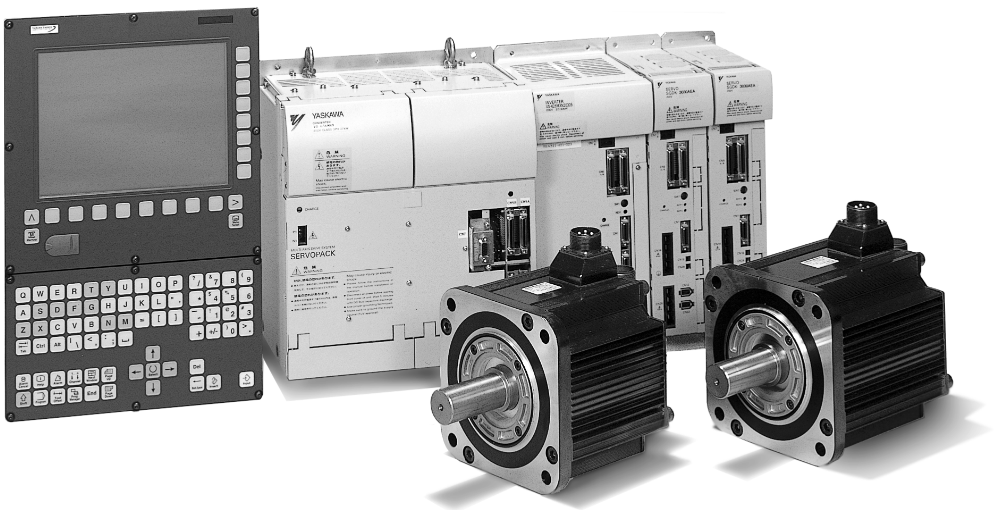
Yaskawa Siemens
numerical controls

Yaskawa Siemens CNC Series

Programming Manual for Lathe



MANUAL No. NCSIE-SP02-21

Yaskawa Siemens Numerical Controls Corp. has been merged to Siemens K.K. and Siemens Japan K.K. as of August, 2010 respectively. "Yaskawa Siemens Numerical Controls Corp." in this manual should therefore be understood as "Siemens Japan K.K."

This manual is intended for both of Yaskawa Siemens 840DI and Yaskawa Siemens 830DI. In this manual, the functional differences of these two models are not taken into account in its description, thus please refer to the catalog (MANUAL No.: NCKAE-PS41-01) for available basic functions and possible optional functions of each model.

Yaskawa Siemes 840DI

Programming Manual for Lathe

Users Manual

Validity

Control System	*Software Version*
Yaskawa Siemens 840DI	

02.2001 Edition

Yaskawa Siemens documentation

Printing history

Brief details of this edition and previous editions are listed below.

The status of each edition is shown by the code in the “Remarks” column.

Status code in the “Remarks” column:

A . . . . . New documentation.
B . . . . . Unrevised reprint with new Order No.
C . . . . . Revised edition with new status.
If factual changes have been made on the page since the last edition, this is indicated by a new edition coding in the header on that page.

Edition	Manual No.	Remarks
02.01	NCSIE-SP02-20	A
		C

Trademarks
Yaskawa Siemens is our trademark.
The other designations in this publication may also be trade marks, the use of which by third parties may constitute copyright violation.

This publication was produced with Interleaf V 7.

Preface

Organization of the Documentation

The Documentation is divided into 3 parts:

- General Documentation
- User Documentation
- Manufacturer/Service Documentation

Target group

This Manual is intended for machine–tool users. It provides detailed information that the user requires to program the Yaskawa Siemens 840DI control system.

Standard scope

This Programming Guide describes the functionality afforded by standard functions. Differences and additions implemented by the machine–tool manufacturer are documented by the machine–tool manufacturer.

More detailed information about other publications relating to Yaskawa Siemens 840DI and publications that apply to all Yaskawa Siemens controls (e.g. Universal Interface, Measuring Cycles...) can be obtained from your local Siemens branch office.

Other functions not described in this documentation might be executable in the control. This does not, however, represent an obligation to supply such functions with a new control or when servicing.

Applicability

Yaskawa Siemens 840DI
with the operator panels OP010/010S/010C/012/015.

Outline

This Programming Guide is intended for use by skilled machine operators with the appropriate expertise in drilling, milling and turning operations. Simple programming examples are used to explain the commands and statements which are also defined according to DIN 66025.

Structure of descriptions

All cycles and programming options have been described according to the same internal structure as far as this is meaningful and practicable. The various levels of information have been organized such that you can selectively access the information you need for the task in hand.

Principle

Your Yaskawa Siemens 840DI has been designed and constructed according to state–of–the–art technology and approved safety regulations and standards.

Additional equipment

The applications of Yaskawa Siemens controls can be expanded for specific purposes through the addition of special add–on devices, equipment and expansions supplied by Yaskawa Siemens NC.

Personnel

Only appropriately trained, authorized and reliable personnel may be allowed to operate this equipment. The control must never be operated, even temporarily, by anyone who is not appropriately skilled or trained.
The relevant responsibilities of personnel who set up, operate and maintain the equipment must be clearly defined; the proper fulfillment of these responsibilities must be monitored.

Behavior

Before the control is started up, it must be ensured that the Operator’s Guides have been read and understood by the personnel responsible. The operating company is also responsible for constantly monitoring the overall technical state of the control (visible faults and damage, altered service performance).

Servicing

Repairs must be carried out according to the information supplied in the service and maintenance guide by personnel who are specially trained and qualified in the relevant technical subject. All relevant safety regulations must be followed.

Note

The following is deemed to be improper usage and exempts the manufacturer from any liability:

Any application which does not comply with the rules for proper usage described above.

If the control is not in technically perfect condition or is operated without due regard for safety regulations and accident prevention instructions given in the Instruction Manual.

If faults that might affect the safety of the equipment are not rectified before the control is started up.

Any modification, bypassing or disabling of items of equipment on the control that are required to ensure fault–free operation, unlimited use and active and passive safety.

Searching aids

In addition to the table of contents we have provided the following information in the appendix for your assistance:

- Index of abbreviations
- Index of terms
- G Code Table
- MDs and SDs
- Data Fields, Lists
- Alarms
- References
- Index

For a complete list and description of Yaskawa Siemens 840DI alarms, please refer to

References: /DA/, Diagnostics Guide

For further useful information on start-up and troubleshooting, please refer to

References: /FB/, D1, “Diagnostics Tools”

Safety Guidlines

This manual contains notices which you should observe to ensure your own personal safety, as well as to protect the product and connected equipment. These notices are highlighted in the manual by a warning triangle and are marked as follows according to the level of danger:

Danger

indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

Warning

indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

Caution

used with the safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

Caution

used without the safety alert symbol indicates a potentially hazardous situation which, if not avoided, may result in property damage.

Note

is an important piece of information about the product, the handling of the product or the respective part of the documentation which should be noted in particular.

Technical information

Trademarks

IBM® is a registered trademark of the International Business Corporation.
MS–DOS® and WINDOWS™ are registered trademarks of the Microsoft Corporation.

Notation

The following notation and abbreviations are used in this document:

- PLC interface signals –> IS “Signal name” (signal data)
  Examples:
  - IS “MMC–CPU1 ready” (DB10, DBX108.2), i.e. the signal is stored in data block 10, data byte 108, bit 2.
  - IS “Feedrate/spindle override” (DB31–48, DBB0), i.e. the signals are stored for specific spindles/axes in data blocks 31 to 48, data block byte 0.
- Machine data –> MD: MD_NAME (English designation)
- Setting data –> SD: SD_NAME (English designation)
- The character “≙” means “corresponds to”.

■

Notes

Table of Contents

Notes

1 Programming Basics

Chapter 1 describes the basic terms used in programming and the feed functions.

1.1 Introductory explanations

1.1.1 Siemens mode

The following conditions apply when Siemens mode is active:

- Siemens G commands are interpreted on the control by default. This applies to all channels.
- It is not possible to extend the Siemens programming system with ISO Dialect functions because some of the G functions have different meanings.
- Downloadable MD files can be used to switch the control to ISO Dialect mode. In this case, the system boots the ISO Dialect mode by default.

1.1.2 ISO Dialect mode

The following conditions apply when ISO Dialect mode is active:

- Only ISO Dialect G codes can be programmed, not Siemens G codes.
- It is not possible to use a mixture of ISO Dialect code and Siemens code in the same NC block.
- It is not possible to switch between ISO Dialect–M and ISO Dialect–T via G command.
- Siemens subprogram calls can be programmed.
- If further Siemens functions are to be used, it is necessary to switch to Siemens mode first.

1.1.3 Switchover

The following two G commands are used to switch between Siemens mode and ISO Dialect mode:

– G290 – Siemens NC programming language active

– G291 – ISO Dialect NC programming language active

The active tool, the tool offsets and the zero offsets are not changed by this action.

1.1.4 G code display

The G code display must always be implemented in the same language type (Siemens/ISO Dialect) as the current block display. If the block display is suppressed with DISPLOF, the current G codes continue to be displayed in the language type of the active block.

Example

The Siemens standard cycles are called up using the G functions of the ISO Dialect mode. DISPLOF is programmed at the start of the cycle, with the result that the ISO Dialect G commands remain active for the display.

```
PROC CYCLE328 SAVE DISPLOF
N10 ...
...
N99 RET
```

Procedure

External main program calls Siemens shell cycle. Siemens mode is selected implicitly on the shell cycle call.

DISPLOF freezes the block display at the call block; the G code display remains in external mode. This display is refreshed while the Siemens cycle is running.

The SAVE attribute resets the G codes modified in the shell cycle to their original state when the shell cycle was called on the return jump to the main program.

