SIEMENS

SINUMERIK 8T/ Sprint 8T

Programming Instructions

SINUMERIK 8T/Sprint 8T

Programming Instructions

Edition 5.84

Siemens Aktiengesellschaft

SINUMERIK - Documentation

Key to editions

Up to the present edition, the editions below have been issued.

In the column "Alterations" the chapters are listed which have been altered with respect to the pre-ceding edition.

Edition	Order Number	Alterations
A.3.80	-	First edition
A.5.80	E321/1719-101	Revised Edition.
A.11.80	E321/1719-101	pp. 1-1, 1-6, 3-27, 3-30, 3-35, 3-40 4-10 5-2, 5-3, 8-17, 8-19, 8-21
A.3.81	E321/1801-101	Revised edition
A.8.81	E321/1873-101	Revised edition
A.10.82	E321/1972-101	Revised edition
A.05.84	E80210-T43-X-A7-7600	Revised edition

Contents

0.1 General notes on the programming instructions

The following assumptions are made in the programmes used for these programming instructions.

1. All machining is carried out from behind the turning axis.

2. The user datum decimal point input is set.

3. The decimal point is written even when it is automatically generated by the control.

4. Block construction is in accordance with DIN 66024, DIN 66025, DIN 66217, ISO R 1056, ISO R 1057 and ISO R 1058.
5. The programming examples are written in ISO code.
6. Diameter programming is used for the X values, with absolute data (machine datum set).

7. All geometric values are metric.
   For conversion into inch see chapter 8.

8. For better understanding preparatory functions are even programmed, if these are commands with reset position.

9. The maximum values given are limit values for the controls.
   They can be limited in practice by the machine, interface and input/output devices.

10. These programming instructions are designed for the maximum functional range of the control. Functions to be realized by options are accordingly marked.

11. The contents of these programming instructions can be found in the fold-out program key.

    Functions not included in this manual may be available in the control. However this does not guarantee that these functions will be available with new equipment or in the case of service.

    We reserve the right to amend these instructions for technical reasons without prior notice.

1.0 Programme construction

1.1 Tape code

The data on the tape is coded in accordance with defined standards, i.e., a given combination of holes always represents the same character. Two code structures are used.
DIN 66025 (IOS)
EIA-RS 244-A

The control recognises automatically the correct code. Code recognition results from the first character read % or EOR or LF or EOB. This last when the appropriate **user** data is set. Each tape must be written in one of the permitted codes. A change of codes on a single tape or joining together of tapes with different codes is not permitted and results in activation of the character parity check.

The characters constituting the two codes have the following common features:

ISO The number of holes is always even.
EIA The number of holes is always odd.
This criterion of odd or even number of holes is used as a simple check of the programme following the second character with a error recognition rate of almost 100 %.

The block parity check monitors that the number of characters in a block is even. Uneven character counts should be completed with "HT" or "SP".
This test can be de-activated.

As an additional test the programme is read into the internal programme store twice and a complete programme comparison made. When the error is detected read-in is stopped and the error indicated on the operators control panel.

Organisation of the word adresses is in accordance with DIN 66025 (ISO).

1.2 Available characters

The control reads every character contained in the tape code. However, only certain characters may be used for defining the programme organisation, as well as geometry and technology of the actual machining programme.

ISO - Code

Adress letters	A,B,F,G,H,I,K,L,M,N,R,S,T,X,Z
Digits Available special characters	0,1,2,3,4,5,6,7,8,9 %, (,), +, -, /, :, ., @
Non-available special characters	HT Tabulator SP Space DEL Delete CR Carriage return LF Line feed

READ IN the following characters are neither stored nor executed	PUNCH-OUT PRINT -O UT The following characters are generated
HT	
SP (except in remarks)	SP (after each word except within remarks)
DEL	
CR (sequence CR LF optional)	LF CR CR Selectable (machine data) CR LF

LF is indicated in the display as *.

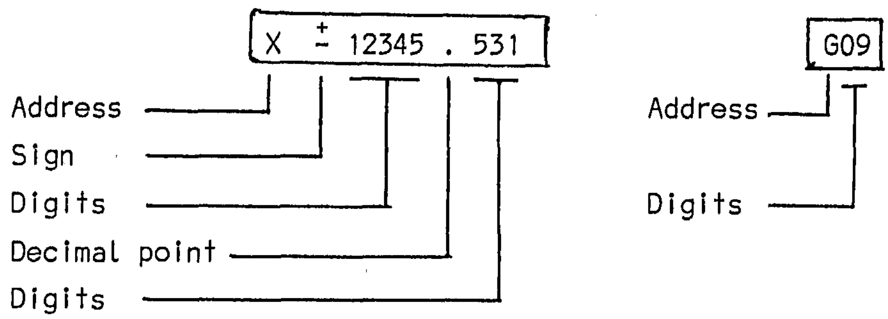
1.3 Word Construction

A word consists of an address letter and a series of digits with or without a sign . (Addressformat)

The word construction and therefore also the input format is exactly defined by and is represented in accordance with DIN 66025 sheet 4.

metric: %04 N04 G02 XL+053 ZL+053 ID053 (AL035 angle BD053 radius: only Sprint 8T)
KD053 F05 L5 S04 T04 H06 R2
RL+08

Word Examples



Word	With decimal point input (user datum)	Without decimal point input (user datum)
1 µm	0.001 or .001	1
10 µm	0.01 or .01	10
100 µm	0.1 or .1	100
1000 µm	1. or 1	1000
10200 µm	10.2	10200
100000 µm	100. or 100	100000

Decimal point input is possible using the following addresses
X,Z,I,K,A,B,F (mm/rev) see 8.2.6

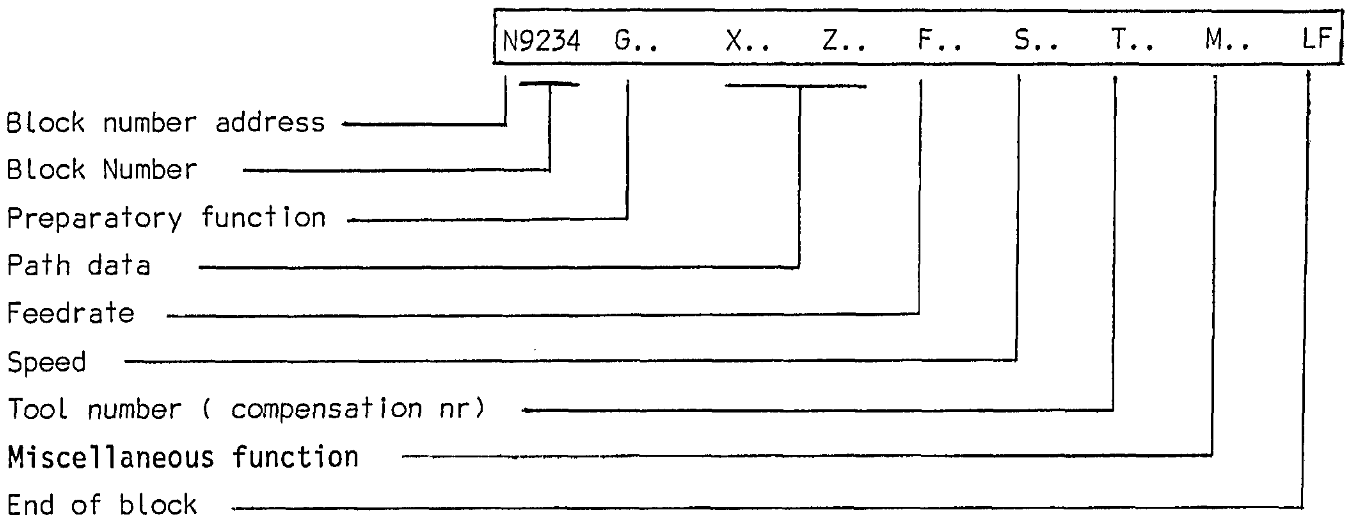
Note: If the setting data decimal point input is changed, the zero point displacement, tool offsets and parameters are correspondingly entered.

1.4 Block construction

Block consists of several words and the character " End of Block"

The block has a maximum possible length of 120 characters.

Example of a block :



Address for the block number : Main block

N Subsidiary block

Main block - in the main block and where appropriate also in the subsequent blocks, all the words must be given which are necessary to be able to start the machining sequence relating to the programme section beginning with this block.

Subsidiary block - Contains all the changing functions

Jumps can occur in the sequence of block numbers, e.g., modified or newly introduced blocks can be designated by introducing a digit into the fourth decade position. By introducing the character"/" in front of the block number the block so designated may be skipped by the control.

This function is dependant on the switch "Skip".

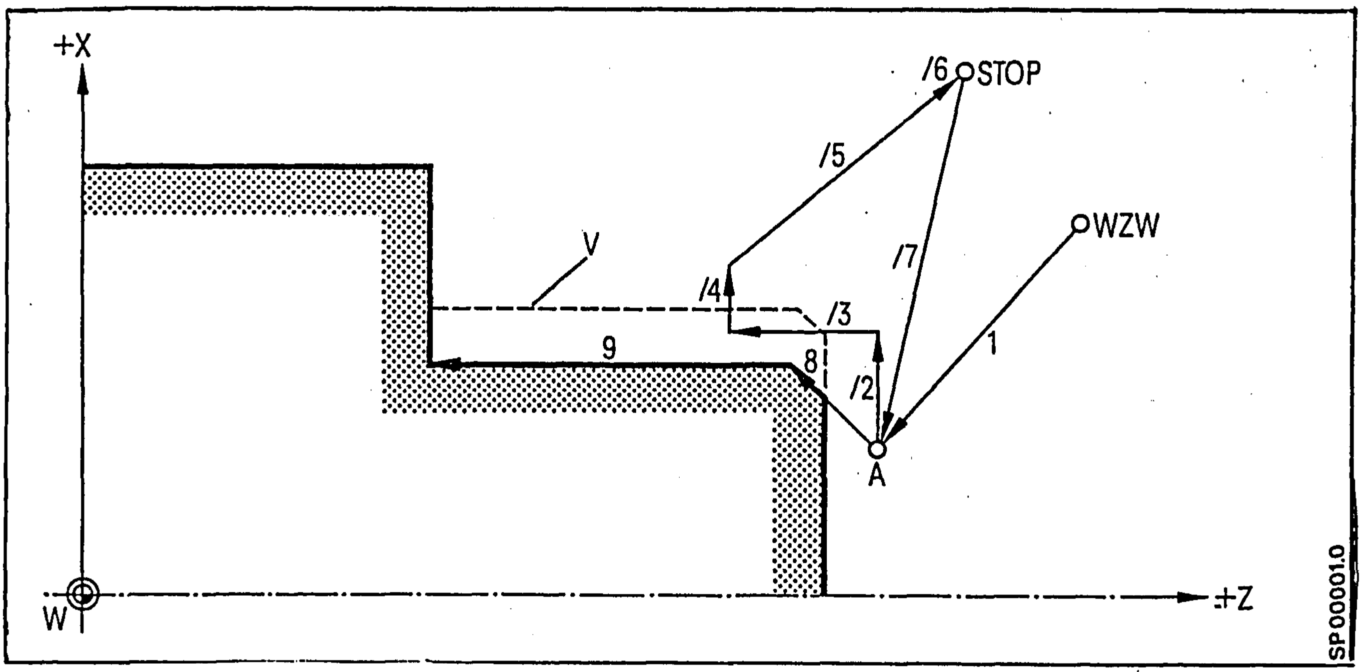
/ : Main block delete
/N Subsidiary block delete

Thus it is possible to delete certain machining sections such as test cuts, dry runs for threadcutting or part cuts necessitated by inaccurately defined offsets, i.e., their execution is not required for each component of the series. L999 must however be programmed before block delete in order to enable delete during machining.

It is important to ensure that block deletes form a closed loop (i.e. with the same starting and finishing point).

A = Starting Point ; V = Premachined contour ;

WZM = Tool Change Point;



If required, the blocks 2 to 7 can be deleted.

1.5 Leader

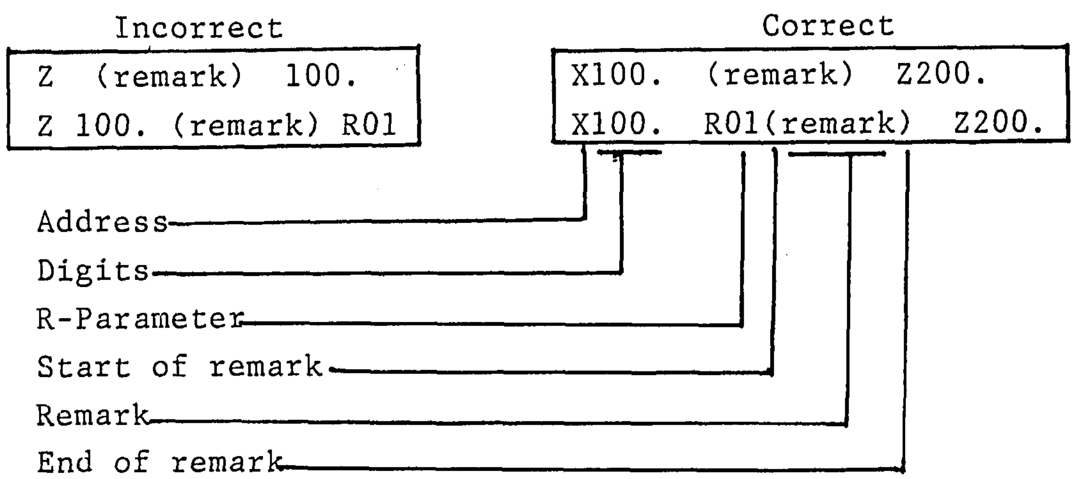
The leader is used to differentiate between different tapes. All characters - except % if the automatic recognition is initiated by %, or LF if the automatic code recognition is initiated by LF, are permitted in the tape reader. During execution of the programme the leader is skipped by the control. The leader is not stored.

1.6 Remark

Programme blocks can be more clearly defined using remarks. In this way it is possible to bring up operator instructions on the display (part programme picture).

A remark must not contain the characters % or LF.
The maximum length for a remark is 29 characters.
A remark between M02 or M30 and another M function, e.g. M30 (comment) M40 LF is not allowed.
If more text than one remark is required, several consecutive remarks can be programmed. Example:

```
N5  M00 L999  LF
N10 G26 X10,52 Z15.305 (MAXIMUM LIMIT VALUES WORKING AREA)
(MANUAL CHANGE POSSIBLE)
```



No remark may be inserted between the address and digits or between a word and its associated parameters.

1.7 Part Programmes

A part programme describes the sequence of a machining process and consists of the part programme itself and any sub-routines and/or cycles which may be contained within it.

The programme store has space for a maximum of 99 part programmes.

Cycles are sub-routines which the machine tool builder enters into a non-volatile area of the memory.

%LF	Programme start when there is only one part programme in the store.
N5 G91 G01 X50 F100 LF	
N10 Z100 LF	
N15 X-30 LF	
N20 Z-10 LF	
N25 M30 LF	

% 1357 LF	Programme start, max. 4 decades (0...9999)
N5 G91 G01 X50 F100 LF	Determination of preparatory function, path
N10 X-30 LF	data, speed, direction etc,
N15 Z-10 LF	
N20 M30 LF	M30 or M02 programme end with return to programme start. In the automatic mode when working directly from the tape reader, M30 initiates tape rewind.

If the programme input is made via the operator panel, the block numbers are automatically generated in multiples of five.A new block number can be entered by clearing the generated block number using the "Clear" button.

1.8 Sub-routines

Repetitive sequences of functions and movements can be entered into the control as sub-routines and called up at any time by part programme or manual data input.

The Nos. L80 to L99 and L900 to L999 are protectable.

Sub-routine definition

The sub-routines are defined :

- under address L by 2 or 3 decades with following nulls
- at choice alone without block numer or together with other functions in first block

The end of the sub-routine is defined :

- by M17 alone in a separate block or in the last block together with other functions (except L address)

For the definition are possible the following versions:

1) Recommended standard version

```
L12300 N5 G00 X.. LF
N10 ...
...
N... G00 G90 X... M17 LF
```

2) Another possible version

```
L12300
N5 G00 X... LF
N10 ...
...
N... G00 G90 X.. LF
N... M17 LF
```

3) Smallest possible sub-routine version 1

```
L12300 N5 G00 G91 X.. LF
N10 M17 LF
```

4) Smallest possible sub-routine version 2.
L12300 LF
N5 G00 G91 X.. M17 LF

5) Curved path machining sub-routine
with fulcrum point calculation without intermediate stop.
(Machine data set for M17 not output to interface).
L12300 N5 G01 XR.. ZR.. LF
N10 M17 LF
Prerequisite: Interface signal "Cycles blocked" is set.

Sub-routine call

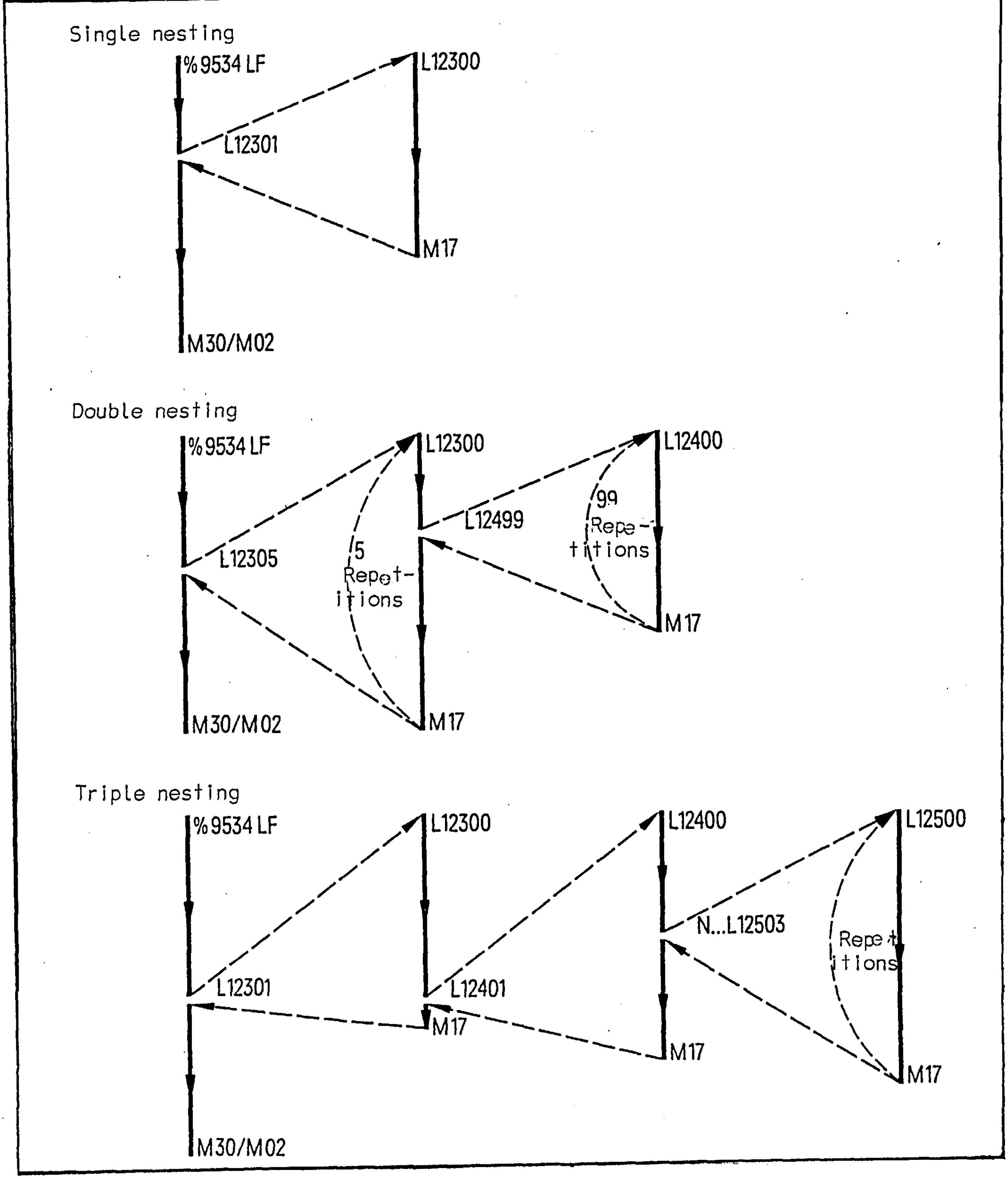
The sub-routine call is the result of a L address in a main programme or sub-routine. Three levels of nesting of sub-routines are possible from the main programme.

L 123 01	Call 2 to 5 decades Number of repeats with 2 decades. No data means one repeat. Number of the sub-routine must be 2-3 decades (01...999).

The block with a sub-routine call may not have M02, M30 or M17 included.
As a result of a sub-routine call with a selected CRC G41/G42, the CRC will be adjusted afterwards, whether or not a path address has been programmed in the first and last sub-routine block (in accordance with para. 8.1.7 "Blocks without path addresses).

1.9 Sub-routine call-up, Sub-routine nesting.

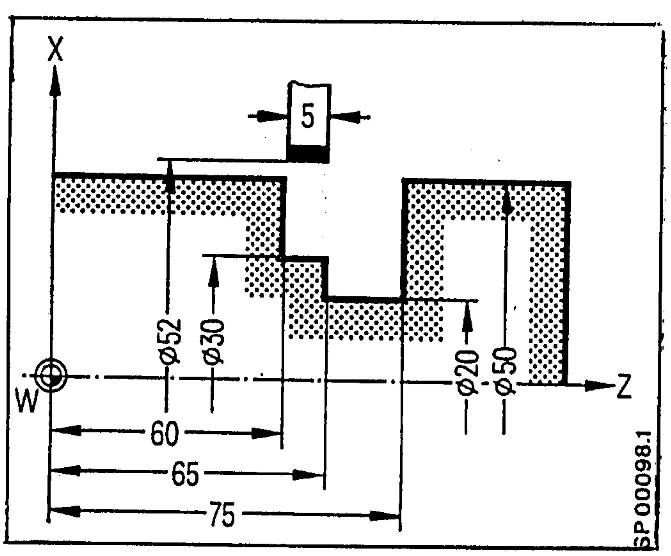
Part programme % 9534 LF	Sub-Routine L 12300	Sub-routine L 12400	Sub-routine L 12500



Sub-Routine Nesting

```
% 4011  LF                          - Part programme 4011
N1 G90 G94 .. S .. T1501 M.. LF
N2 G00 X52. Z60. LF
N3  L 12301  LF                     - Call up sub-routine 123,
                                      single cycle ( L12301)

N90 M30 LF

L12300                              - Sub-routine 123
N1 G91 G01 X-11. LF
N2     G00 X 11. LF
N3 L  12402      LF                 - Call up sub-routine 124,
                                      2 repetitions ( L12402)

M17              LF                 -  End of sub-routine

L12400
N1 G91 G00 Z5.  LF                  -  Sub-routine 124
N2 G01     X-16 LF
N3 G00     X 16.LF
N4 M17          LF                  - End of sub-routine.
```

Programme execution



1.10 Tape formats

SUB ROUTINES | % {SP} LF | L12300 LF | N1....LF | N2 M17 LF

Leader | Rewind stop & block start | Subroutine 123 | End of sub routine

L12400 LF | N1 .... LF | N2(Drilling cycle)... LF | N.. M17 LF

Sub routine 124 | (remark) | End of sub routine

L12500 LF | N.. .... LF | N.. .... LF | N.. M17 LF | N.. M02 or M30 LF

Sub routine 125 | End of sub routine | End of subroutine block

Previous part programm | % {1234} LF | (CARRY OUT MEASUREMENT)

Leader | Part programme 1234 | (remark)

N.. .... LF | N.. .... LF | N200 M02 or M30 LF

Part programme | End of part programme

% TO LF | G92 T01 .... LF | G92 T02 .... LF | M02 or M30 LF

Tool compensations | End of tool compensation block

% ZO LF | G59 N1 .... LF | G59 N12 .... LF | M02 or M30 LF

Zero offset | End of zero offset block

{ } Characters in brackets can be omitted

SP - Subroutine (Sub - Programme)

The sequence in which the above tapes are entered is un-important.

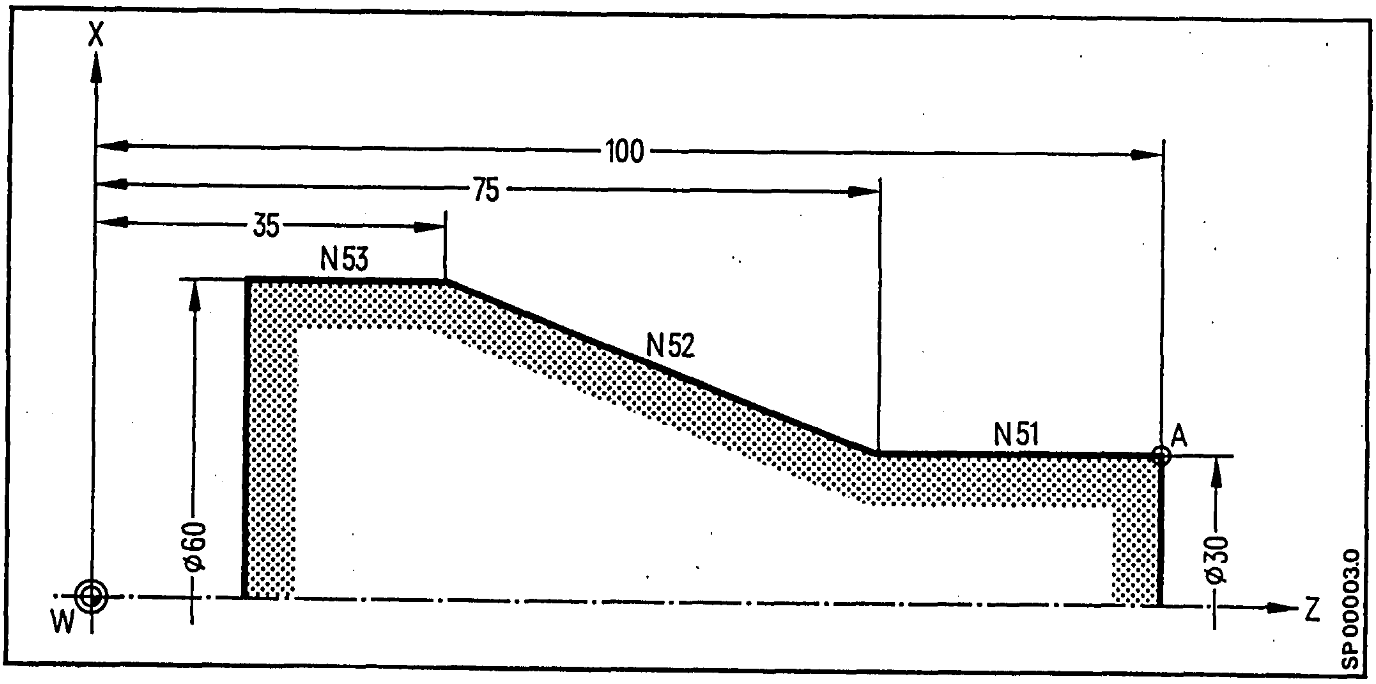
Division of the memory into the area for part programs and the area for sub-routines is automatic.

Tool offsets and zero offsets are entered in the appropriate memory areas using the designations TO (Tool Offset) and ZO (Zero Offset).

2.0 Path Data

2.1 X , Z axis commands

The address for the axis commands X and/or Z determine the axis, which is to be traversed in accordance with the associated digital value. When using absolute data input (G90), the values for the X axis are diameter values or radius values (machine parameter) Using G91 the values always refer to the radius.



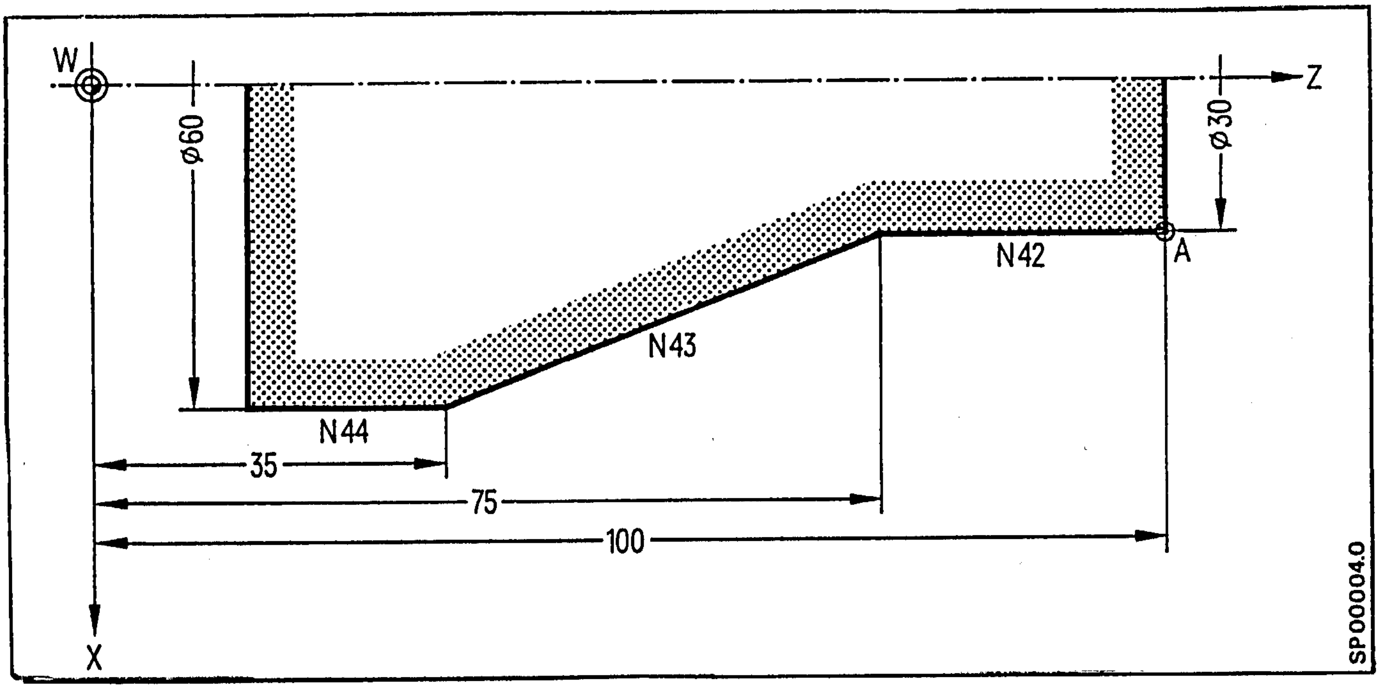
Machining behind the Turning axis.

Absolute data input

```
N51  G01  G90  Z75. LF
N52        X60.Z35. LF       X values denote diameter.
N53 ...             LF
```

Incremental data input

```
N51  G01  G91  Z-25.LF
N52  X15  Z-40.     LF       X values denote radius
N53...              LF
```



Machining in front of the turning axis.

Absolute data input

```
N42  G01  G90 Z75.   LF
N43       X60 Z35    LF        X values denote diameter
N44 ....             LF
```

Incremental data input

```
N42  G01  G91  Z-25  LF
N43  X15       Z-40  LF        X values denote radius
N44 ....             LF
```

2.2 Mirror Image

Using the input signals "Mirror Image X" and/or "Mirror image Z" the following values will be inverted or interchanged in the control:

X Axis

Mirror image of

- Programmed axis command values with sign (inc. G92 displacement)
- Cutter radius compensation G41 — G42 or G42 — G41 (see section 4.3.2)
- Tool length compensation (see section 4.3)
- Position of tool cutter point (see 4.3.2)
- Direction of movement G02 — G03; G03 — G02

There is no mirror image of

- Zero point offset
- Differential resolver offset (DRF)
- Preset offset

Z Axis

Mirror image of

- Values programmed with sign (inc. G92 offset)
- Cutter radius compensation G41 — G42, G42 — G41
- Direction of movement G02 — G03 ; G03 — G02

There is no mirror image of

- Zero point offset
- Tool length compensation
- Position of the tool cutter point
- Differential resolver offset (DRF)
- Preset offset

The mirror image for the X axis is always the axis.

The mirror image for the Z axis is always the workpiece.

The machine manufacturer decides upon the M-function to realize the "mirror" signals.

If there is a change to "mirror image" during a program, L999 must be called up after selecting "mirror image".

3.0 Preparatory functions

The preparatory functions describe the movement of the machine slides, type of interpolation, type of dimensioning, time-related influences and activate certain operating conditions within the control.

The preparatory functions are assigned to the groups G1 to G14 (see programme key).

Only one preparatory function from each of the fourteen groups may be included otherwise only the last value programmed is recognised. Preparatory functions designated by * in the program key, must not be included in a block with another function.

The reset data are only effective after switching on the control, resetting or at the end of the programme. They do not have to be programmed.

Modal preparatory functions can only be overwritten by words from the same group.

3.1 G90/G91: Absolute and incremental data programming

Absolute Data Input G90

When using absolute data input all data are referred to the fixed zero point of the workpiece.
The X values are recognised as diameter and radius (machine parameter).

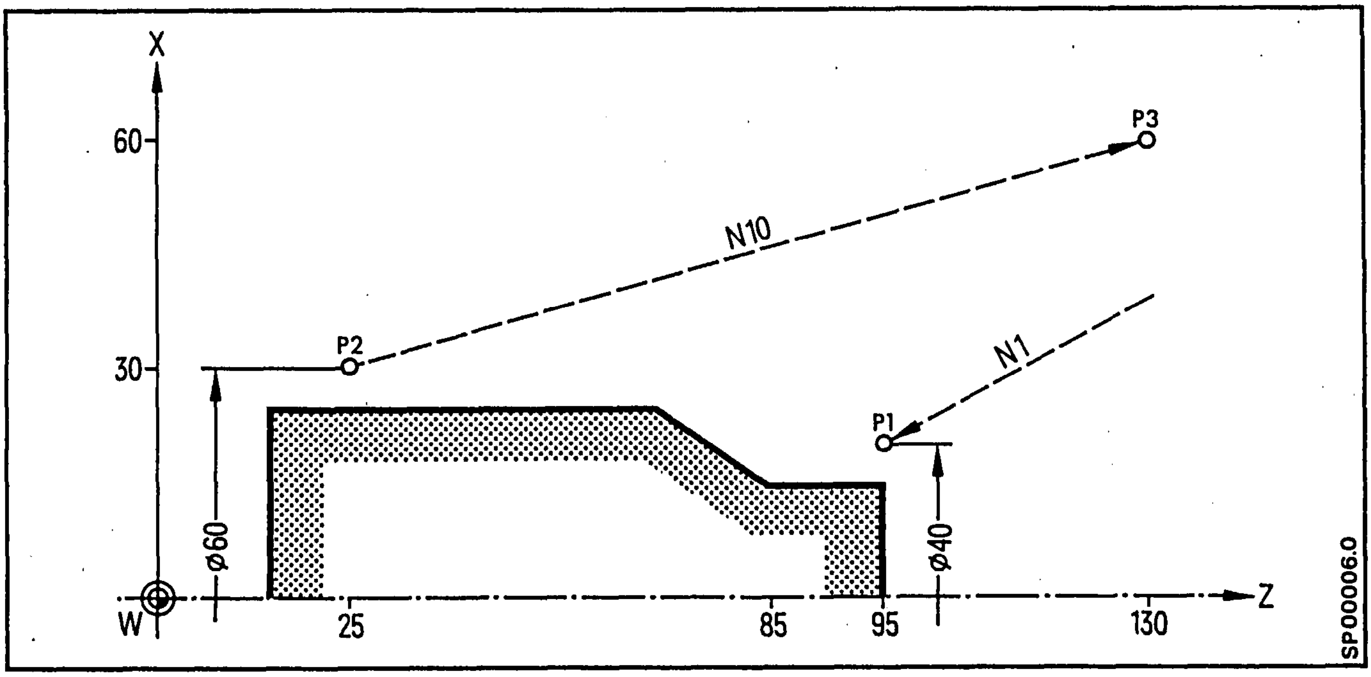
Absolute data input simplifies entry to and exit from a programme.
It may also simplify editing of the geometry of the programme.

Incremental Data Input G91

Incremental position data means that each dimension corresponds to the distance to be traversed. For this reason one talks of incremental measurement or of incremental data input. The numerical value of the axis command indicates the distance to be traversed in order to reach the end position. Incremental data are preferred for use with sub-routines which have to be called up in different operating positions of the machine (recessing, etc.).

Note: Tool offsets are taken into account for both incremental and absolute programming (see also block increment calculation in section 8.2.3).

G90/G91 Absolute / Incremental data programming



Absolute data input G90

.

.

N 1 G90 G00 X40 Z95 .... Tool travels from any position to the point P1
N10 ...

Incremental Data Input G91

.

.

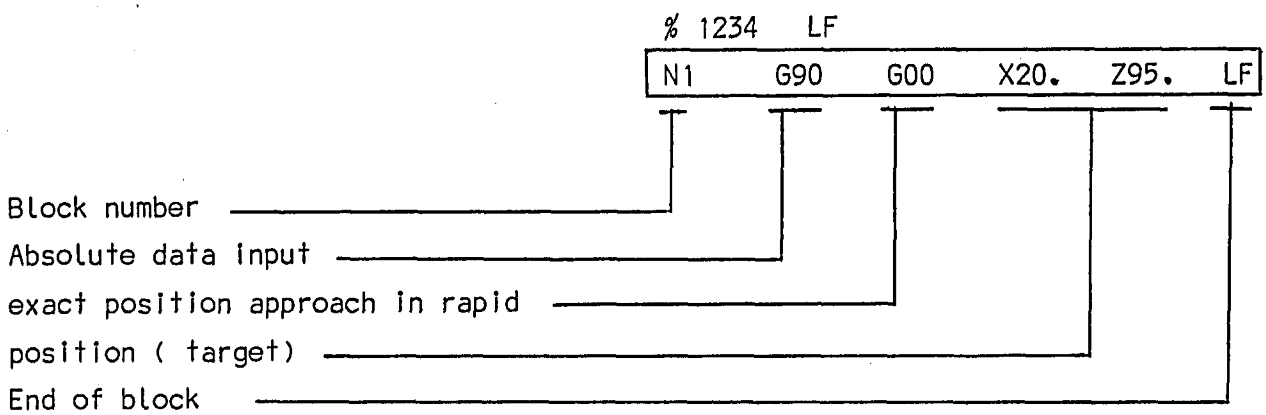
N1 G90
N10 G91 G00 X 30. Z105 .... Tool travels from P_2 to P_3.

3.2 G00 Rapid Traverse

The distance programmed in a block with G00 is traversed at the highest possible rate, i.e. at rapid traverse, in a straight line. At the same time the control monitors the axes in order to prevent them exceeding the maximum permissible rate (machine parameter).

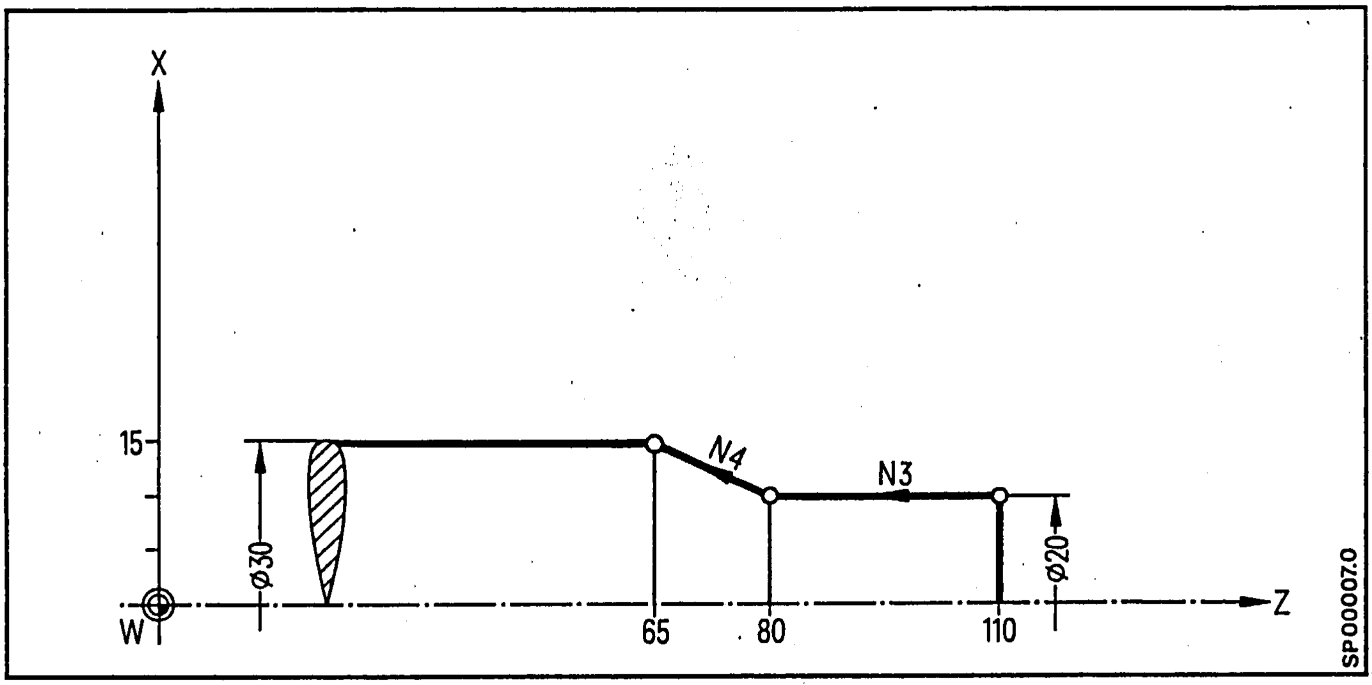
The preparatory function rapid traverse (G00) initiates automatic exact stop (G09). When G00 is programmed, the value of the feedrate programmed under address F remains stored and can be reactivated using e.g., G01.

Example : Position approach using rapid traverse. (see G90/G91).



