR, CONTACT EMCO!

ALARM 995: FATAL ERROR, CONTACT EMCO!

These alarms with the numbers 970 to 999 should never occur! If they occur repeatedly, please contact your nearest EMCO agency.