1.1.5 Maximum number of axes / axis designation

In ISO Dialect–T the maximum number of axis is 8. Axis designation for the first two axes is fixed to X and Z. Further axes can be designated Y, A, B, C, U, V, W.

1.1.6 Selection of G code system A, B, or C

ISO Dialect T distinguishes between G code system A, B, and C. G code system B is default setting. The G code system in use is selected by MD $MN_MM_EX–TERN_GCODE_SYSTEM as follows:

$MN_MM_EXTERN_GCODE_SYSTEM = 0: G code system B
$MN_MM_EXTERN_GCODE_SYSTEM = 1: G code system A
$MN_MM_EXTERN_GCODE_SYSTEM = 2: G code system C

G Code system A

If G code system A is active, G91 is not available. In this case, incremental axes movement for axis X,Y, and Z is programmed by address U, V, and W. U, V, and W are not available as axis designation in this case resulting in a maximum axes number of 6.
Address H is used for programming incremental movement of the C axis in G code system A.

Note

- If not otherwise noted, the manual in hand describes G code system B.
- For the differences between G code system A, B, and C refer to the G code list in the appendix.

1.1.7 Block skip (/0 to /7)

In ISO Dialect mode, a skipped block is represented by ”/”. This block is skipped when the relevant skip level is active. A block that is skipped must still be syntactically error–free. Skip levels /1 to /9, which are possible in ISO Dialect original mode, are mapped onto Siemens skip levels /0 to /7.
If the skip character ”/” is programmed alone, without a level, level 1 is active by default in ISO mode.
An alarm is issued in ISO Dialect mode if the skip identifier is in the middle of the block.

Note

- “1” can be omitted for “/1”.
- The optional block skip function is processed when a part program is read to the buffer register from either the tape or memory. If the switch is set ON after the block containing the optional block skip code is read, the block is not skipped.
- The optional block skip function is disregarded for program reading (input) and punch out (output) operation.

1.2 Basics of feed function

This section describes the feed function that specifies feedrate (distance per minute, distance per revolution) of a cutting tool.

1.2.1 Rapid traverse

Rapid traverse is used for positioning (G00) and manual rapid traverse (RAPID) operation. In the rapid traverse mode, each axis moves at the rapid traverse rate set for the individual axes; the rapid traverse rate is determined by the machine tool builder and set for the individual axes by using parameters. Since the axes move independently of each other, the axes reach the target point at different time. Therefore, the resultant tool paths are not a straight line generally.

1.2.2 Cutting feed (F command)

The feedrate at which a cutting tool should be moved in the linear interpolation (G01) mode or circular interpolation (G02, G03) mode is designated using address characters F. The axis feed mode to be used is selected by designating the feed function G code (G94 or G95) as indicated in Table 1-1. Select the required feed mode by designating the feed function G code before specifying an F code.

Table 1-1 Cutting feed mode G codes

G code	Function	Group
G94	Designation of feed per minute (mm/min) mode	05
G95	Designation of feed per revolution (mm/rev) mode	05

See 1.2.3 “Switching between feed per minute mode and feed per revolution mode” for details of these G codes. The F code is modal and once designated it remains valid until another F code is designated. If feed mode designation G codes are switched between G94 and G95, however, it is necessary to designate the F code again. If no new F code is designated, alarm 10860 “No feedrate programmed” occurs.

Feed per revolution mode (G95)

A feedrate of a cutting tool per revolution of the spindle (mm/rev, inch/rev) can be designated by a numeral specified following address character F.

Note: The upper limit of feedrates could be restricted by the servo system and the mechanical system. For the actual programmable feedrate range, refer to the manuals published by the machine tool builder.

An F command specified in the simultaneous 2-axis linear interpolation mode or in the circular interpolation mode represents the feedrate in the tangential direction.

Example of programming (linear interpolation mode)

G95 S1000 (r/min);
G91 G01 X60. Z40. F0.5;

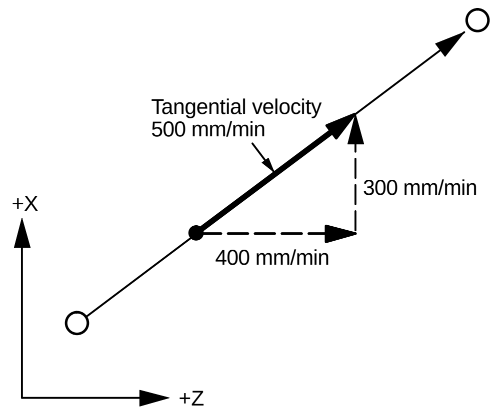


Fig. 1-1 F command in simultaneous 2-axis control linear interpolation (feed per revolution)

Example of programming (circular interpolation mode)

G95 S1000 (r/min);
G91 G03 X ... Z ... I ... F0.2;

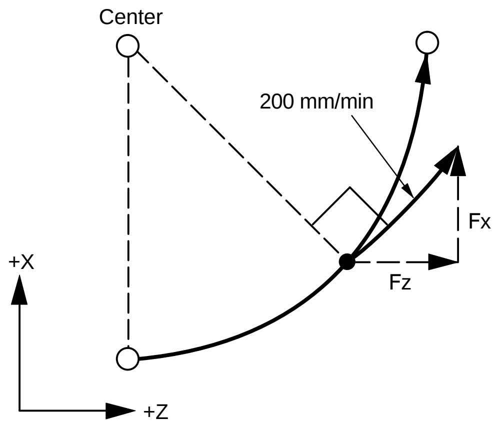


Fig. 1-2 F command in the simultaneous 2-axis control circular interpolation (feed per revolution)

Note

- An F0 command causes an input error.
- A feedrate in the X-axis direction is determined by the radial value.

A feedrate of a cutting tool per minute (mm/min, inch/min) can be designated by a numeral specified following address character F.

Note: The upper limit of feedrates could be restricted by the servo system and the mechanical system. For the actual programmable feedrate range, refer to the manuals published by the machine tool builder.

Simultaneous 2-axis control

An F command specified in the simultaneous 2-axis linear interpolation mode or in the circular interpolation mode represents the feedrate in the tangential direction.

Example of programming (linear interpolation mode)

```
G94;
G91 G01  X60.  Z40.  F500.;
```

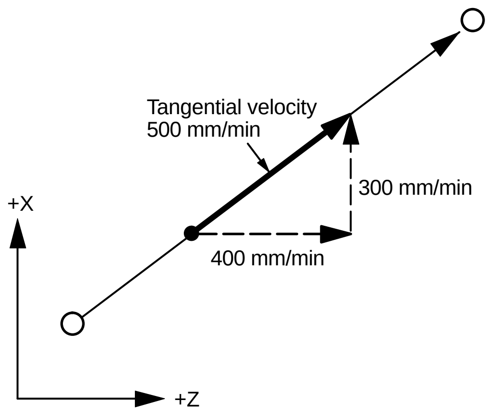


Fig. 1-3 F command in simultaneous 2-axis control linear interpolation (feed per minute)

Note

- An F0 command causes an input error.
- A feedrate in the X-axis direction is determined by the radial value.

Example of programming (circular interpolation mode)

```
G94;
G91 G03  X ...  Z ..   I ...  F200.;
```

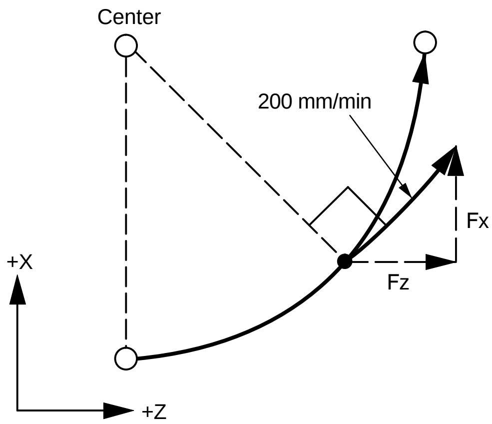


Fig. 1-4 F command in the simultaneous 2-axis control circular interpolation (feed per minute)

Note

Do not specify a negative value for an F command.

Rotary axis and linear axis

An F command specified in the interpolation mode between a rotary axis and a linear axis represents the feedrate in the tangential direction.

Example of programming

```
G94 G91;
G01  Z10.  C60.  F100.;
```

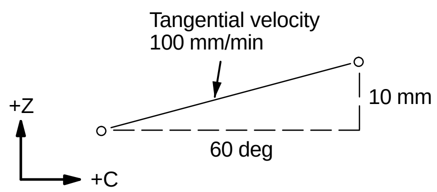


Fig. 1-5 F command in interpolation between rotary axis and linear axis (feed per minute)

1.2.3 Switching between feed per minute mode and feed per revolution mode (G94/G95)

Before specifying a feedrate command (F), a G code that determines whether the specified feedrate command is interpreted as feed per minute value or feed per revolution value should be specified. These G codes (G94, G95) are modal and once they are specified they remain valid until the other G code is specified. When the feed mode designation G code is specified, the presently valid F code is canceled. Therefore, an F code must be specified newly after switching the feed mode by designating G94 or G95 command. The initial status that is established when the power is turned on is set by MD 20154, EXTERN_GCODE_RESET_VALUES[4].

Table 1-2 MD EXTERN_GCODE_RESET_VALUES[4] and initial status

MD 20154	Initial G code
MD EXTERN_GCODE_RESET_VALUES[4]=1	G94
MD EXTERN_GCODE_RESET_VALUES[4]=2	G95

Feed per minute mode (G94)

By specifying “G94;”, the F codes specified thereafter are all executed in the feed per minute mode.