3.3 G01 Linear Interpolation

The tool traverses at the preset feedrate along a straight line to the required final position.
Paraxial and traverse movements at any angle may be executed.



Absolute data input

N2 ...

N3 G01 G90 Z80 F10 LF

N4 X30 Z65 LF

.

Incremental data input

N2 ...

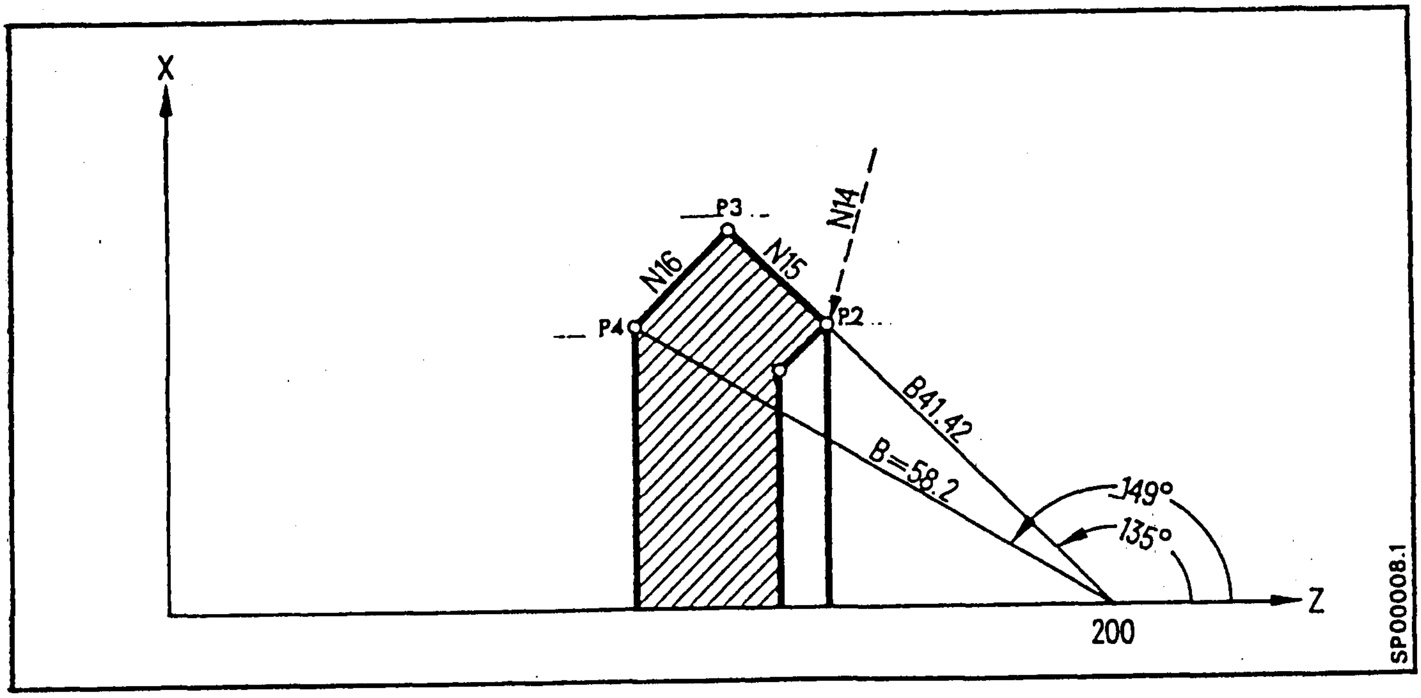
N3 G01 G91 Z-30. F10. LF

N4 X5. Z-15 LF

3.4 G10/G11 Programming in polar co-ordinates (only with Sprint 8T; option)

G10 Linear interpolation rapid traverse

G11 Linear interpolation feedrate (F)



N14	G90	G10	Z200	X0	B41.42	A+135	LF	(P2)	A = Angle
N15		G11			B56.56		LF	(P3)	B = Radius
N16					B58.2	A+149	LF	(P4)	

The angle always refers to the axis programmed first in the block in the positive direction (in this example from +Z to +X in the shortest direction).

The positive direction of the first programmed axis corresponds to an angle of 0°.

The positive direction of the second programmed axis corresponds to an angle of 90°.

The angular value is always absolute and positive.

The centre point is modal until re-programmed. At the end of the programme (M02/M30) the centre point is reset.

When programming in polar co-ordinates for the first time, both centre point co-ordinates must be entered in absolute data.

The incremental data input of the centre point (using G91) always refers back to the last programmed centre point.

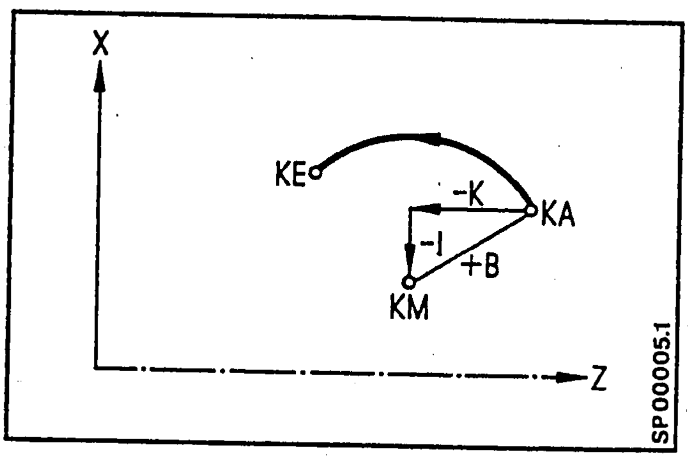
3.5 G02/G03 Circular interpolation

Together with axis commands the interpolation parameters determine the circle or arc. The starting point "KA" of the circle or arc is determined by the previous block and the end point "KE" is fixed by the axis values X and Z. The interpolation parameter determines the circle centre "KM" :

a) Either using the vectors I and K with sign over a range from 0 to 360°. I in X direction, K in Z direction: The sign is determined by the co-ordinate direction from the starting point to the centre point.

b) Or directly using the radius B (only with Sprint 8T ; option)
+B angle less than or equal to 180°
-B angle greater than 180°

Do not program radii if the traverse angle is 0° or 360°.
Thus circles must be programmed using the interpolation parameters I and K.



An interpolation parameter I or K with value 0 need not be programmed (except in contour description, see chapter 6.1).
If the end point co-ordinate has not changed with respect to the circle start, it does not have to be programmed. For circles at least one axis must be programmed (X0 or Z0).

The direction in which the arc is traversed is determined by G02 or G03.

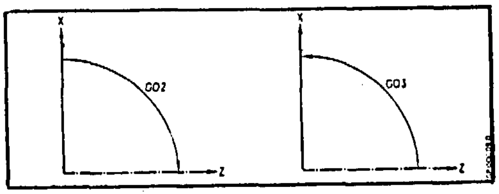
G02/G03 determine the traverse direction of the circle fixed by X,Z,I,K, or R

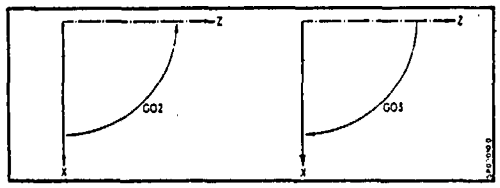
Right hand co-ordinate system (DIN 66025)

G02 in clockwise direction

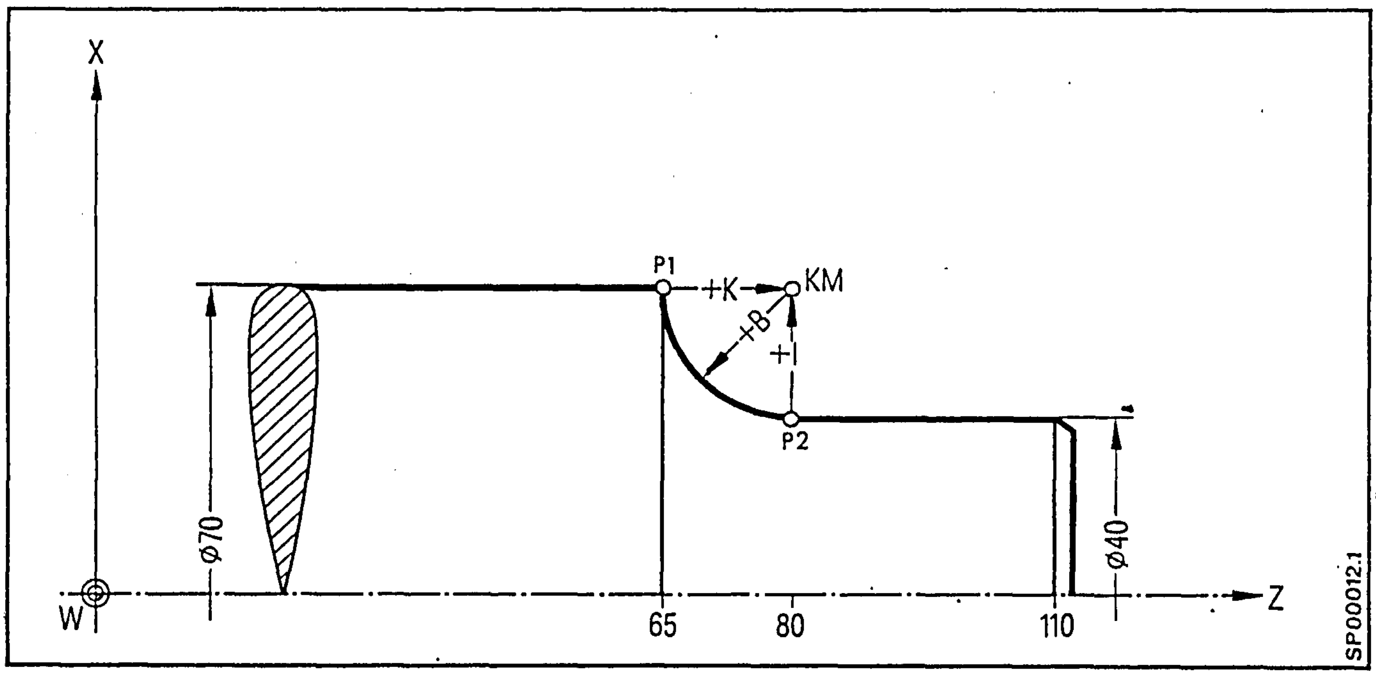
G03 in counter-clockwise direction

Operating area





Behind the turning axis In front of the turning axis



N5 G03 G90 X40 Z80 K15 I 0. LF - Tool traverses from P1 to P2

N10 G02 X70 Z65 K0. I 15 LF Tool traverses from P2 to P1

or

N5 G03 G90 X40 Z80 B+15.LF Tool traverses from P1 to P2

N10 G02 X70 Z65 B+15.LF Tool traverses from P2 to P1

3.6 G33 Thread cutting with constant pitch

Single pass or multi-pass turning, facing and conical threads with constant pitch can be machined using G33.

G33 links the main spindle speed to the feedrate. An encoder generates a no. of pulses per spindle revolution and these are decoded in the control and fed to the feed drives. Thus the feedrate is directly linked with the spindle speed and no feedrate programming is required.

During threadcutting the feedrate override switch, the "Feed off" button, the spindle speed override switch, and the operating mode " Single Block" are all non-operational. However, the feedrate under F remains stored and is re-activated the next time G01, G02, G03 and G11 are programmed.
In order to produce threads requiring several passes the feedstart is initiated by the zero mark of the encoder. This ensures that the threadcutting is always started at the same angular position of the workpiece relative to the tool. All cuts must be carried out at the same spindle speed in order to avoid variations in the pitch.

The direction of rotation and spindle speed must be programmed before the threadcutting instruction in order to allow the spindle to reach its operating speed.
Further threading blocks G33 immediately after each other (thread to thread) are possible. To ensure safe hand over, so that all spindle pulses are calculated, adhere to the smallest block length "Smin" having the following relationship:

$$S_{min}\ (mm) = 1{,}7 \cdot 10^{-5} \cdot n\left(\frac{rev}{min}\right) \cdot L\left(\frac{mm}{rev}\right) \cdot t_A\ (ms)$$

n = Spindle speed, L = thread pitch,
t_A = 12 ms or length of auxiliary function output.

The thread length is entered using the path addresses X and / or Z taking into account the acceleration distance and overrun distance in which the feedrate is increased or decreased.
The values may be entered as absolute or incremental data.

The thread pitch is entered using addresses I and K.

For longitudinal threads the pitch is entered using K, for facing threads using I and for taper threads using both I and K. I and K values are always entered as incremental data without sign.

Thread pitch : 0.001 mm to 2000mm , max to 10000
Using M37 the programmed thread pitch can be reduced by a factor 1 : 100.

Right or left hand threads are programmed using the spindle rotation direction M03,M04.

3.6.1 Thread cutting with constant pitch on variable angle tapers

The angle of the taper, on which the thread is to be cut, can be changed in steps. In the case of turned threads this permits a smooth run out. An example for programming such a taper is given in section 3.6.8.

3.6.2 Multi-start threads

Thread cutting always starts at the synchronised point by the marker on the encoder. The feed is not enabled until this signal is received. An offset can be programmed for the position of the starting point for the thread. It is thus possible to cut threads using the right or left flank of the cutter as well as varying the offset to cut single, double or multi-start threads. A single pass for a multi start thread is programmed in the same way as the single start thread.
After completing the first pass, the starting point is offset by h and the next pass initiated.

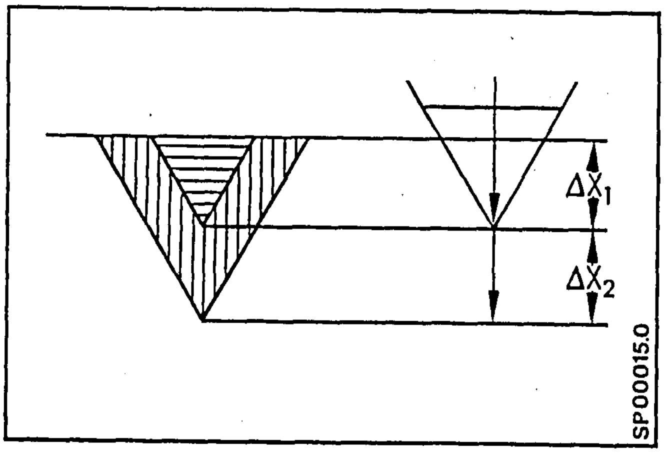
$$\Delta h = \frac{\text{Thread pitch}}{\text{Number of passes}}$$

See also example "multi-start threads" (3.6.7)

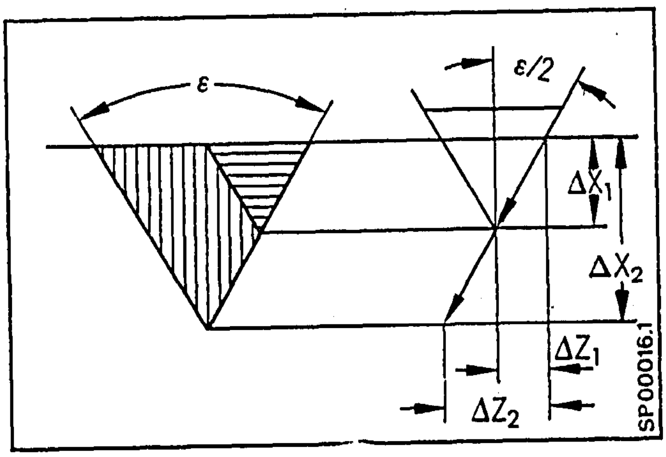
The individual passes must be executed at the same spindle speed in order to avoid different thread pitch errors.

3.6.3 Infeed directions

The tool can approach at right angles to the cutting direction or along the flank.



Infeed direction " at right angles to the cutting direction"

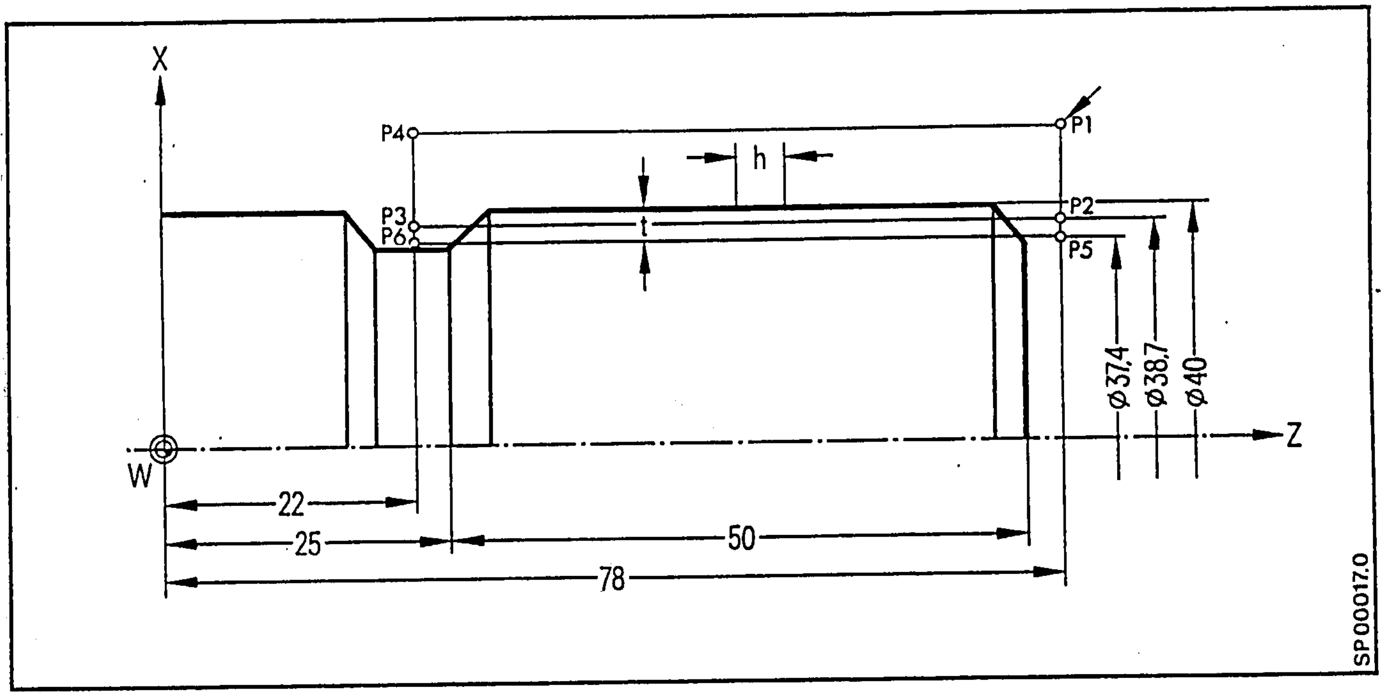


$$\Delta Z = \Delta X \cdot \tan \varepsilon/2$$

"Infeed along flank.

3.6.4 Single start, cylindrical thread

Pitch h = 2 mm ; thread depth t = 1.3 mm
Radial infeed direction



Absolute data input

```
N20  G90             S...      LF
N21  G00 X 46        Z 78.     LF      (P1)
N22      X 38.7                LF      (P2)
N23  G33             Z 22  K.2 LF      (P3)
N24  G00 X 46.                 LF      (P4)
N25                  Z 78      LF      (P1)
N26      X 37.4                LF      (P5)
N27  G33             Z22   K.2 LF      (P6)
N28  G00 X46.                  LF      (P4)
```

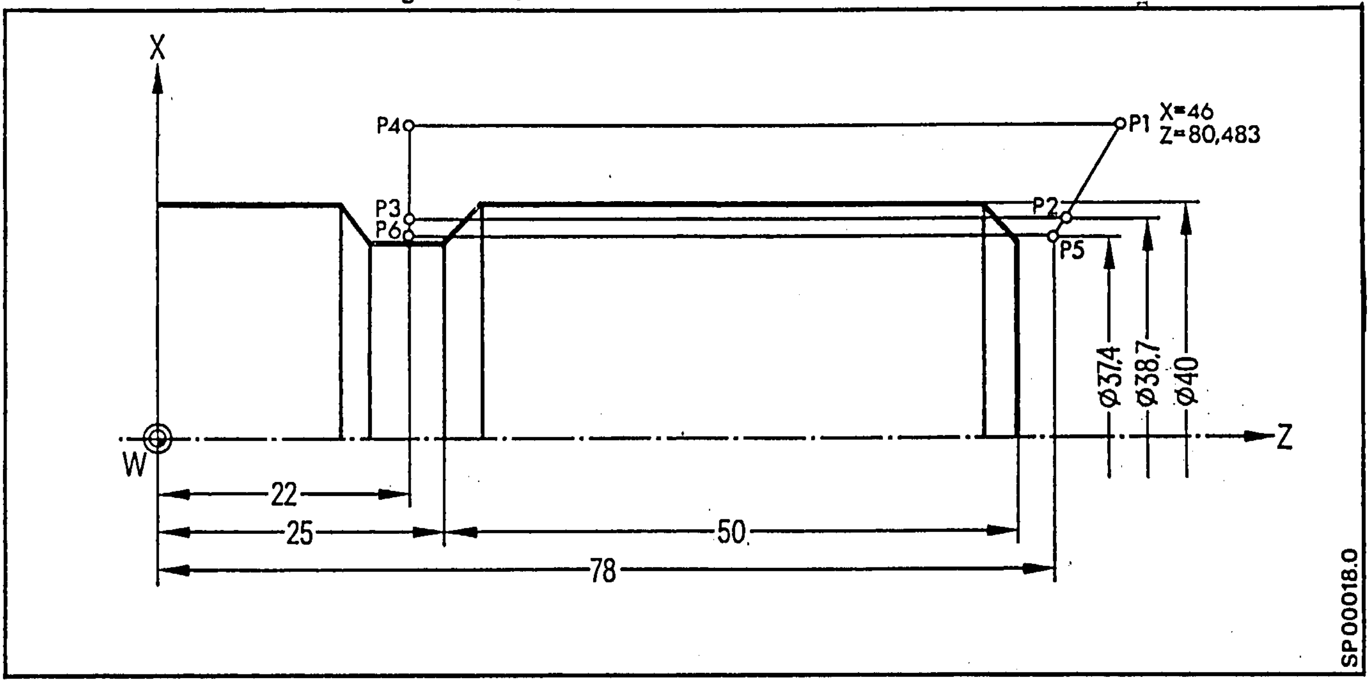
Incremental data input

```
N20 G91          S...              LF
N21 G00 X-...     Z-...            LF       (P1)
N22     X-3.65                     LF       (P2)
N23 G33          Z-56.       K 2.  LF       (P3)
N24 G00 X 3.65                     LF       (P4)
N25               Z 56.            LF       (P1)
N26      X-4.3                     LF       (P5)
N27 G33           Z-56 K 2.    LF           (P6)
N28 G00 X 4.3                      LF.      (P4)
```

3.6.5 Single start longitudinal thread with constant pitch

Pitch h = 2 mm; Thread depth t = 1.3mm ; infeed angle $\mathcal{E}$ = 30°

Infeed direction along flank.



ΔX = (46-38,7) /2 = 3.65	ΔX = (3.65 + 1.3)/2 = 4.3
ΔZ = 3.65 .tan $\mathcal{E}$ /2	ΔZ = 4,3 . tan $\mathcal{E}$ /2
ΔZ = 3.65 .tan 30°	ΔZ = 4,3 . tan 30°
ΔZ = 3.65 .0,5774	ΔZ = 4.3 . 0,5774
ΔZ = <u>2.108</u>	ΔZ = <u>2,483</u>

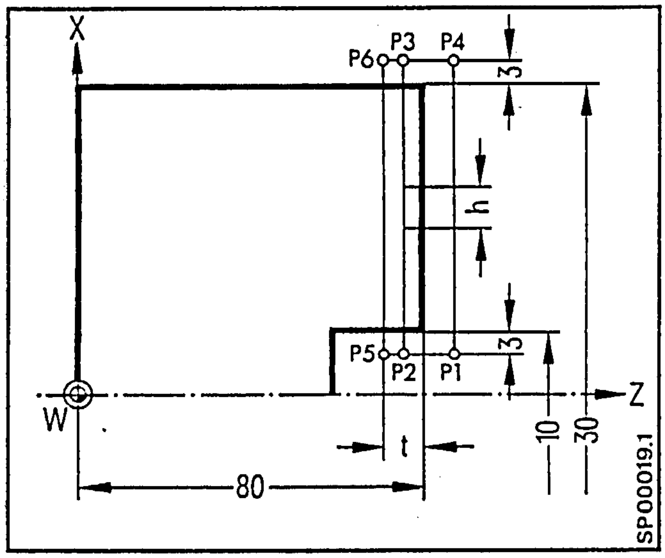
Absolute data input

N33	G90		S ...		LF	
N34	G00	X46.	Z 80.483		LF	(P1)
N35		X38.7	Z 78.375		LF	(P2)
N36	G33		Z 22.	K2	LF	(P3)
N37	G00	X46.			LF	(P4)
N38			Z80.483		LF	(P1)
N39		X37.4	Z78.		LF	(P5)
N40	G33		Z22	K2	LF	(P6)
N41	G00	X46			LF	(P4)

3.6.6 Single start facing thread with constant pitch

Pitch h = 2mm ; thread depth t = 1.3mm

Infeed direction at right angles to cutting direction.



Absolute data input

N41	G90		S...		
N42	G00	X 4.	Z 82		LF (P1)
N43			Z 79.35		LF (P2)
N44	G33	X 36.		12	LF (P3)
N45	G00		Z82.		LF (P4)
N46		X 4			LF (P1)
N47			Z 78.7		LF (P5)
N48	G33	X 36		12	LF (P6)
N49	G00		Z82.		LF (P4)
N50					

Incremental data input

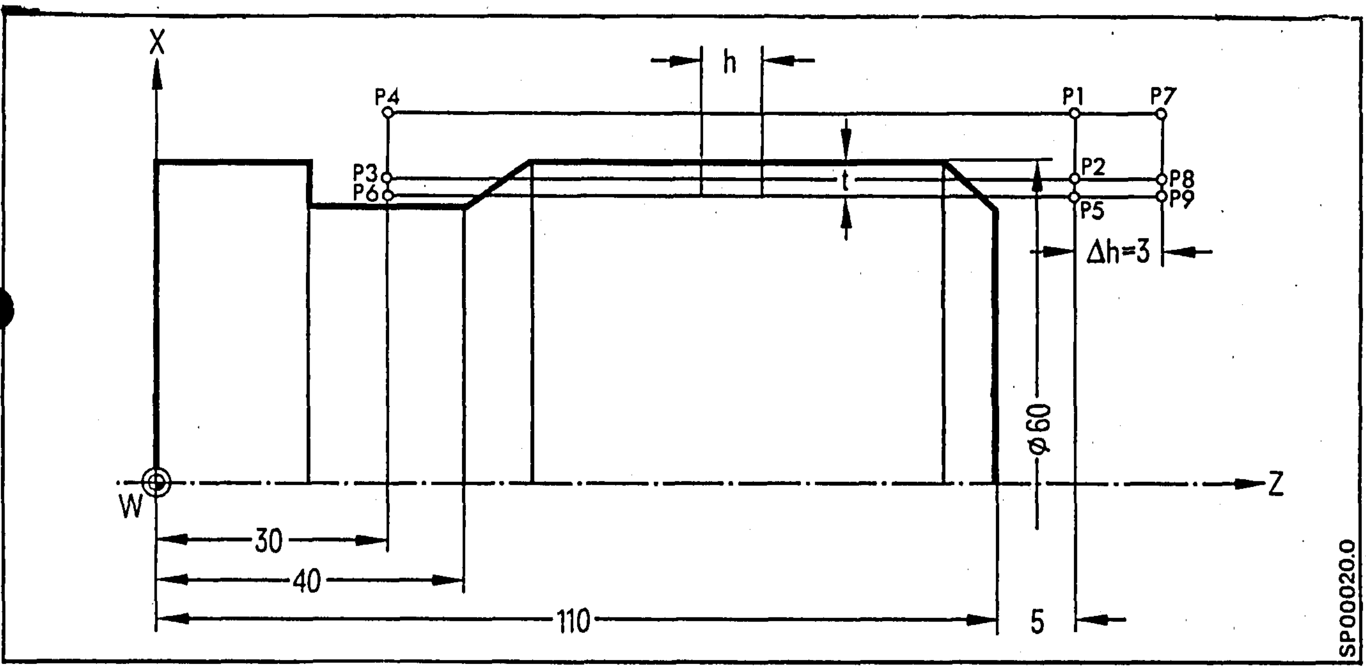
N41	G91				LF
N42	G00	X-...	Z-...		LF (P1)
N43			Z-2.65		LF (P2)
N44	G33	X16.		12	LF (P3)
N45	G00		Z 2.65		LF (P4)
N46		X16			LF (P1)
N47			Z 3.3		LF (P5)
N48	G33	X16		12	LF (P6)
N49	G00		Z 3.3		LF (P4)
N50					

3.6.7 Multi-start thread with constant pitch

Radial infeed direction ; pitch h = 6mm, thread depth t = 3.9 mm ; double start.

In the example, each thread start is machined in two passes. After the first start has been machined the second start is machined with the starting point displaced Δ h

Δ h = thread pitch / number of starts = 6/2 = 3mm



Absolute data input

```
N35 G90   S ...
N36 G00   X 66.       Z 115.          LF  (P1)
N37       X 56.                       LF  (P2)
N38 G33               Z 30.      K 6. LF  (P3)
N39 G00   X 66.                       LF  (P4)
N40                   Z 115.          LF  (P1)
N41       X 52.2                      LF  (P5)
N42 G33               Z 30.      K 6. LF  (P6)
N43 G00   X 66.                       LF  (P4)
N44                   Z 118           LF  (P7)
N45       X 56.                       LF  (P8)
N46 G33               Z 30.      K 6. LF  (P3)
N47 G00   X 66.                       LF  (P4)
N48                   Z 118.          LF  (P7)
N49       X 52.2                      LF  (P9)
N50 G33               Z 30.      K 6. LF  (P6)
N51 G00   X 66.                       LF  (P4)
```

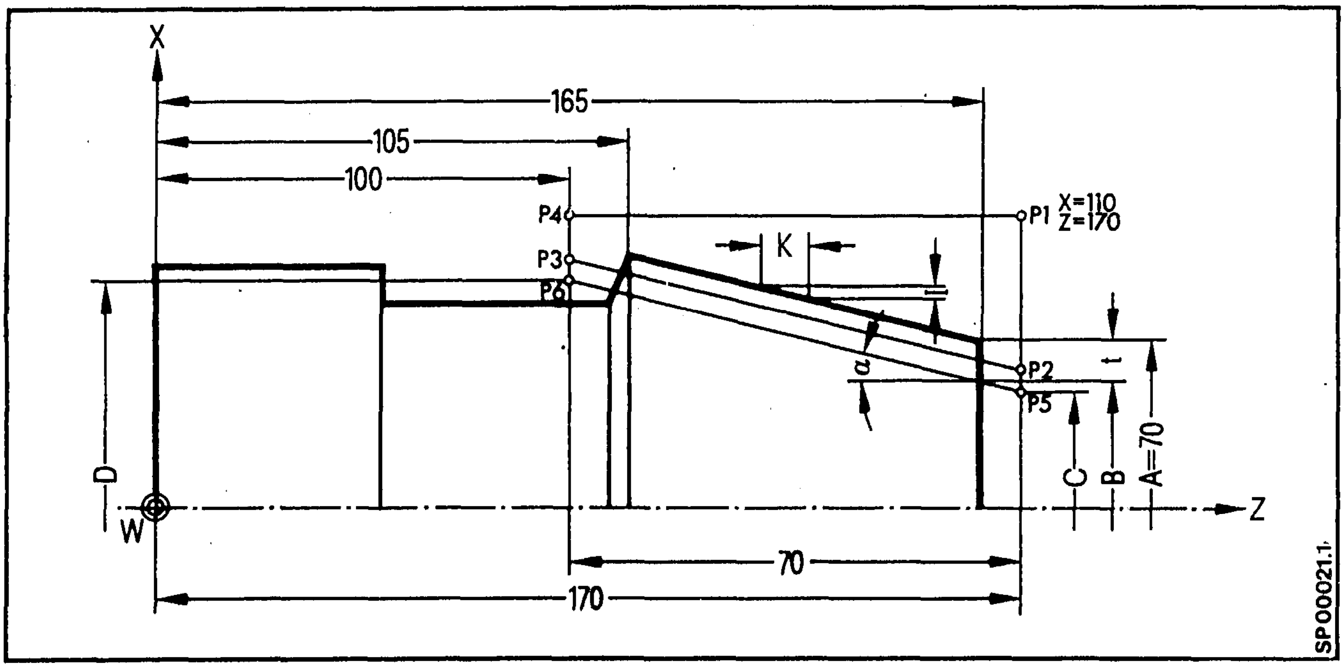
3.6.8 Single start taper thread with constant pitch

Radial infeed direction

Pitch h = 5mm thread depth t = 1,73mm $\alpha = 15^\circ$

Both end point coordinates must be written.

The pitch for the leading axis only must be programmed (assignment: X corresponds to I and Z corresponds to K). The leading axis is the one with the largest travel. If both are of equal length, the X axis is the leading axis. With angles (α) smaller than 45° the pitch will be programmed using K. With angles greater than or equal to 45° the pitch will be programmed using I.



Calculation of the start and end point co-ordinates for the thread

A = 70

B = A - 1,73 . 2

B = 70 - 3,46

B = 66,54 mm

C = B - (5 . tan α) . 2

C = 66,54 - (5 . tanα) . 2

C = 63,86 mm

D = C + (70 . tanα) . 2

D = C + (70 . 0,2679) . 2

D = 63,86 + 37,506

D = 101,366 mm

Absolute data input

```
N31 G90    S ...                                LF
N32 G00    X 110          Z 170.                LF    (P1)
N33        X 65.86                              LF    (P2)
N34 G33    X 103.366      Z 100.       K 5.     LF    (P3)
N35 G00    X 110.                               LF    (P4)
N36                       Z 170.                LF    (P1)
N37        X 63.86                              LF    (P5)
N38 G33    X 101.366      Z 100.       K 5.     LF    (P6)
N39 G00    X 110.                               LF    (P4)
```

Calculation of the points P2 and P3

X(P2) = C + 2 mm
X(P2) = 63.86 mm + 2 mm
X(P2) = 65.86 mm

X(P3) = D + 2 mm
X(P3) = 101.366 mm + 2 mm
X(P3) = 103.366 mm

3.7 G 34 thread cutting (increasing pitch)

The thread pitch increases per rev by the value programmed under F up to the maximum possible pitch.

e.g.,

N..	G34	G 90	Z17.	K 2.	F0.1	LF

Starting pitch 2mm —— (K 2.)
Pitch change + 0.1mm per rev —— (F0.1)
i.e., after five turns the thread pitch is;
2mm + (5 x 0.1mm) = 2.5mm

3.8 G35 thread cutting (decreasing pitch)

The thread pitch per rev decreases by the value programmed under F to the minimum possible pitch.

N..	G35	G90	Z17.	K200.	F0.5	LF

Starting pitch 200 mm —— (K200.)
Pitch change 0.5mm per turn —— (F0.5)
i.e., after ten turns the thread pitch is ;
200 mm - (10 x 0.5mm) = 195 mm

The use of M37 is not permitted with G34 and G35.

The examples for G34/G35 correspond to those given for G33. The thread cutting blocks must be programmed to correspond to decreasing or increasing thread. The maximum pitch change is 16 mm.
The value is calculated by the known start and end pitch as follows:

$$F = \left| \frac{\text{start pitch}^2 - \text{end pitch}^2}{2 \cdot \text{thread pitch}} \right|$$

The value is used without sign.

3.9 Smoothing and feedrate ramp time for threadcutting

For thread cutting purposes a feedrate ramp time can be programmed for synchronisation with the main spindle. The programmed value G92 T.. defines the smoothing time and causes the spindle speed during this value to be averaged. The ramp time is determined by the available approach distance. The shorter the available approach distance, the shorter must be the ramp time. The longer the available approach distance, the longer the ramp time recommended to spare the machine. The programmed time simultaneously generates the smoothing time. The actual spindle speed will be averaged over this value to give a continuous feedrate limit.

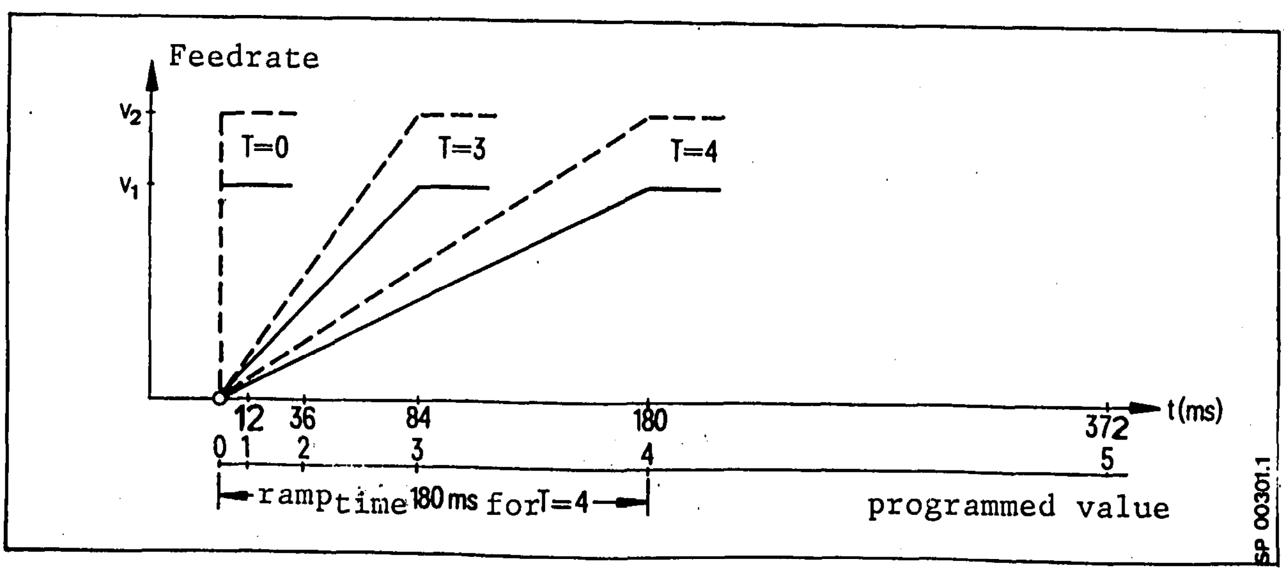
The ramp time is programmed in its own block or entered by the operator as follows:

N.. G92 T. LF

six values are available:

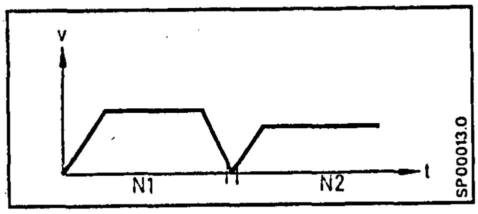
Programme value with G92 T. LF	0	1	2	3	4	5
Ramp time to thread cutting feedrate (ms)	0	12	36	84	180	372

For normal operation T = 3 = 84 ms is recommended.



3.10 G09 Exact stop (non modal)
G60 Exact stop (modal)

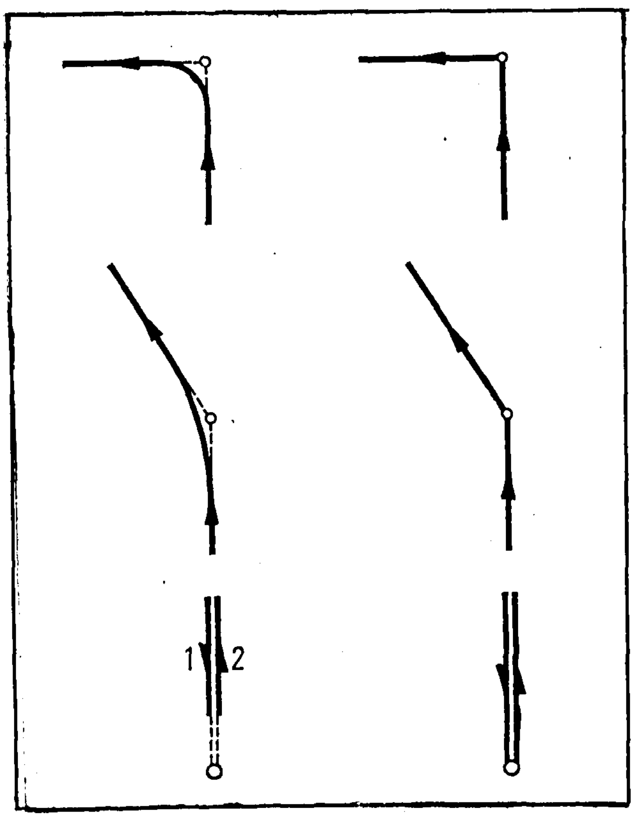
During the function G09 or G60 it is possible to traverse exactly to a target



position (within the exact stop limit). In this case the feedrate and following error are reduced to zero.

The function G09 or G60 can be used, e.g., when sharp corners are to be machined or for cutting grooves or reversing direction. In blocks with G00 it is not necessary to write G09 since this is contained within G00.
G09 is effective in blocks non-modal
G60 is modal and is cancelled using G64 (a continuous path operation).
Examples of direction changes with and without speed reduction.

Without G09/G60 With G09/G60



3.11 G63 Tapping using compensated tap holder

The preparatory function G63 is programmed for tapping operations using a compensated tap holder. There is no functional relationship between spindle speed and feedrate.

Spindle speed is programmed using address S and a matching feedrate using address F. The longitudinal compensating holder must be able to absorb the tolerance between feedrate and spindle speed as well as the spindle deceleration after reaching position.

With G63 the feedrate override switch is non-operational. Depending upon the design of the interface control the spindle is also stopped "Feed Hold". The spindle speed override switch is effective.

It is only possible to use G63 in blocks with linear interpolation G01.
G64 is used to cancel G63.