Table 1-3 Meaning of G94 command

G94	Meaning
mm input	mm/rev
inch input	inch/rev

Feed per revolution mode (G95)

By specifying “G95;”, the F codes specified thereafter are all executed in the feed per revolution mode.

Table 1-4 Meaning of G95 command

G95	Meaning
mm input	mm/rev
inch input	inch/rev

Commands Calling Axis Movements 2

Chapter 2 describes the interpolation commands, thread cutting function, and reference point return function.

2.1 Interpolation commands

This section describes the positioning commands and the interpolation commands that control the tool path along the specified functions such as straight line and arc.

2.1.1 Positioning (G00)

The G00 command moves a tool to the position in the workpiece system specified with an absolute or an incremental command at a rapid traverse rate. In the absolute command, coordinate value of the end point is programmed. In the incremental command the distance the tool moves is programmed.

For calling the positioning, the following G code can be used.

Table 2-1 G code for positioning

G code	Function	Group
G00	Positioning	01

Format

G00 X... Z... ;

When “G00 X(U)... Z(W)... (C(H)... Y(V)...);” is designated, positioning is executed. The program advances to the next block only when the number of lag pulses due to servo lag are checked after the completion of pulse distribution has reduced to the permissible value.

In the G00 mode, positioning is made at a rapid traverse rate in the simultaneous 2-axis control mode. The axes not designated in the G00 block do not move. In positioning operation, the individual axes move independently of each other at a rapid traverse rate that is set for each axis. The rapid traverse rates set for the individual axes differ depending on the machine. For the rapid traverse rates of your machine, refer to the manuals published by the machine tool builder.

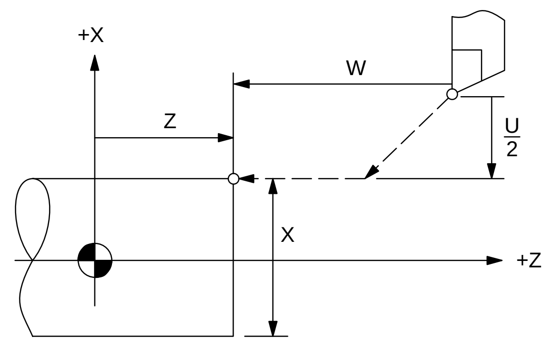


Fig. 2-1 Positioning in simultaneous 2-axis control mode

Note

- In the G00 positioning mode, since the axes move at a rapid traverse rate set for the individual axes independently, the tool paths are not always a straight line. Therefore, positioning must be programmed carefully so that a cutting tool will not interfere with a workpiece or fixture during positioning.
- The block where a T command is specified must contain the G00 command. Designation of the G00 command is necessary to determine the speed for offset movement which is called by the T command.

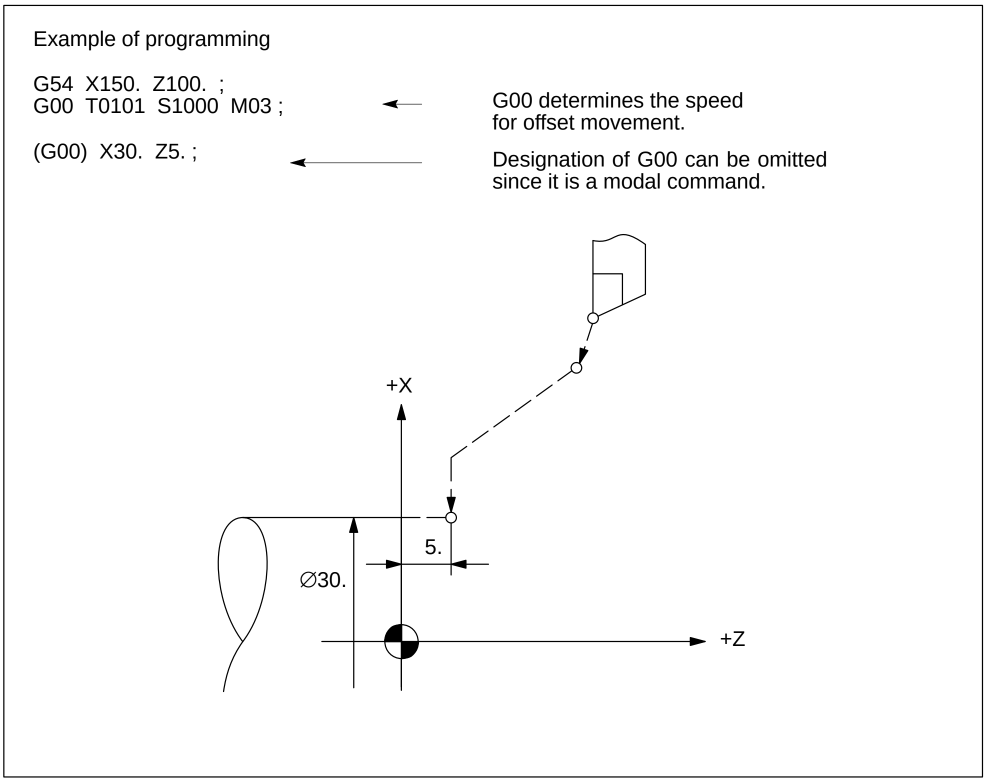


Fig. 2-2

G0 Linear Mode

The G0 linear mode is valid if MD $MC_EXTERN_G0_LINEAR_MODE is set. In this case, all programmed axes move in linear interpolation and reach their target position at the same point of time.

2.1.2 Linear interpolation (G01)

Format

G01 X... Z... F... ;

With the commands of “G01 X(U)... Z(W)... (C(H)... Y(V)...) F...;”, linear interpolation is executed in the simultaneous 2-axis control mode. The axes not designated in the G01 block do not move. For the execution of the linear interpolation, the following commands must be specified.

Command format

To execute the linear interpolation, the commands indicated below must be specified.

- Feedrate

  Feedrate is designated by an F code. The axes are controlled so that vector sum (tangential velocity in reference to the tool moving direction) of feedrate of the designated axes will be the specified feedrate.
  (Fx: feedrate in the X-axis direction)

- With an F code, axis feedrate is specified in either feed per spindle revolution (mm/rev or inch/rev) or feed per minute (mm/min or inch/min).

Note

For the C-axis, a feedrate cannot be specified in the feed per minute mode.

- End point

  The end point can be specified in either incremental or absolute values corresponding to the designation of an address character or G90/G91. For details, see 3.2.1, “Absolute/incremental designation”.

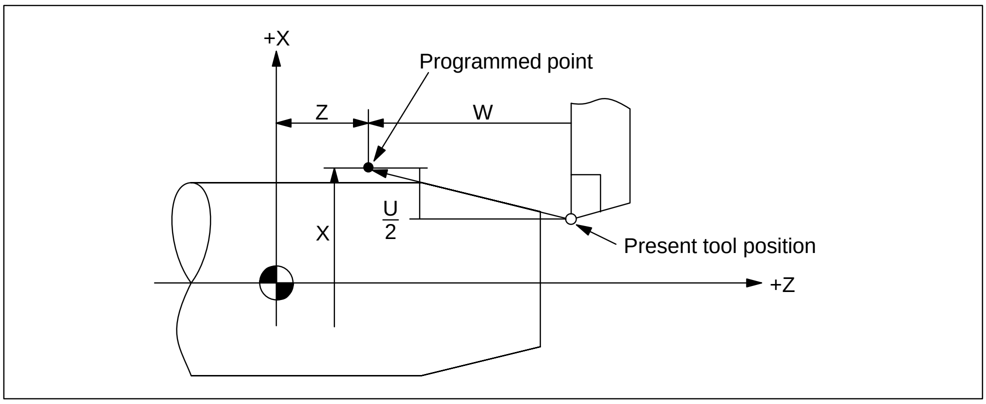


Fig. 2-3 Linear interpolation

$$F\ (mm/min) = \sqrt{Fx^2 + Fz^2 + (Fc^2)}$$

Example of programming

```
G54  X100.  Z60.;
G00  T0202  S600  M03;
     X35.  Z5.;
G01  Z0  F1.;
     X60.  F0.2;
```

$\}$ Axes are moved in the G01 linear interpolation mode.

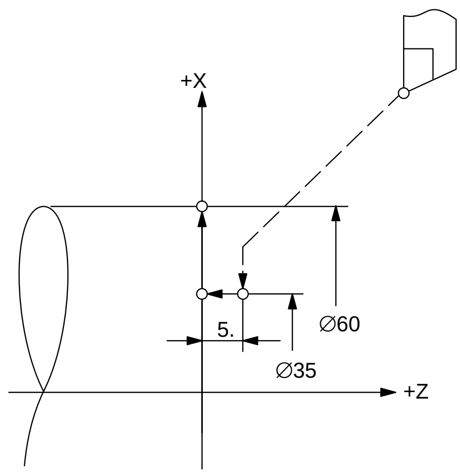


Fig. 2-4 Example of programming

2.1.3 Circular interpolation (G02, G03)

Format

By specifying the following commands in a program, the cutting tool moves along the specified arc in the ZX plane so that tangential velocity is equal to the feedrate specified by the F code.

G02(G03) X(U)... Z(W)... I... K... (R...) F... ;

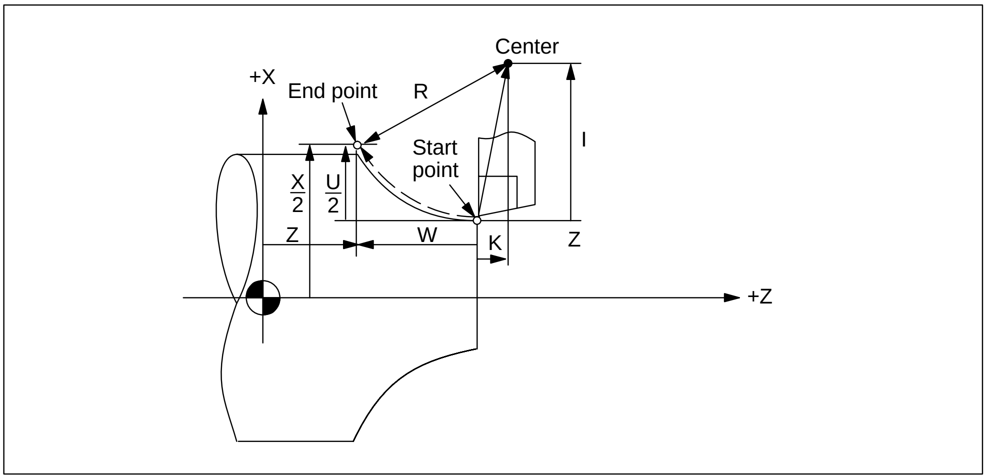


Fig. 2-5 Circular interpolation

Command format

To execute the circular interpolation, the commands indicated in Table 2-2 must be specified.

Table 2-2

Item	Address	Description
Direction of rotation	G02	Clockwise (CW)
	G03	Counterclockwise (CCW)
End point position	X (U)	X coordinate of arc end point (diametric value)
	Z (W)	Z coordinate of arc end point
	Y (V)	Y coordinate of arc end point
Distance from the start point to the center	I	Distance along the X-axis from the start point to the center of arc (radial value)
	K	Distance along the Z-axis from the start point to the center of arc
	J	Distance along the Y-axis from the start point to the center of arc
Radius of circular arc	R	Distance to the center of arc from the start point

Rotation direction

The direction of arc rotation should be specified in the manner indicated in Table 2-3.

Table 2-3

G02	Clockwise direction (CW)
G03	Counterclockwise direction (CCW)

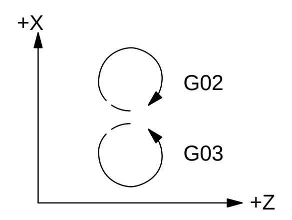


Fig. 2-6 Rotation direction of circular arc

End point

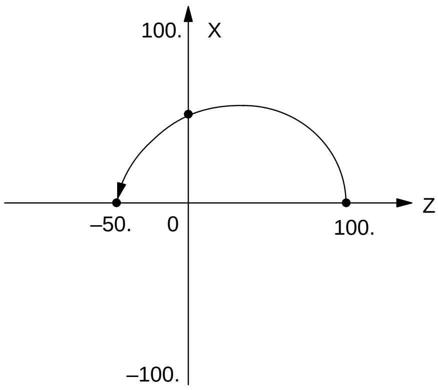
The end point can be specified in either incremental or absolute values correspon-ding to the designation of G90 or G91.

If the specified end point is not on the specified arc, the arc radius is gradually changed from the start point to the end point to generate a spiral so that the end point lies on the specified arc.

Example of programming

```
G01  Z100.  X0  F10.;
G03  Z–50.  K–100.;
```

(a) End point positioned inside the circumference



Example of programming

```
G01  Z50.  X0;
G03  Z–100.  K–50.;
```

(b) End point lying outside the circumference

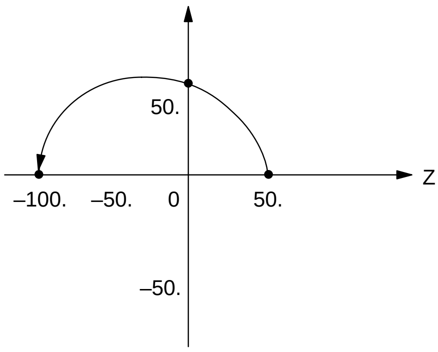


Fig. 2-7 Interpolation with end point off the specified arc

Center of arc

The center of arc can be specified in two methods – designation of the distance from the start point to the center of the arc and designation of the radius of the arc.

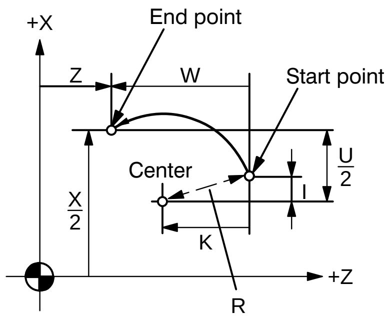


Fig. 2-8

- Specifying the distance from the start point to the center

  Independent of the designated dimensioning mode (G90 or G91), the center of an arc must be specified in incremental values referenced from the start point.

- Specifying the radius

  When defining an arc, it is possible to specify the radius by using address R instead of specifying the center of the arc by addresses I or K. This is called “circular interpolation with R designation” mode.

  For the circular arc with the central angle of 180 deg. or smaller, use an R value of “R > 0”.

  For the circular arc with the central angle of 180 deg. or larger, use an R value of “R < 0”.

Example of programming

G02 X(U) · · · Z(W) · · · R ± · · · F · · ·;

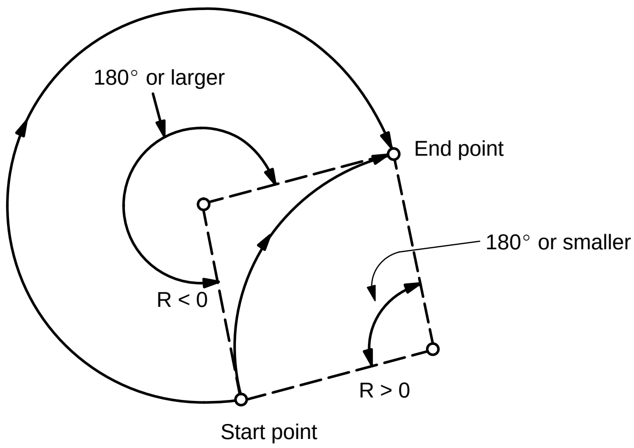


Fig. 2-9 Circular interpolation with radius R designation

Supplements to circular interpolation

A circular arc extending to multiple quadrants can be defined by the commands in a single block.

Example of programming

```
G01 Z · · · F · · ·;
G02 X60. Z–46.6 I20. K–19.596 F · · ·;
```

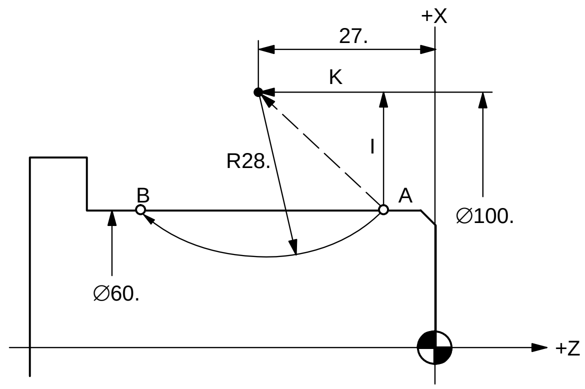


Fig. 2-10 Circular interpolation over multiple quadrants

Center of arc	(10000, 2700)
I value	$\frac{100 - 60}{2} = 20$ mm
K value	$-\sqrt{28^2 – 20^2} = -\sqrt{384} = –19.596$ mm

It is possible to insert chamfering and corner rounding blocks automatically between the following items:
– Linear interpolation and linear interpolation blocks
– Linear interpolation and circular interpolation blocks
– Circular interpolation and linear interpolation blocks
– Circular interpolation and circular interpolation blocks

Format

, C...; Champfering
, R...; Corner rounding

Explanations

A chamfering or corner rounding block is inserted whenever the above specification is added to the end of a block that specifies linear interpolation (G01) or circular interpolation (G02 or G03). It is possible to specify blocks applying chamfering and corner rounding consecutively.

Example

N10 G1 X10. Z100. F1000 G18
N20 A140 C7.5
N30 X80. Z70. A95.824, R10

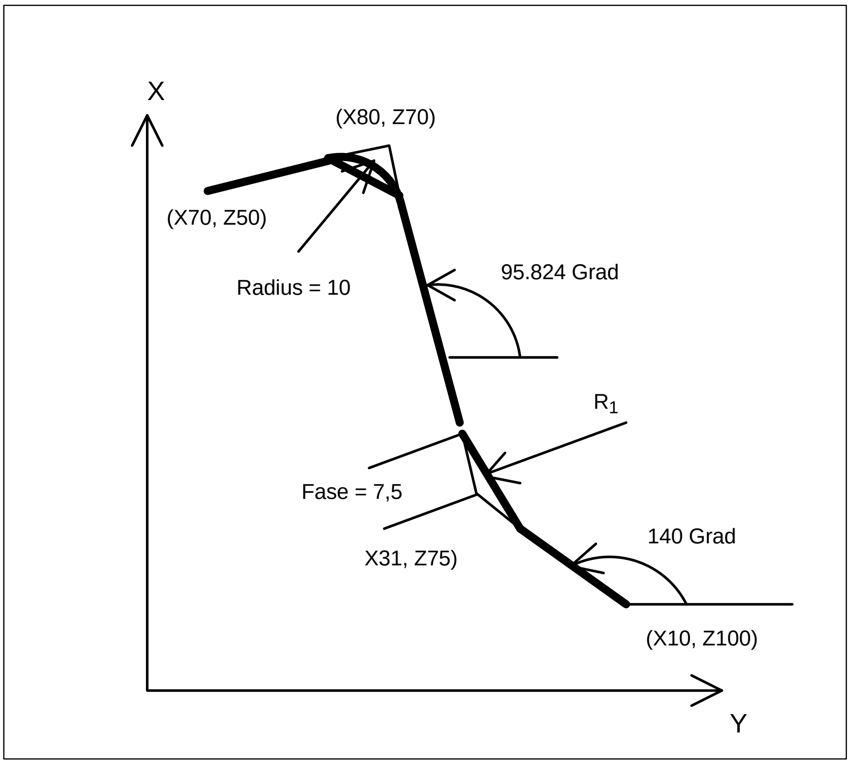


Fig. 2-11

Restrictions

ISO dialect mode

Address C is used in ISO Dialect mode both as an axis identifier and as an identifier for a chamfer on the contour.
Address R can be a cycle parameter or an identifier for the radius in a contour.
In order to distinguish between these two options, a “,” must be placed in front of the C or R address during contour definition programming.