3.12 G64 Continuous path operation

The preparatory function G64 is used in order to produce continuous transitions from block to block thus avoiding relief cuts. Moreover, transitions involving alterations of the tangents can be smoothed (see fig. without G09/ G60).
G64 is modal (reset state) and cancels G60.

3.13 G04 Dwell Time

The dwell time is normally programmed using address X. It can also be entered using address F.

for X between 1 ms and 99999 999 ms
for F between 1 ms and 99 999 ms

With the exception of G04 no further preparatory functions may be written in a block containing a programme dwell.

e..g.,

N..	G04	X11.5	LF

Dwell 11.5 s (→ X11.5)
always without sign

If required a sequence of blocks with dwell times may be written.

Dwells may be required for relief cuts or perhaps during changes in speed and machine switching functions (steadies, tail stock etc.,).

G04 is non modal.

3.14 G70/G71

G70 Inch input system

G71 Metric input system

The reset state is determined by machine parameter during commissioning. A change of input system during operation of a part programme is not possible. Any change must be programmed in the first programme block, i.e. the change must be inserted before the comment. The range of values for the individual input systems is given in chapter 8.2. The display refers to the respectively valid input system (see Operating Instructions page 2-4).

When changing from G70, G71 , it is the responsibility of the operator or programmer to ensure that the appropriate user data (see Operating Insutructions pages 4-17 and 4-18) have previously been set in the appropriate input system.

3.15 G25/G26 Programmable working area limit

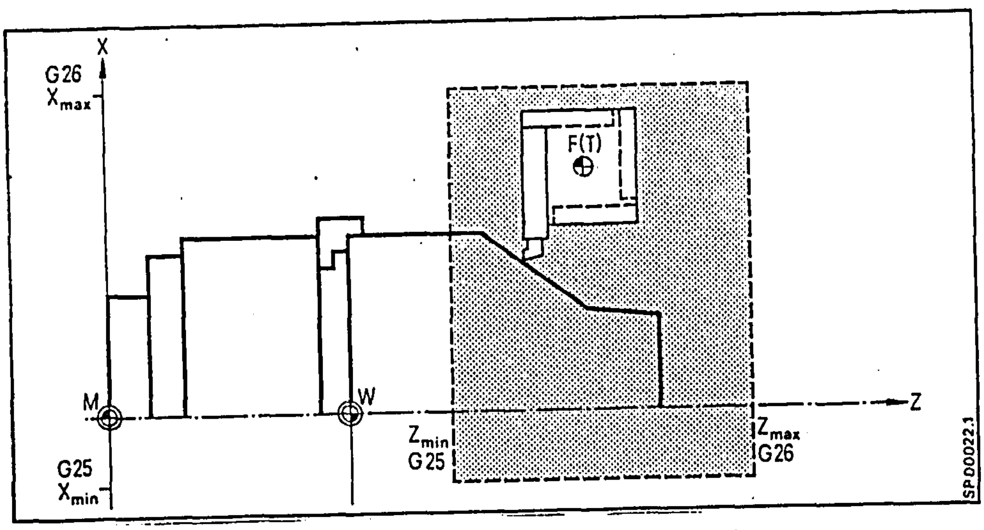
The programmable safe working area gives machine protection against faulty programming or operation. On reaching the limit of the working area the path data input is interrupted (programme stop and alarm). The existing following error is processed.

The programmable working area is only effective in the automatic mode. The effect is that of the software limit switch (see interface description 2.9.7) . The working area limits refer to the machine zero point. The value in the X direction always refers to a radius value.

N... G26 X.. Z.. LF

G25 minimum Work area
G26 maximum limitation

Co-ordinate values for the limitation



The point F (slide reference point) may operate within the shaded area.

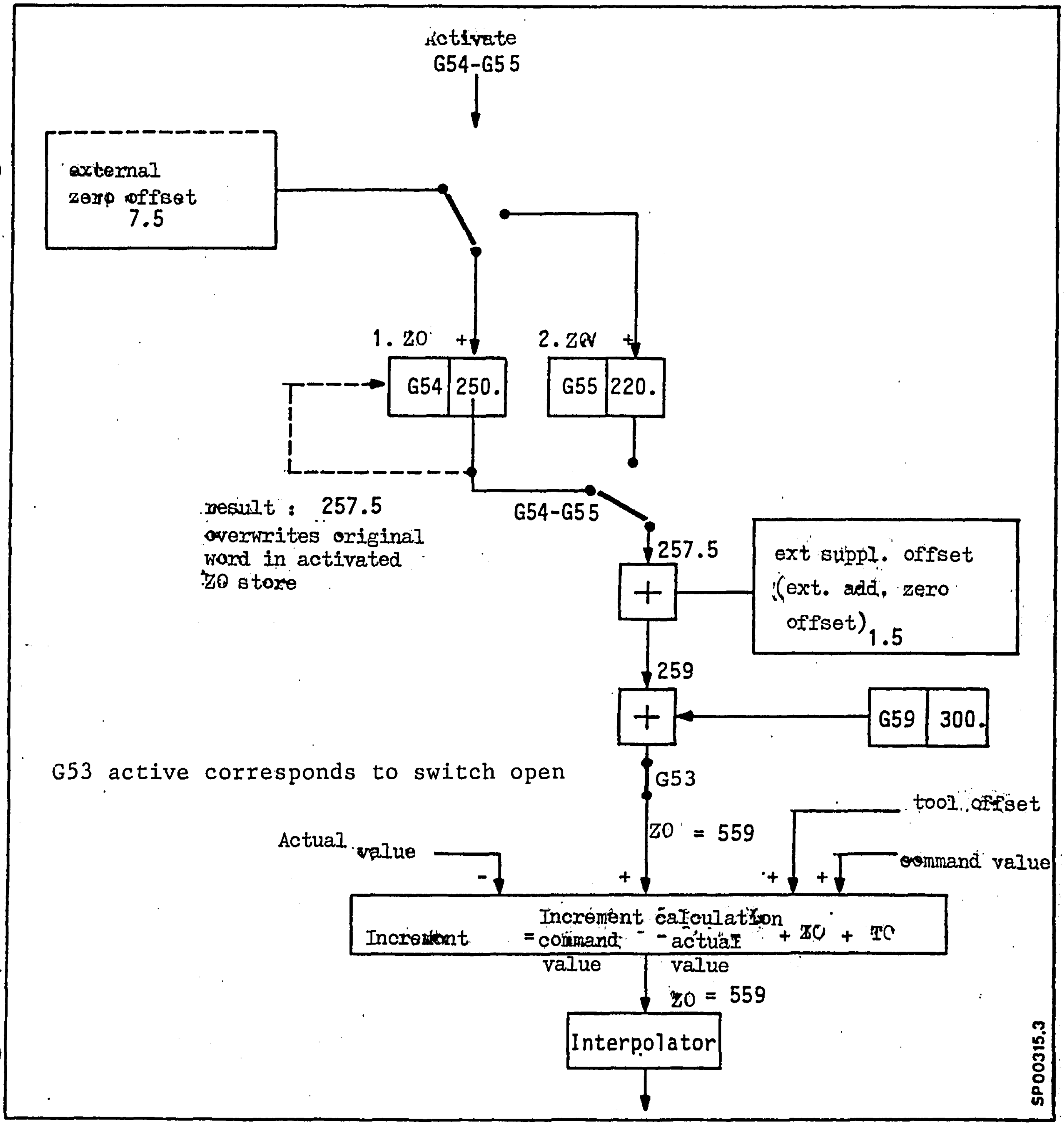
3.17 Zero point offsets (ZO)

ZO = set. ZO (G54-55) + add. ZO (G59) + ext. ZO(PC) + add. ZO

The zero point offset is the difference between the workpiece zero point (to which the measurements are related) and the machine zero point.

Note: When CRC is applied zero point offset cannot be changed.

Example of zero offset in one axis with Sprint 8T, 8T.



3.17.1 G54/55 Adjustable zero point offset

Values for the zero point offset for each axis can be entered into the control manually via the operator's panel, through user program or through tape.

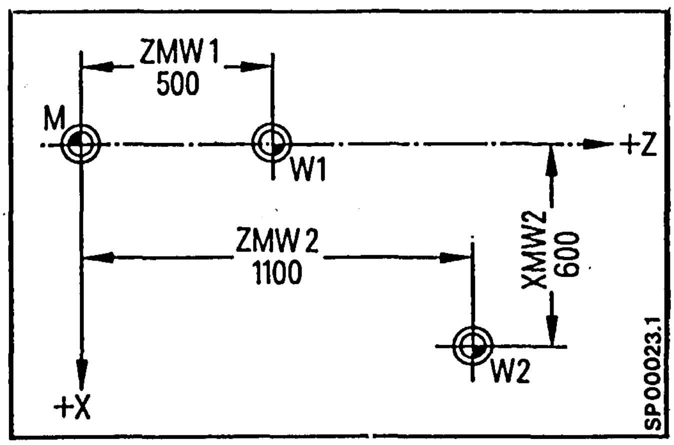
Absolute data blocks (G90) are used to calculate the final block point when the associated axis is programmed. With incremental data blocks (G91) any change in zero point offsets is taken into account.

Example:

Change from G54 to G55 in an incremental data block. The resulting difference between ZO (G55) and ZO (G54) is included in the calculation (see block increment calculation chapter 8.2).

Two adjustable zero point offsets per axes (G54/55) can be selected.

When a zero point offset is included in the calculation, the external zero point offset originating in the interface control for the corresponding axis is also taken into account (external ZO plus supplementary offset).



G54: Input values $XMW_1 = 0$

$ZMW_1 = 500$

G55: Input values $XMW_2 = 600$

$ZMW_2 = 1100$

Selecting the settable zero point offset using G54/G55

```
.
N150  G54  ....  LF - selection of zero point offset 1
.
.
N180  G55  .... LF  - selection of zero point offset 2
```

Entering the adjustable zero point offset via extra tape

```
% ZO LF        (zero offset)
N5  G59  N1  X... Z... LF    - 1st ZO per axis  - selection by G54
N10 G59 N2   X... Z... LF    - 2nd ZO per axis  - selection by G55
N15 M02 or M30         LF
```

3.17.2

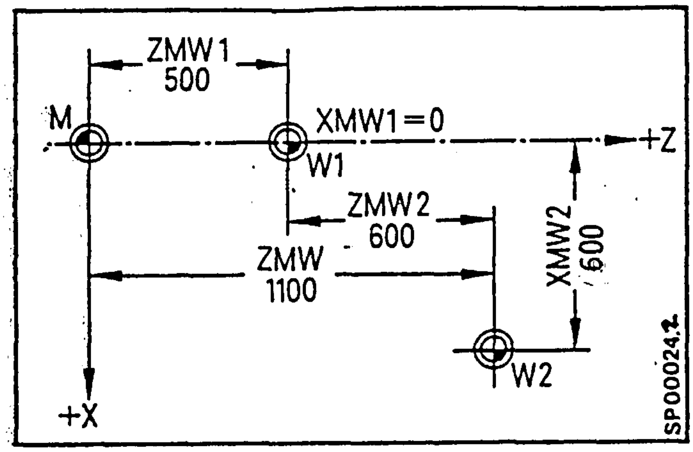
The settable zero point offset can also be loaded in the user programme.

```
% 1234 LF
.
N5  ......       LF
N10 G59 N1  X... Z... LF    - 1st ZO per axis  -  selection by G54
N15 G59 N2  X... Z... LF    - 2nd ZO per axis  -  selection by G55
N20 .....
```

In the above examples, the second addresses N1 or N2 designate the zero point offset 1 and the zero point offset 2. In both these cases the address N is not a block number. The programming sequence within a block must be maintained.

3.17.3 G59 Programmable additive zero point offset

G59 together with addresses X and Z can be used to programme a additional zero point offset. These programmed values are then added during the calculations to the values of the adjustable zero point offset.



Adjustable zero point offset :

Input values XMW_1 = 0
ZMW_1 = 500

Programmable additive zero point offset:
Input values XMW_2 = 600
ZMW_2 = 600

Total zero point offset:

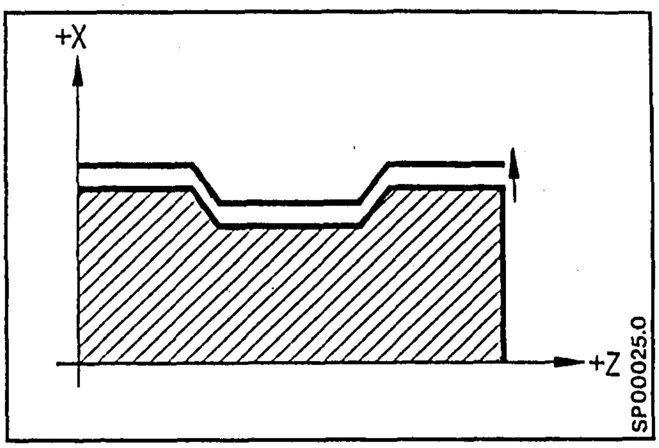
XMW = 600
ZMW = 1100

In a block with G59 X.. Z.. no other information may be programmed.

Application example

The contour has been programmed exclusively in absolute data. In order to achieve a finishing depth cut the entire contour can be offset in the co-ordinate direction X using programmable zero offsets (active).

Select:	N .. G59 X ... LF
Cancel:	N .. G59 X 0 . LF



Programmable additive zero point offset only in X

Following end of programme M02, M30 or programme interrupt the values of the programmed zero point offsets for X and Z are automatically cleared since they will be reset at the programme start or following programme restart.

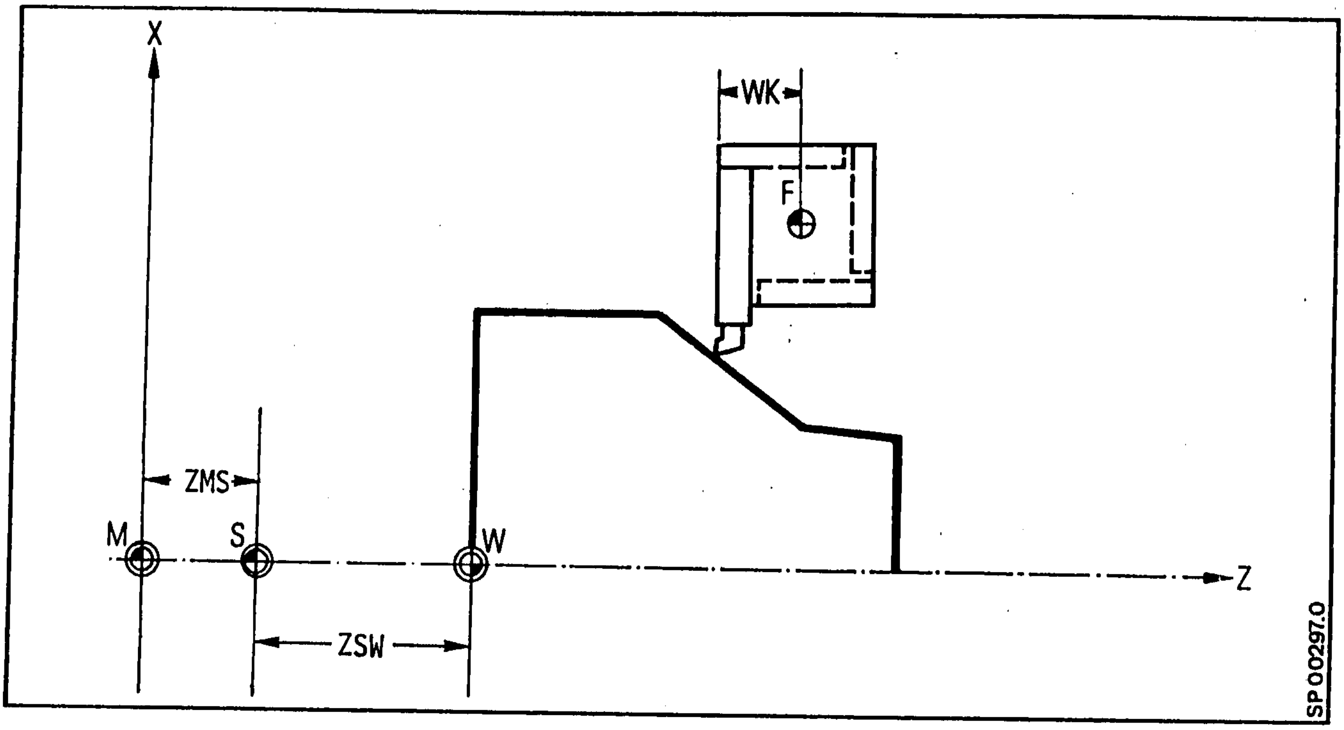
3.18 G53 Inhibit zero point offset

By machine data selection G53 has two different effects:

Machine data N424 bit 2=1 Reference to machine zero point	Machine data N424 bit 2=0 Reference to control zero point
Blockwise suppression of: Settable ZO (G54-G55) Programmable additive ZO (G59) External ZO External additive ZO PRESET-offset G92-offset DRF-offset* (Handwheel shift)	Blockwise suppression of: Settable ZO (G54-G55) Programmable additive ZO (G59) External ZO External additive ZO
Remaining effective: Selected tool offset	Remaining effective: Selected tool offset G92 offset PRESET offset DRF offset (Handwheel shift)

* After programming G53 the remaining DRF-offset is ineffective until reset or end of programme.

Reference point for G53



M = Machine zero point

S = Control zero point

ZMS = Sum of the offsets through DRF, PRESET, G92

ZSW = Sum of:
- settable ZO (G54-G55)
- programmable additive ZO (G59)
- external ZO
- external additive ZO

WK = Sum of all the tool offsets Z axis

F = Slide reference point

The actual value display is related to the control zero point.

Example 1:

Reference to machine zero point	Reference to control zero point
N1232 T1300 Cancel TO N1234 G53 X.. Z.. Travel to the zero point in the machine actual value system	N1232 T1300 Cancel TO N1234 G53 X.. Z.. Travel to the zero point in the control actual value system.
	As above but with cancellation of G92-offsets.
	N1232 T1300 Cancel TO N1233 G92 Cancel G92-offsets N1234 G53 X..Z.. Travel to the zero point in the control actual value system (PRESET/DRF)

The position value G53 X.. is effective in diameter measurements when the machine data "diameter programming" is set.
It is wise to cancel the TO if the need arises, otherwise by programming G53 X.. or G53 Z.. both axes travel to remove the zero point offsets.

Example 2:

Set actual value after reference to machine zero point	Set actual value after reference to control zero point
N1232 T1300	N1232 T1300
N1234 G53 X... Z...	N1233 G92
N1235 G53 G92 X... Z...	N1234 G53 X... Z...
	N1235 G53 G92 X... Z...

The block G53 G92 is compulsory.
Depending on machine data "Diameter programming X axis with G92", G53 G92 X... programmed value will relate to diameter or radius.

3.19 G92 Setting actual value store

The function G92 should only be used for special applications. For normal applications it is recommended to use the settable zero point offset G54/55, the programmable zero point offset G59 and the tool length preset T... (separately adjustable from tool wear). No additional character may be written in a block with G92 X... Z... (Exception: setting of actual value after reference to machine zero point (3.18) G53 G92 X... Z...). Exception: If G02/G03 is still effective in the 1st G group, G00 or G01 must be written before G92.

Without G92 the control zero point (S) and machine zero point (M) coincide. The control zero point is the reference point for all internal control calculations. Using G92 X... Z... the control zero point can be displaced with reference to the machine zero point. This function is particularly advantageous when no program interrupt and restart within the program is anticipated, e.g., machining of batch components with short program cycle times.

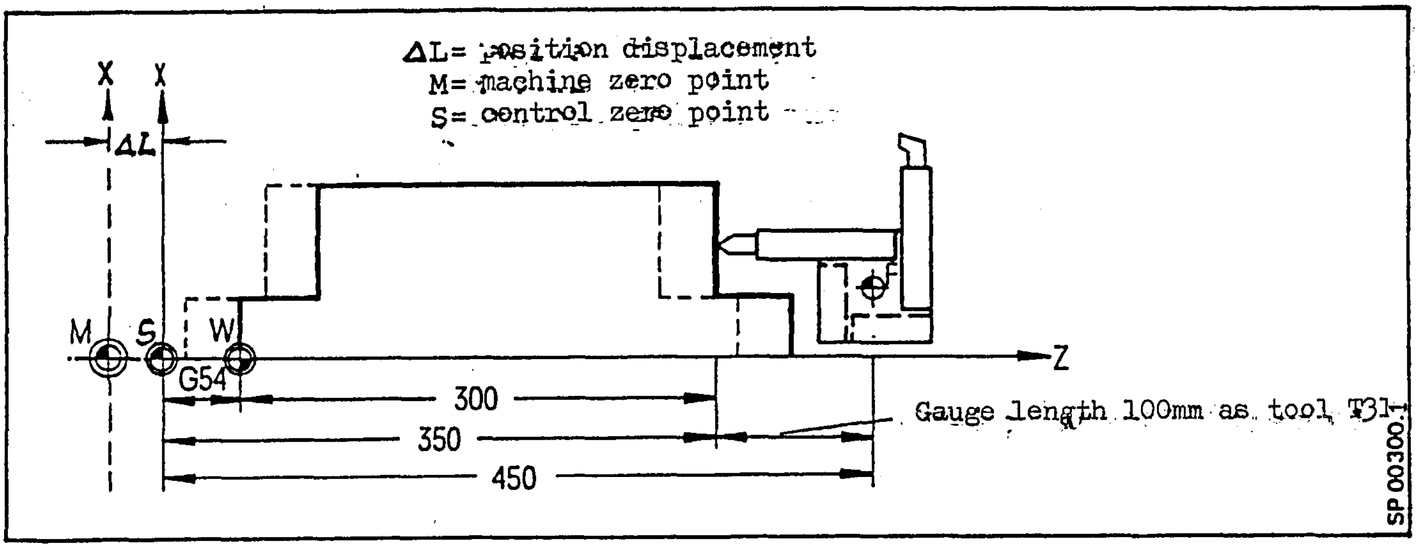
Resetting all G92 offsets
If G92 is programmed alone, i.e. without X and Z address, all summated G92 offsets for each axis are reset. The control zero point (S) again coincides with the machine zero point (M).

Example: N... G92 LF

The setting of the position memory is cleared also at end of program (M02 or M30) and with reset.

Example

The position of the shoulder on each turned component of a series varies more in the longitudinal axis due to automatic chucking than the available machining offset. To prevent the operator from having to continually adjust the zero point offset, a guage is moved in until it touches the shoulder and the operative block is interrupted. Using G92 and taking gauge length into account, this position referred to the workpiece is set as a tool offset and the workpiece length set as the Z position. Only then does the actual machining programme start (all dimensions in mm).



Programming :

```
                                        LF
N...    T3131
N ... G54              Z-999.           LF   -  (Traverse into workpiece until gauge
                                                 contact interrupts block).
N ... G92              Z 300.           LF
N ....  (MACHINING PROGRAMME)  ...
```

The control sets the actual value to 450 using the following calculation:

Calc. of TO, ZO WRT Datum	Example from diagram with ZO with TO	Example without ZO	Example without ZO , without TO
G54 Zero point offset	50	0	0
+Z Setting position	300	300	300
+ TO gauge longitudinal offset	100	100	0
Actual value Z	450	400	300

The setting of the actual value store using G92 is cancelled at the end of the programme (M02 or M30) and with RESET.

3.20 G94/G95/G96/G97 Feedrate F, M36, M37

Feedrate F in mm/min or in mm/rev.
G95 (F in mm/rev.) is Always a reset state

The programmed speed is maintained on the path of the cutter point "P" (imaginary tool point) also with tool nose radius compensation.

Using a feedrate override switch on the operators panel the programmed feedrate can be adjusted between 1% and 120%. The 100% position corresponds to the programmed value. During threadcutting the feedrate input is de-activated and the feedrate override switch is non-operational.

The programmed feedrate can be reduced by 1:100 using M37.
With M36 feedrate is as programmed using F.

e.g., N5 ... G94 M36 F10 LF

G94 Feedrate F in (mm/min)
G95 Feedrate F in (mm/rev)
G96 Feedrate F in (mm/min) and
Constant cutting speed S (in m/min)

M36 need only be programmed if M37 has been previously programmed.

Feedrate

The relationship between the rotational feedrate and the spindle speed and limit values is shown in the Fig. "Limit data for rotational feedrate"(see chapter 8.2).

3.21 G96 S.. Constant Cutting Speed (v = constant) (option)

Depending on the programmed cutting speed the control determines the appropriate spindle speed for the continuously changing diameter.

```
N5  G01 G96 X.. S.. F.. LF
```

Constant cutting speed in (metre/min).

This correlation of turning diameter, spindle speed and feed rate ensures optimum matching of the programme to the machine, the workpiece material and the tool.

The zero point for the X axis must be the turning axis. This is ensured by returning to datum. The NC derives the position of the tool tip from the actual value, the tool length compensation nos. 1-32 and the supplementary tool length compensation G92 I... K... and calculates from this the spindle speed. Any actual value displacement by G92 X... or PRESET or zero offset from handwheel is ignored for v = const. The actual value displayed refers to the radius. In G96 selection block the X axis must also be programmed.
When function G97 is programmed the constant cutting speed is switched off and the actual speed value frozen. This function is used to avoid undesirable changes in speed in the event of intermediate blocks in the X direction without machining.

Change of Gear range

A gear is selected and used for constant cutting speed. It is acceptable at any time to change gear at an appropriate part in the programme.

3.22 G92 S.. Limitation of spindle speed

It may become necessary (e.g., with constant cutting speed G96) to limit the spindle speed at some point, i.e., to continue machining at a certain point with a constant spindle speed. The limitation is programmed in a separate block using address S in rpm prior to the programme section in which it is to be effective. The function G92 S... may be used more than once in the programme.

N..	G92	S300	LF

No further commands in the same block

Spindle speed limited to 300 rpm.

This limitation is not effective with G94 or G95.

Cancellation of this limitation also results from using G92 S...., whereby the maximum speed for the selected gear stage must be written under S. G92 will stop the spindle.

3.23. G26 S... Monitoring of the actual spindle speed

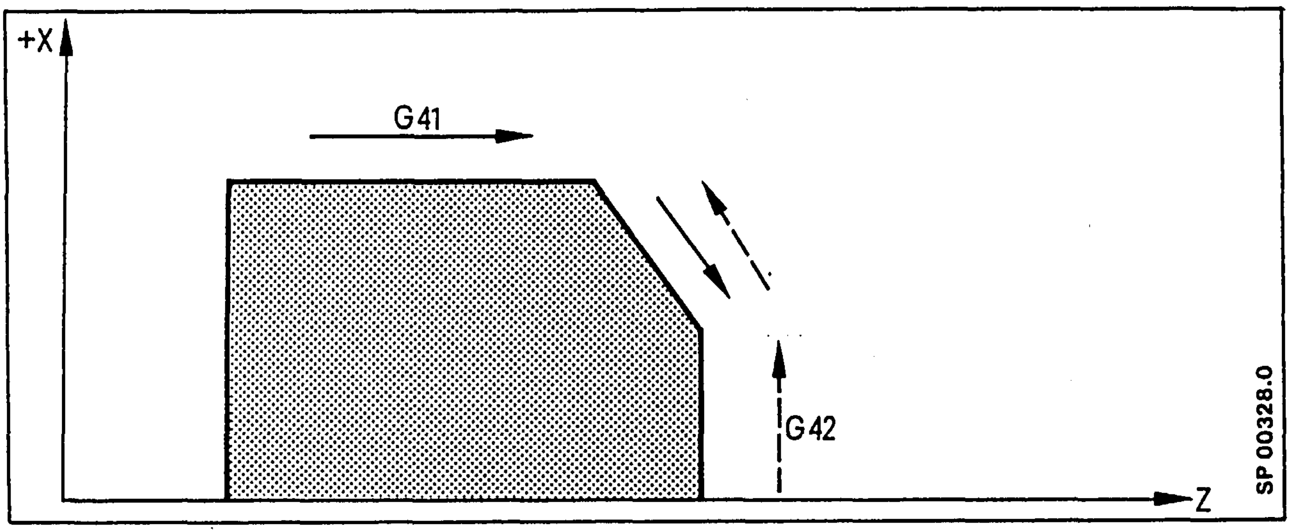
The speed monitoring G26 S... is a workpiece or chuck- dependant safety speed for the complete part program, independent of speed programmed using G94 to G97. G26 S... limits the speed command value and monitors the speed actual value. The function G26 S... has been created primarily for the operator.

G26 S... See operating manual for input.

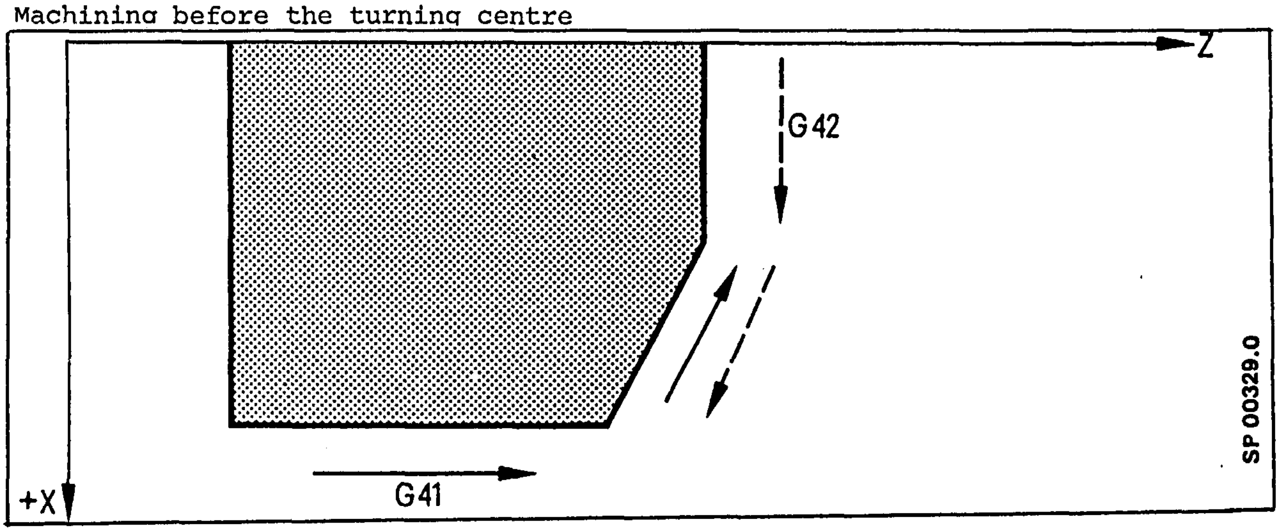
3.24 G40/41/42 Cutter radius compensation (CRC)

	Machining behind the turning centre	Machining before the turning centre
G40	Cancellation of CRC	Cancellation of CRC
G41	Tool left from workpiece	Tool right from workpiece
G42	.... right	.... left ....

Machining behind the turning centre



Machining before the turning centre



Using CRC the workpiece contour is programmed. The control calculates the path for the cutter radius centre (equidistant calculation) and the intersection points of these equidistant paths at acute transitions. CRC can be used for inside and outside contours, obtuse and acute angles and at acute and tangential transitions. There are no contouring errors.

The functions G 40 / 41 / 42 can be entered with or without path data. After programming G41 or G42 the cutter radius compensation becomes effective in the machining plane with the first movement (X and / or Z). Activation is only effective in blocks with G00 or G01. Within the programme (from block to block) a change from G41 to G42 or vice versa is possible. In this case the CRC does not have to be de-activated using G40.

In a stock removal cycle the correct correlation of G41 / 42 is automatic. At the end of the cycle G40 is automatically effective (see section 7.3).

Example: activate, change, de-activate.

```
N180  G01   G41   X... Z...      -  activate
N190        G42   X...           -  change
N200              X... Z...
N210        G40                  -  de-activate.
```

Example : activation and de-activation of the cutter radius compensation

```
%5551  LF
N10                                        T 1212 M.. M..   LF - Select tool
N20   G00   X100        Z212               S190       M04   LF   and offset
N30   G01   G41   X90                      F5               LF - Activation of CRC;
                                                                 the corrected path
                                                                 is reached at the
                                                                 end of this block .
N80   G00   G40   X200  Z350                                LF - Deactivation of CRC
N90                                        T2222M..M..      LF   Tool change
N100  G00         X90   Z21.3   S220          M04           LF
N110  G42   G01   X80           F5                          LF - Activation of CRC ;
N120                    Z150    F...                        LF   at the end of this
                                                                 block the corrected
                                                                 path is reached.
N210  G00   G53   X...  Z...    S00     T00                 LF - Cancellation of
N220  M30                                                   LF   tool offset using T..00
```

Note:
If CRC is selected (G41/G42),G59, G25, G26, G33, G34, G35, G92 and @31 or L999 may not be programmed.

4.0 Switching and miscellaneous functions S,H,T,M,

The switching and miscellaneous functions are given out in the block in which they are programmed. A maximum of 3 M functions, 1 S, 1 T and 1 H, can be programmed in a single block. The output to the interface is in the following sequence:

- simultaneous output of all programmed functions with the exception of a second or third M function which may be programmed.

- output of the second M function
- output of the third M function.

A machine parameter is used to determine whether or not the functions are outputted before or during axes movement. Exact specifications are given by the machine tool builder. The following applies for output of the functions during axes movement:

If a new value is to become effective before axes movement, the new function must be written in the previous block.

4.1 S Word

The S word can be used in the following ways :

Spindle speed in coded values
Spindle speed in rpm or 0.1 rpm.
(Determined during commissioning).
Cutting speed in metre/min or 0.1 metre/min
(determined during commissioning).

The use of both formats for spindle speed and cutting speed is not possible.

4.2 Auxiliary functions H

For switching functions on the machine or movements, which are not under numerical control, one auxiliary function per block is available using the address H and this can be programmed using up to a maximum of 6 decades. For further details see the programming instructions of the machine tool builder.

4.3 T Word tool command

The tool command determines the tool (tool number) required for a particular machining section and also the relevant tool data (tool offset number).

T 12 32

- Tool command address → T
- Tool number 00..99 (1 or 2 decades) → 12
- Tool offset number → 32
  (TO select : 01. . 32 or 1...9)
  (TO cancel : 00 or 0)
  (1 or 2 decades depending on machine parameter)

Tool offset number

The tool data are stored under a tool offset number. Offset values for up to 32 tools can be stored.

Tool offset:

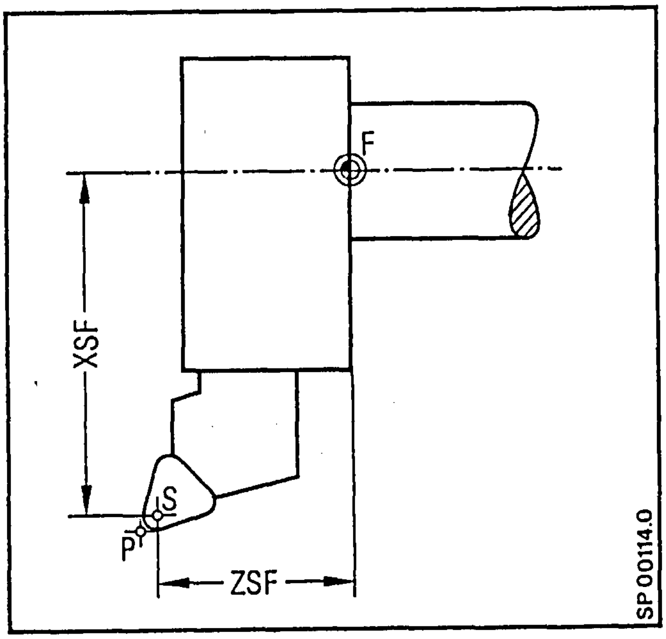
Each of the 32 tool offsets consists of:

Tool length compensation X axis Tool length compensation Z axis	Wear compensation
Tool geometry X axis Tool geometry Z axis Cutter radius Position of the tool tip	Insert

Additional programmable tool length compensations G92 I... K... (see 4.3.1).

4.3.1 Tool offset without using CRC

The effective tool offset is the sum of the tool geometry and the tool length compensation. This sum corresponds to the dimension XSF or ZSF.

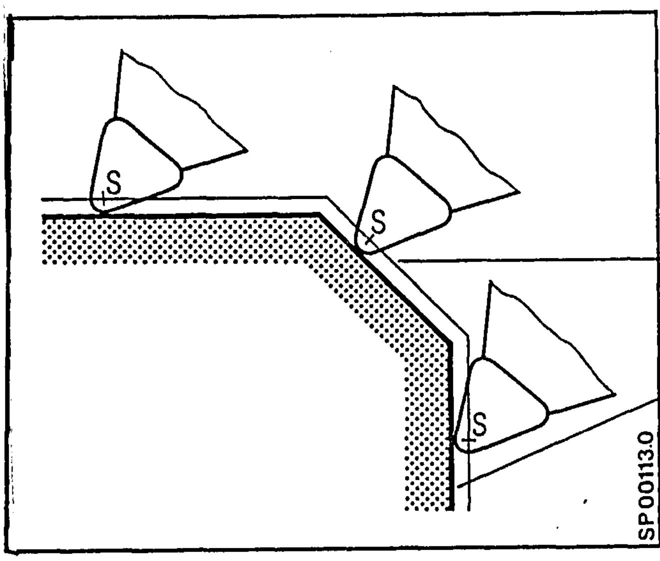


P = theoretical tool tip point

S = cutter radius centre

F = slideway reference point

The programme is written for the path of the cutter radius centre S. The length compensation refers to the cutter radius centre.



<u>Offset calculation :</u>

When changing the tool offset number the difference between the old and new values is calculated.
The following is determined during commissioning:

The resulting difference
- is traversed directly after the change in offsets; no axis command is necessary for traverse of the tool offset or the difference;
- is first considered when traversing of the appropriate axis occurs.

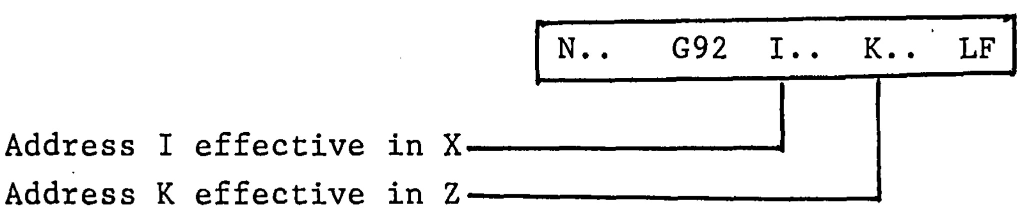
<u>Note</u>: With CRC (G41, G42) the difference is traversed <u>in both</u> axes in addition to the cutter radius.

Any external tool length compensation is added to the wear compensation value effective at that time. The new value is then entered in the active tool length compensation store (wear). The new tool offset becomes effective in the next block, if L999 has been programmed in the preceding block (empty intermediate memory, see 7.6).

<u>Additional programmable tool length compensation;</u>

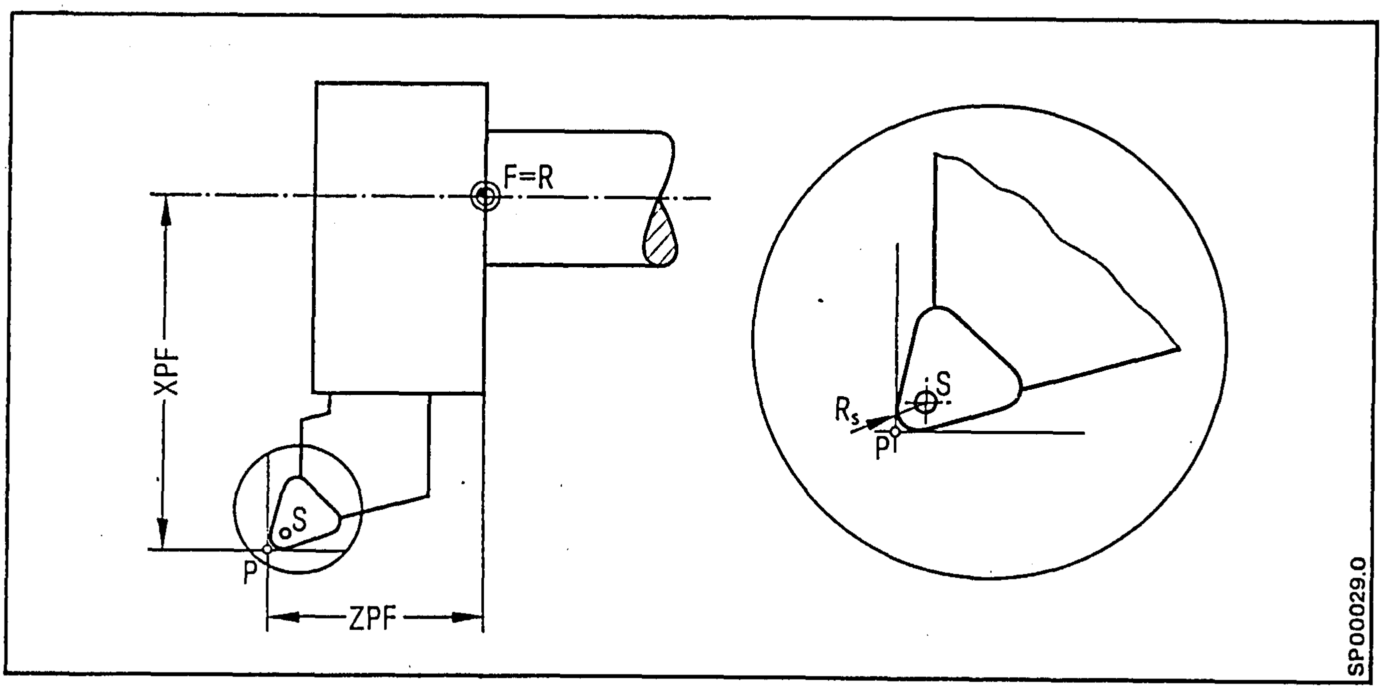
This facility allows the values for the tool geometry to be changed in a part programme. When this additional tool length compensation is programmed it is automatically added with the correct sign to the tool geometry values in the X and Z axes. So with G92 I.. K.. the presented value will

be loaded into the correction memory T33 and automatically added to the selected tool offset compensation.