2.1.4 Cylindrical interpolation (G07.1)

The cylindrical interpolation function allows programming of machining on a cylindrical workpiece (grooving on a cylindrical workpiece) in the manner like writing a program in a plane using the cylinder developed coordinate system. This functions allows programming both in absolute commands (C, Z) and incremental commands (H, W).

The following G code is used for cylindrical interpolation.

Table 2-4 G codes used for cylindrical interpolation

G code	Function	Group
G07.1	Cylindrical interpolation mode	18

Format

G07.1 C... r ;

Starts the cylindrical interpolation mode (enables cylindrical interpolation).

G07.1 C0 ;

The cylindrical interpolation mode is cancelled.

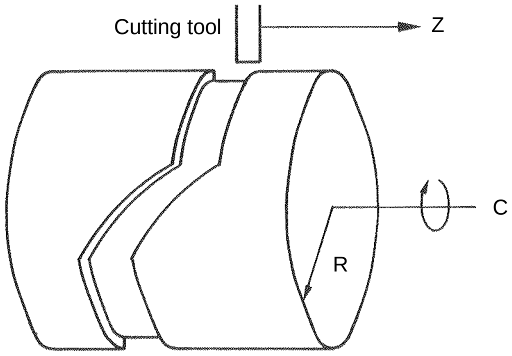
C: The rotation axis
r: The radius of the cylinder
Specify G07.1 C... r ; and G07.1 C0 ; in separate blocks.

Note

- G07.1 is based on the Siemens option TRANSMIT. The relevant machine data need to be set accordingly.
- For details refer to the manual “Extended Functions”, chapter M1, 2.1 ff.

Specify G07.1 in a block without other commands. G07.1 is a modal G code of group 18. Once G07.1 is specified, the cylindrical interpolation mode ON state remains until G07.1 C0 is commanded. When the power is turned ON or the NC is reset, the cylindrical interpolation mode OFF state is set.

Example



G00 X44.0 C0 ;	Positioning at the start point of cutting
G07.1 C45.0 ;	Cylindrical interpolation mode ON
G01 G42 Z47.5 F100 ;	Machining program
C60.0 ;	
Z32.5 C120.0 ;	
C240.0 ;	
G03 Z40.0 C249.549 R7.5 ;	
G02 Z47.5 C259.099 R7.5 ;	
G01 C360.0 ;	
Z44.0 ;	
G07.1 C0 ;	Cylindrical interpolation mode OFF
M30 ;	

Fig. 2-12 Coordinate system for cylindrical interpolation

In the cylindrical interpolation mode, program restart is not possible. If program restart is attempted from a block in the cylindrical interpolation mode, an alarm occurs. However, program restart is allowed for blocks in which the cylindrical interpolation mode blocks are included.

2.1.5 Polar coordinate interpolation (G12.1, G13.1)

The polar coordinate interpolation function allows programming of machining that is executed by the combination of tool movement and workpiece rotation in a virtual rectangular coordinate system.

In the machining accomplished by the combination of a linear axis and a rotary axis, the rotary axis is assumed to be a linear axis that is perpendicular to the linear axis. By assuming a rotary axis as a linear axis, machining an arbitrary shape that is defined by the linear and rotary axis can be programmed easily in the rectangular coordinate system. In this programming, both of absolute commands and incremental commands can be used.

Programming format

When G12.1 is specified, the polar coordinate interpolation mode is established and the virtual coordinate system is set in the plane represented by a linear– and a rotary axis with the origin of the absolute coordinate system taken as the origin of this coordinate system. Polar coordinate interpolation is executed in this plane. Note that polar coordinate interpolation starts when G12.1 is specified assuming the present position of the rotary axis to be “0”.

Note

Return the rotary axis to the origin of the absolute coordinate system before specifying G12.1.

Features of G12.1 and G13.1

The following G codes are used to turn ON/OFF the polar coordinate interpolation mode.

Table 2-5 G codes used for turning ON/OFF the polar coordinate interpolation

G code	Function	Group
G12.1	Polar coordinate interpolation mode ON	21
G13.1	Polar coordinate interpolation mode OFF	21

Specify G12.1 and G13.1 in a block without other commands.

G12.1 and G13.1 are modal G codes of group 21. Once G12.1 is specified, the polar coordinate interpolation mode ON state remains until G13.1 is specified. When the power is turned ON or the NC is reset, the G13.1 (polar coordinate interpolation mode OFF) state is set.

Note

- The Polar Coordinate Interpolation is based on the Siemens option TRACYL. The relevant machine data need to be set accordingly.
- For details refer to the manual “Extended Functions”, chapter M1, 2.2 ff.

Restrictions when selecting

- An intermediate motion block is not inserted (phases/radii).
- A spline block sequence must be terminated.
- Tool radius compensation must be deselected.
- The frame which was active prior to TRACYL is deselected by the control (corresponds to “Reset programmed frame” G500).
- An active working area limitation is deselected by the control for the axes affected by the transformation (corresponds to programmed WALIMOF).
- Continuous path control and rounding are interrupted.
- DRF offsets must have been deleted by the operator.
- In the case of cylinder generated surface curve transformation with groove wall compensation (axis configuration 2, TRAFO_TYPE_n = 513), the axis used for the correction (TRAFO_AXES_IN_n[3]) must be set to zero (y = 0) so that the groove is machined in the center of the programmed groove center line.

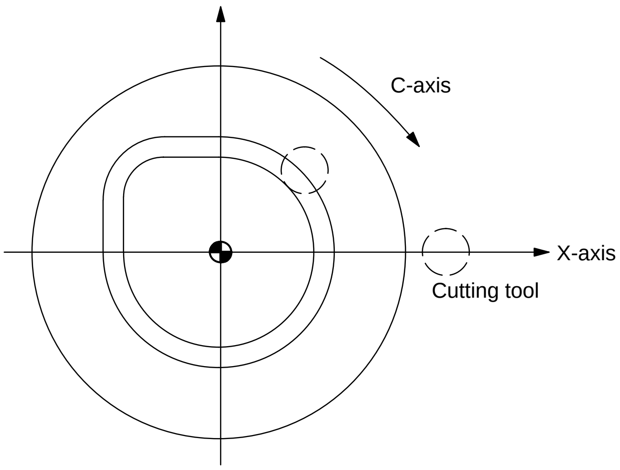
Restrictions when delecting

- The same points apply as for selection.

Restrictions when in polar coordinate interpolation

- Tool change:
  Tools may only be changed when the tool radius compensation function is deselected.
- Work offset:
  All instructions which refer exclusively to the base coordinate system are permissible (work offset, tool radius compensation). Unlike the procedure for inactive transformation, however, a work offset change with G91 (incremental dimension) is not specially treated. The increment to be traversed is evaluated in the workpiece coordinate system of the new work offset – regardless of which work offset was effective in the previous block.
- Rotary axis:
  The rotary axis cannot be programmed because it is occupied by a geometry axis and cannot thus be programmed directly as a channel axis.

Example of programming



Example of programming

```
G291 ;
G94 ;
T0101 ;
G00  X120.0  C0 ;                 ← Positioning at the cutting start point
G12.1 ;                           ← Polar coordinate interpolation mode ON
G01  G42  X40.0  F100.0 ;         }
G03  X0  C40.0  I–20.0 ;          }
G01  X–25.0 ;                     }
G03  X–40.0  C25.0  K–15.0 ;      }  Machining program using the polar coordi-
G01  C0 ;                         }  nate interpolation function
G03  X20.0  I20.0 ;               }
G01  G40  X120.0 ;                }
G13.1 ;                           ← Polar coordinate interpolation mode OFF
M30 ;
```

Fig. 2-13 Coordinate system for polar coordinate interpolation

---

Note

- Cylindrical interpolation mode must be deselected before the tool radius compensation and length compensation are deselected.

2.2 Using the thread cutting function

2.2.1 Thread cutting and continuous thread cutting (G33)

Format

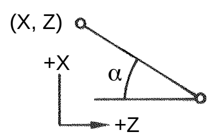
G code system A	G code system B	G code system C
G32	**G33**	**G33**

With the commands of “G... X (U)... Z (W)... F... ;”, it is possible to cut straight thread, tapered thread, or scroll thread in the lead specified by an F command to the point specified by absolute coordinate values (X, Z) or incremental coordinate values (U, W).

Direction of thread lead

The direction of thread lead specified by the F commands is indicated in Table 2-6.

Table 2-6 Direction of thread lead

		Direction of thread lead
	a ≦ 45°	Lead in the Z-axis direction should be specified.
	a > 45°	Lead in the X-axis direction should be specified.

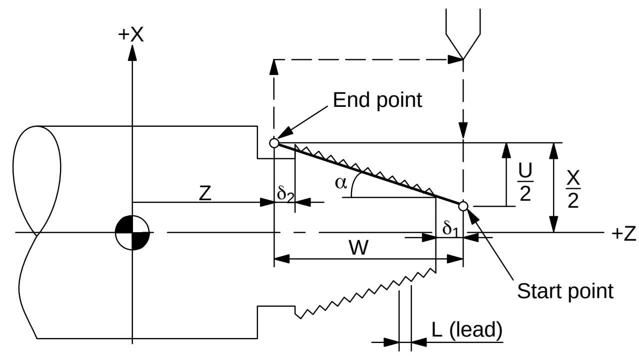


Fig. 2-14 Thread cutting

Programming formats

G33 X... Z... F... ;

X, Z : End point
F_ : Lead of the long axis
(always radius programming)

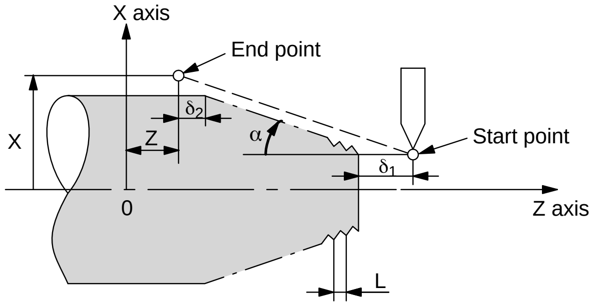


Fig. 2-15

Example of programming for cutting straight thread (G code system A)

Thread lead L = 5.0 mm
δ_1 = 5.0 mm
δ_2 = 3.0 mm
Depth of cut per pass = 1.0 mm

```
G00    U -42. ;
G32    W -68.  F5.0 ;
G00    U  42. ;
       W  68. ;
       U-44. ;
G32  W-68. ;
G00  U  44. ;
  .
  .
```

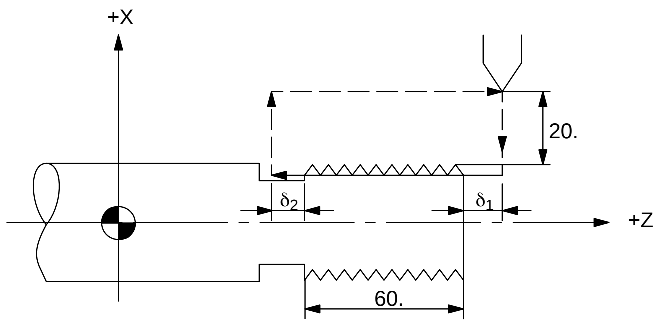


Fig. 2-16 Example of programming for cutting straight thread

Example of programming for cutting tapered thread (G code system A)

Thread lead	L	= 4.0 mm
	δ_1	= 3.0 mm
	δ_2	= 2.0 mm
Depth of cut per pass		= 1.0 mm

```
G00       X13.
G32       X38.  W-35.  F4.0 ;
G00       X60. ;
          W35. ;
          X11. ;
G32  X36.  W–35. ;
G00  X60. ;
   .
   .
```

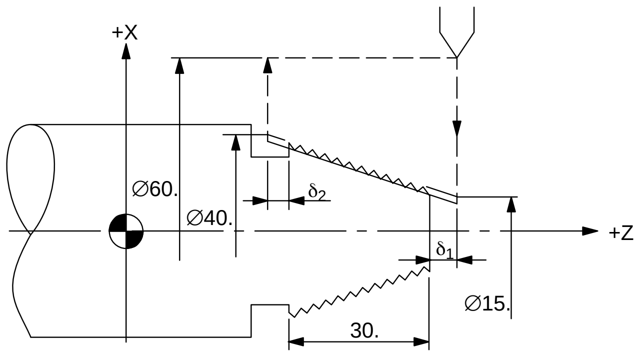


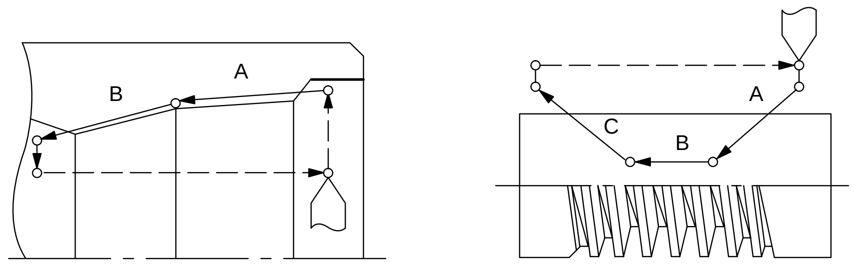
Fig. 2-17 Example of programming for cutting tapered thread

2.2.2 Continuous thread cutting

Since the NC has buffer register, designation for continuous thread cutting is possible. In addition, continuous threads can be cut smoothly because the block-to-block pause time is “0” for thread cutting command blocks.

Example of programming

```
G33  X (U) · · · Z (W) · · · F  · · · ;      ←  A
G33  X (U) · · · Z (W) · · · ;               ←  B
G33  X (U) · · · Z (W) · · · ;               ←  C
   .
   .
   .
```



(a) Reinforced pipe coupling

(b) Worm screw

Fig. 2-18 Continuous thread cutting

Note

If designation of thread lead (F) is changed during thread cutting cycle, lead accuracy is lost at joints of blocks. Therefore, thread lead designation must not be changed during thread cutting cycle.

If continuous thread cutting is specified, M codes must not be specified. If an M code is specified, the cycle is suspended at the specified block and continuous thread cannot be cut.

Margin for incomplete thread portions (δ1, δ2)

At the start and end of thread cutting, lead error is generated. Therefore, margins δ_1 and δ_2 should be given at the start and end portions in thread cutting.

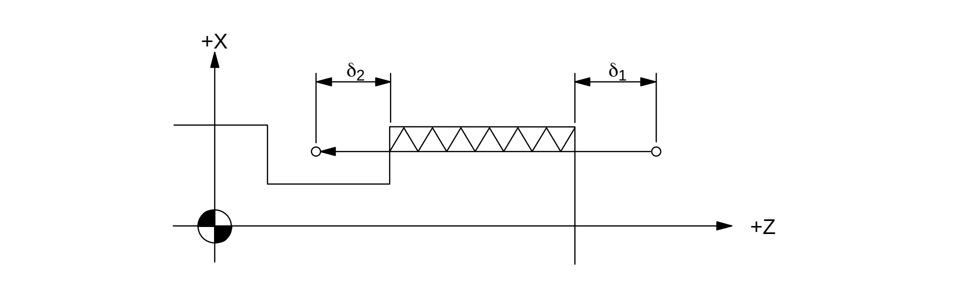


Fig. 2-19 Margins for incomplete threads

Note

Keep the spindle speed at the same value until one thread is cut. If the spindle speed is not maintained constant, accuracy could be lost due to servo lag.

Note

During thread cutting, override operation and feed hold operation are disregarded.

If G33 is specified in the G94 (feed per minute) mode, an alarm occurs.

2.2.3 Multiple-thread cutting (G33)

G code system A	G code system B	G code system C
G32	G33	G33

Multiple-thread cutting (multiple threads in a lead) is possible without shifting the thread cutting start point. In thread cutting operation, axis feed starts in synchronization with the start-point pulse (1 pulse/turn) output from the spindle pulse generator attached to the spindle. Therefore, the thread cutting start point is always at the same point on the workpiece circumference. In multiple-thread cutting operation, axis feed starts when the spindle rotates by a certain angle after the output of the start-point pulse from the spindle pulse generator.

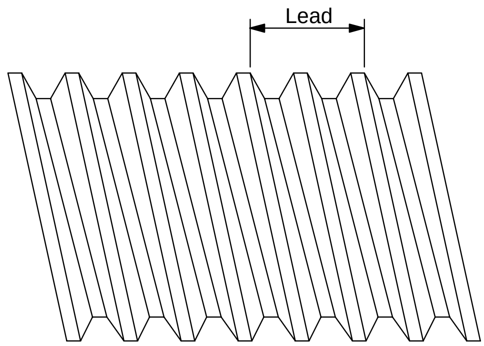


Fig. 2-20 Double-start thread

Format

With the commands of “G... X (U)... Z (W)... F... Q... ;”, the spindle rotates by the angle specified by address Q after the output of the start-point pulse of the spindle pulse generator. After that thread cutting starts toward the point specified by X (U) and Z (W) at the lead specified by an F command.

Table 2-7 Address Q specified in multi–thread cutting

Least input increment	:	0.001°
Programmable range	:	0 ≦ B < 360.000

If decimal point input is used, "Q1." is equal to 1° (Q1. = 1°). Q commands are non-modal and valid only in the specified block.

Number of threads and Q command

In general, the thread cutting start points lie on the workpiece circumference; the intervals of these points are calculated by dividing 360° by the number of threads. Examples of multiple threads (double-start, triple-start, and quadra-start threads) are shown in Fig. 2-21.