4.3.2 Tool offset using cutter radius compensation (option)

Using cutter radius compensation the workpiece contour can be programmed. The tool length compensation to be inputted refers to the cutter point "P". Both the cutter radius and the position of the cutter point must be entered. The control then calculates the path to be traversed. No contour errors occur.
The cutter radius compensation is effective after execution of the block in which it is programmed (G41, G42) , i.e., the following block is traversed properly

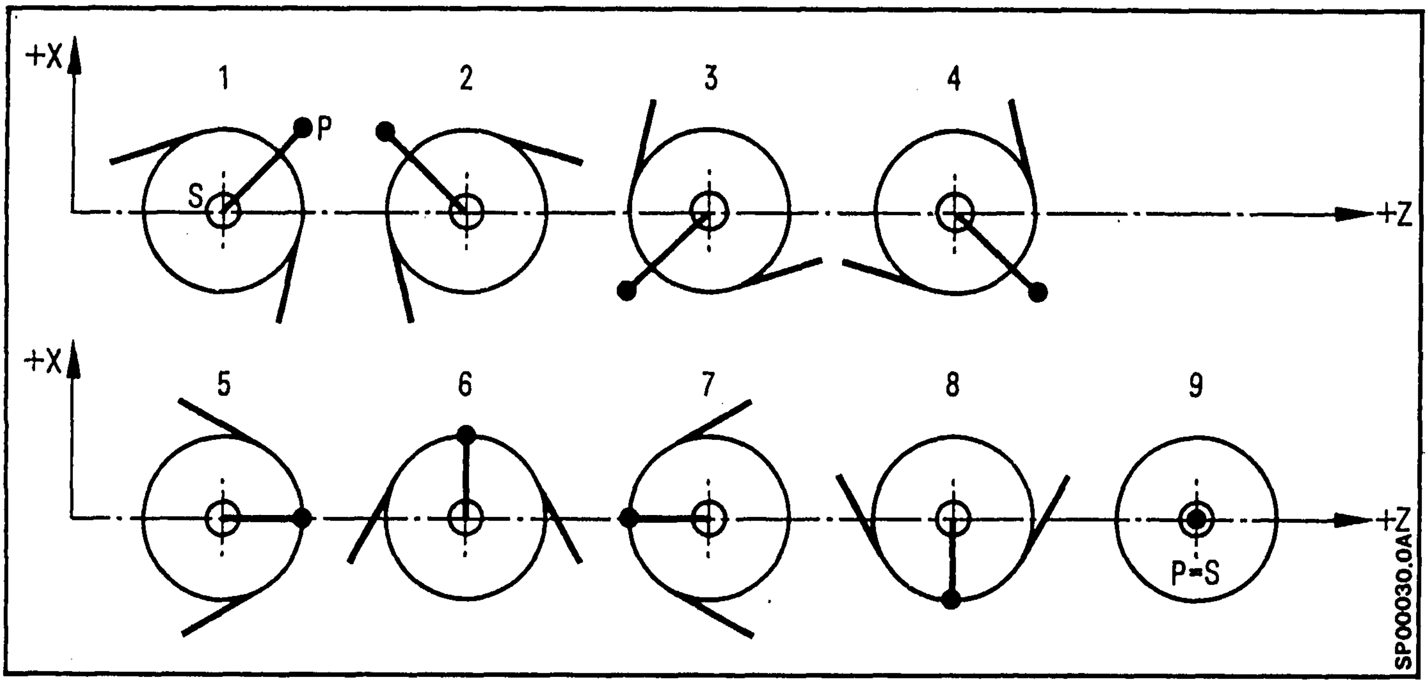


P = theoretical tool tip
S = cutter radius centre
R_S= cutter radius
F = machine slide reference point

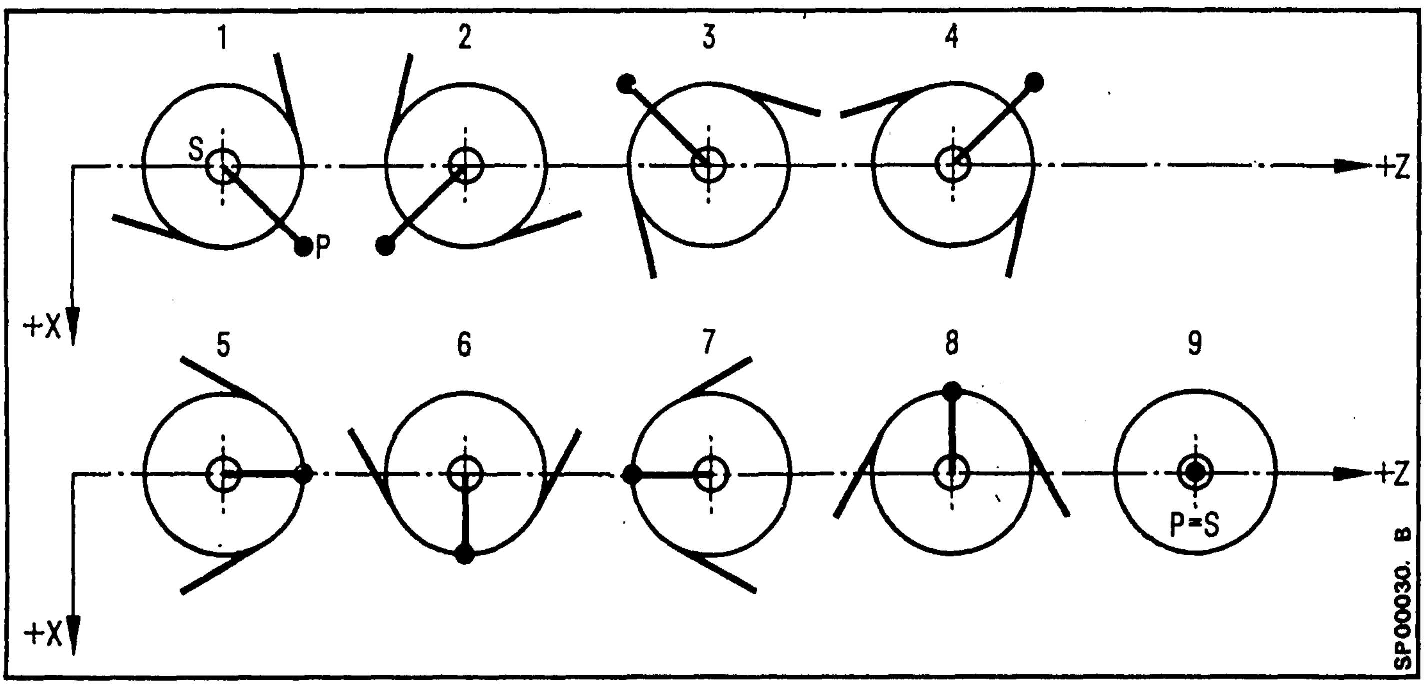
In order to calculate the cutter radius compensation, the control requires some indication of the position of the tool cutter point. A total of nine designations are used to represent the theoretical line between the tool tip "P" to the cutter radius centre "S".

The line of sight is always from S to P.

Machining behind the turning centre



Machining before the turning centre



When XSF and ZSF are selected as tool dimensions instead of XPF and ZPF (dimensioning cutter centre point - slide reference point), identification 9 must be used for each tool.

4.3.3 Tape format for input of tool offsets

Tool offset values can be entered not only via the operator panel but also via the data input interface.

TOOL OFFSETS FOR ...	Remark
% TO LF (TOOL OFFSET)	Offset start and rewind stop

```
N5  G92 T1  X.. Z.. B.. A.. LF
N10 G92 T32 X.. Z.. B.. A.. LF
N15 G92 T13 X.. Z.. B.. A.. LF
N..
N.. M02 oder M30 LF
```

T = Tool offset number

X = Tool geometry in X axis

Z = Tool geometry in Z axis

B = Cutter radius

A = Tool tip position

4.3.4 Programming of the tool offsets within a programme

Tool offsets can also be entered within a user programme.

In this case, the start condition % TO LF and the finish condition M02 or M30 LF are omitted.

```
% 1234
N5  G92  T1   X..  Z..  B..  A..  LF
N10 G92  T32  X..  Z..  B..  A..  LF
N15 G92  T13  x..  Z..  B..  A..  LF
N..
N.. M02 oder M30 LF
```

4.4 Miscellaneous functions M

M00 Programmed stop (unconditional)

M00 makes it possible to interrupt the programme in order to make a measurement or for some other similar purpose. After making the measurement the machine can be restarted by operating the start key " cycle ". The information stored in the control is maintained. The miscellaneous function M00 is operational in all automatic modes. Whether or not the spindle drive will be stopped will be stated in the special programming instructions from the machine tool manufacturer. M00 is also effective in a block without position data. It is possible to programme "M0" or only "M". The subsequent programming of L999 enables the display of a text written after L999. Selection by AUT PP.

M01 Programmed Stop (conditional)

M01 acts like M00 but only when the switch "Optional Stop" is actuated.
It is also permissible to write "M1".

M02 End of programme

M02 with programme return to programme start is written in the last block of the programme.
M02 can be programmed on its own or together with other functions.
A reading-in procedure is stopped by M02.

Writing M2 is also permitted. The control is put into the reset state (see programme key).

M17 End of sub-routine

M17 is written in the last block (alone or with other functions) of a sub-routine. M17 represents only the end of a sub-routine. M17 may not be programmed in the same block in which a subroutine is called up (when nesting.)

M30 End of programme with rewind

M30 acts like M02, except that in automatic mode from tape reader it initiates tape rewind to rewind stop "%".

M03 M04, M05, M19 Main spindle control

If the NC is equipped with analogue spindle speed output (option), certain M words are used for spindle control :

M03 Direction of spindle rotation clockwise
M04 Direction of spindle rotation counter-clockwise
M05 Spindle stop
M19 Orientated spindle stop (only with encoder)

Using M19 S it is possible to stop the spindle in a pre-defined position. Positioning from a stopped spindle is also possible. The angle is programmed using S in degrees (distance from the marker pulse in the M03 direction). The angle programmed using address S is modal. When M19 is programmed without S the stored value becomes effective for the angle. A block containing M19 is only finished when the signal "spindle stop" is received from the interface. Any of the functions "M3, M4, M5" is allowed. M19 or M19 S... must be programmed as a separate block. Positioning of the spindle occurs parallel to the axis motion independent of block boundaries.

M36, M37 Decreasing the Feedrate

The feedrate programmed under F in mm/min or mm/rev can be reduced by the ratio 1 : 100 using a further M function.

- M36 Feedrate remains as programmed under F
- M37 Feedrate is reduced by a ratio of 1 : 100

Unassigned miscellaneous functions

All miscellaneous functions except M00, M02, M03, M04, M05, M17, M19, M30, M36 and M37 are unassigned. Exact information regarding the application of the individual functions is given in the programme key specific to the machine. A partial definition of this function is given in DIN 66025.

5.0 Parameters

In part programmes it is possible to apply the parameters R00 to R99 (the machine tool builder can inhibit R50 to R99) to all addresses except N and @ instead of numeric values. For each parameter a certain numeric value is defined in the part programme or sub-routine. The dimension of the R parameter depends upon its associated address. R parameters are always written in 2 decades. A maximum of 10 parameters may be programmed per block.

Example:

```
L51000              LF          Parameters R01, R05 and R49
N1 Z-R49 SR05       LF          used in sub-routine.
N2 X100. - R01      LF
.
.
N50  M17            LF

% 5772 LF
N1 ... LF
.
.
N37  R01 10. R49-20.05 R05 500 LF
N38 L51002    LF                Call up of subroutine    510
                                2 repititions
                                R01   =   10
                                R05   =   500
                                R49   =   -20.05
```

5.1 Parameter definition

Defining the R parameter means that each R parameter is given a certain numeric value with sign.
The definition of the R parameter can be made in the part programme and/or subroutine. Up to 10 parameter definitions can be programmed in any one block.

Programmed Operation	Execution	Result
R01 10.78	R01 + 10.78	R01 = +10.78
R02 95.34	R02 + 95.34	R02 = +95.34
R03-555.1	R03 - 555.1	R03 = -555.1

5.2 Assignment of the parameter in the programme

Direct assignment An address is assigned directly to the value defined for an R parameter

Programmed operation	Execution	Result
F R01	F R01	F = +10.78
Z R02	Z R02	Z = +95.34
X-R03	X R03	X = +555.1

Additive assignment
The defined value of an R parameter with the correct sign is assigned to the digital value of an address.

Programmed operation	Execution	Result
X 20.78-R01	X (+20.78-10.78)	X = 10
Z 44.9-R03	Z 44.9-(-555.1)	Z = 600
F10.1 R02	F + 10.1+(+95.34)	F = 105.44

The sequence address, digital value, diameter must be maintained.
No sign is interpreted as (+) plus.

5.3 Parameter linking

Type of calculation	Programmed calculation	Execution	Result found in
Adding	R01 R02	R01+R02	R01
Subtracting	R01-R02	R01-R02	R01
Multiplying	R01.R02	R01.R02	R01
Dividing	R01/R02	R01:R02	R01
Square root	@10R01	$\sqrt{R01}$	R01
Sine	@15R01	sinR01	R01
Definition + addition	R01 10 R02	R01 +10 R01+R02	R01
Definition + subtraction	R01-10-R02	R01-10 R01-R02	R01

During multiplication and division the linking of R parameters and absolute numbers is not permitted. The symbol for multiplication corresponds to that of the decimal point and the symbol for division corresponds to that for block delete. The sequence of links is critical and each of @ 10 and @15 must be written in its own. When calculating the sine the angle must not exceed ± 360 °. The calculation time per link is approx 20 ms. One link may be programmed ber block.

Range of values : smallest value : 1.10^{-8}
largest value : $2^{27}-1$

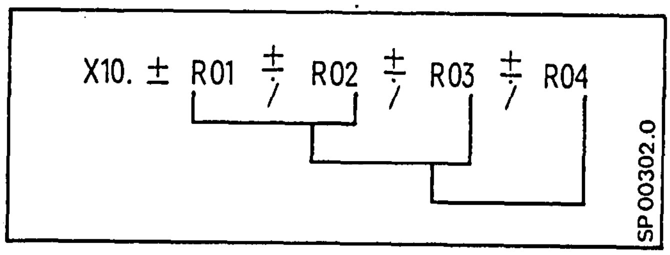
Display : Floating point (± .8) to (± 8.)

* Note : These functions are only possible with Sprint 8T.

5.4 Parameter chaining

Using parameter chaining certain values can be continually changed during multiple repitition of a programme section or sub-routine. A calculation is made during each programmed repetition of a chain.

Four-fold chaining of parameters is the maximum possible.



Calculation:

During calculation of a new R parameter value it is only the arithmetic sign between 2 parameters, which is important.

Example of chaining 2 parameters.

R01 + R02	new new	R01 = R01+R02 R02 = R02
-R01 + R02	new new	R01 = R01+R02 R02 = R02
R01 - R02	new new	R01 = R01-R02 R02 = R02
-R01 - R02	new new	R01 = R01-R02 R02 = R02
R01 . R02	new new	R01 = R01.R02 R02 = R02
-R01 / R02	new new	R01 = R01/R02 R02 = R02

Example of chaining four parameters

-R01+R02.R03-R04

new	R01	=	R01+R02
new	R02	=	R02.R03
new	R03	=	R03-R04
new	R04	=	R04

Otherwise the value statements for the individual parameters are still written with sign.

Note : chaining with multiplication and division is only possible with the SPRINT 8T.

Example :

% 9534 LF

.

N1 L0105 R01-10. R02 81 R03 3. LF

N6 L0206 R04 -1.R05 4. R06 -1. LF

.

N100 M30 LF

Sub-routine for the fifth or five-fold repitition and definition of the parameter.

R01 = - 10. R02=81. R03 =3

R04 = - 1. R05= 4. R06 = -1

L00100

N5 X 1000. - R01 + R02/R03 LF

N10 M17 LF

L00200

N1 Z100. + R04. R05+R06 LF

N20 M17 LF

Using the parameters in the called up sub-routine.

The following digital values result

L00100		X	- R01+R02	/	R03	
Repitition	Definition	1000.	-10.	81.	3.	
1	Assigned value	1010.	71	27.	3.	
2	Assigned value	929.	98.	9.	3.	
3	Assigned value	902.	107.	3.	3.	
4	Assigned value	893.	110.	1.	3.	
5	Assigned value	890.	111.	0.333	3.	

L00200		Z	+ R04.	R05+	R06	
Repitition	Definition	100.	-1.	4.	-1.	
1	Assigned value	99.	-4.	3.	-1.	
2	Assigned value	96.	-12.	2.	-1.	
3	Assigned value	88.	-24.	1.	-1.	
4	Assigned value	76.	-24.	0.	-1.	
5	Assigned value	76.	0.	-1.	-1.	
6	Assigned value	100	0.	-2.	-1.	

At the end of the programme, the parameters are given the last value to be ascertained. This value is maintained until a new definition or a further change of parameters occurs.

5.5 Correlation of a R parameter to the address "L"

The R parameters are used to define the sub-routine number as well as the number of repititions of sub-routines.

Examples:

Sub routine callup	Subroutine number	Number of repetitions
N13 L123	123	1
N12 .....R01 0 N13 L123 R01	123	0
N12 .... R01 99 N13 L123 R01	123	99
N12 ... R01 150 1) N13 L123 R01	124	50
N12 R01 12365 * N13 LR01	123	65
N12 R01 1236 * N13 LR01	12	36
N12 R01 12.69 N13 L123 R01	123	12

* Note : In these cases (LR01) the parameter value must be 4 or 5 decades.

1) When the repetition number is larger than 99, the rub-routine number is increased by the respective value from the 3rd digit seen from the right side, e.g. by 12 with a repetition number of 1234.

6.0 Short description of the contour (Only for Sprint 8T; option)

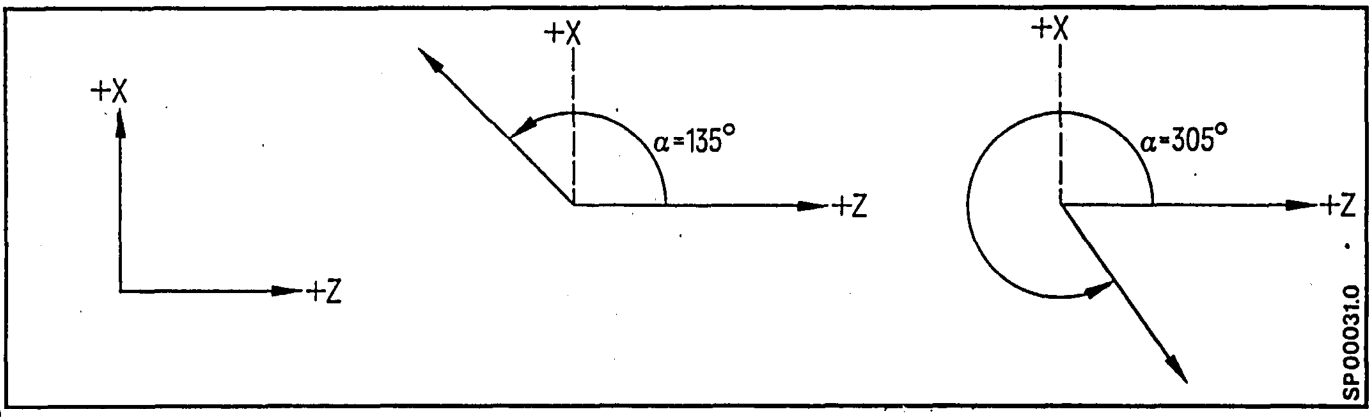
For the purpose of describing the contour multi-point traces are provided for direct programming from the workpiece drawing. The intersections of the straight sections are entered as co-ordinate values or as angles.

The individual straight line sections can intersect directly in the form of a corner or rounded using radii or chamfered. Chamfers and transition radii are programmed only together with their magnitudes. The control calculates all the geometry. Programming of the various end point co-ordinates is possible either in absolute data or incremental data. The first block of a contour section must always start with a linear interpolation G00 or G01. Then may follow G02/G03 with a circular block.

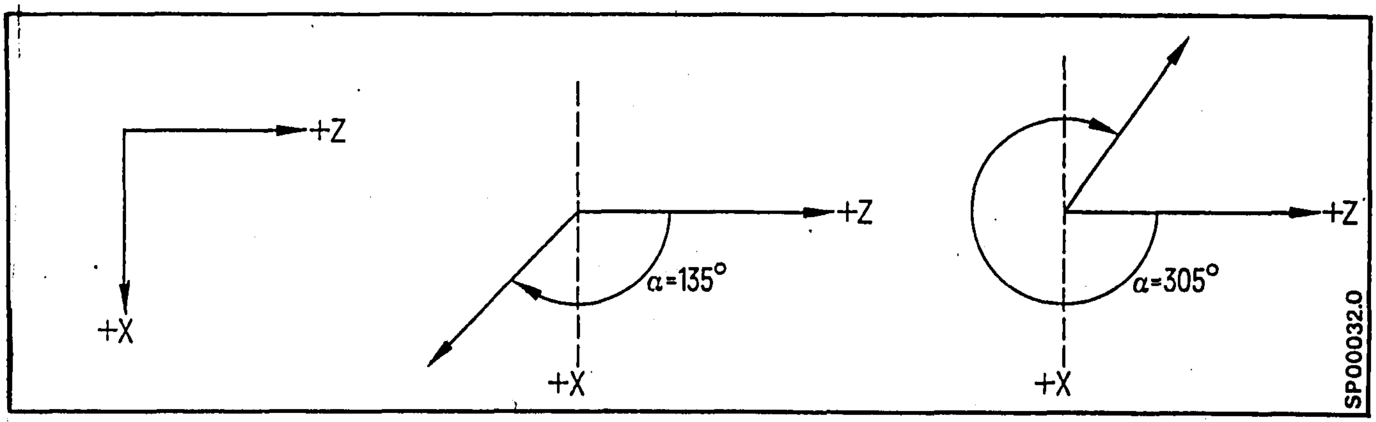
Angle (A) : Input resolution 0.00001 corresponding to 10^{-5} degrees.

The angle defined (max 359.999^0) always refers to the positive direction of the Z axis.

Right hand co-ordinate system and machining area behind the turning axis :



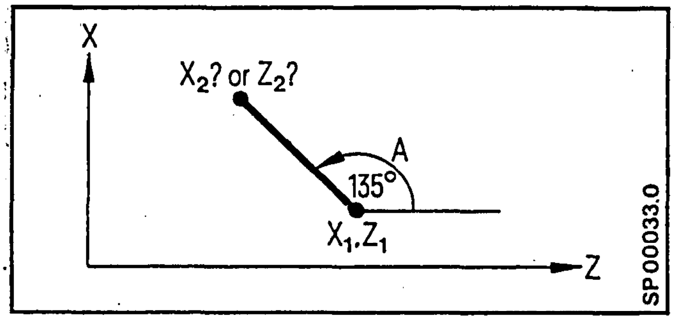
Right hand coordinate system and machining area in front of the turning axis



6.1 Programming contour sections

Examples (1) to (8) represent the basic elements of programming contour sections. These basic elements can be combined in a variety of ways (see pages 6-4 and 6-5).

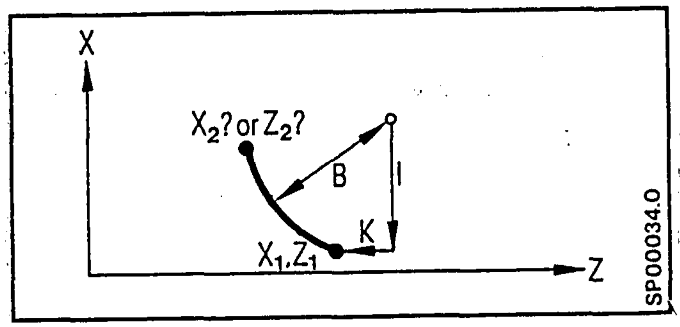
(1) 2 point section



N...A...X_2... (or Z_2)

The second end point co-ordinate is calculated by the control.

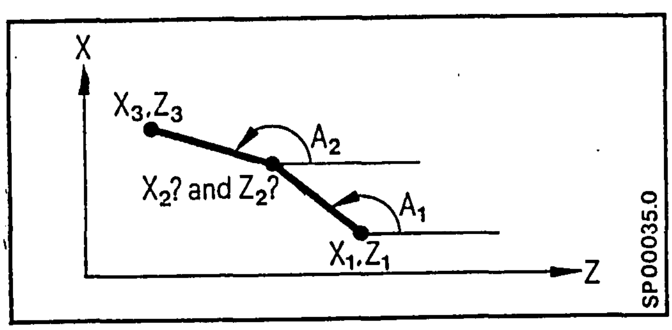
(2) Arc



N..G02 (or G03) I..K..B..X_2 (or Z_2)

The arc is restricted to one quadrant. The second end point co-ordinate is calculated by the control. The parameters I and K must both be programmed in the contour section, even if their value is zero.

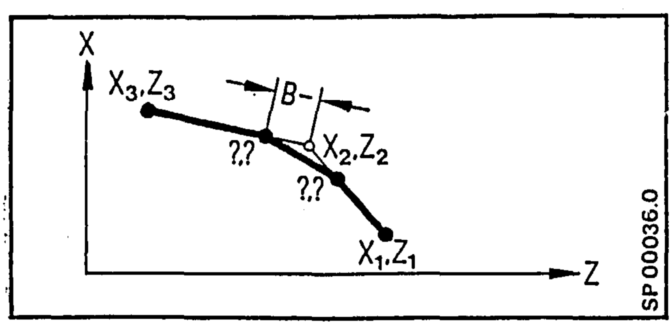
(3) 3 point section



N..A_1...A_2...X_3...Z_3...

The control calculates the co-ordinates (X_2, Z_2) and generates 2 blocks. The angle A_2 refers to the intermediate point (X_2, Z_2) which has not been programmed.

(4) Chamfer



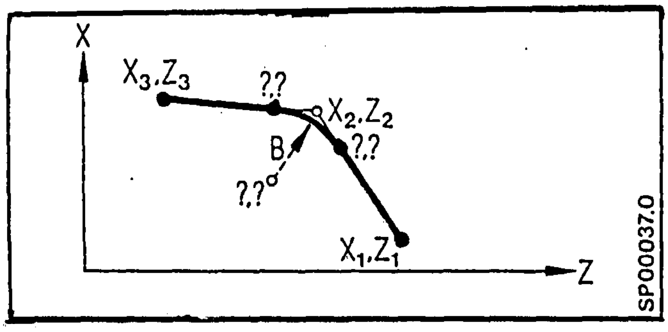
N151 X_2... Z_2... B-...
N152 X_3... Z_3... *

B - ... designates insertion of chamfer
B ... designates insertion of a radius

The "-" sign is no prefix in this case, but signifies the special designation of B as chamfer.

* the second block can also be a contour section.

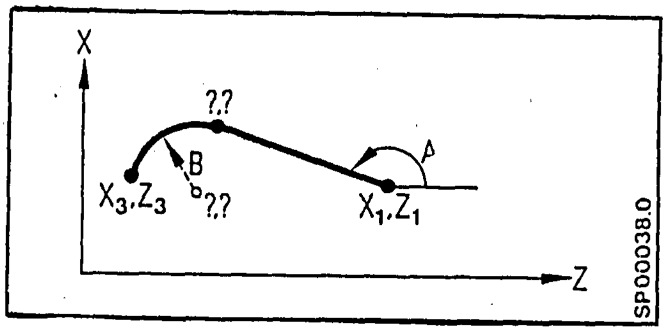
⑤ Radius



N...X_2...Z_2...B...
N...X_3...Z_3... *

The radius inserted must not be greater than the smaller of the two sections.

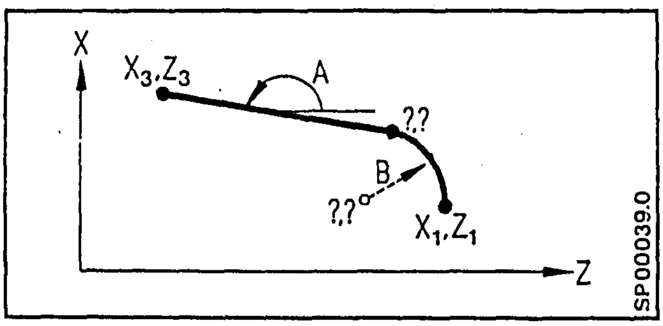
⑥ Straight line - arc (tangential)



N..G02(or G03)A..B..X_3..Z_3..

Arc not above 180°.
The sequence A (angle) and B (radius) must be observed.

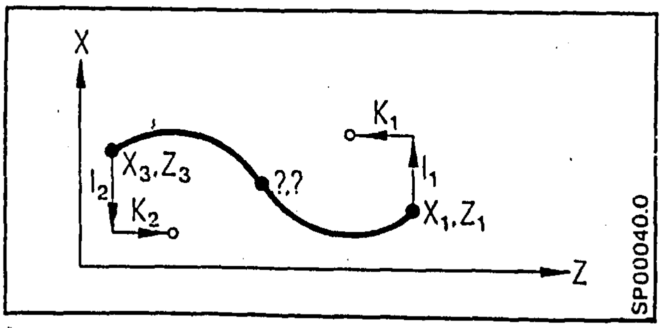
⑦ Arc - straight line (tangential)



N..G02(or G03)B..A..X_3..Z_3..

Arc not above 180°.
The sequence B, A must be observed.
No radius can be inserted in X_3, Z_3

⑧ Arc - Arc (tangential)

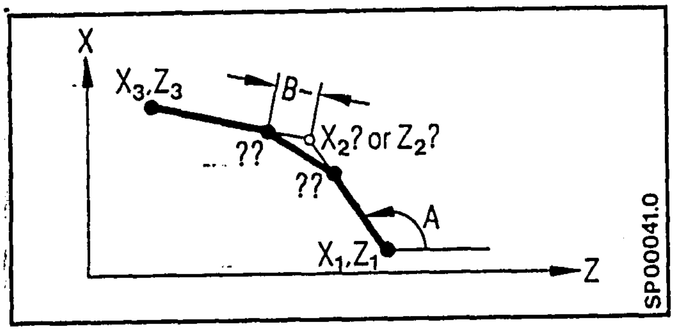


N..G02(or G03)I_1..K_1..I_2..K_2..X_3.Z_3
circle 1

The preparatory function is programmed for the first arc. The second preparatory function is a reversal of the first and is not programmed.

* the second block can also be a contour section.

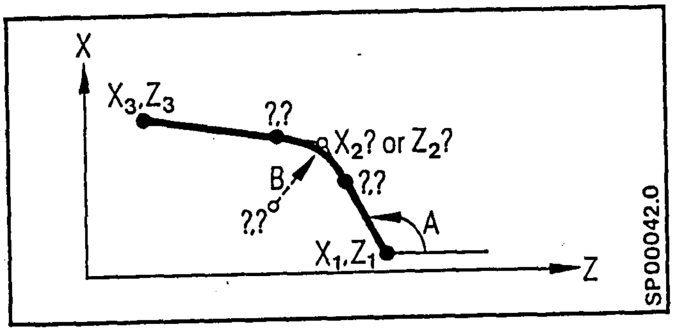
① + ④ 2 point sections + chamfer



N15 A..X_2 (or Z_2..) B-..

N16 X_3..Z_3... *

① + ⑤ 2 point sections + radius

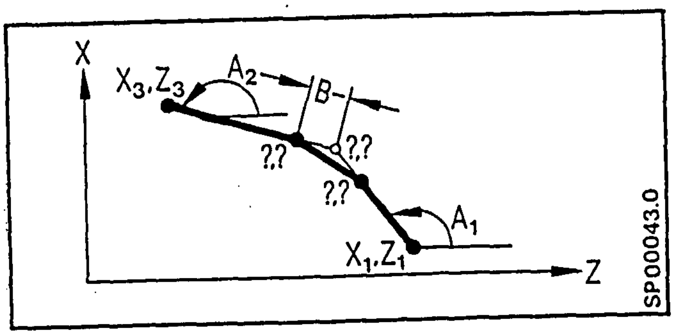


N15 A...X_2...(or Z_2...)B...

N16 X_3..Z_3... *

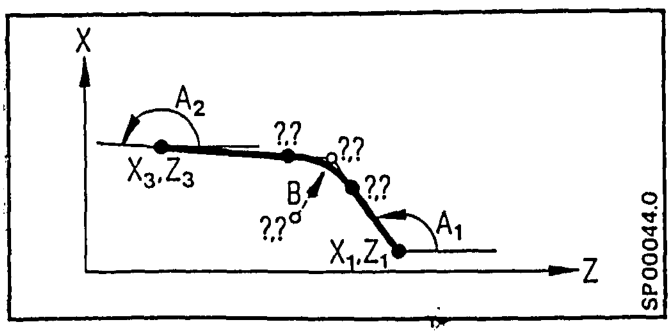
The radius inserted must not be larger than the smaller of the two sections.

③ + ④ 3 point sections + chamfer



N15 A_1..A_2..X_3..Z_3..B-..

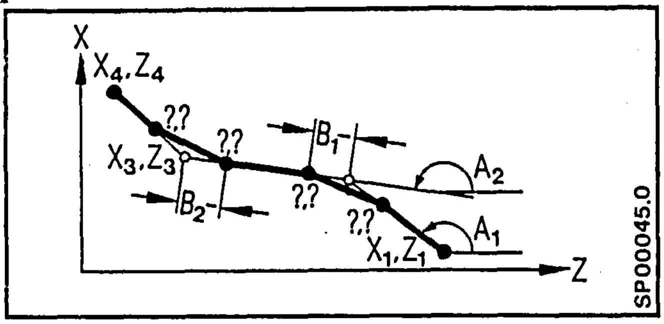
③ + ⑤ 3 point sections + radius



N15 A_1..A_2..X_3..Z_3..B...

* the second block can also be a contour section.

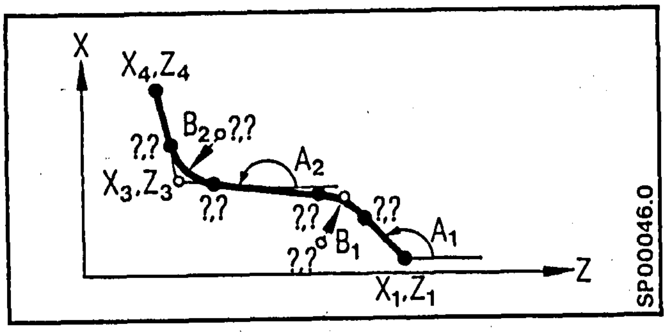
③ + ④ + ⑤ 3 point section + chamfer + chamfer



N15 A_1..A_2..X_3..Z_3..B_1-..B_2-..
N16 X_4..Z_4.. *

Insertion of a second chamfer at the end pont (X_3, Z_3,)

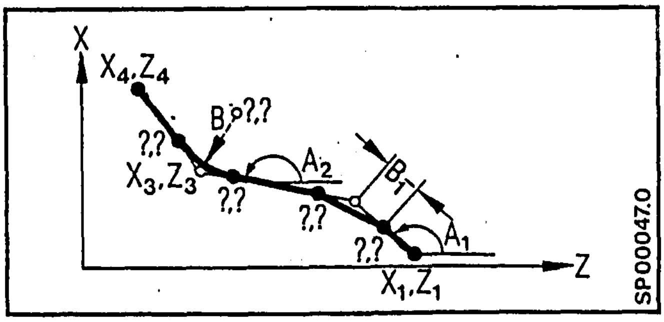
③ + ⑤ + ③ 3 point section + radius + radius



N15 A_1..A_2..X_3..Z_3..B_1..B_2..
N16 X_4..Z_4.. *.

Insertion of a second radius at the end point (X_3, Z_3)

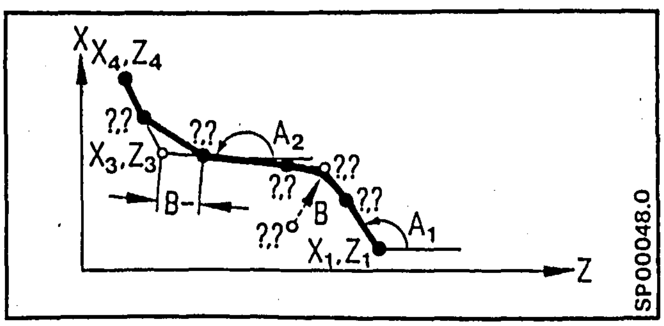
③ + ④ + ⑤ 3 point section + chamfer + radius



N15 A_1..A_2..X_3..Z_3..B_1-..B_2..
N16 X_4..Z_4.. *

Insertion of a radius at end point (X_3, Z_3).
In each case the next block is taken into account automatically.

③+⑤+④ 3 point section + radius + chamfer



N15 A_1..A_2..X_3..Z_3..B..B-..
N16 X_4..Z_4.. *

Insertion of a chamfer B- at the end point

* the second block can also be a contour section.

At corners where no chamfer or radius is to be inserted B0 should be programmed, if a radius or chamfer follows in the contour section. (Note: This programming will cause the control to generate a block with a path = 0. These blocks must take into account the effectiveness of the CRC. See para. 8.1.7).
B-0 is interpreted as B0.
A radius or a chamfer can be inserted when the subsequent block is not a circular block.

The sequence of addresses A, X, Z, B, F etc., is not important, however, angles and radii must have sequence as defined above(first angle before second angle, first radius before second radius in the direction of machining).

6.2 Mode of operation of functions G09, F,S,T,H,M, in the contour section

If a G09 is programmed in the contour section block, this only becomes effective at the end of the block, i.e., when the final position is reached.

At discontinuities (corners, edges) within the contour section, the control automatically generates a G09.

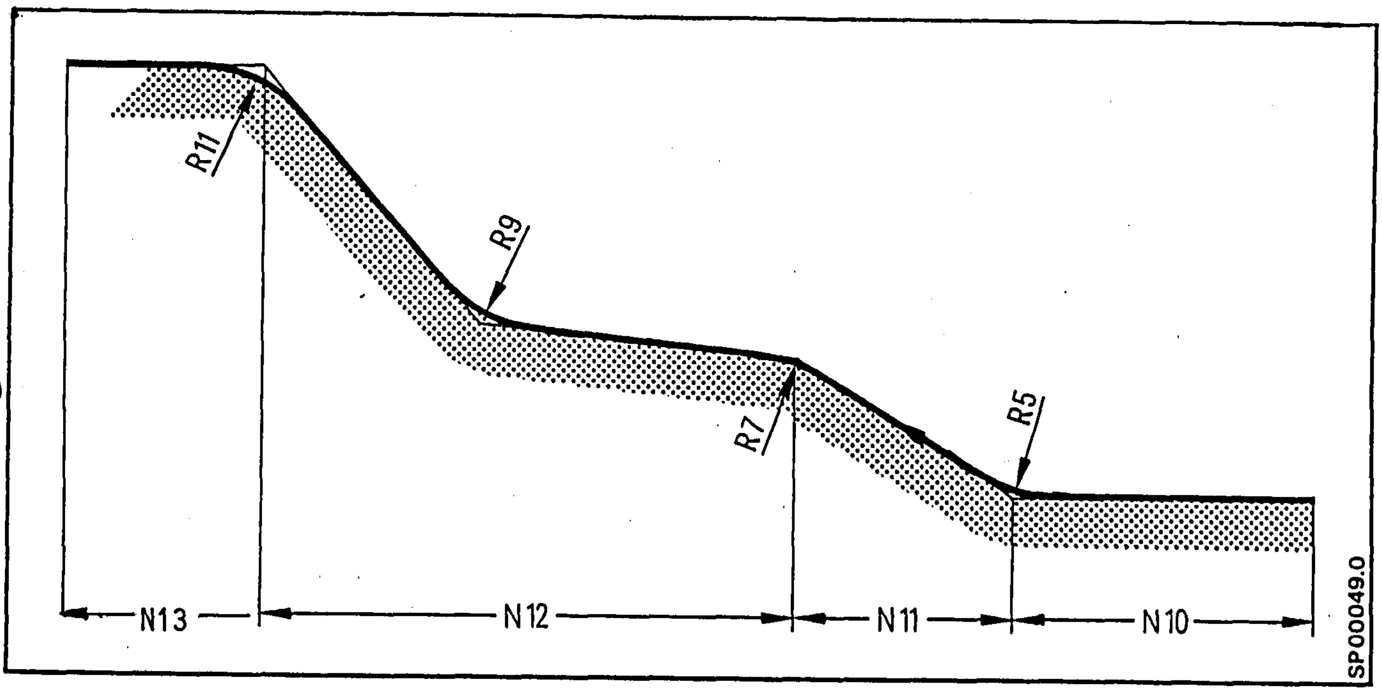
Where F,S,T,H,M are programmed in a contour section block, these are effective at the start of the block.

Where M00, M01, M02, M17, M30 are programmed in a contour section, these are effective at the end of the block.

6.3 Chaining of blocks

Blocks may be chained with and without angle data and with inserted radii or chamfers in any sequence.