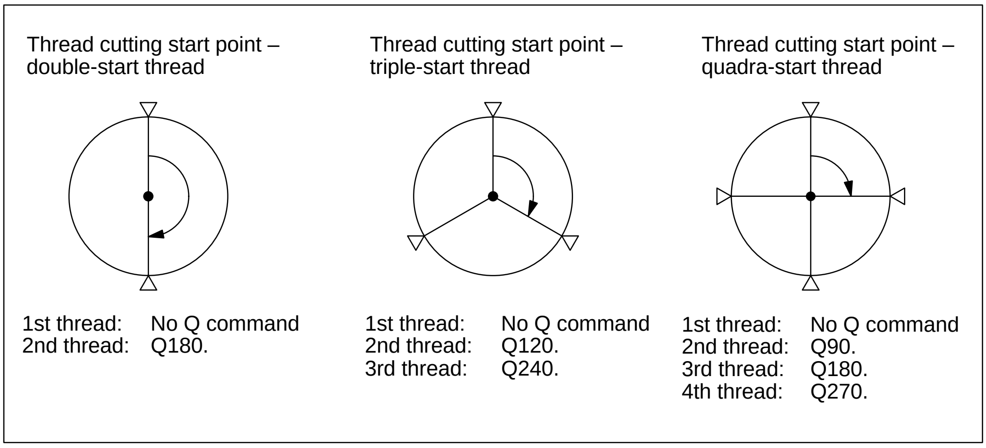


Fig. 2-21 Number of threads and Q commands

Spindle rotating angle from start-point pulse specified by Q command (G code system A)

Example of programming

```
G00  U · · · ;
G32  W · · · F · · · ;
G00  U · · · ;
W · · · ;
U · · · ;
G32  W · · · ;
   .
   .
   .
G00  U · · · ;
G32  W · · · Q180. ;
```

Thread cutting of thread A

```
G00  U · · · ;
W · · · ;
U · · · ;
G32  W · · · Q180. ;
   .
   .
   .
```

Thread cutting of thread B

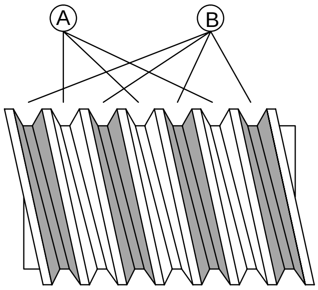


Fig. 2-22   Spindle rotation angle from start-point pulse by Q command

---

Note

If a Q command is specified for multiple-thread cutting, continuous thread cutting is not possible.

G33W   Q90

G33W   ← . . . Since the operation is suspended at this block to wait for the start-point pulse, continuous thread cannot be cut.

The spindle rotation angle from the start-point pulse is specified using a Q command (0 to 360°) disregarding of the spindle rotating direction.

2.2.4 Variable lead thread cutting (G34)

Format

G code system A	G code system B	G code system C
G34	**G34**	**G34**

G34 X... Z... F... K... ;

With the commands of “G34 X (U)... Z (W)... F... K... ;”, variable lead thread can be cut; thread lead variation per one spindle rotation is specified by address K.

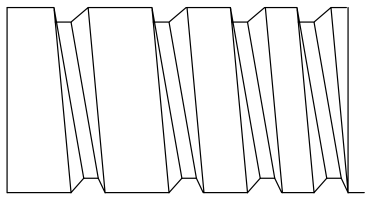


Fig. 2-23 Variable lead thread

Table 2-8 Upper limit of feedrate at end point

	Upper limit
mm output	500 mm/rev
inch output	50 inch/rev

$S \times (F + \frac{K}{2} + KN) \leqq$ max. cutting feedrate

Feedrate at end point

Specify the commands so that the feedrate at the end point will not be a negative value.

$(F + \frac{K}{2})^2 + 2KW > 0$

Note

In the continuous block thread cutting for variable lead thread cutting, distribution of command pulses is interrupted at joints between blocks.

If a K command is outside the programmable range, an alarm occurs.

If address Q is designated in the G34 block, an alarm occurs.

2.3 Reference point return

2.3.1 Automatic return to reference point (G28)

Format

G28 X... Z... ;

With the commands of "G28 X(U)... Z(W)... (C(H)... Y(V)...);", the numerically controlled axes are returned to the reference point. The axes are first moved to the specified position at a rapid traverse rate and then to the reference point automatically. The axes not designated in the G28 block are not returned to the reference point.

In case incemental encoders are used, manual reference point return needs to be carried out before using G28.

Reference position

The reference position is a fixed position on a machine tool to which the tool can easily be moved by the reference position return function. For example, the reference position is used as a position at which tools are automatically changed. Up to four reference positions can be specified by setting coordinates in the machine coordinate system in MD 34000, REFF_SET_POS.

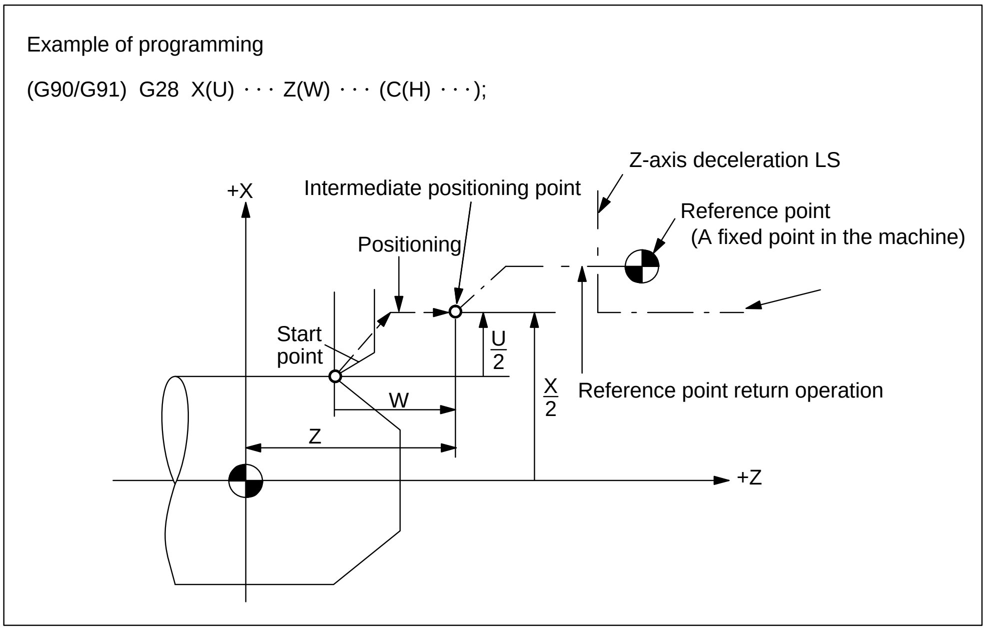


Fig. 2-24 Reference point return

Reference point return operation

Reference point return operation is the series of operations in which the axes return to the reference point after the reference point return operation has been started manually.

The reference point return is executed in the following manner.

- After the positioning at the intermediate positioning point B, the axes return directly to the reference point at a rapid traverse rate. The axes can be returned to the reference point in a shorter time compared to the normal reference point return operation that uses a deceleration limit switch for the individual axes.
- Even if point B is located outside the area in which reference point return is allowed, the reference point return specification allows the axes to return to the reference point.
- Automatic reference point return is valid only when reference point return is called by G28, and it does not influence manual reference point return operation.

Note

- Before specifying the G28 command, the tool position offset mode and nose R offset mode should be canceled. If the G28 command is specified without canceling these modes, they are canceled automatically.

2.3.2 Reference point return check (G27)

Format

G27 X... Z... ;

This function checks whether the axes are correctly returned to the reference point at the completion of the part program which is created so that the program starts and ends at the reference point in the machine by specifying the commands of "G27 X(U)... Z(W)... (C(H)... Y(V)...);".

In the G27 mode, the function checks whether or not the axes positioned by the execution of these commands in the simultaneous 2-axis control mode are located at the reference point. For the axes not specified in this block, positioning and check are not executed.

Operation after the check

When the position reached after the execution of the commands in the G27 block agrees with the reference point, the reference point return complete lamp lights. The automatic operation is continuously executed when all of the specified axes are positioned at the reference point. If there is an axis that has not been returned to the reference point, reference point return check error (alarm 61816, "axes not reference") occurs and the automatic operation is interrupted.

Supplements to the reference point return check command and other operations

- If G27 is specified in the tool position offset mode, positioning is made at the position displaced by the offset amount and the positioning point does not agree with the reference point. It is necessary to cancel the tool offset mode before specifying G27. Note that the tool position offset function is not canceled by the G27 command.
- The reference point return check is not executed if G27 is executed in the machine lock ON state.

2.3.3 Second to fourth reference point return (G30)

Format

G30 Pn X... Z... ;

With the commands of “G30 Pn X(U)... Z(W)... (C(H)... Y(V)...);”, the axes are moved to P2 (second reference point), P3 (third reference point*), or P4 (fourth reference point) in the simultaneous 3-axis control mode after the positioning at the specified intermediate positioning point. If “G30 P3 U–40. W30.;” is specified, the X- and Z-axis return to the third reference point. If “Pn” is omitted, the second reference point is selected. The axes not specified in the G30 block do not move.

Reference point positions

The position of each reference point is determined in reference to the first reference point. The distance from the first reference point to each of the reference points is set for the following machine data.

Table 2-9 Reference points

2nd reference point	REFP_SET_POS[1]
3rd reference point	REFP_SET_POS[2]
4th reference point	REFP_SET_POS[3]

Supplements to the 2nd to 4th reference point return commands

- For the points to be considered to for the execution of G30, refer to the supplements in 2.3.1, “Automatic return to reference point (G28)”.
- For the execution of G30, reference point return must have been completed after power-ON either manually or by the execution of G28. If an axis for which reference point return has not been completed is included in the axes specified in the G30 block, alarm 61816 “axes not reference” occurs.

2.4 Tool retract (G10.6)

To replace the tool damaged during machining or to check the status of machining, the tool can be withdrawn from a workpiece. In fact, a machine specific sequence can be initiated. Therefore, please refer to the machine tool builders documentation for details.

Format

G10.6 X... Z... ; Activation

G10.6 ; Deactivation

X, Z :
In incremental mode, retraction distance from the position where the retract signal is turned on. In the absolute mode, retraction distance to an absolute position.

Warning

The retraction axis and retraction distance specified in G10.6 need to be changed in an appropriate block according to the figure being machined. Be very careful when specifying the retraction distance;
An incorrect retraction distance may damage the workpiece, machine, or tool.

Notes

Movement Control Commands 3

Chapter 3 describes the procedure used for setting and selecting the coordinate system and the programming for controlling the movement of a cutting tool.