Example:

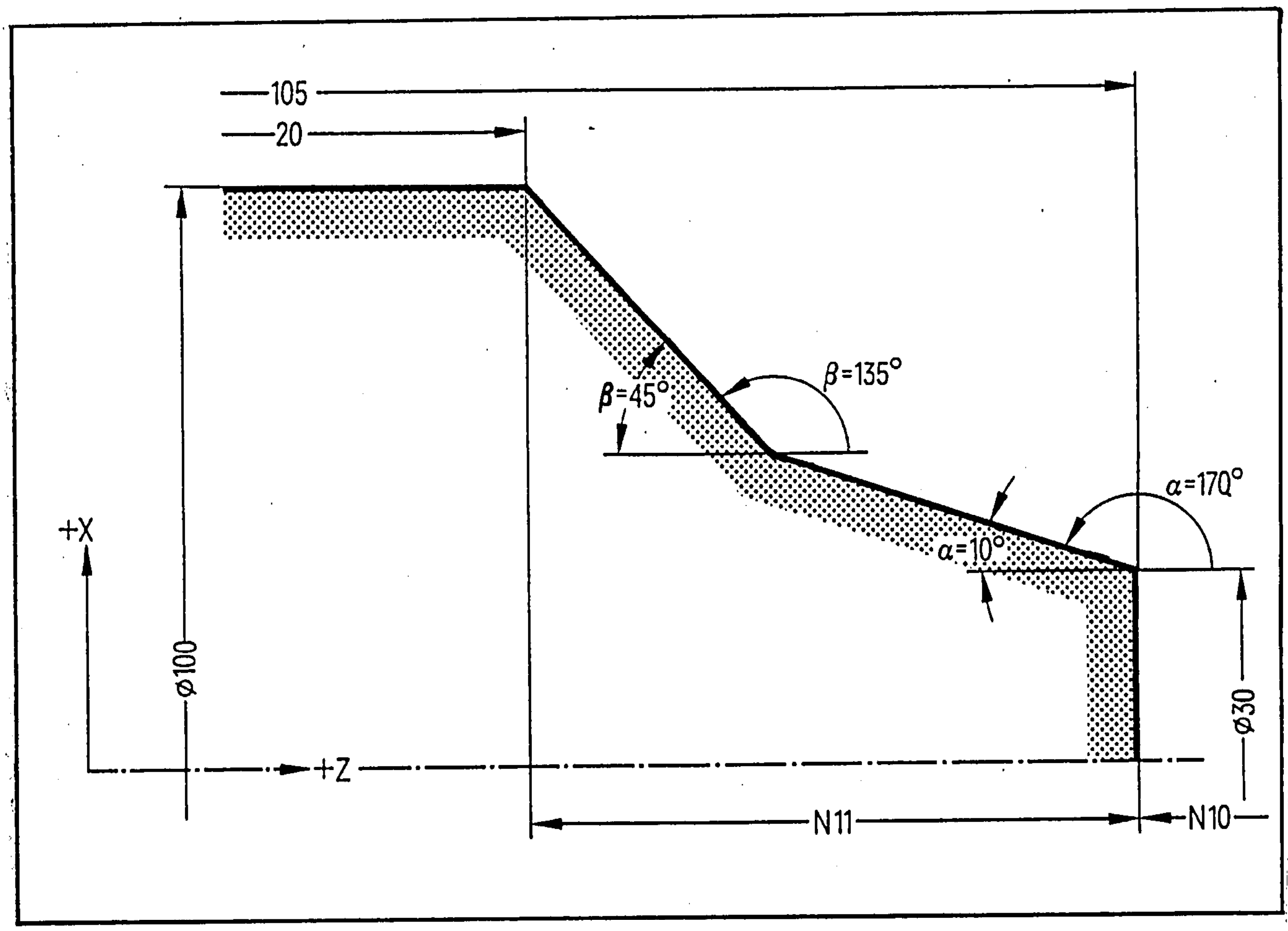



```
N10  Z...  B5    LF
N11  A...  X...     B7.  LF
N12  A...  A...  X...  Z...  B9.  B11.  LF
N13  Z...  LF
```

6.4 Examples

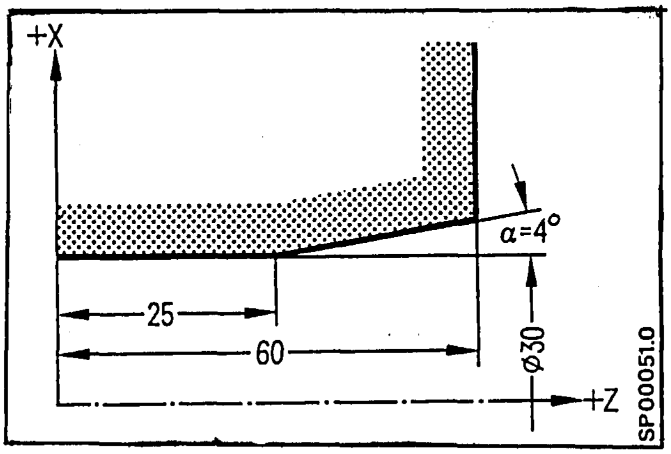
The angle α refers to the starting point, the angle β to the missing intermediate point.
The end point can be programmed in absolute data G90 or incremental data G91.
The end point co-ordinates must be stated. The control calculates the intermediate point from the known starting point, the two angles and the end point.

Example : Outside contour



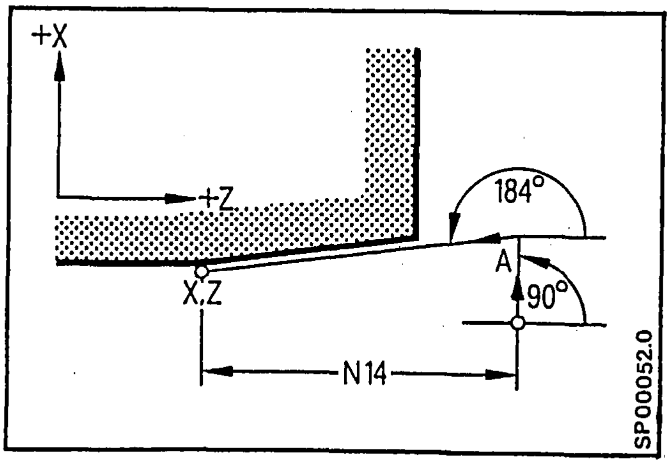

```
N10  G00  G90  X30.   Z105.   LF
N11  G01  A170. A135.  X100.   Z20.  F...  LF
```

Example : inside contour



Drawing dimensions

The start point is defined anywhere outside the internal taper.



The perpendicular to the start point and the extension of the inside taper gives the intersection A.

The programme is as follows:

.

.

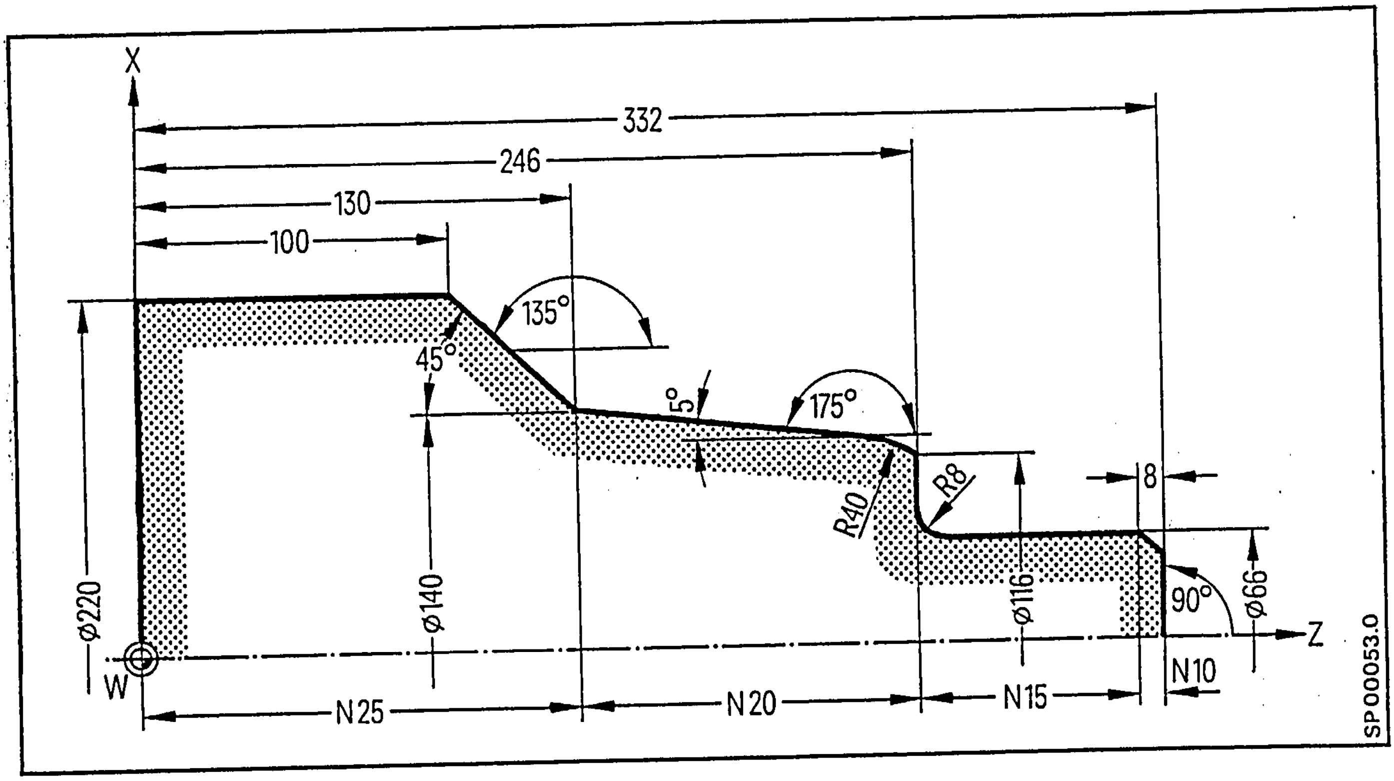
N13 G00 X start Z start LF

N14 G01 A 90 A 184. X... Z... LF

.

.

Example : Contour section programming



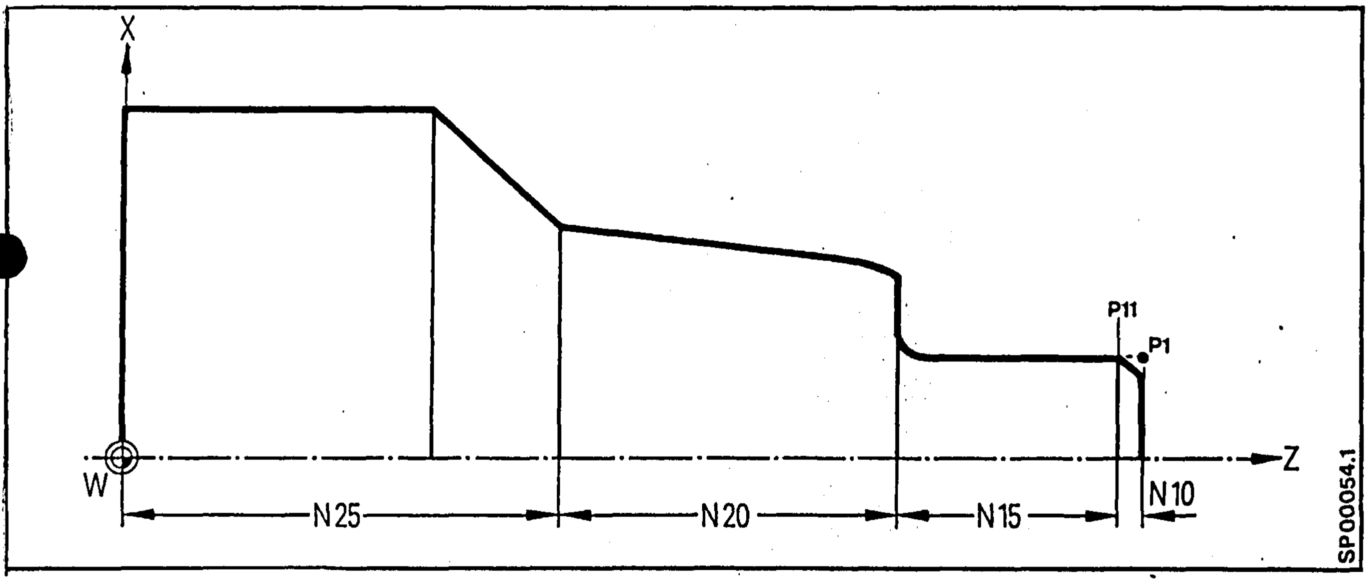
Programming using short contour description

```
L10500
N5  G00 G90 X0. Z332. LF
N10 G01 G09 A90. X66. B-8. F0.2 LF ┐  Linking using B - 8
N15 A180. A90. X116. Z246. B8. LF  ┘
N20 G03 B40. A175. X140. Z130. LF
N25 G01 A135. A180. X220. Z0.  LF
N30 M17 LF
```

6.5 Switching and auxiliary functions in linked blocks

Linked blocks result whenever blocks intersect due to radii or chamfers.

Example:



For programming see 6.10

A block with switching and auxiliary functions may be inserted between linked blocks.

Example : see above and page 6.10

```
N10   G01   G09   A90.   X66.  B-8. LF  (P11)
N101  M..   H..            LF
N15  A180   A90  X116   Z246          B8    LF
```

The switching and auxiliary functions become effective at point P11 (see above). Thus the cut leaves the contour at point P11. The F value programmed in block N10 becomes effective at the start of this block.

7.0 Cycles

7.1 Machining cycles (option)

Machining (canned) cycles are available as permanently - stored sub-routines for use as standard machining processes which are often repeated. (input via tape).

These cycles can be adapted to any particular machining problem by writing in the parameters.

Machining cycles can be applied to :
Tool change
Pattern repeat
Thread cutting
Deep hole boring.

Machine cycle call-up

A machine cycle is called up in the part programme or sub-routine.

e.g., N... L 91 LF

The allocation of a value to a particular R parameter can be made together with machine cycle call-up or in the previous blocks. For further details see the appropriate examples.

Continuation of programming after cycle call-up

All cycles end with the G functions G00, G40, G64, G90. Any other G functions required at program continuation have to be programmed anew.

The following machining cycles are available:

L91 Retract cycle for tool change - inside contour
retract sequence Z or Z - X

L92 Retract cycle for tool change - outside contour
retract sequence X or X-Z

L95 Stock removal with pattern repeat - parallel to axis

L950 Stock removal with pattern repeat - parallel to contour (for premolded workpieces)

L97 Thread cutting cycle

L98 Deep hole boring cycle

The description now follows for each of the cycles

If necessary, the described cycles can be changed. To be noted are additional indications from the tool manufacturer.

7.2 L91/L92 Retract cycles for tool change (revolver lathes)

Parameter R 18 / R 19

Using the parameters R18/R19 a protective zone around the workpiece can be written at the start of the programme. This protective zone is then active throughout the programme. R18/R19 may not therefore be used in any further programme. The parameters refer to the workpiece zero point.

R18 Protective zone in X
R19 Protective zone in Z

If the value 0 is programmed for parameter R18 or R19, this is interpreted as an unlimited protective zone in the associated axes direction. (see examples).

Note : Clearance movements for rotation of the tool holder may have to be programmed together with R18, R19.

Call-up :

L91 or L92 is called up, the control automatically calculates the optimum retract position. Before the call-up the tool must be located in such a position that no collision is possible during retract. For the purpose of calculating the optimum tool change point the control takes into account the greatest tool length of the tools being used and the active zero point offset.

During retract to the defined retract position the active tool offset remains stored depending on whether the call-up is in X or Z. Outside the protective zone all tool offsets are automatically de-activated and the machine traverses to the calculated tool change point.

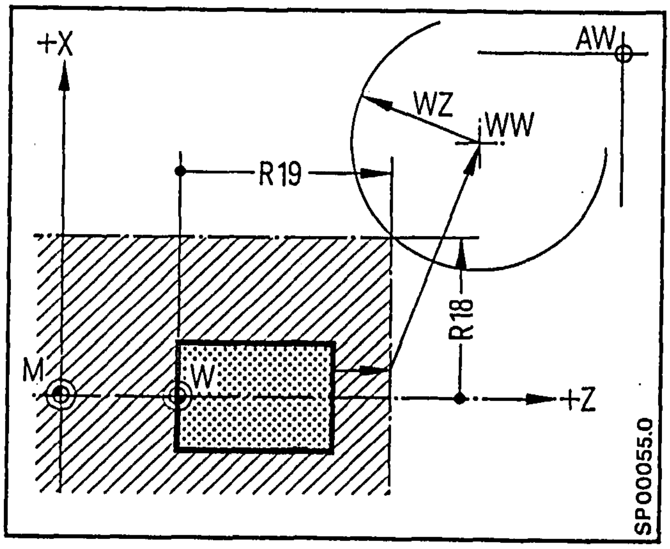
Limiting conditions

In the event that the calculated retract position "WW" exceeds the co-ordinates of the absolute tool change point "AW", the pull-out is only made to this point or to the corresponding co-ordinate (If a parameter has been programmed as 0).

If L91 or L92 is called up and no tool lengths have been entered in the memory, the retract movement is always made to the absolute tool change point "AW".

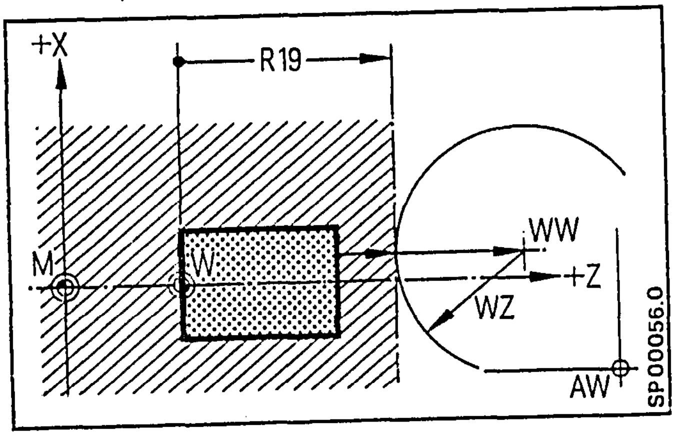
Example of inside contour machining L91

a) R18 and R19 are programmed. A no-go area is defined in the X and Z axes. Retract occurs within this area only along the Z axis and outside this area in the X and Z axes.



N1 R18 40. R19 65. LF
N2 T1 M15
.
.
N25 L91 LF (Pull out)
N26 T10 M15 LF (Tool change).

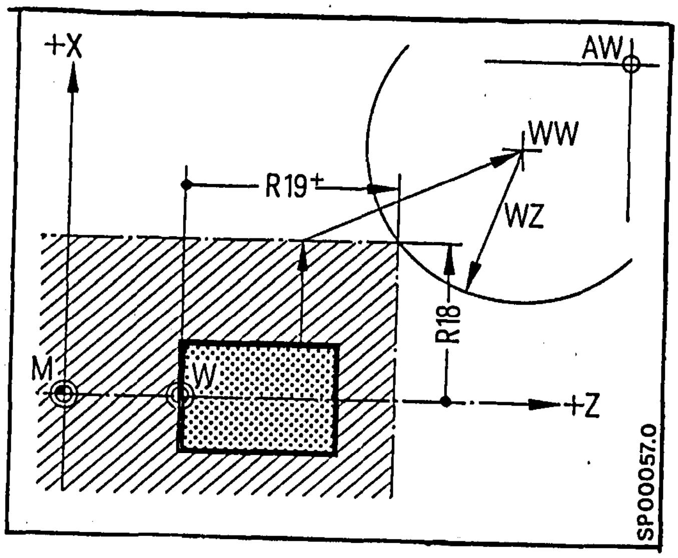
b) R18 = 0 and R19 are programmed. The no-go area is defined only in the Z axis. Retract occurs only along the Z axis.




```
N1   T1   R18   0.  R19  65.  LF
N2   T1             M15  LF
.
.
N25 L91   LF        (Pull  out )
N26 T10   M15  LF   (Tool change)
```

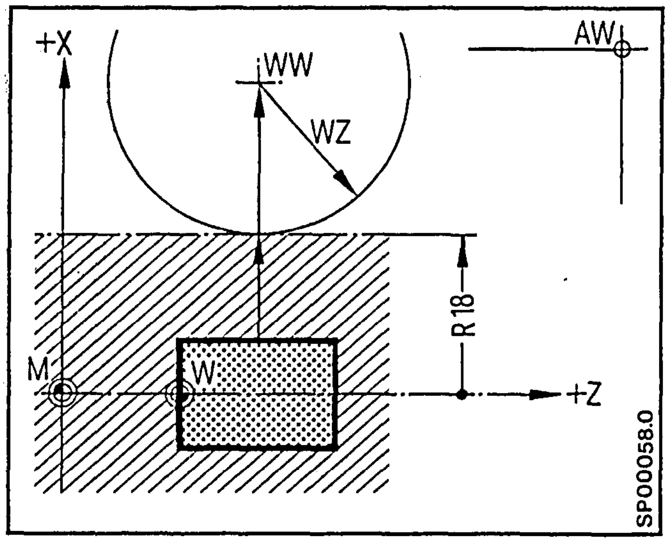
Examples of outside contour machining L92

a) R18 and R19 are programmed. The no-go area is defined in both the X and Z directions. Retract within the area only occurs in the X direction. Outside this area retract follows in both X and Z directions.



N1 R18 40. R19 65. LF
N2 T1 M15 LF
.
.
N25 L92 LF (Pull out)
N26 T2 M15 LF (Tool change.)

b) R18 and R19 = 0 are programmed. The no-go area is defined only in the X axis. Retract occurs only along the X axis.



N1 R19 0. R18 40. LF
N2 T1 M15 LF
•
•
N25 L92 LF (Pull out)
N26 T2 M15 LF (Tool change)

7.3 L95 stock removal cycle

7.3.1 L95 stock removal cycle (paraxial roughing)

R20 Sub-routine number under which the contour is defined

R21 Starting point of the contour in X (absolute).

R22 Starting point of the contour in Z (absolute)

R24 Depth of finishing cut in X (incremental)

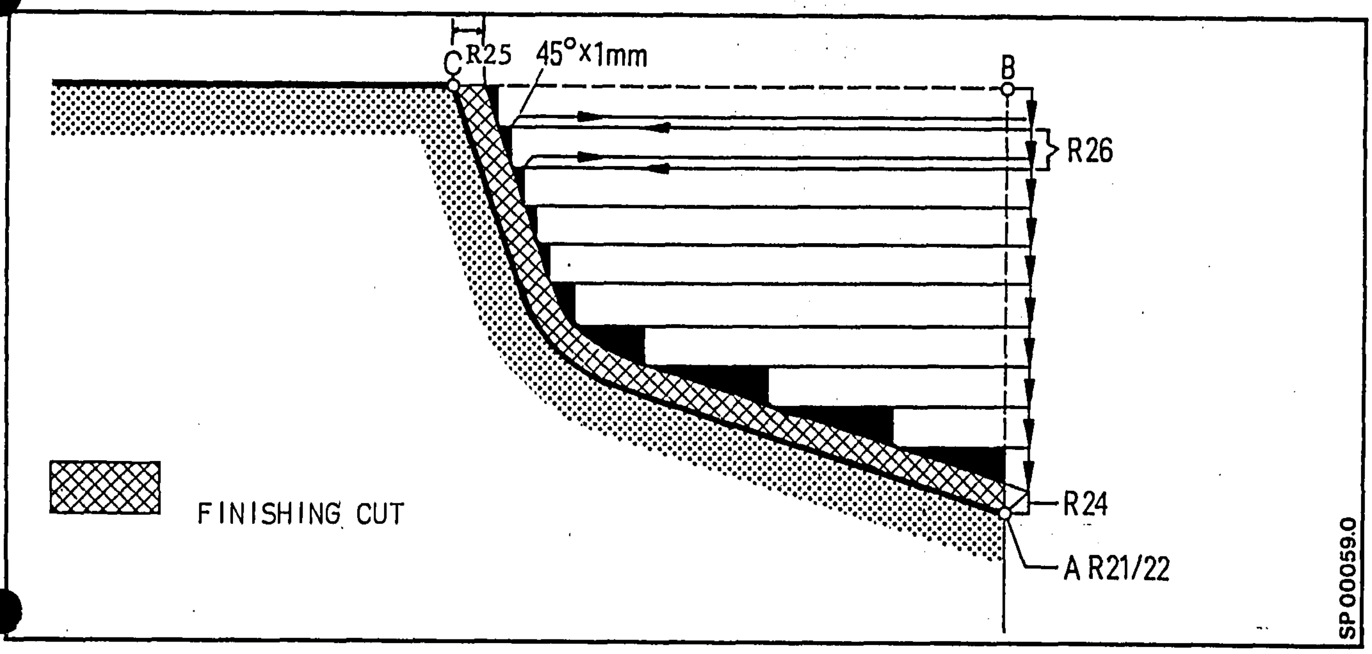
R25 Depth of finishing cut in Z (incremental)

R26 Roughing depth in X or Z (incremental)
(not required for final pass using R29 21, R29 22,)

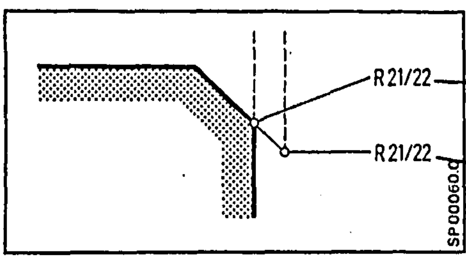
R27 Cutter radius compensation (41, 42 or 46)

R29 Form determination for roughing and finishing.

The parameters to be entered are shown in the diagram below.



The starting points R21 (X) and R22 (Z) are entered with reference to the contour. For the roughing pass the control automatically displaces the traverse by the finishing cut depth R24, R25 + 1 mm safety distance. However, if this distance is not sufficient the starting point for the contour R21, R22 should be correspondingly displaced.



Starting point of original contour

Starting point with displacement.

The machining cycle L95 can be called up from any collision-free position With the aid of the final contour description the control automatically determines the starting points.

The R parameters previously referred to are modal, i.e., the parameters R20, R21 and R22 need not be re-entered for a subsequent finishing pass cycle following a roughing cycle with L95.

The parameter values which are changed are

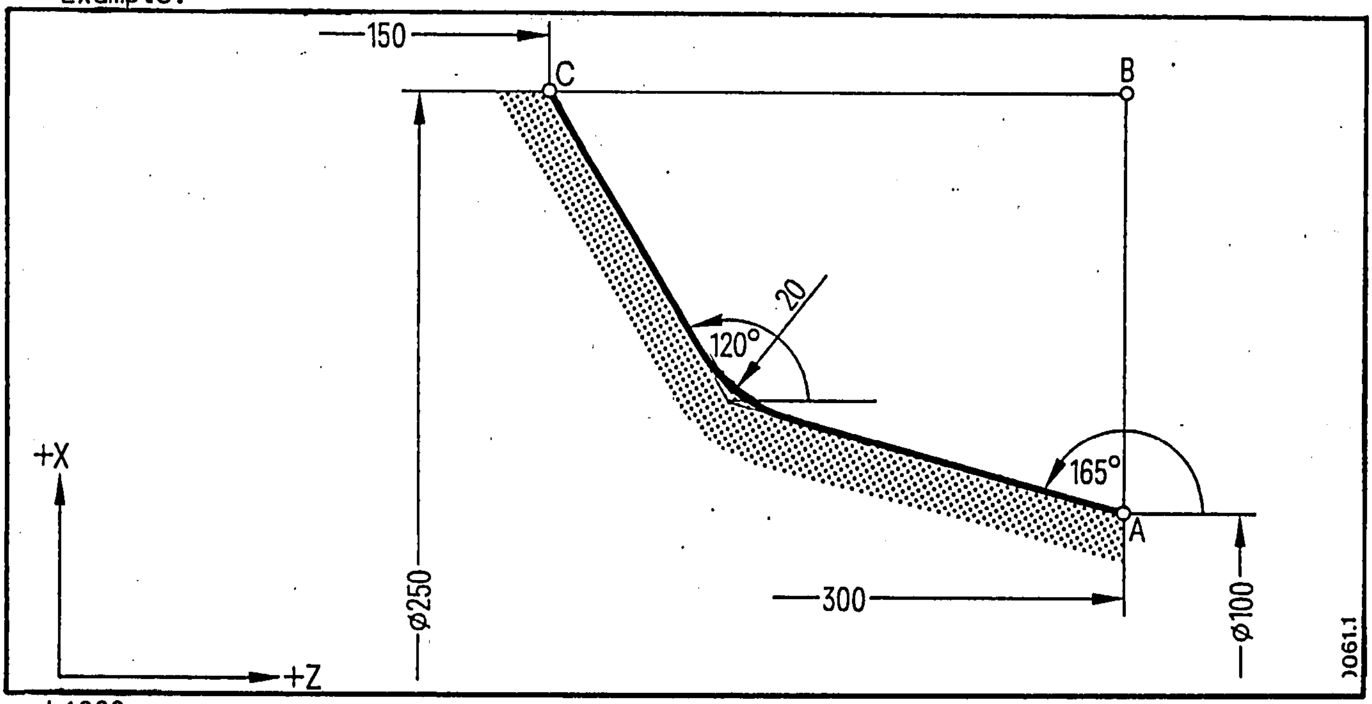
R24	Cut depth in X = 0
R25	Cut depth in Z = 0
R29	Type of machining "finishing".

Contour (R20)

For the purpose of automatic stock removal, the final contour of the component must be described. This is written as a sub-routine and called up within the stock removal cycle.

There is no limit to the number of blocks for the sub-routine. However, in each block there must be a co-ordinate value. No recess cutting is permitted within the roughing cycle. The final contour description can take the form of a short description of the contour. (Only Sprint 8T). The starting point, fixed by R21, R22, may not be programmed in the first programme block of the contour.

Example:



L1000

N1 G90 A165. A120. X250. Z150. B20. M17 LF

With cycle call up L95 the contour is registered using R20 10. The corner point B also represents the guide point for the finishing cycle. The values for the points A and C are determined from the cycle. The end of the cycle is point B.

M17 must be written in the last block of the subroutine (no separate block). Deletable blocks in the contour are permitted.

Type of machining (R29)

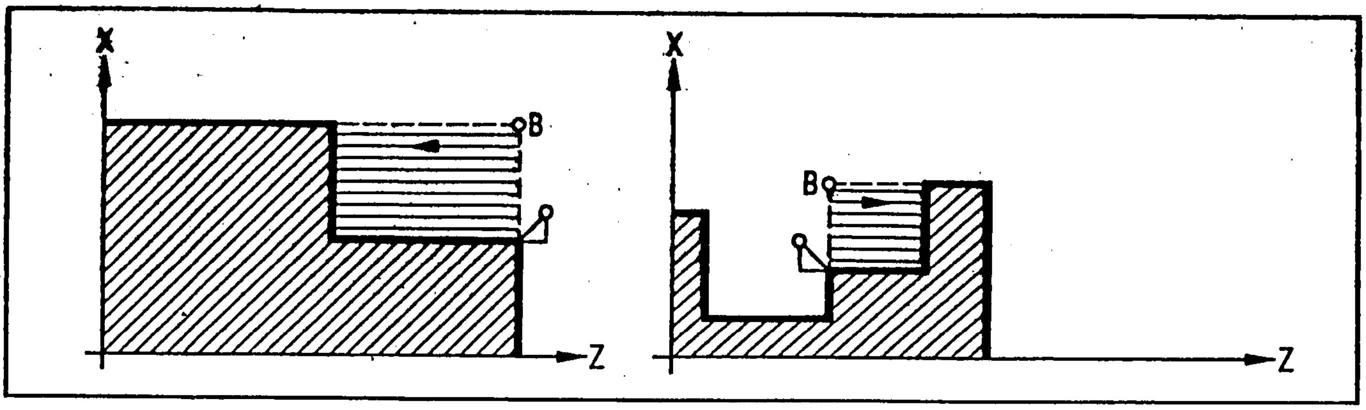
The parameter R29 describes the type of stock removal ; roughing or finishing whether inside or outside contour machining, the form of the pattern repeat whether longitudinal or facing.

R29	11	longitudinal	(Z)	outside	Roughing
R29	12	facing	(X)	outside	
R29	13	longitudinal	(Z)	inside	
R29	14	facing	(X)	inside	
R29	21			outside	*21,23 finishing to final pass depth.
R29	23			inside	
R29	31	longitudinal	(Z)	outside	*Paraxial roughing with one final cut parallel to the contour down to finish cut depth
R29	32	facing	(X)	outside	
R29	33	longitudinal	(Z)	inside	
R29	34	facing	(X)	inside	
R29	41	longitudinal	(Z)	outside	*Refer to R 2931 - R2934 and then finally one cut parallel to the contour down to the final contour.
R29	42	facing	(X)	outside	
R29	43	longitudinal	(Z)	inside	
R29	44	facing	(X)	inside	

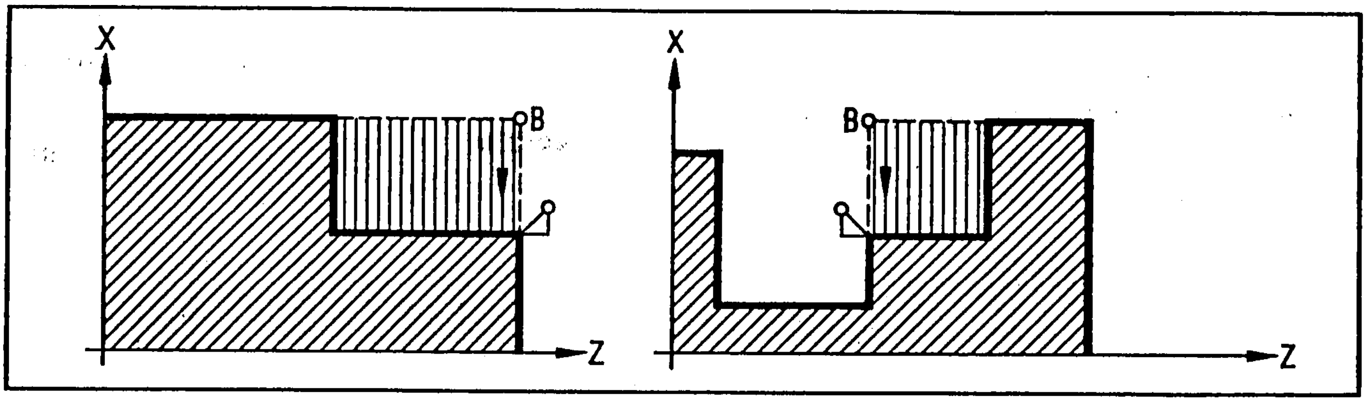
* In these cases the control automatically activates the cutter radius compensation in the correct direction providing that previously this has been activated using R27 (41 or 42).

With paraxial roughing the cutter radius compensation is suppressed internally. At the end of the cycle it is de-activated and if necessary must be reprogrammed. The correct sequence in time of activation and deactivation of cutter radius compensation within the cycle is controlled automatically by the cycle itself.

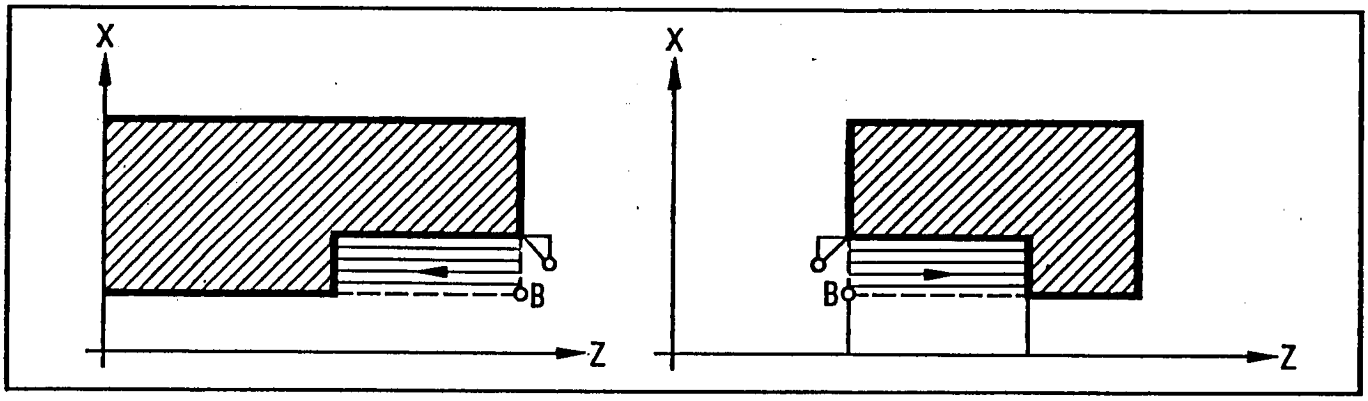
Example:



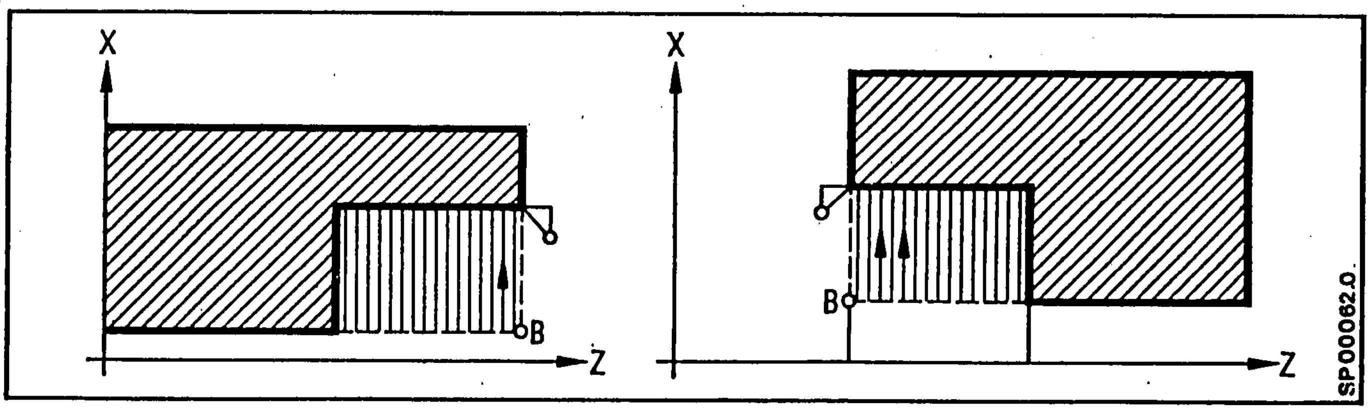
R29 11 - Outside contour roughing longitudinal (Z)



R29 12 - Outside contour roughing facing (X)

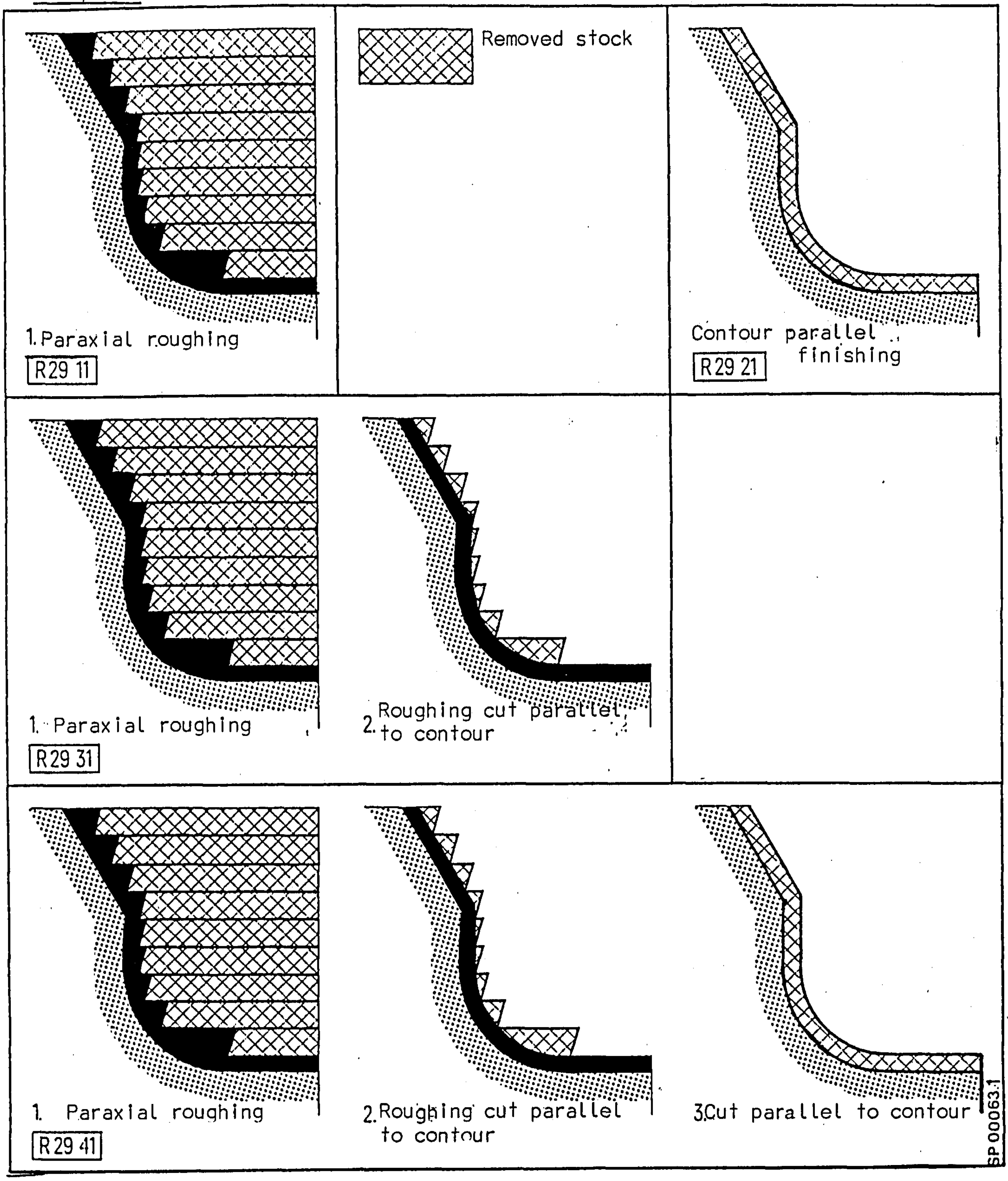


R29 13 - Inside contour roughing longitudinal (Z)

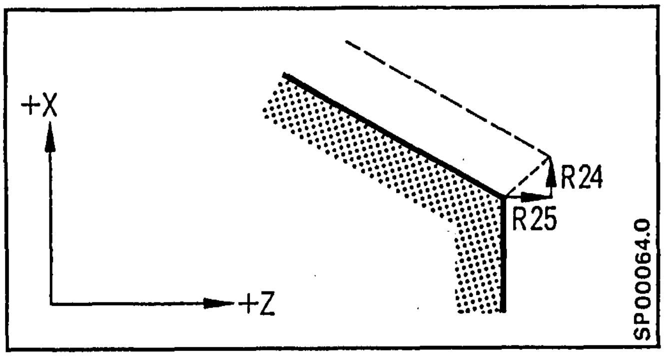


R29 14 - Inside contour roughing facing (X)

Examples :


1. Paraxial roughing
R29 11
Removed stock
Contour parallel finishing
R29 21
1. Paraxial roughing
R29 31
2. Roughing cut parallel to contour
1. Paraxial roughing
R29 41
2. Roughing cut parallel to contour
3. Cut parallel to contour
SP00063.1

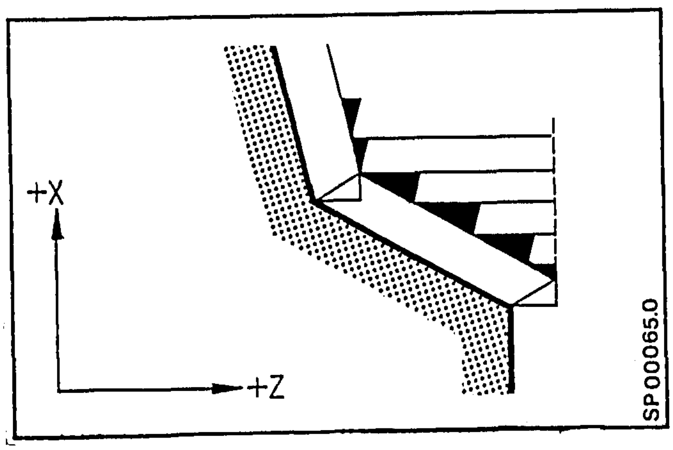
Depth of finish cut (R24, R25)



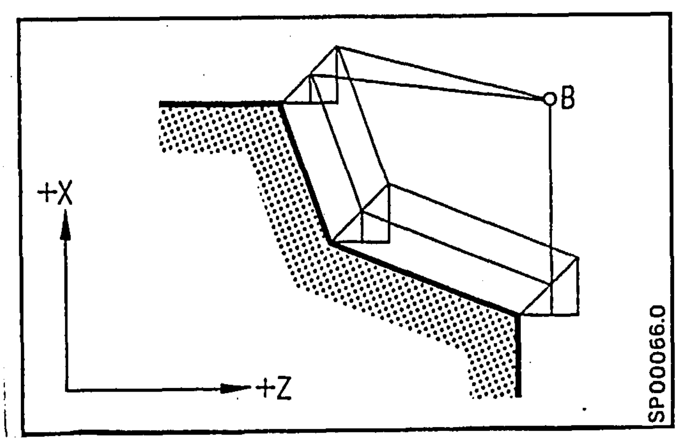
The contour is displaced by the depth of finishing cut programmed using R24 and R25.

e.g., R24 0.3
R25 0.3

During the roughing cycle, rough machining occurs down to this depth.



In the finishing cycle, machining to the cut depth which has been programmed:



Finishing cut depth R24 0, R25 0 corresponds to the final contour

Using several finishing cuts it is possible e.g., to turn down the corners left after the rough machining with the aid of a copy tool, in the event that the cycles R29 31 - R29 34 or R29 41 - R29 44 are not utilised.

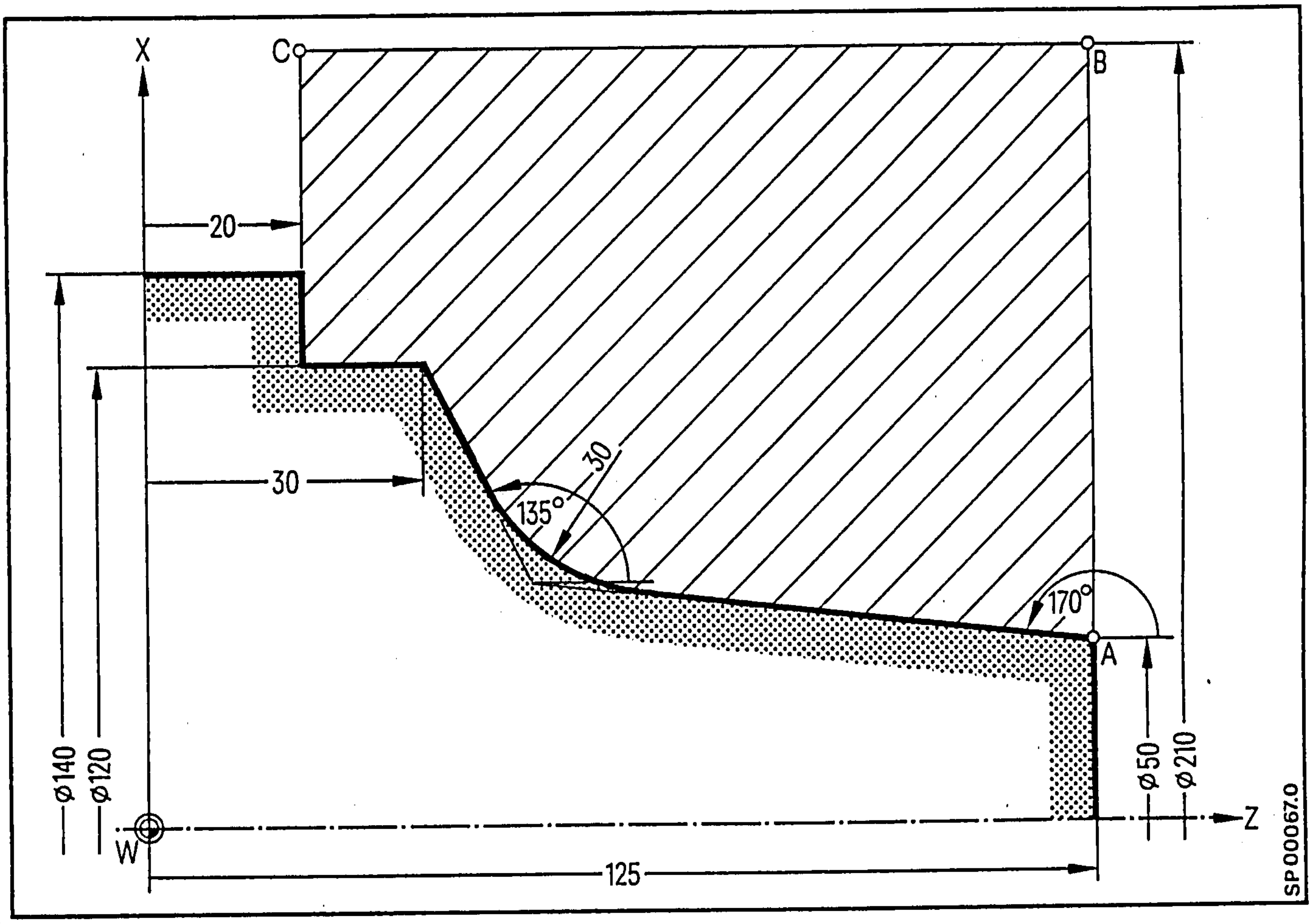
Stock removal at constant cutting speed

If the machining cycle is to be executed with constant cutting speed, the function "Constant Cutting Speed G96" must be selected before the cycle is called up.

e.g.,

```
N..    G96   S180   LF
N..    R20 ..   R21..   R22..   R24..  R25..  R26..   R27..   R29..    LF
N..    L95          F...  LF
N..    L91  LF
N..  T....  LF
```

Example:



The contour for which stock is to be removed is written in a sub-routine.

```
L70000
N1  G90  A170  A135.  X120.  Z30.  B30.  LF
N2                    Z20.             LF
N3             X210.  M17              LF
```

Parameter definition

A Starting point	X	50mm	R21 50
	Z	125mm	R22 125
Finishing cut	X	0.3mm	R24 .3
Finishing cut	Z	0.3mm	R25 .3
Stock removal depth		5mm	R26 5
Contour L7000			R 2070

Call-up in the part programme

N16 R20 70 R21 50. R22 125. R24 .3	
R25 .3 R26 5. R27 42 R29 11	LF R29 11 Roughing
L95 F...	LF
N17 L92	LF Retract to tool change
N18 T..	LF Tool change
N19 R24 2.5 R25 2.5 R27 42 R29 21	R29 21 finishing
L95 F...	LF R24 2.5
	R25 2.5 finishing
	depth 2.5 mm
N20 R24 .05 R25 .05 R27 42 L95	LF R24 .05
	R25 .05 finishing
	depth 0.050 mm
N21 R24 0. R25 0. R27.42 L95 F.. ..	LF R24 0
	R25 0 Finishing
	depth 0 = final contour

or roughing and finishing in a single call-up :

N15 R20 70 R21 50 R22 125 R24 .3	
R25.3 R26 5 R27 42 R29 41 LF	Paraxial roughing
N16 T.. L95 F... LF	1 cut parallel to the contour
	1 cut to the final contour

7.3.2 L950 Stock removal cycle - roughing parallel to contour

L950 is derived from L95.
Roughing passes are executed parallel to the contour during which the contour is projected onto point B.
As with L95 it is necessary to define the R parameter before calling up L950.
The R parameter corresponds to that with L95 with the exception of R29 and R26.

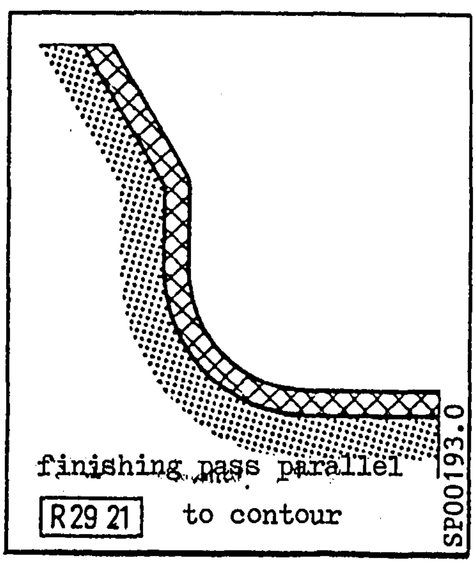
Type of machining (R29)

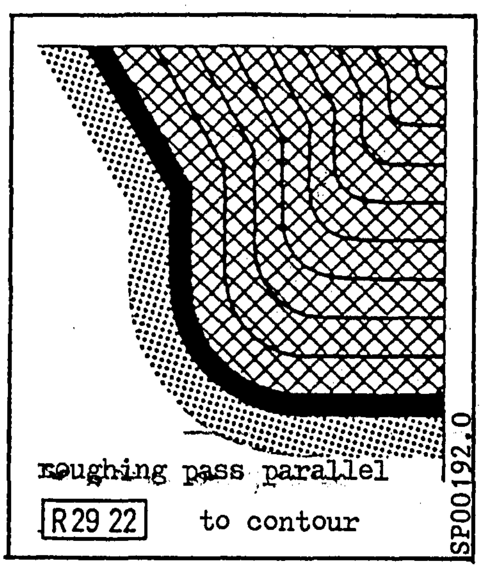
The following types of machining are permitted:

R29 21 outside *	finishing pass on final contour (R24 and R25 are ignored)
R29 22 outside	roughing to final pass depth
R29 23 inside *	finishing pass on final contour (R24 and R25 are ignored)
R29 24 inside	roughing till finishing cut

* In these cases the cutter radius comp. is activated if R27 (41 or 42) has been effected.

Examples:

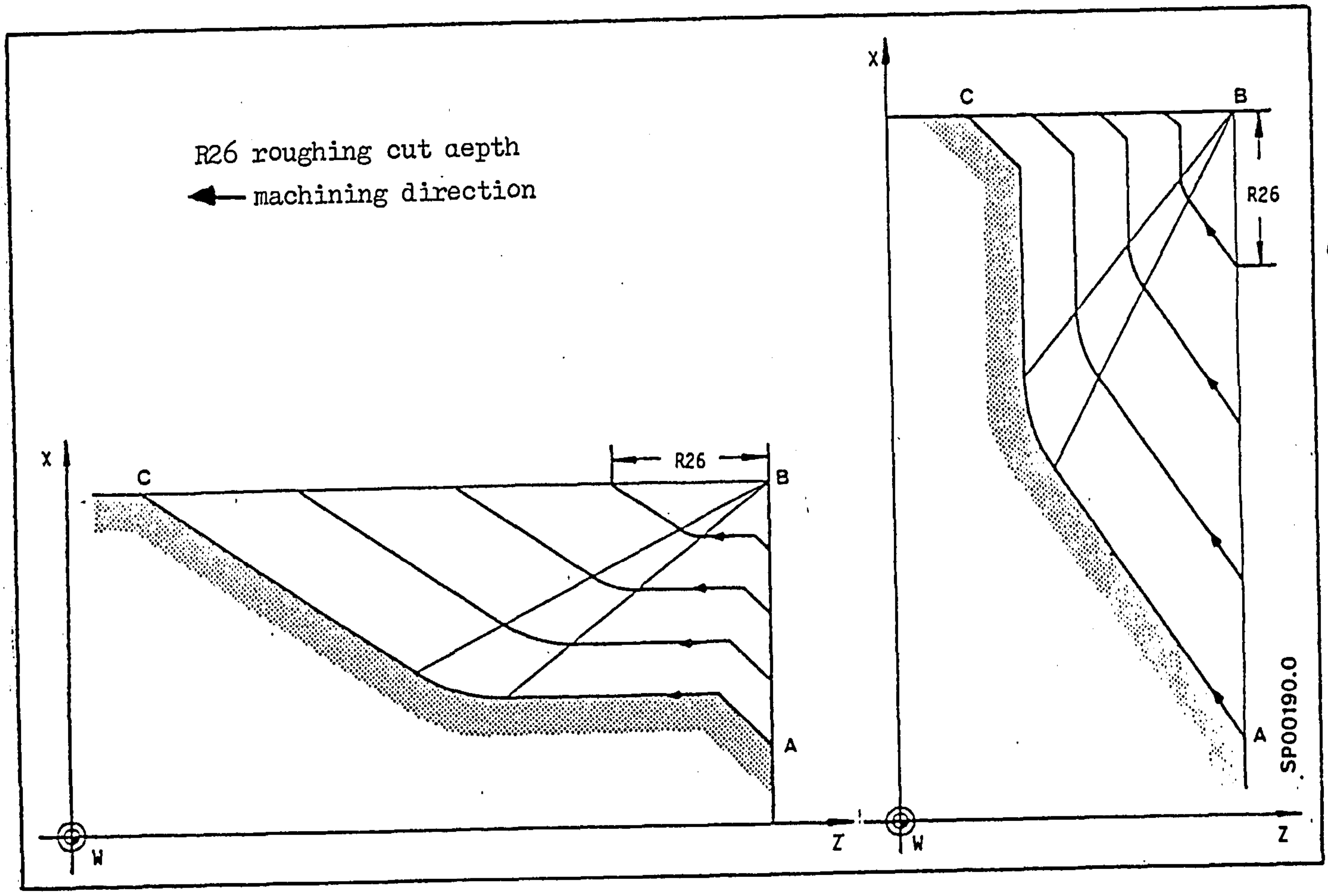




Roughing cut path (R26)

The control defines the roughing cut depth in the direction of the max. difference between the starting and finishing point of the contour.
The contour is always machined in the programmed sequence.

Examples :



Example

The contour to be cut is written in a subroutine.

L7000

N1	G90	A170.	A135.	X120.	Z30.	B30.	LF
N2				Z20.			LF
N3			X210.	M17			LF

M17 must be programmed in the last block of the subroutine (no separate block).

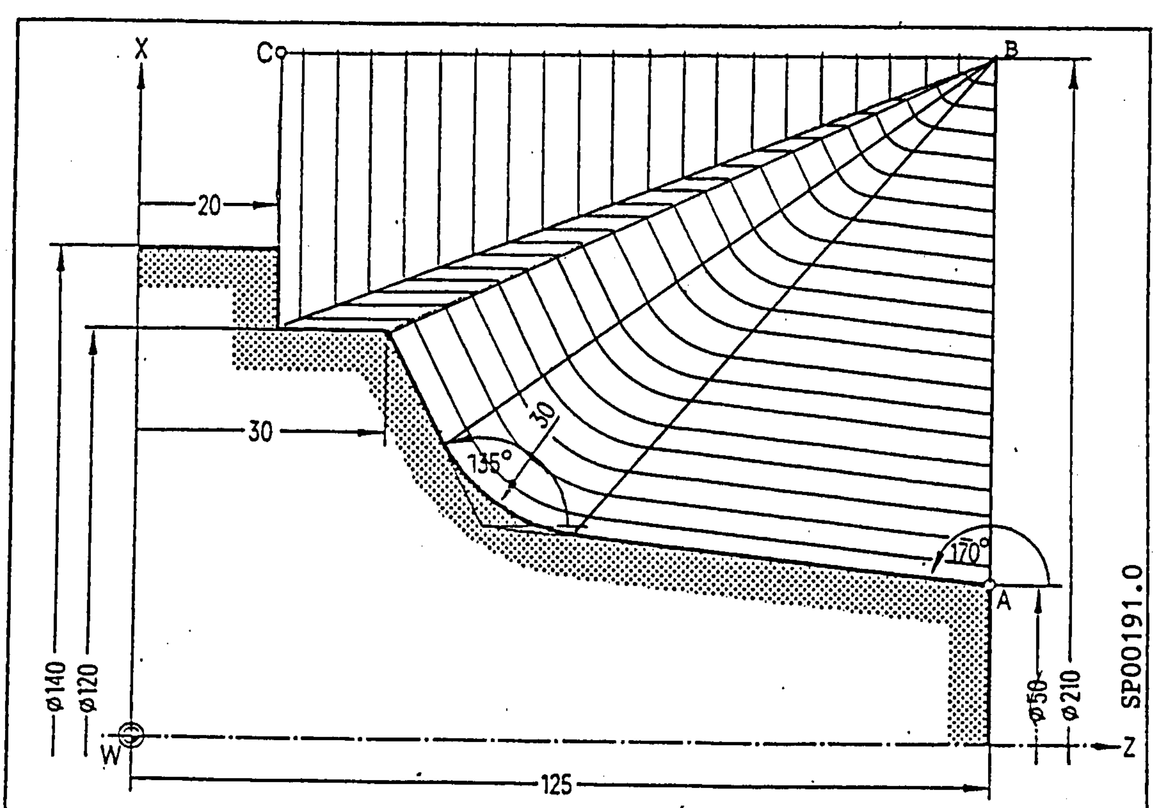
Parameter definition:

A starting point	X	50 mm	R21 50
	Z	125 mm	R22 125
Finish pass depth	X	0,3 mm	R24 .3
Finish pass depth	Z	0,3 mm	R25 .3
Depth of cut		5 mm	R26 5
Contour L7000			R20 70

Call-up in part program:

N20
N25 R20 70 R21 50 R22 125 R24.3 R25.3 R26 5 LF
N30 L950 F... LF

Example:



7.4 L 97 Thread cutting cycle

This cycle is used for cutting outside threads, inside threads and taper threads.
The tool infeed is automatic and takes the form of a diminishing quadratic with the result that the stock removal rate remains constant.
Before calling up cycle L97 a value must be assigned to the following R parameters:

R21	Starting point for the thread in X (absolute)
R22	Starting point for the thread in Z (absolute)
R23	Number of compound feeds
R24	Thread depth (incremental), sign required to define inside or outside thread, + = inside thread, - = outside thread.
R25	Finishing cut depth
R26	Approach path
R27	Run out path
R28	Number of roughing cuts
R29	Infeed angle
R31	Endpoint of thread in X, (absolute).
R32	End point of thread in Z (absolute)

Only 10 R parameters per block are permitted so that two blocks are required for the value.

Example:

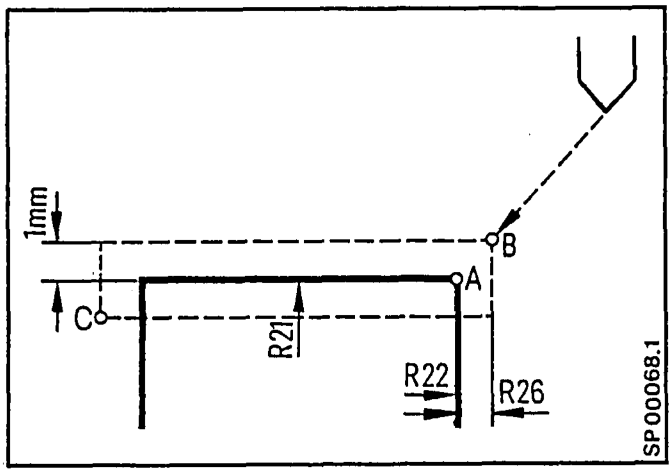
```
N10 R21 ... R22 ... R23 ... R24 ... R26 ... R27 ...  LF
N15 R28 ... R29 ... R31 ... R32 ... L97 LF
```

The individual parameter values are represented in the following sketches.

R20 Thread pitch

The parameter represents the value of the thread pitch. It is always written as a paraxial value without sign..
min 0.001 mm
max. 2000 mm

R21 and R22 Start Point for thread



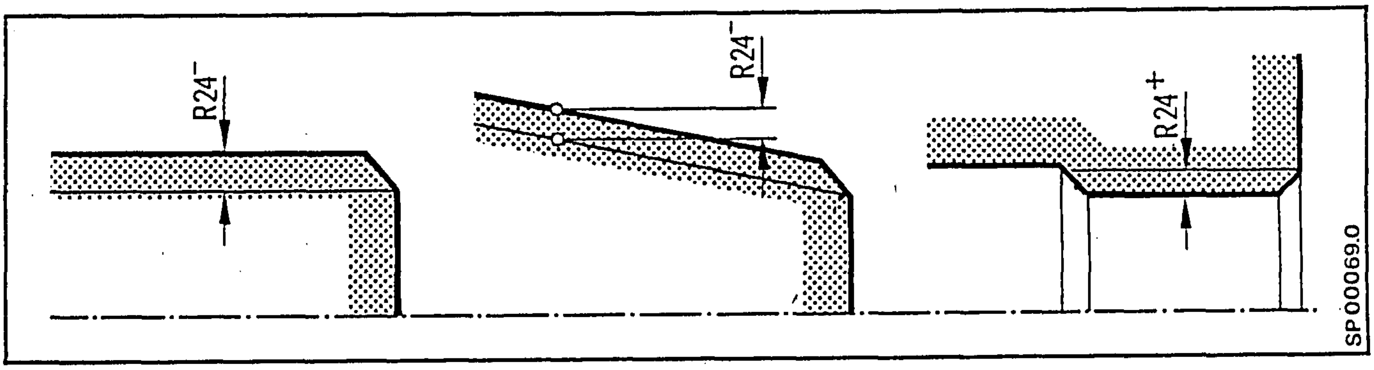
Parameters R21 and R22 define the original starting point for the thread (A). The starting point for the thread cycle is B which is located in front of the thread starting point at a distance given by parameter R26 (approach path). In the X axis the starting point B is located 1mm above the parameter value R21. This raised plane is generated automatically in the control.
The thread cutting cycle can be called up independantly of the tool position and infeed to point B is effected at rapid traverse rate.

R23 Compound feeds

Any number of compound feeds can be selected. They are entered using parameter R23.
e.g., 3 compound feeds R23 3.

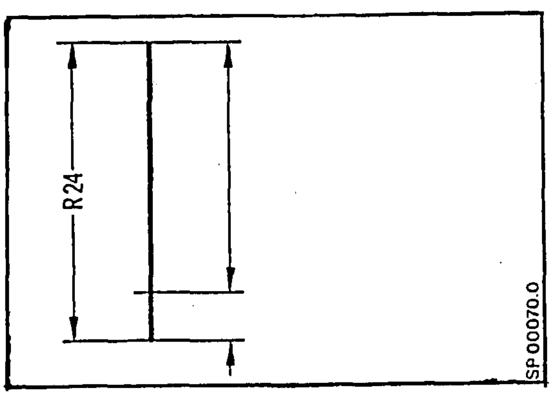
R24 Thread Depth

The depth of thread is entered using parameter R24.. and the sign determines the infeed direction, i.e., whether it is an outside or inside thread. (+ inside thread, - outside thread).



R25 Finish cut depth

R25 gives the finish cut depth. When a finish cut depth is programmed, this depth is subtracted from the thread depth and the remaining value divided into roughing cuts. After the roughing cuts have been completed a finishing cut is made followed by a certain number of cleaning cuts programmed under R23.



The roughing depth is automatically calculated and divided into roughing cuts.

Finish cut depth R25.

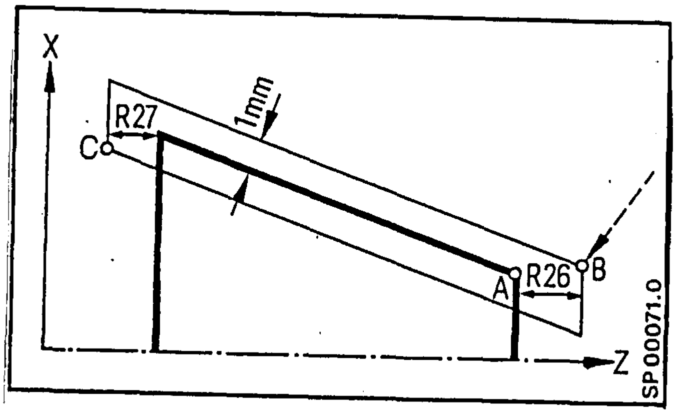
R26 Approach path

R27 Run-out path

The approach and run-out paths are programmed without sign.

The parameters represent paraxial, incremental values.

In the case of taper threads the control calculates the approach and run-out paths with regard to the taper and determines the corner points B and C.



R28 Number of roughing cuts.

The number of thread roughing cuts is determined by the parameter value. The control automatically calculates the individual infeed depths for constant stock removal. This ensures that the cutting pressure remains constant throughout the roughing cuts.

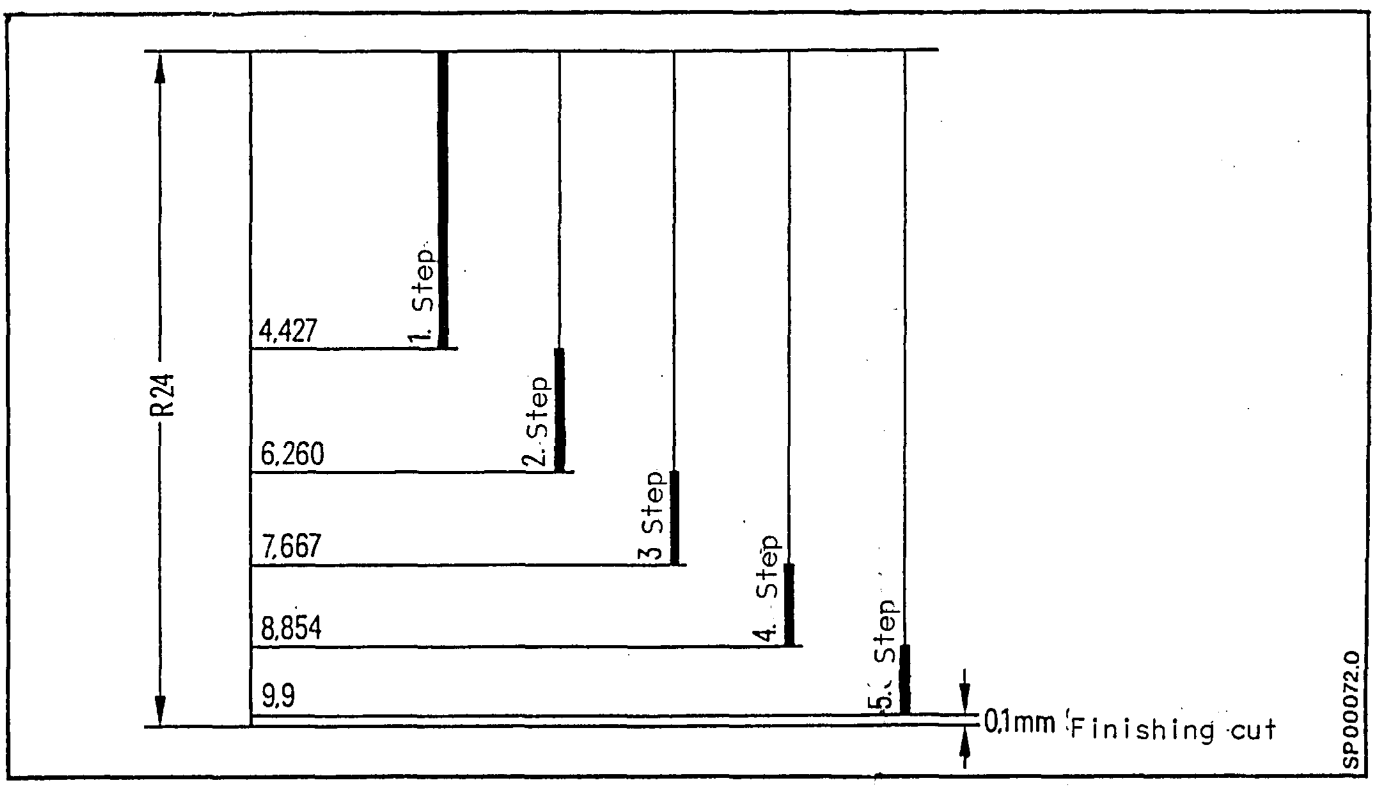
The desired cutting depth Δt is calculated using the following formula.

$$\Delta t = \frac{t}{\sqrt{R28}} \cdot \sqrt{i}$$

t = R24 - R25
i = actual cut

Example:

Thread depth t = 10
Number of roughing cuts = 5
Finishing cut depth = 0.1

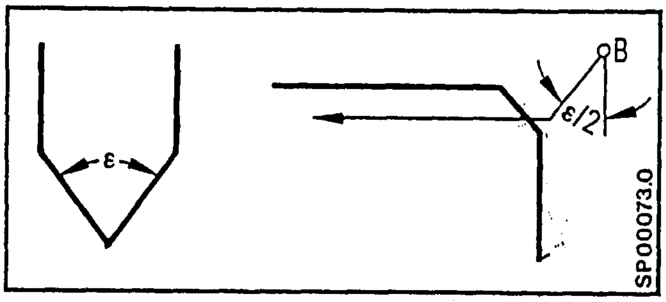


R29 Infeed angle for longitudinal or facing leads

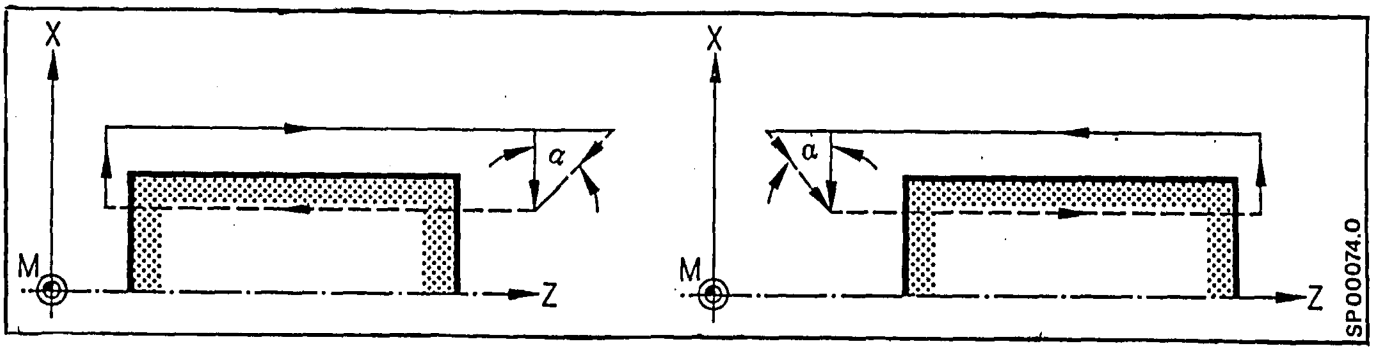
For longitudinal or facing threads the cutter approach can be made at any angle. For conical leads oblique feed-in is not possible.

No sign is required with the angle data

e.g., $\varepsilon/2$ = 30° R29 30



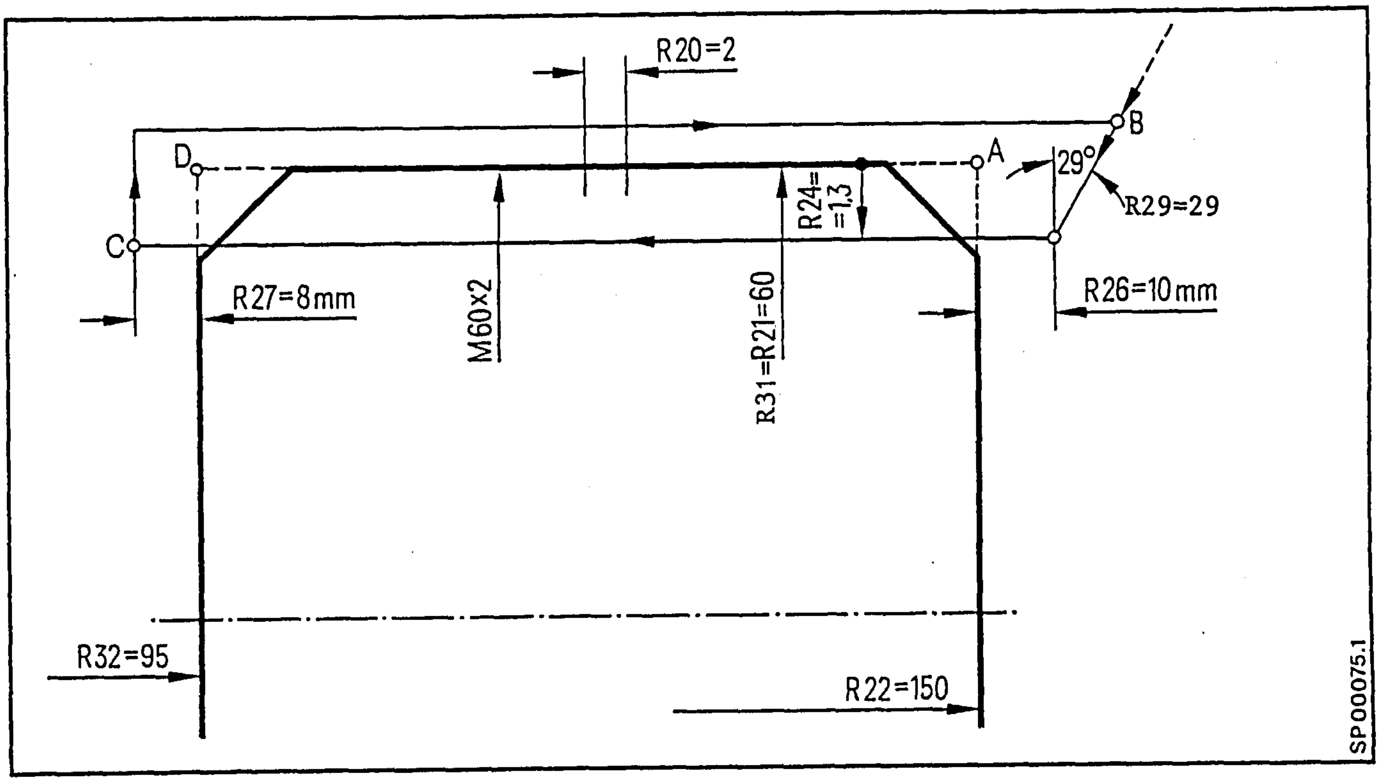
In the cycle the angle is written in accordance with the machining directions.



R31 and R32 Thread end point

The original end point of the thread is represented by the parameters R31 and R32 (see page 7-28 point D).

Example : External lead



End point X R31 = 60 mm = R21

End point Z R32 = 95 mm

Number of clearing cuts R23 = 2

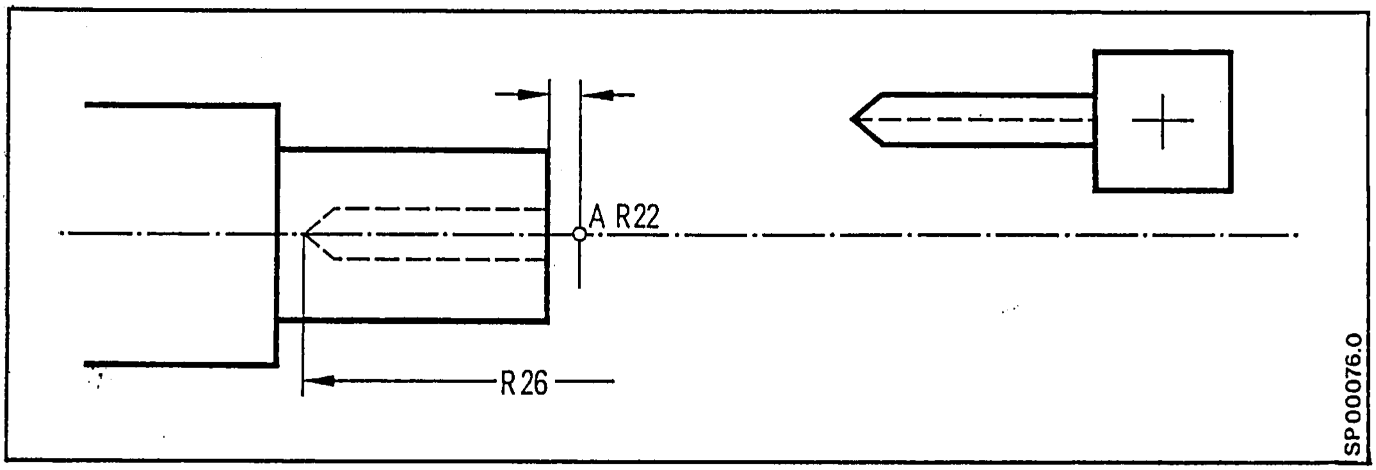
Finishing cut depth R25 = 0

Number of roughing cuts R28 = 7

Infeed angle R29 = 29°

```
.
.
N130  T5   M15                                        LF

N140  R20 2.  R21 60. R22 150. R23 2. R24 -1.3
      R25 0.  R26 10. R27 8.   R28 7. R29 29          LF
N150  R31 60. R32 95. L97                             LF
N160  G00 X...    Z...                                LF
```

7.5 L98 Deep hole drilling cycle



A value must be assigned to the following parameters before calling up cycle L98.

R22 Start point in Z direction, values entered as absolute.

R24 Degression magnitude is programmed incrementally without sign

R25 The first drilling depth is programmed as incremental without sign.

R26 Final drilling depth (absolute)

R27 Dwell time at start point (for chip removal)

R28 Dwell time at bottom of drill hole (chip breaking).

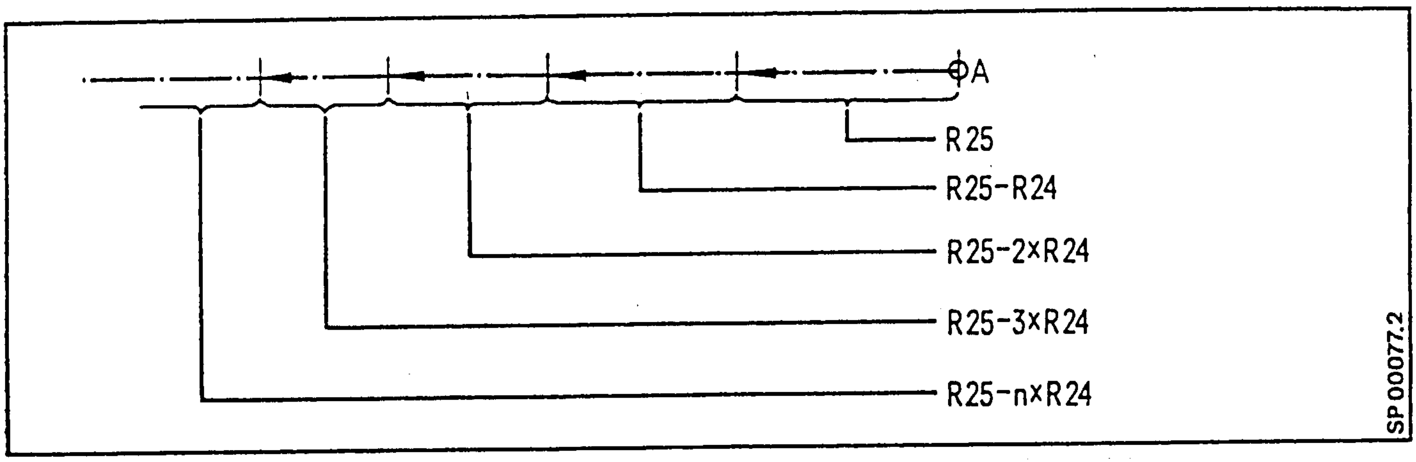
R26 Final Drill Depth

The drill depth is reduced by a constant diminishing amount until the end point R26 is reached.

If a particular drill depth is theoretically less than the diminishing value, it is maintained at this magnitude.

If the remaining infeed depth is less than twice the degression magnitude, the remaining amount is halved and the last two infeeds are executed with this new value. This ensures that the last infeed is not executed with a value which is too low. This calculation always results in a minimum infeed of half the degression magnitude.

At the end of the cycle, the drill point is located at start point A.

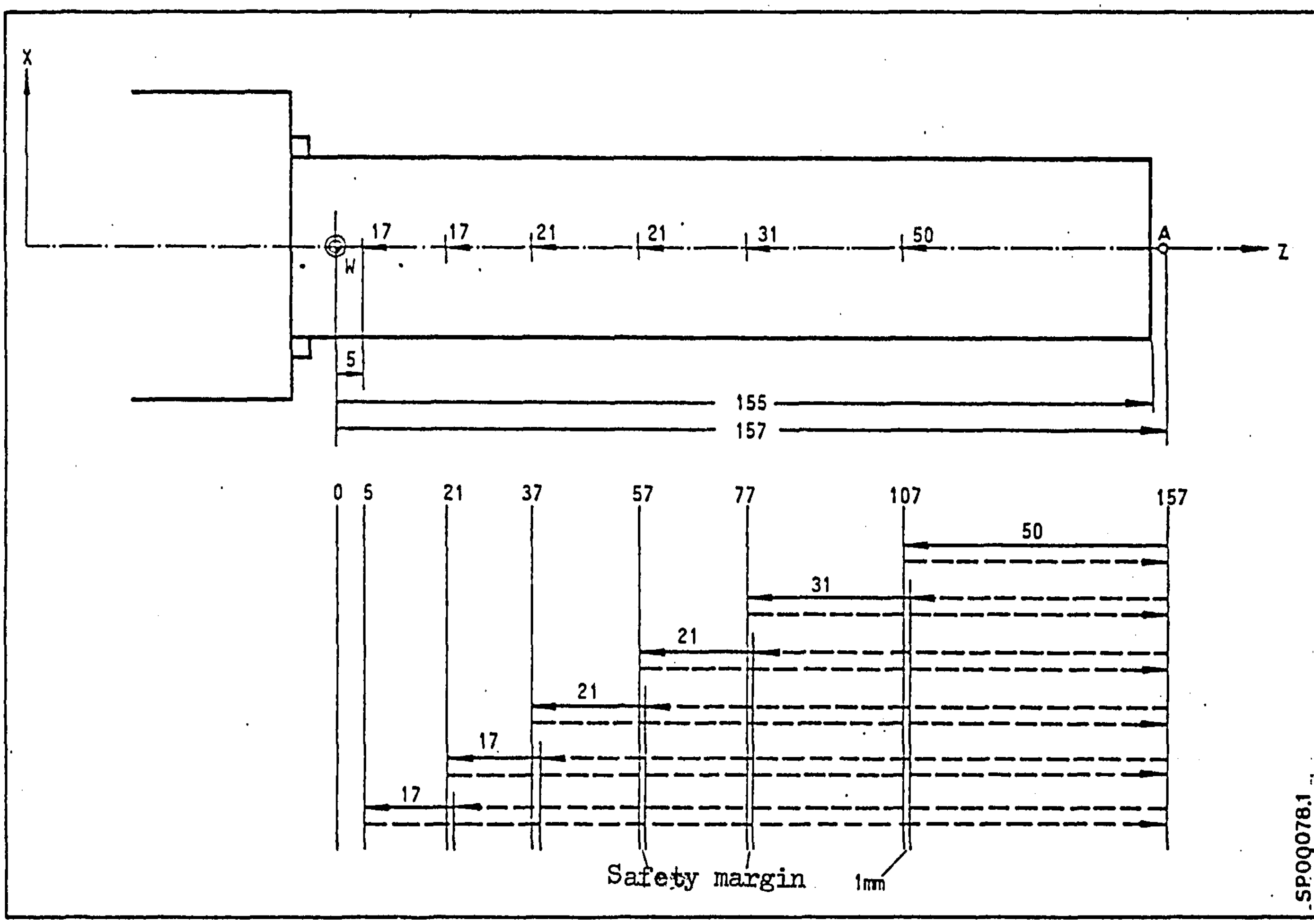


Example:

Start point	Z = 157 mm	R22 157
Diminishing value	20 mm	R24 20
1st drill depth	50 mm	R25 50
Final drill depth	5 mm	R26 5
Dwell time at start point	2s	R27 2
Dwell time at drill depth	1s	R28 1

Call up in program:

```
N...  R22. 157. R24 20. R25 50. R26 5. R27 2. R28 1. LF
N...  L98       F...  S...  M...                  LF
```



—— Feedrate

– – – Rapid traverse

7.6 L999 de-activation of read-in of NC blocks

A series of functions from the operator's panel or interface control is not directly registered in the active store of the NC but indirectly via buffer store. To these functions belong the following:

- mirror image
- R parameter input
- external additive zero offset
- external zero offset
- external tool offset
- clear text remarks
  for the operator after programmed stop M00
- block delete after programme stop M00

If these functions which are actuated in the active program are to be effective in the block following their selection, the block buffer stores must be emptied. Alternatively the selected control signal only becomes active a few blocks later.

In each program the buffer store can be emptied by a single call-up of the subroutine L999. The subroutine L999 must be defined as follows:

```
L99900 LF
@ 31M17 LF
```

The control registers the status "Buffer store empty" in the interface control and the selected control signal or the required external data input can be enabled.

Example 1:

Activation of external tool offset, e.g. after a measurement of the tool

N15 M ...	Read-in activation of external tool offset
N20 L999	Empty buffer store (no further calculations are carried out before execution of block 15)
N25	The new tool offset is calculated.

Example 2:

Clear text remarks for the operator after M00

N ... M00 L999 LF

(OPERATING NOTE) note can be read in PP picture

N ...

8.0 Appendix

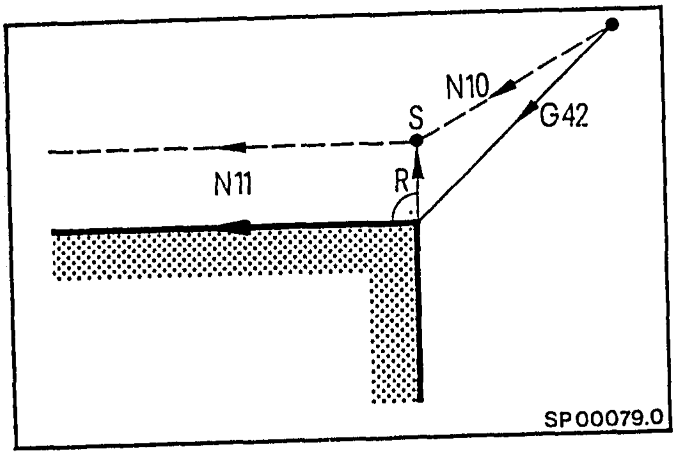
8.1 Cutter radius compensation (CRC)

In the following all the stop points are indicated by the character S.

In all figures the cutter is shown as starting from position 9.

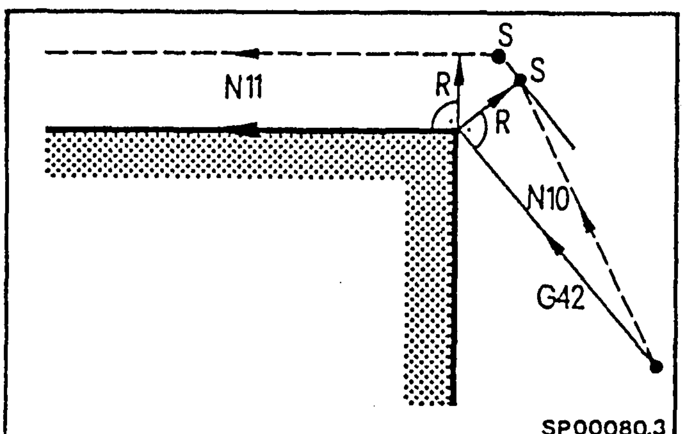
8.1.1 Activation of CRC

- for inside contours (angle between block N10 and N11 < 180°)



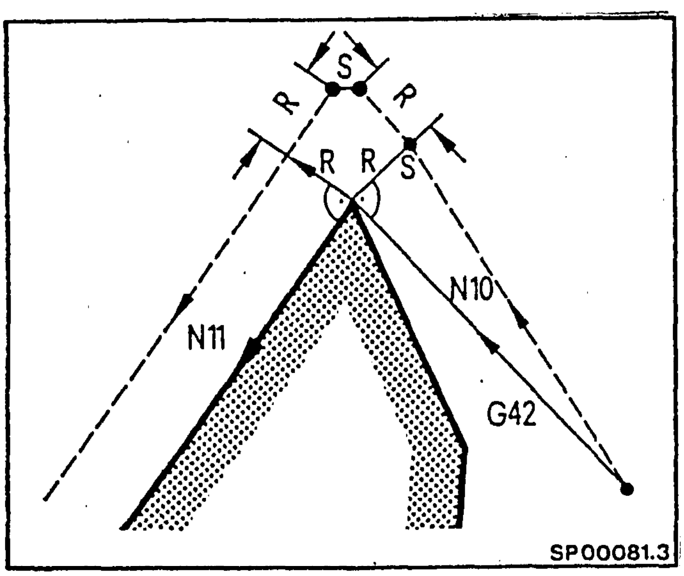
Start Point.
In the block following activation block start vector (length R) is constructed perpendicular to the programmed path.
Block start vector.

- for outside contours (angle between block N10 and N11, 180° to 270°)



Start point

(Angle between block N10 and N11 > 270°)

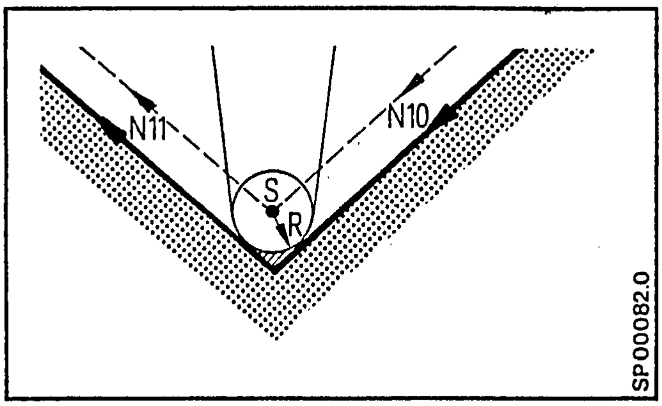


Start point

8.1.2 CRC in the programme

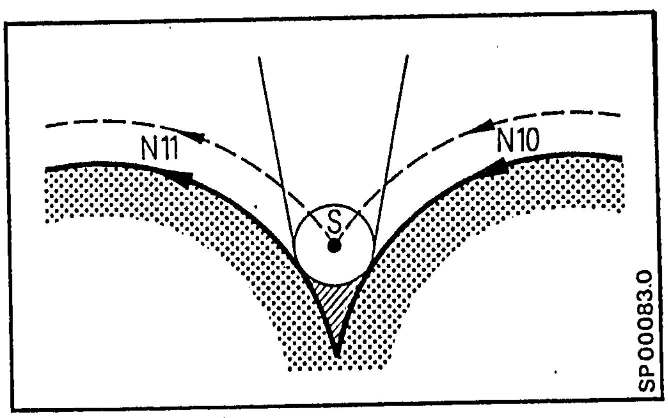
- inside contour (angle between 2 blocks < 180°)

Straight line - straight line



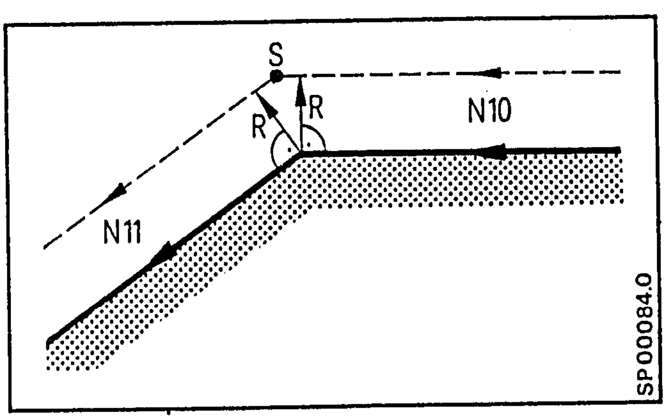
In each case the cutting point of the corrected path is calculated

Circle - Circle



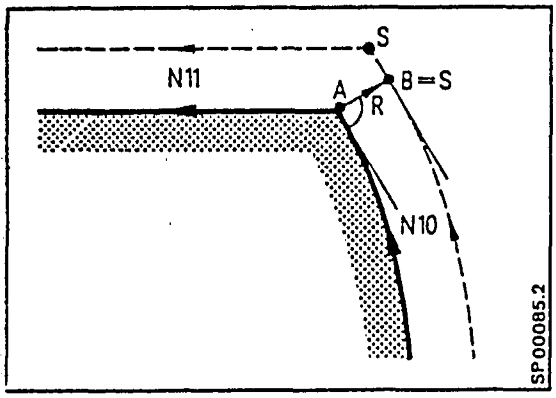
- outside contour (angle between 2 blocks 180° to 270°)

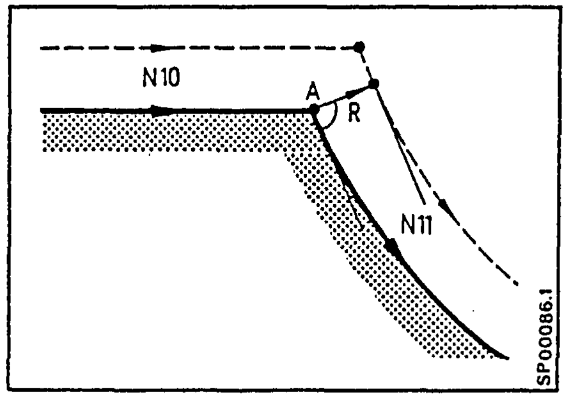
Straight line - straight line



The cutting point of the corrected path is calculated.

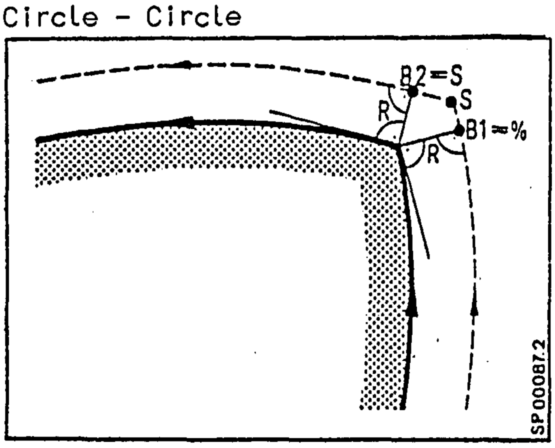
Straight line - circle





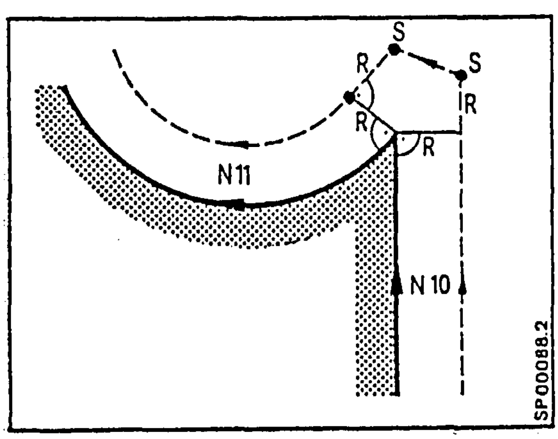
A vector (length R) is constructed perpendicular to the end point of the circle (or start point) A. The cutting point is calculated between the <u>tangent</u> at point B and the corrected path in block N11 (or N10).

Circle - Circle



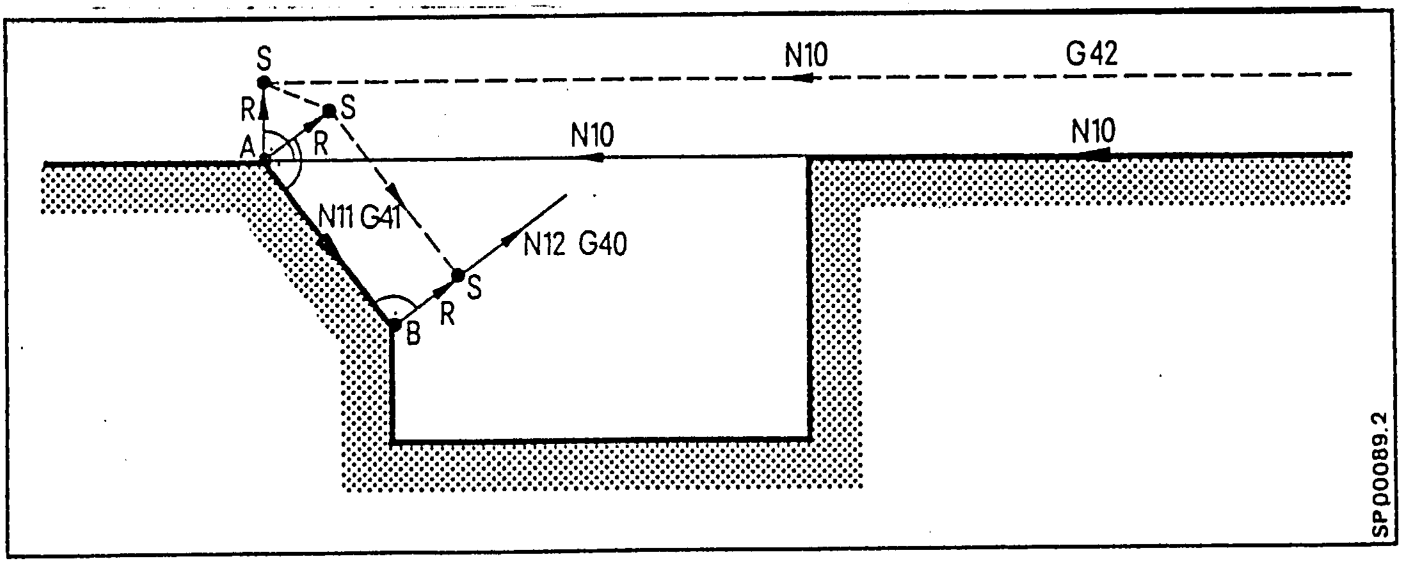
A vector of length R is constructed at right angles to the block end point or start point.
The cutting point is calculated between the tangents and point B1 and B2.

Straight line - circle arc > 270°



A vector of length R is constructed at right angles to the end point or start point of block N10 or N11.
A compensating traverse is made beyond the programmed contour by the cutter radius R to prevent damage to the workpiece.

Change of compensation direction



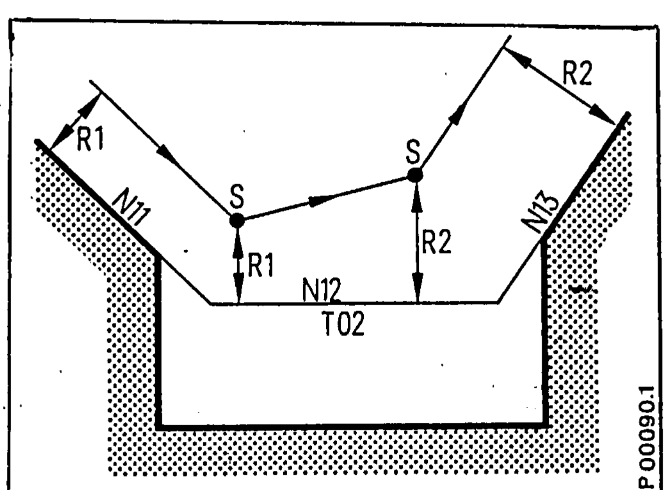
A perpendicular vector of length R is constructed at the end point of the block with the old compensation direction (here G42) and at the start point of the block with the new compensation direction (here G41).

A small chamfer is made at point A because at this point no cutter point calculation is made.

In order to obtain exact machining at point B in this example, it would be necessary to de-activate in block N12 using G40.

Change of tool offset number (T..., T...)

The following applies when the tool offset is changed :

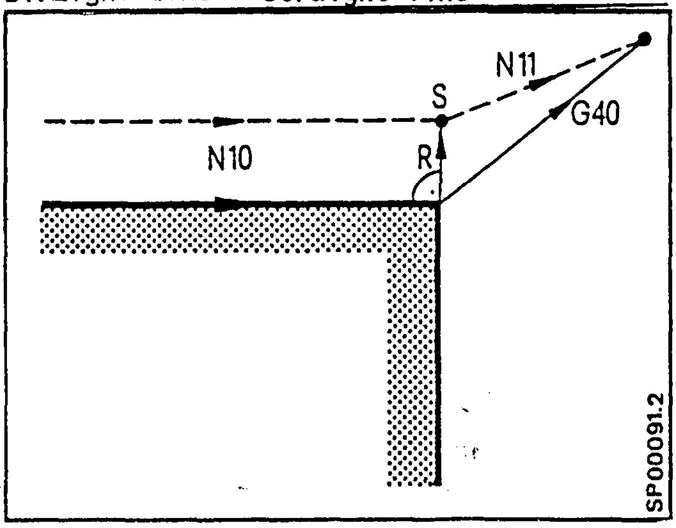


1. The cutter point at the start of the block is calculated using the old tool offset.
2. The cutter point at the end of the block is calculated using the new tool offset.

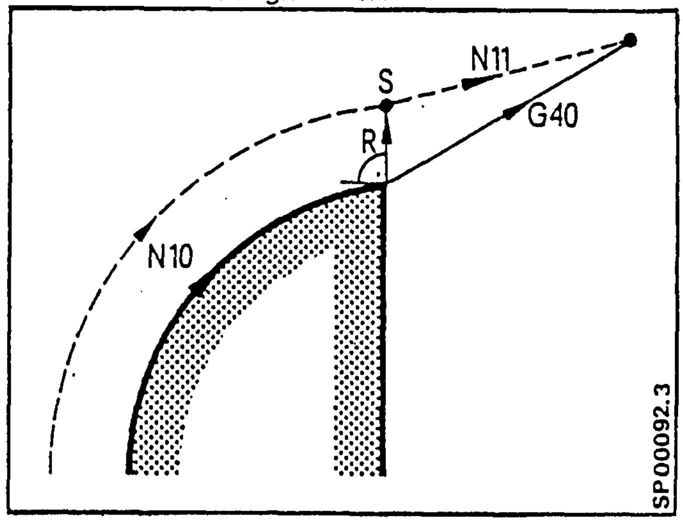
8.1.3 Deactivation of CRC

- for inside contours (angle between block N10 and N11 $< 180^{o}$)

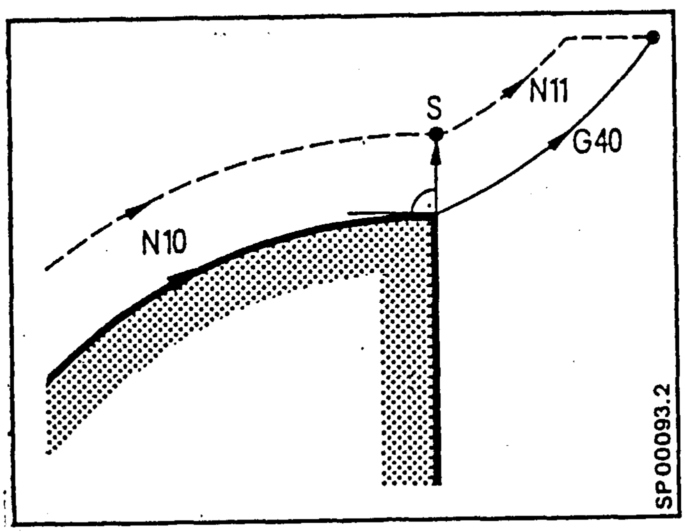
Straight line - straight line



Circle - straight line



Circle - Circle

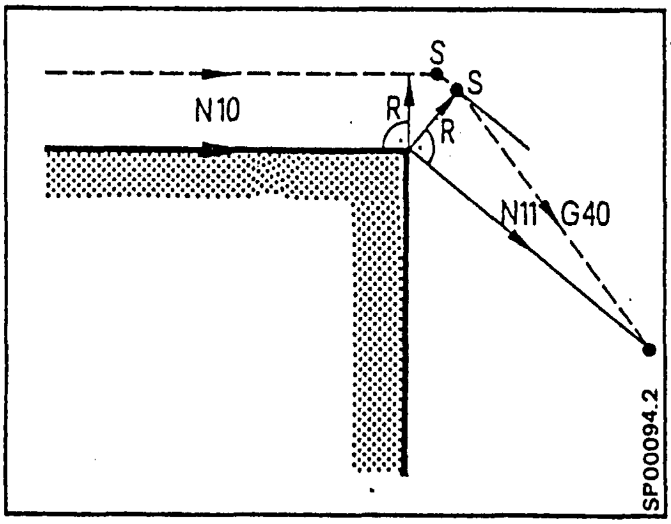


In the last block with CRC a vector of length R is constructed at right angles to the programmed path.

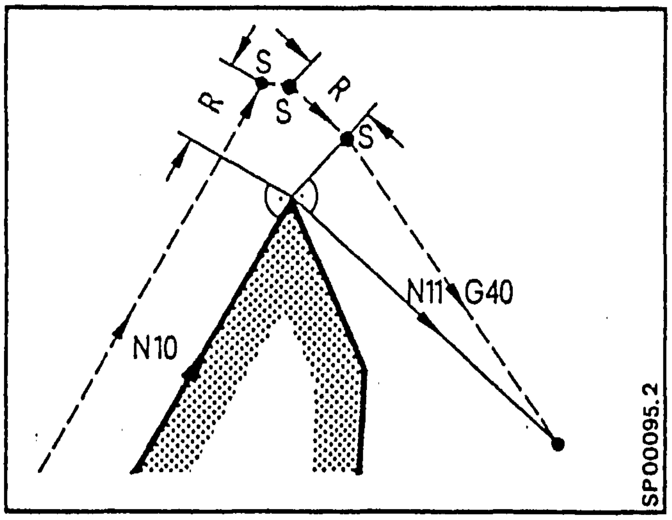
During transition to a straight line the programmed end point is approached directly. During the transition to a circle the programmed circle is approached with offsets until the perpendicular to the circle end point is reached. The approach is then made along the perpendicular to the end point.

If an end point co-ordinate (X or Z) is reached before the perpendiculars, the other co-ordinate is approached directly (see diagram above).

- for outside contours (angle between block N 10 and N 11 , 180° to 270°)



(angle between block N10 and N11 > 270°)



The corrected path is calculated and traversed up to the cutting point at the start of the block in which the CRC is de-activated.

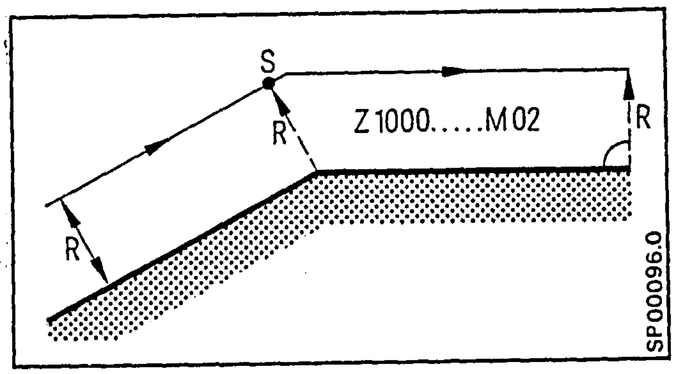
G40: De-activation of cutter radius compensation (CRC)
T00: De-activation of cutter radius comp. and tool length comp.

The CRC must be de-activated in the last block at the latest.

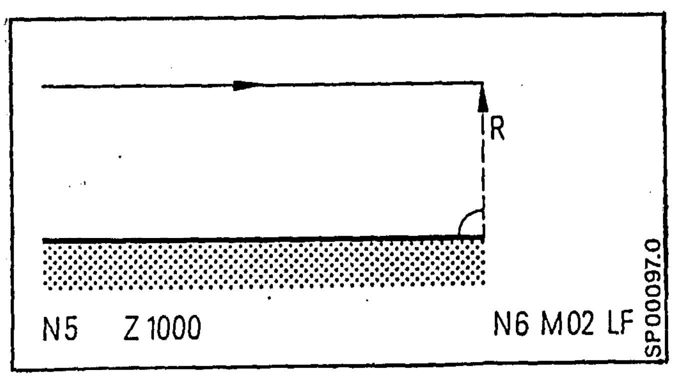
8.1.4 M00, M01, M02 and M30 with CRC activated

M00, M01 : The NC stops at point S for single block (the points are shown in the diagrams.)

M02, M30 : - with at least one active axis.



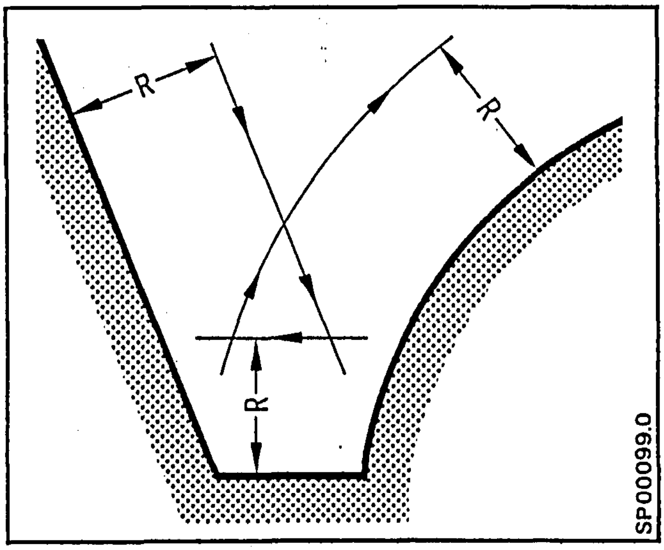
- without movement



In neither case is the CRC traversed.

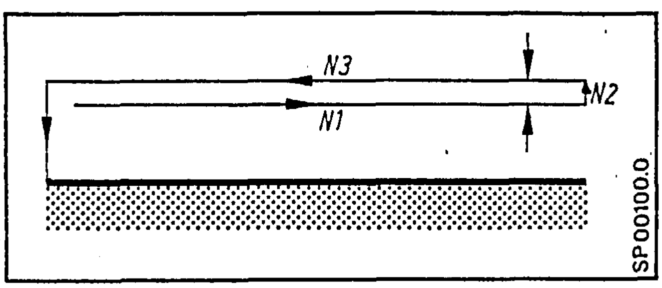
8.1.5 Special cases with CRC

Since the NC only uses the next block for calculation of the cutting point, the following contour errors can result with inside contours:



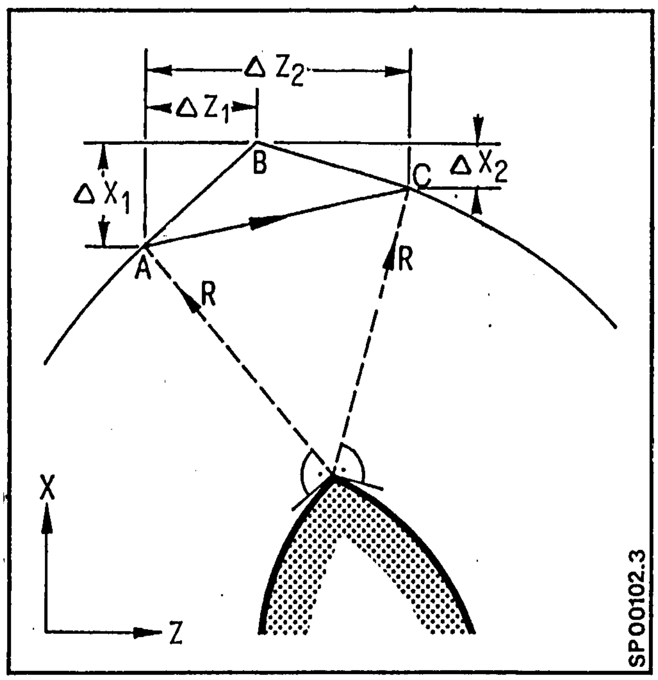
Intermediate block for the activated compensation is too small. Machining is not interrupted. An alarm (506) is displayed and cancelled at the end of the programme.

The compensation direction for the CRC is maintained and the traverse direction reversed.



The pull out path in N2 must be greater than twice the cutter radius (otherwise the tool would execute the movement in the wrong direction).

- outside contour and acute angle

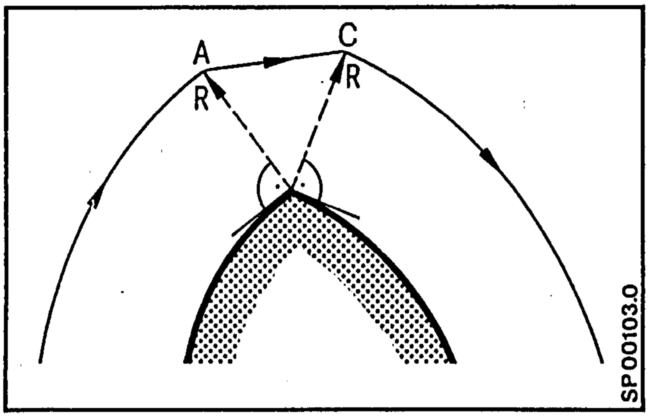


In order to prevent a conditional hold in continuous path operation resulting from intermediate blocks which are too small, the path followed is dependent on a tolerance d determined during commissioning (max. 32000 µm),and will be as follows:

$|\Delta x_1| + |\Delta z_1| \geq d$

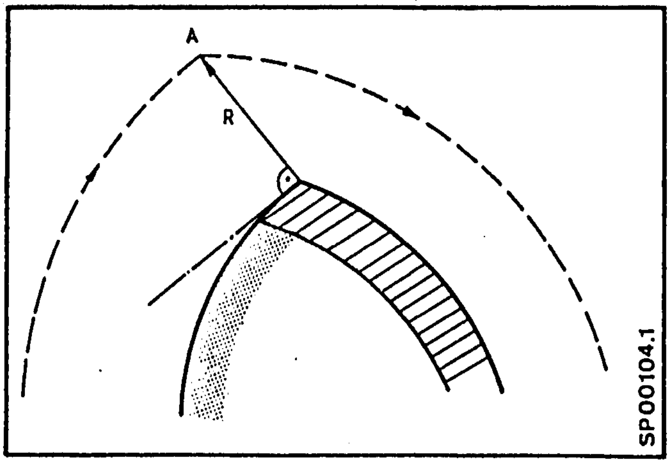
$|\Delta x_1| + |\Delta x_2| + |\Delta z_1| + |\Delta z_2| \geq d$

Travel will be A, B, C.



$|\Delta x_1| + |\Delta z_1| < d$

Travel will be direct from A to C



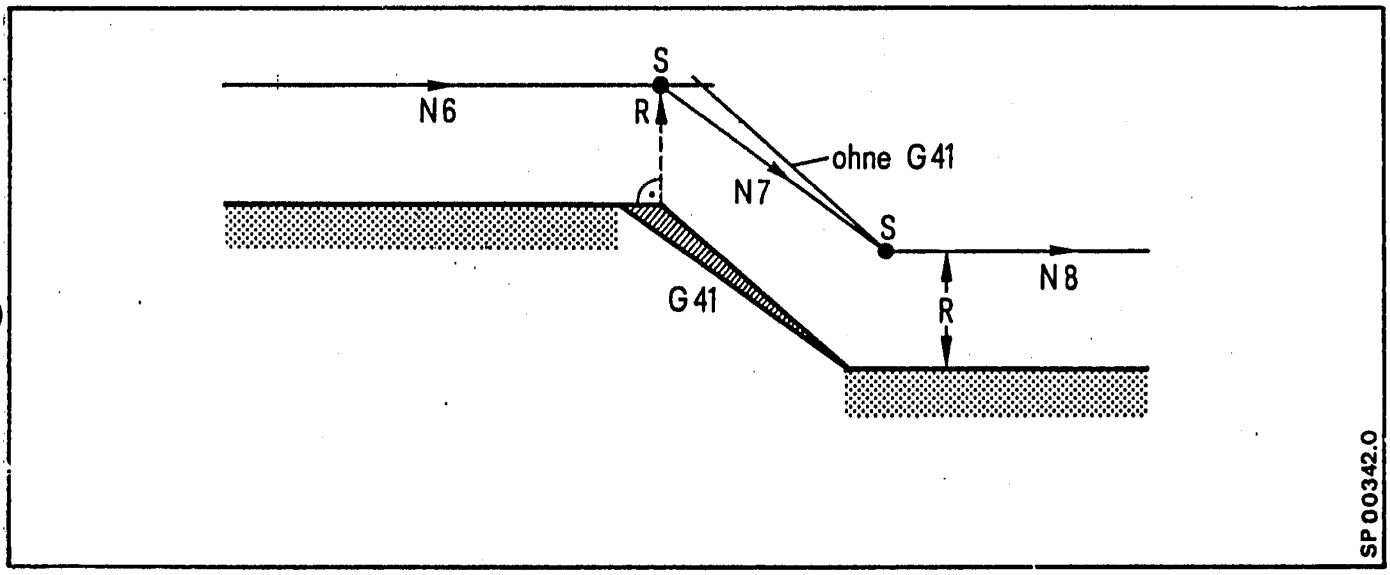
$|\Delta x_1| + |\Delta x_2| + |\Delta z_1| + |\Delta z_2| < d$

Does not cause a compensated motion. From point A an arc will be described around the programmed middle point.

/// Contour error.

8.1.6 Repetition of previously activated G function (G41, G42) with the same offset number

If a previously programmed G41, G42 is repeated, a vector with the length R perpendicular to the programmed path is constructed at the starting point of the following block.



N6	G41 G91	Z....	T1313	LF	
N7	G41	X....	Z....	LF	- Error: G41 repeated!
N8		Z....		LF	

8.1.7 CRC for combination of several block types

Type: Paths in CRC plane

Example: N.. G91 X100 LF

Paths = 0

Path addresses have been programmed, but there will be no movements, since the path equals 0.

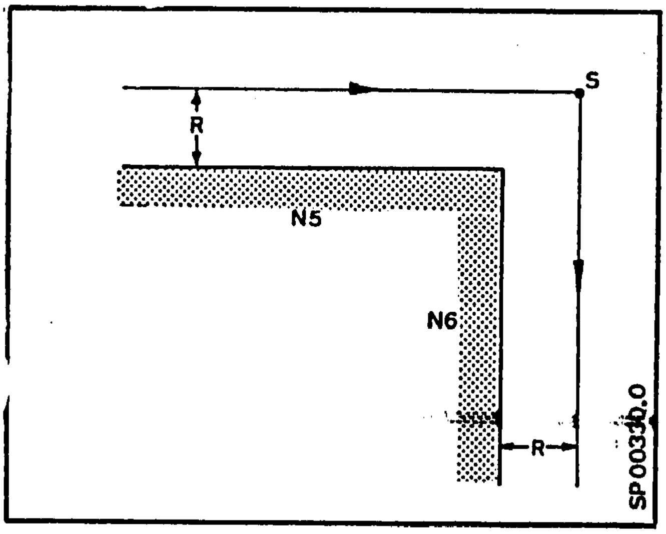
Example: N.. G91 X0 LF

Block without path addresses (block of auxiliary functions)

No path addresses have been programmed, but only auxiliary functions, dwell, setting functions, subroutine definitions, subroutine end alone in the block

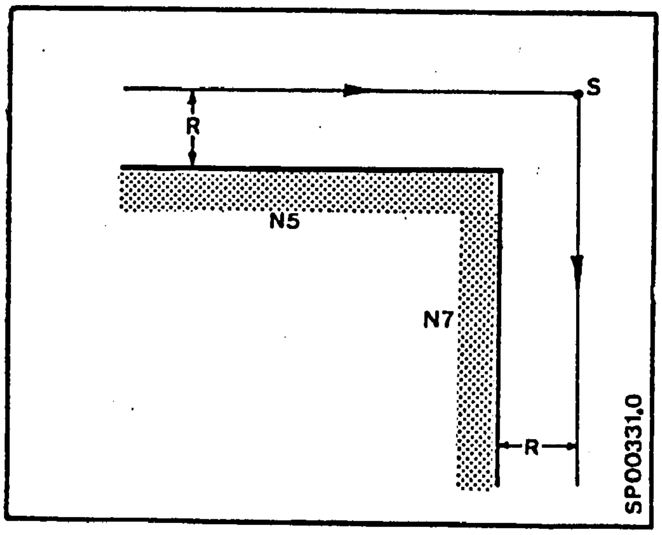
Example:	N... X100		LF
	N... M08		LF
	N... G04	X10	LF
	N... T0101		LF

Two paths



N5 G91 Z100 LF
N6 X-100 LF

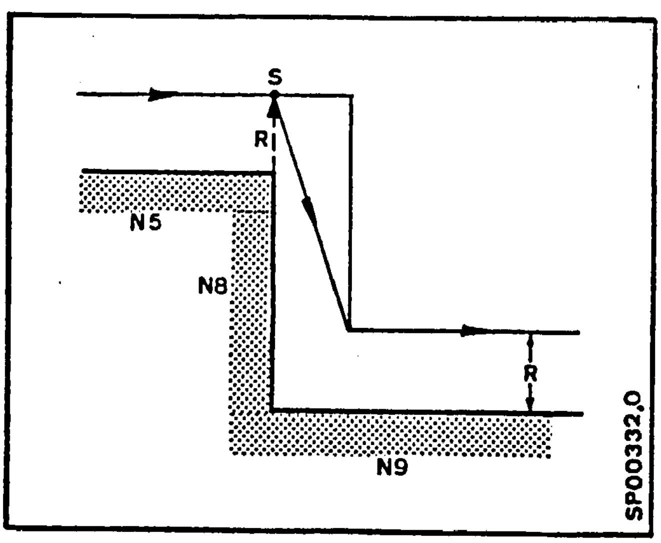
One "auxiliary function block" between paths



N5 G91 Z100 LF
N6 M08 LF
N7 X-100 LF

Block N6 is executed at point S.

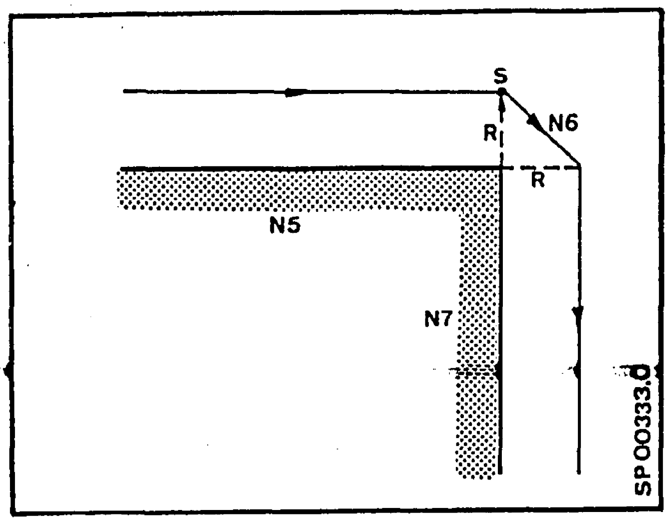
Two "auxiliary function blocks" between paths



N5 G91 Z100 LF
N6 M08 LF
N7 M09 LF
N8 X-100 LF
N9 Z100 LF

The blocks N6 and N7 are executed at point S. There is a contour error, except at tangential passages.

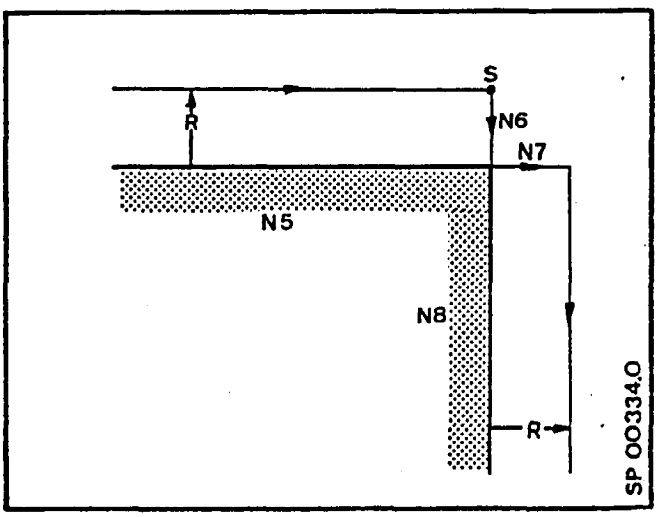
One block "path = 0" between paths



N5	G91	Z100	LF
N6		Z0	LF
N7		X-100	LF

There is a contour error, except at tangential passages.

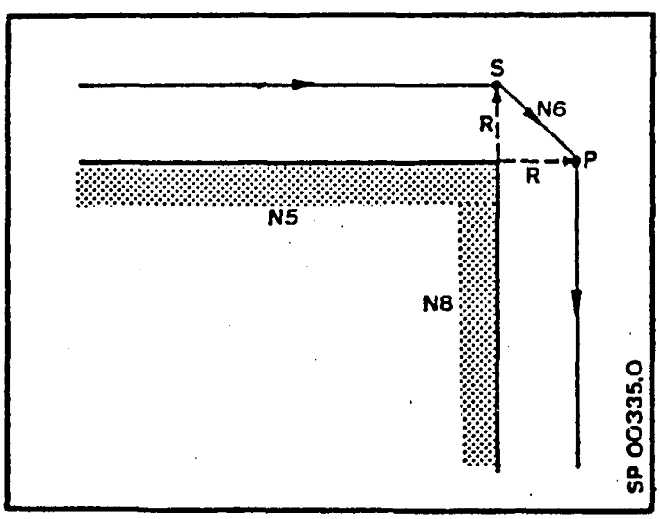
Two blocks "path = 0" between paths



N5	G91	Z100	LF
N6		Z0	LF
N7		Z0	LF
N8		X-100	LF

There is a contour error, except at tangential passages.

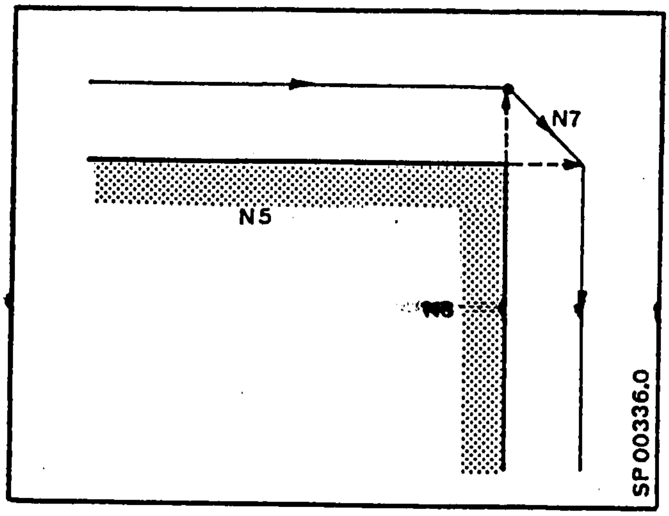
One block "path = 0" and one "auxiliary function block" between paths



N5	G91	Z100	LF
N6		Z0	LF
N7	M08		LF
N8		X-100	LF

The block is executed at point P. There is a contour error, except at tangential passages.

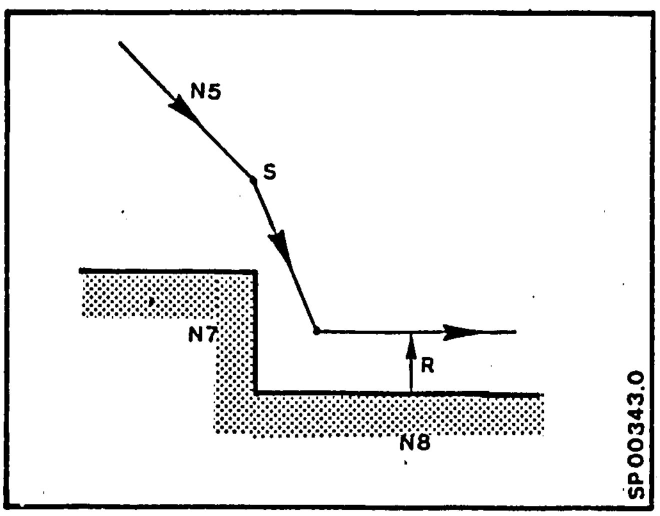
One "auxiliary function block" and one block "path = 0" between paths




```
N5   G91   Z100    LF
N6   M08           LF
N7         Z0      LF
N8         X-100   LF
```

The block is executed at point S. There is a contour error, except at tangential passages.

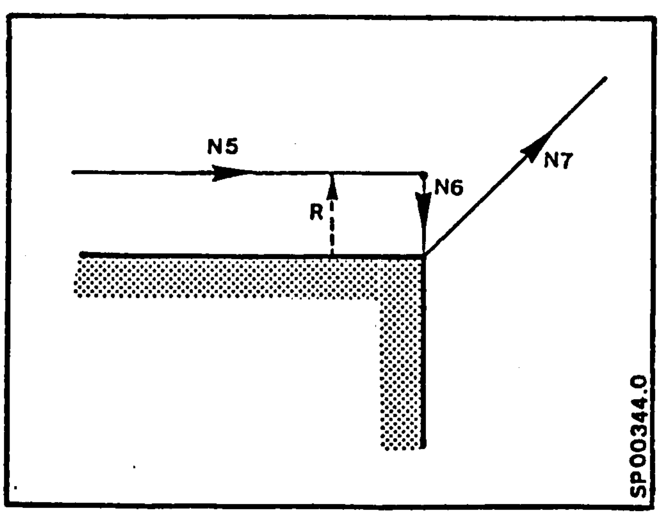
Activation of CRC in a block "path = 0"




```
N5   G91   X-100          LF
N6   G41   T0101   Z0.    LF
N7   X-100.               LF
N8   Z100.
```

There is a contour error!

De-activation of CRC in a block "path = 0"




```
N5   G91    Z100         LF
N6   G40    Z0           LF
N7   Z100   X+100        LF
```

There may be a contour error!

8.2 Input systems, diagrams and tables

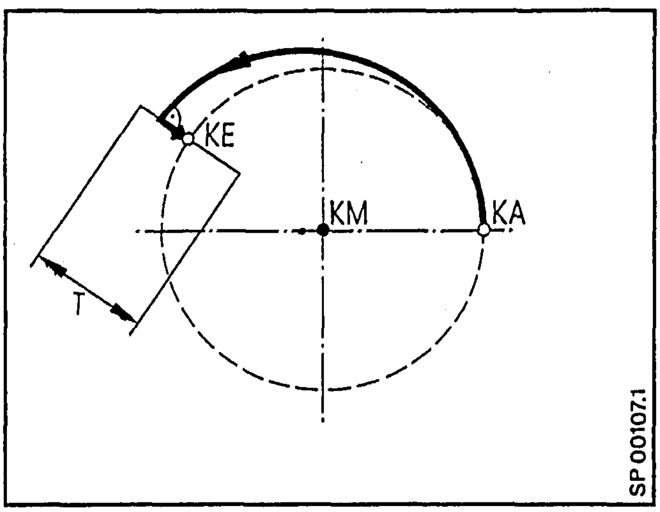
8.2.1 Inexact input of interpolation parameter or radius

The circle end point is monitored and any programming error (assuming it is outside the tolerance range) is recognised. Alarm 308 is displayed and circular interpolation cannot start.

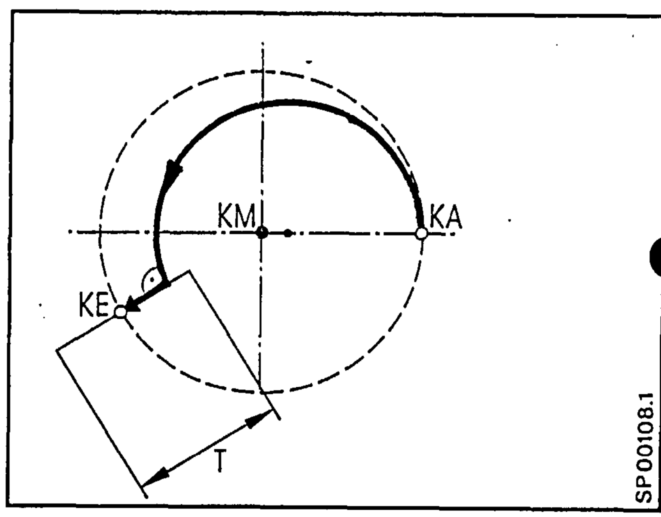
When the programming error lies within the tolerance range, the traverse is made exactly to the end point of the arc: but then the path between start and end point is as follows:

Interpolation parameter or radius

too large

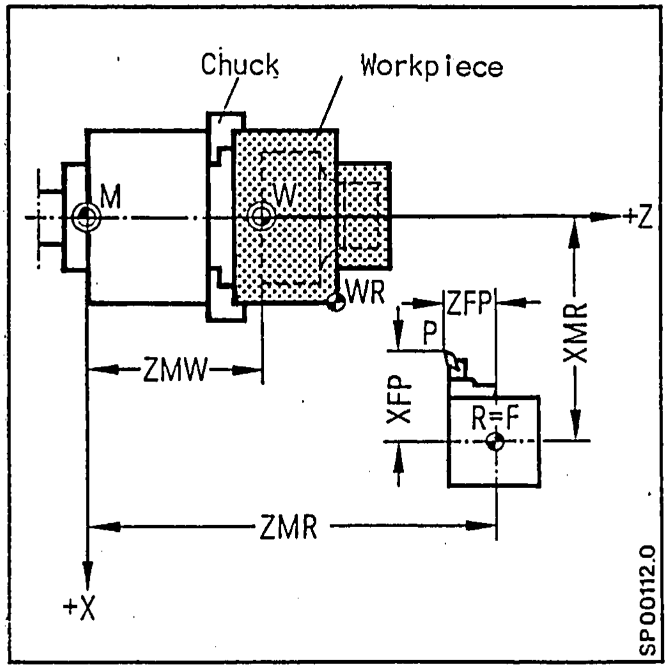


too small



The setting range (machine datum) for the tolerance "T" around the circle end point "KE" is ± 1 micron to ± 32000 microns This monitoring of the circle end point can be supressed by using a large value. The tolerance range is entered as a magnitude without sign.

8.2.2 Reference points



P	=	Tool setting point
M	=	Machine zero point
W	=	Workpiece zero point
R	=	Machine reference point
F	=	Machine slide reference point
WR	=	Workpiece reference point
XMR, ZMR, etc	=	Reference point co-ordinate for each axis
XMW, ZMW, etc	=	Sum of zero offsets for each axis
XFP, ZFP	=	Workpiece dimensions.

8.2.3 Path calculation

G91 in the first block with motion

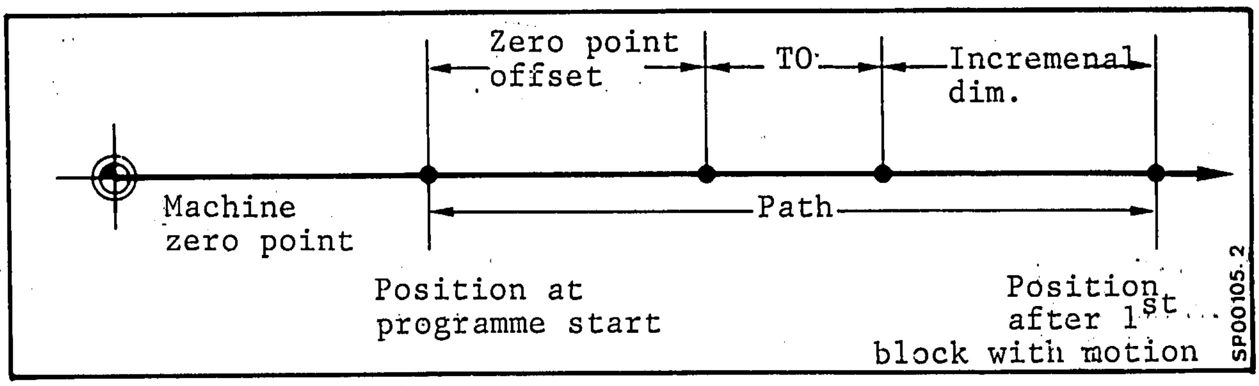
Path = Incremental dim. + ZO + TO

G91 from the second block with a motion

Path = incremental dim. + $ZO_{(new)}$ - $ZO_{(old)}$ + $TO_{(new)}$ - $TO_{(old)}$

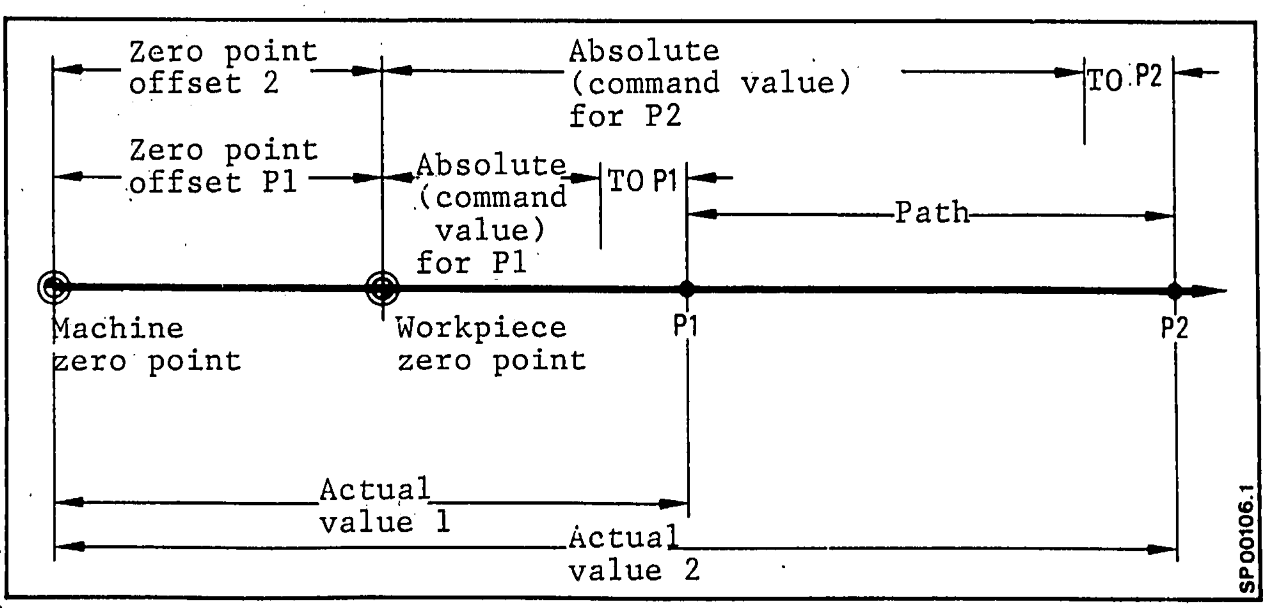
When the ZO and TO are not changed, the formula is simply:

Path = Incremental dimension

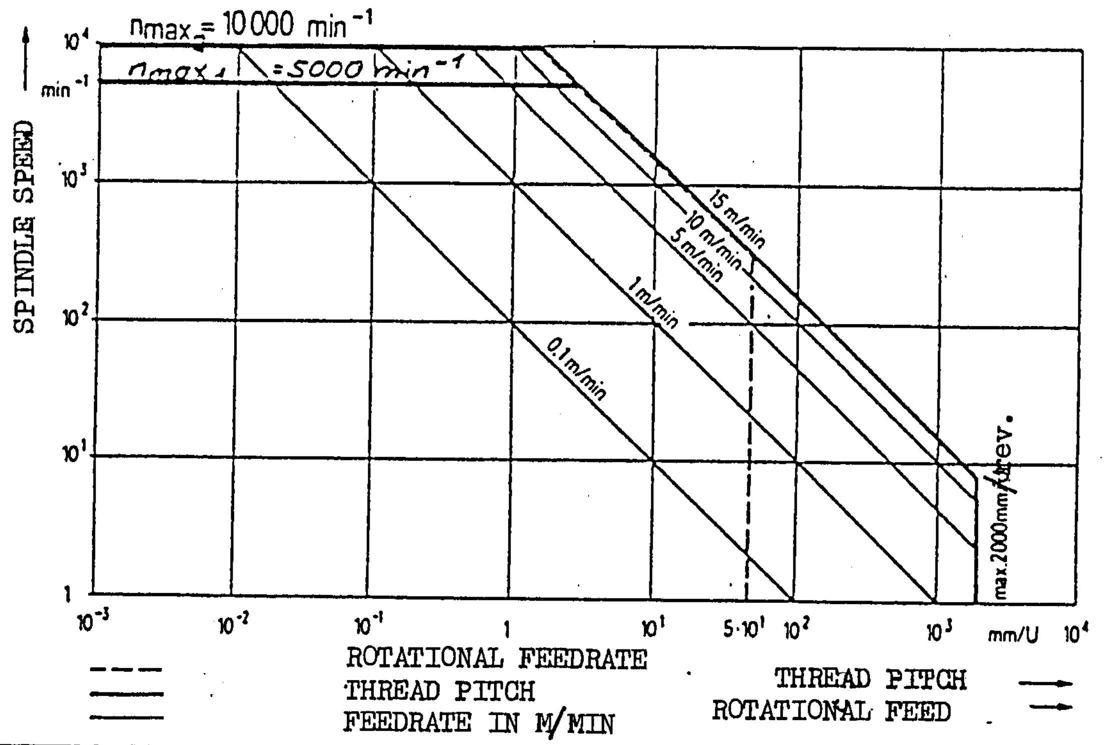


G90 in any block with a motion

Path = Absolute $dim._{(new)}$-absolute $dim._{(old)}$+$ZO_{(new)}$-$ZO_{(old)}$+ +$TO_{(new)}$-$TO_{(old)}$



8.2.4 Limit data for rotational feedrate



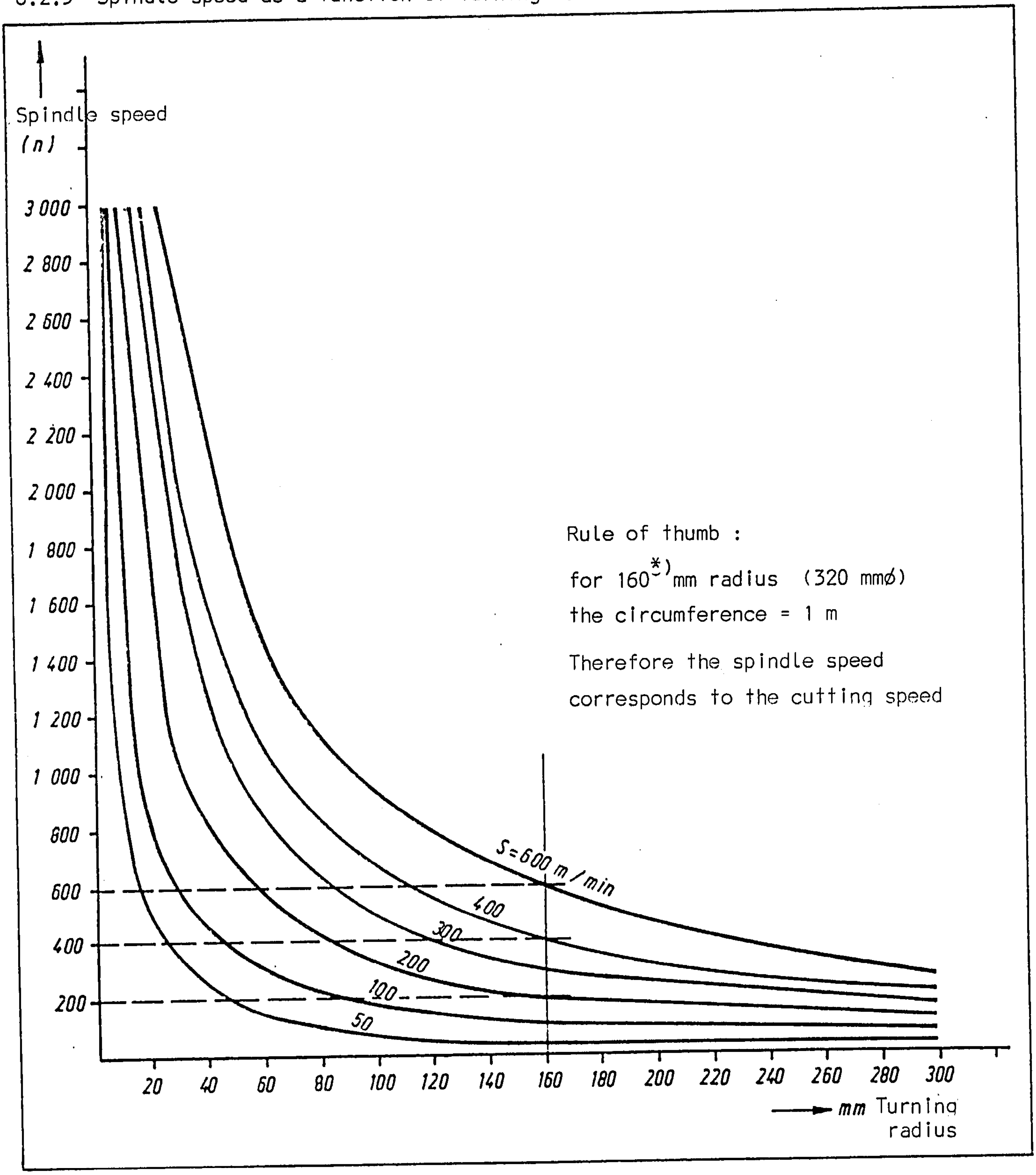
Relationship between rotational feedrate and spindle speed

Relationship between pitch and spindle speed (thread cutting G33)

$n\ max._1$ can be achieved with ROD encoder connected 1:1

$N\ max._2$ can be achieved with ROD encoder connected 1:2

8.2.5 Spindle speed as a function of turning radius for V = constant.



*) exactly 159.164 mm

8.2.6 Input format

Address definition		Metric		Inch				Degrees	
		Decades	smallest increment	Decades	smallest increment	Decades	smallest increment	Decades	smallest increment
Path data (linear axes) Interpolation parameter Radius		± 5.3	10^{-3} mm	± 4.4	10^{-4} inch	± 3.5	10^{-5} inch	-	10^{-3} degrees
Path data (rotary axes)		-		-		-		± 5.3	
Chamfer		- 5.3		- 4.4		- 3.5		-	
Work area limitation Zero point offset		± 5.3		± 4.4		± 3.5		± 5.3	
Thread pitch		4.3		3.4		2.5		-	
Cutting speed S (Weighting factor set during commissioning) Spindle speed S		4.0	1 or 0,1 m/min 1 or 0,1 min^{-1}	4.0	1 or 0.1 Ft/min 1 or 0.1 min^{-1}	4.0	1 or 0.1 Ft/min 1 or 0.1 min^{-1}		
Linear feedrate (F)		5.0	mm/min	4.1	10^{-1} inch/min	4.1	10^{-1} inch/min	5.0	degrees/min
Thread pitch increase or decrease F Rotary feedrate (F)		2.3	10^{-3} mm/rev	1.4	10^{-4} inch/rev	1.4	10^{-4} inch/rev		
Tool offset	Lgth add lgth	± 4.3	10^{-3} mm	± 3.4	10^{-4}	± 2.5 Inch	10^{-5}		
	Radius	± 3.3		± 2.4		± 1.5			
	Wear	± 1.3		± 0.4		± 0.04			
	Form	1		1		1			
Factor (G92)		3.5		3.5		3.5			
Dwell	X	5.3	10^{-3} sec	5.3	10^{-3} sec	5.3	10^{-3} sec		
	F	2.3		2.3		2.3			
Angle								3.5	10^{-5} degree
Angle for oriented spindle stop								3.0	1 degree
R - Parameter		Dimension depends on associated (internal floating point) all combinations (2 decades for call-up)							
G. prep. function		2		2		2			
M. functions		2		2		2			
H. functions		1 to 6		1 to 6		1 to 6			
Block number		1 to 4		1 to 4		1 to 4			
Special functions @		2		2		2			

Using inch input (G70) the smallest input increment can be changed from 10^{-4} inch to 10^{-5} inch by modification of setting datum.

The parameters (R00-R99) and special functions @ 00 - @ 99 are always written as 2 decades. For all other functions (except address L) the leading zeros can be omitted.

8.2.7 Code table

Code table

ISO DIN 66024											
Bit no. Ptestbit	P	7	6	5	4		3	2	1		
Track no. Sprocket	6	7	6	5	4	T	3	2	1		
Charac.	Hole combination									Allowed in Prog.	Leader
NUL						•				X	X
BS	●				●	•				–	X
HT					●	•			●	X	X
LF					●	•		●		X	-
CR	●				●	•	●		●	X	X
SP	●		●			•				X	X
(			●		●	•				X	X
)	●		●		●	•			●	X	X
%	●		●			•	●		●	X	–
:			●	●	●	•		●		X	X
/	●		●		●	•	●	●	●	X	X
+			●		●	•		●	●	X	X
–			●		●	•	●		●	X	X
.			●		●	•	●	●		X	X
0			●	●		•				X	X
1	●		●	●		•			●	X	X
2	●		●	●		•		●		X	X
3			●	●		•		●	●	X	X
4	●		●	●		•	●			X	X
5			●	●		•	●		●	X	X
6			●	●		•	●	●		X	X
7	●		●	●		•	●	●	●	X	X
8	●		●	●	●	•				X	X
9			●	●	●	•			●	X	X
@	●	●				•				X	X
A		●				•			●	X	X
B		●				•		●		X	X
C	●	●				•		●	●	X	X
D		●				•	●			X	X
E	●	●				•	●		●	X	X
F	●	●				•	●	●		X	X
G		●				•	●	●	●	X	X
H		●			●	•				X	X
I	●	●			●	•			●	X	X
J	●	●			●	•		●		X	X
K		●			●	•		●	●	X	X
L	●	●			●	•	●			X	X
M		●			●	•	●		●	X	X
N		●			●	•	●	●		X	X
O	●	●			●	•	●	●	●	–	X
P		●		●		•				X	X
Q	●	●		●		•			●	X	X
R	●	●		●		•		●		X	X
S		●		●		•		●	●	X	X
T	●	●		●		•	●			X	X
U		●		●		•	●		●	X	X
V		●		●		•	●	●		X	X
W	●	●		●		•	●	●	●	X	X
X	●	●		●	●	•				X	X
Y		●		●	●	•			●	X	X
Z		●		●	●	•		●		X	X
DEL	●	●	●	●	●	•	●	●	●	X	X

EIA 244 A											
Bit no.	8	7	6	P	4		3	2	1		
Track no.	8	7	6	5	4	T	3	2	1		
Charac.	Hole combination									Allowed in Prog.	Leader
						•				–	X
RT			●		●	•		●		–	X
TAB			●	●	●	•	●	●		X	X
<=EOB	●					•				X	–
LC)		●	●	●	●	•		●		–	X
ZWR				●		•				X	X
(≡				●	●	•		●		X	X
) ≡		●			●	•		●		X	X
EOR					●	•		●	●	X	–
.		●	●		●	•		●	●	X	X
/			●	●		•			●	X	X
+		●	●	●		•				X	X
–		●				•				X	X
0			●			•				X	X
1						•			●	X	X
2						•		●		X	X
3				●		•		●	●	X	X
4						•	●			X	X
5				●		•	●		●	X	X
6				●		•	●	●		X	X
7						•	●	●	●	X	X
8					●	•				X	X
9				●	●	•			●	X	X
@ ≡		●	●		●	•	●		●	X	X
a		●	●			•			●	X	X
b		●	●			•		●		X	X
c		●	●	●		•		●	●	X	X
d		●	●			•	●			X	X
e		●	●	●		•	●		●	X	X
f		●	●	●		•	●	●		X	X
g		●	●			•	●	●	●	X	X
h		●	●		●	•				X	X
i		●	●	●	●	•			●	X	X
j		●		●		•			●	X	X
k		●		●		•		●		X	X
l		●				•		●	●	X	X
m		●		●		•	●			X	X
n		●				•	●		●	X	X
o		●				•	●	●		X	X
p		●		●		•	●	●	●	X	X
q		●		●	●	•				X	X
r		●			●	•			●	X	X
s			●	●		•		●		X	X
t			●			•		●	●	X	X
u			●	●		•	●			X	X
v			●			•	●		●	X	X
w			●			•	●	●		X	X
x			●	●		•	●	●	●	X	X
y			●	●	●	•				X	X
z			●		●	•			●	X	X
IRR		●	●	●	●	•	●	●	●	X	X
UC		●	●	●	●		●			–	X

8.2.8 Block preparation time

The block preparation time is the time needed by the control to prepare one block, so that it can be accounted for in the working block.

The block preparation time for the 8T and Sprint 8T is:

80ms without CRC	)	Working from memory
100ms with CRC	)	

When working from the tape reader an additional 4ms per character has to be calculated.

To avoid free cutting (as in feed interruption on the contour) the feedrate or distance travelled per block must be selected so that the block preparation time is not undercut. That means:

t is equal to or greater than $\frac{s}{v}$

s =	Path	(mm)
v =	Speed	(m/min)
t =	80 or 100	(ms)

As the control has 4 - 8 blocks permanently in buffer behind the working block, individual blocks may undercut the block preparation time.

8.3.1 Program key Sprint 8T

Group	EIA	ISO	Code	Chapter	Function and Meaning
	EOR	%		1.	Rewind stop, program start for tape read-in
	EOR EOB	% ..LF	0 to 9999	1.7	Program number
	o n /o /n	: N /: /N	1 to 9999	1.4	Main block Subordinate block Deletable main block Deletable subordinate block
G1	g	G	00 01 • 10 11 02 03 33 34 35	3.2 3.3 3.4 3.4 3.5 3.5 3.6 3.7 3.8	Rapid traverse Linear interpolation Polar coordinate programming rapide Polar coordinate programming linear interpol. Circular interpolation clockwise Circular interpolation counter-clockwise Thread cutting constant pitch Thread cutting linear increase Thread cutting linear decrease
G2	g	G	■ 04 *	3.13	Dwell, predetermined using addresses X or F; own block
G3	g	G	■ 09	3.10	Speed reduction for exact stop
G5	g	G	■ 25 * ■ 26 *	3.15 3.15 3.23	Minimum limit; machining area X,Z Minimum limit; machining area X,Z Comand value check S
G6	g	G	40 • 41 42	3.24 3.24 3.24	No cutter radius compensation CRC left hand side CRC right hand side
G7	g	G	■ 53	3.18	Zero offset suppression
G8	g	G	54 • 55	3.17 3.17	Zero offset 1 Zero offset 2
G9	g	G	■ 59 *	3.17 3.17	Programmable additive zero offset Loading of zero offsets G59 N..
G10	g	G	60 63 64 •	3.10 3.11 3.12	Exact stop Tapping with compensated tap holder Continuous path operation
G11	g	G	70 71	3.14 3.14	Input system inch) Reset state via machine Input system metric) parameters
G12	g	G	90 • 91	3.1 3.1	Absolute data input Incremental data input
G13	g	G	■ 92 *	3.19 3.19 3.22 3.9 4.3.1 4.3.4	Setting actual data stores with X,Z Resetting actual data stores Command value, limitation with S Smoothing time for thread cutting Additional programmable tool offset with I+K Loading of Tool offset G92 T..X..Z..B..A..
G14	g	G	94 95 • 96 97	3.20 3.20 3.20 3.21 3.21	Feedrate using address F in mm/inch Feedrate using address F in mm/rev Feedrate using address F in min/rev and constant cutting speed (S = m/min) for G96 Deactive G96, store last speed command value
	x	X	0.001 to ±99999.999	2.1	Path data in mm
			0.001 to +99999.999	3.13	Dwell time in sec
	z	Z	0.001 to ±99999.999	2.1	Path data in mm
	a	A +)	0 to 359.99999 0 to 359.99999	6.1 3.4	Angle degrees for contour section Angle in degrees for polar coordinates
	b	B +)	0,-0 +0.001 to +99999.999 -0.001 to -99999.999 ±0.001 to ±99999.999	6.1 3.4/6.1 6.1 3.5	Corner for contour section Radius for contour section and polar coordin. Chamfer for contour section in mm) in mm Radius for circular interpolation in mm
	i	I	0.001 to ±99999.999	3.5	Interpolation parameters for X axis in mm
			0.001 to 2000.000	3.6	Thread pitch in mm
			0.001 to ±9999.999	3.6	Length compensation in X
	k	K	0.001 to ±99999.999	3.5	Interpolation parameters for Z axis in mm
			0.001 to 2000.000	3.6	Thread pitch in mm
			0.001 to ±9999.999	3.6	Length compensation in Z

Program key Sprint 8T (continuation)

Group	EIA	ISO	Code	Chapter	Function and meaning
	r	R	00 to 49	5.0	Parameter
	f	F	1 to 15 000	3.20	Feedrate in m/min (see 4.4)
			0.001 to 99.999	3.13	Dwell in sec
			0.001 to 50.000	3.20	Feedrate in mm/rev
			0.001 to 16.000	3.7/3.8	Thread pitch increase or decrease in mm/rev
	s	S	1 to 9999	4.1 3.21	Spindle speed rpm or 0.1 rpm or Constant cutting speed min/mm or 0.1 m/min
			1 to 9999	3.23	Spindle speed limitation in rpm or 0.1 rpm
			0 to 359	4.4	Spindle orientation in degrees from encoder marker pulse
	t	T	00.. to 99..	4.3	Tool number
			..01 to ..32 ..1 or to..9	4.3	Tool offset activation
			..00 or .0	4.3	Tool offset de-activation
			0,1,2,3,4,5	3.9	Feed ramp time
	h	H +)	1 to 999999	4.2	Auxiliary function
	l	L	001.. to 999..	1.8	Subroutine number
			...01 to ...99	1.9/5.5	Number of subroutine repetitions
M1	m	M	■ 00 01	4.4 4.4	Programmed stop, unconditional Programmed stop, conditional
M2	m	M	02 17 30	4.4 4.4 4.4	Program end without rewind) written in the Subroutine end } last block of pro- Program end with rewind) gram (subroutine)
M3	m	M	03 04 05 ● 19	4.4 4.4 4.4 4.4	Spindle rotation clockwise Spindle rotation counter-clockwise Spindle stop Oriented spindle stop angle in degrees using S
M4	m	M	36 37	3.20/4.4 3.20/4.4	Feedrate programmed using F) Effective also Feedrate in mm/min or mm/rev) with G33
M5	m	M	00 to 99	4.4	Misc. function unassigned except grps (M1 - M4)
	l	L	91 92	7.2 7.2	Retract cycle for tool change sequence Z-X Retract cycle for tool change sequence X-Z
	l	L	95	7.3	Stock removal cycle
	l	L	97	7.4	Thread cutting cycle
	l	L	98	7.5	Deep hole boring cycle
	l	L	999	7.6	Clear active stores
	5-4-2 1) 7-4-2 1)	( )		1.6 1.6	Remark start Remark end
	EOB	LF		1.4	Block end

1) Punched tracks
* No other preparatory functions may be written in this block
● Reset state (ground state after reset M02/M30 control switch on)
■ In blocks, all others modal
+) Other addresses selectable (A.B.C.U.V.W.O.E.P.H)

8.3.2 Program key 8T

Group	EIA	ISO	Code	Chapter	Function and Meaning
	EOR	%		1.	Rewind stop, program start for tape read-in
	EOR EOB	% ..LF	0 to 9999	1.7	Program number
	o n /o /n	: N /: /N	1 to 9999	1.4	Main block Subordinate block Deletable main block Deletable subordinate block
G1	g	G	00 01 ● 10 11 02 03 33 34 35	3.2 3.3 3.4 3.4 3.5 3.5 3.6 3.7 3.8	Rapid traverse Linear interpolation Polar coordinate programming rapid Polar coordinate programming linear interpol. Circular interpolation clockwise Circular interpolation counter-clockwise Thread cutting constant pitch Thread cutting linear increase Thread cutting linear decrease
G2	g	G	■ 04 *	3.13	Dwell, predetermined using addresses X or F; own block
G3	g	G	■ 09	3.10	Speed reduction for exact stop
G5	g	G	■ 25 * ■ 26 *	3.15 3.15 3.23	Minimum limit Minimum limit; Comand value check S
G6	g	G	40 ● 41 42	3.24 3.24 3.24	No cutter radius compensation CRC left hand side CRC right hand side
G7	g	G	■ 53	3.18	Zero offset suppression
G8	g	G	54 ● 55	3.17 3.17	Zero offset 1 Zero offset 2
G9	g	G	■ 59 *	3.17 3.17	Programmable additive zero offset Loading of zero offsets G59 N..
G10	g	G	60 63 64 ●	3.10 3.11 3.12	Exact stop Tapping with compensated tap holder Continuous path operation
G11	g	G	70 71	3.14 3.14	Input system inch) Reset state via machine Input system metric) parameters
G12	g	G	90 ● 91	3.1 3.1	Absolute data input Incremental data input
G13	g	G	■ 92 *	3.19 3.19 3.22 3.9 4.3.4	Setting actual data stores with X,Z Resetting actual data stores Command value, limitation with S Smoothing time for thread cutting Loading of tool offset G92 Too X..Z..B..A..
G14	g	G	94 95 ● 96 97	3.20 3.20 3.20 3.21 3.21	Feedrate using address F in mm/inch Feedrate using address F in mm/rev Feedrate using address F in min/rev and constant cutting speed (S = m/min) for G96 Deactive G96, store last speed command value
	x	X	0.001 to ±99999.999	2.1	Path data in mm
			0.001 to +99999.999	3.13	Dwell time in sec
	z	Z	0.001 to ±99999.999	2.1	Path data in mm
	i	I	0.001 to ±99999.999	3.5	Interpolation parameters for X axis in mm
			0.001 to 2000.000	3.6	Thread pitch in mm
			0.001 to ±9999.999	3.6	Length compensation in X
	k	K	0.001 to ±99999.999	3.5	Interpolation parameters for Z axis in mm
			0.001 to 2000.000	3.6	Thread pitch in mm
			0.001 to ±9999.999	3.6	Length compensation in Z

Program key 8T (continuation)

Group	EIA	ISO	Code	Chapter	Function and meaning
	r	R	00 to 49	5.0	Parameter
	f	F	1 to 15 000	3.20	Feedrate in mm/min (see 4.4)
			0.001 to 99.999	3.13	Dwell in sec
			0.001 to 50.000	3.20	Feedrate in mm/rev
			0.001 to 16.000	3.7/3.8	Thread pitch increase or decrease in mm/rev
	s	S	1 to 9999	4.1 3.21	Spindle speed rpm or 0.1 rpm or Constant cutting speed min/mm or 0.1 m/min
			1 to 9999	3.23	Spindle speed limitation in rpm or 0.1 rpm
			0 to 359	4.4	Spindle orientation in degrees from encoder marker pulse
	t	T	00.. to 99..	4.3	Tool number
			..01 to ..32 ..1 or to ..9	4.3	Tool offset activation
			..00 or .0	4.3	Tool offset de-activation
			0,1,2,3,4,5	3.9	Feed ramp time
	h	H +)	1 to 999999	4.2	Auxiliary function
	l	L	001.. to 999..	1.8	Subroutine number
			...01 to ...99	1.9/5.5	Number of subroutine repetitions
M1	m	M	■ 00 01	4.4 4.4	Programmed stop, unconditional Programmed stop, conditional
M2	m	M	02 17 30	4.4 4.4 4.4	Program end without rewind } written in the Subroutine end } last block of pro- Program end with rewind } gram (subroutine)
M3	m	M	03 04 05 ● 19	4.4 4.4 4.4 4.4	Spindle rotation clockwise Spindle rotation counter-clockwise Spindle stop Oriented spindle stop angle in degrees using S
M4	m	M	36 37	3.20/4.4 3.20/4.4	Feedrate programmed using F } Effective also Feedrate in mm/min or mm/rev } with G33
M5	m	M	00 to 99	4.4	Misc. function unassigned except grps (M1 - M4)
	l	L	91 92	7.2 7.2	Retract cycle for tool change sequence Z-X Retract cycle for tool change sequence X-Z
	l	L	95	7.3	Stock removal cycle
	l	L	97	7.4	Thread cutting cycle
	l	L	98	7.5	Deep hole boring cycle
	l	L	999	7.6	Clear active stores
	5-4-2 1) 7-4-2 1)	( )		1.6 1.6	Remark start Remark end
	EOB	LF		1.4	Block end

1) Punched tracks

* No other preparatory functions may be written in this block

● Reset state (ground state after reset M02/M30 control switch on)

■ In blocks, all others modal

+) Other addresses selectable (A.B.C.U.V.W.O.E.P.H)

Issued by Siemens AG,
Logic and Process
Control Systems Department
Postfach 3240, D-8520 Erlangen

Subject to change without prior notice

Siemens Aktiengesellschaft

Order No. E80210-T43-X-A7-7600
Printed in the Federal Republic of Germany
082301 PA 3850.5

progress in automation: Siemens