3.1 The coordinate system

A tool position is clearly determined by coordinates within a coordinate system. These coordinates are defined by program axes. For example, if there are 3 program axes involved designated as X, Y, and Z, the coordinates are specified as:

X... Z...

The above command is called a dimension word.

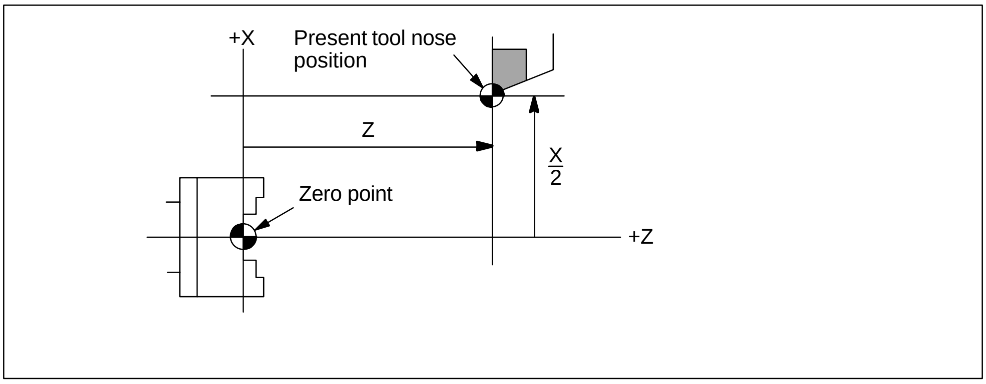


Fig. 3-1 Tool position specified by X... Z...

The following three coordinate systems are used to determine the coordinates:

1. Machine coordinate system
   G code system A, B, C: G53
2. Workpiece coordinate system
   G code system A: G50
   G code system B, C: G92
3. Local coordinate system
   G code system A, B, C: G52

3.1.1 Machine coordinate system (G53)

The machine zero point represents the point that is specific to a machine and serves as the reference point of the machine. A machine zero point is set by the MTB for each machine tool. A machine coordinate system consists of a coordinate system with a machine zero point at its origin.

A coordinate system with a machine zero point set at its origin is referred to as a machine coordinate system. By using manual reference position return after power-on, the machine coordinate system is set. Once set, the machine coordinate system remains unchanged until power–off.

Format

G53 X... Z... ;
X, Z ; absolute dimension word

How to select a machine coordinate system (G53)

Once a position has been determined in terms of machine coordinates, the tool moves to that position in rapid traverse. G53 is a one–shot G code. Thus, any command based on the selected machine coordinate system is effective only in the block where G53 is issued. The G53 command has to be determined by using absolute values. Program the movement in a machine coordinate system based on G53 whenever the tool should be moved to a machine–specific position.

Compensation function cancel

When the G53 command is specified, cancel the tool nose radius compensation and tool offset.

G53 specification right after power–on

At least one manual reference position return must be applied after power–on, since the machine coordinate system must be set before the G53 command is determined.
If an absolute position detector is attached, this is not required.

Reference

A machine coordinate system is set whenever manual reference position return is applied after power–on, so that the reference position is at the coordinate values set using MD 34100, REFP_SET_POS.

3.1.2 Workpiece coordinate system (G92)

Prior to machining, a coordinate system for the workpiece, the so called workpiece coordinate system, needs to be established. This section describes the various methods how to set, select, and change a workpiece coordinate system.

How to set a workpiece coordinate system

The following two methods can be used to set a workpiece coordinate system:

1. Using G92 (G50 in G code system A) in the program
2. Manually, using the HMI panel

Format

G92 (G50) X... Z... ;

Explanations

The coordinate system for a workpiece is set in such a way that a point on the tool, for example, the tip of the tool, is regarded as positioned to determined coordinates. Assuming “X.. Z...” is an incremental command value, the work coordinate system is defined in such a way that the current tool position is identical with the sum of the specified incremental values and the coordinates of the previous tool position.

3.1.3 How to select a workpiece coordinate system

As described below, the user may choose from predefined workpiece coordinate systems.

1. G92 (G50)

   Absolute commands work with the workpiece coordinate system once a workpiece coordinate system has been selected.

2. Selecting from workpiece coordinate systems previously set up by using the HMI panel.

   A workpiece coordinate system can be selected by determining a G code from G54 to G59, and G54 P{1...100}.
   Workpiece coordinate systems are set up subsequently to reference position return after power–on. The default coordinate system after power–on is G54.

Examples

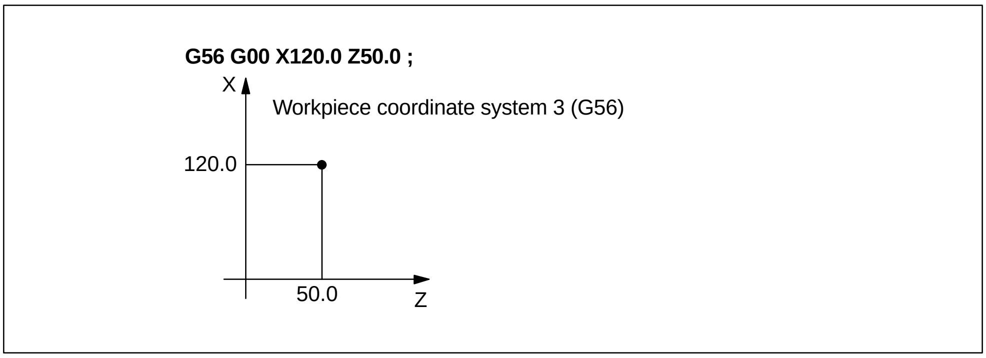


Fig. 3-2 Workpiece coordinate system G56

3.1.4 How to change a workpiece coordinate system

By changing an external workpiece zero point offset value or workpiece zero point offset value, the workpiece coordinate systems determined through G54 to G59 as well as G54 P{1 ... 100} are changed.

In order to change an external workpiece zero point offset value or workpiece zero point offset value, two methods are available.

1. IEntering data using the HMI panel
2. By program command G10 or G92

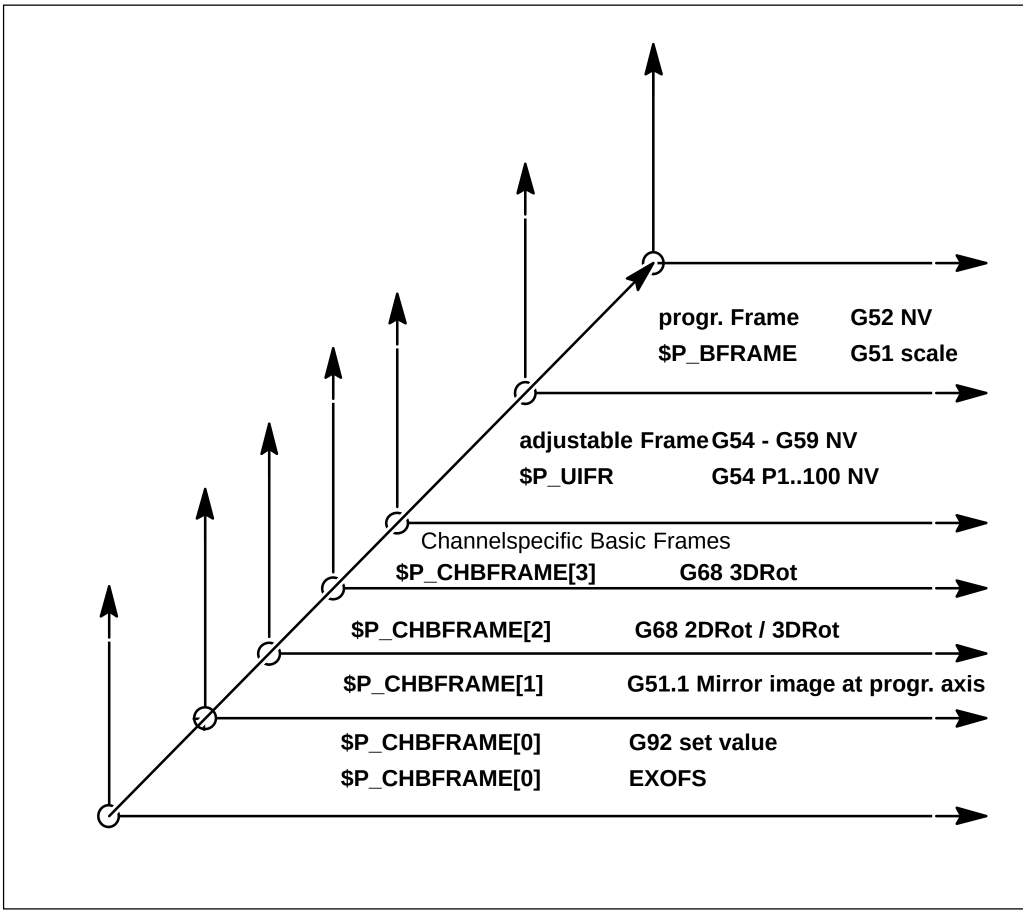


Fig. 3-3 ISO-dialect coordinate systems

Format

Changing by G10:

G10 L2 Pp X... Z... ;

p=0:	External workpiece zero point offset value (EXOFS)
p=1 to 6:	Workpiece zero point offset value correspond to workpiece coordinate system G54 to G59
X, Z:	For an absolute command (G90), workpiece zero point off–set for each axis. For an incremental command (G91), value to be added to the set workpiece zero point offset for each axis (the sum is set as the new offset).

G10 L20 Pp X... Z... ;

p=1 to 100:	Workpiece zero point offset value correspond to additional workpiece coordinate systems G54 P1 ... P100
IP:	For an absolute command (G90), workpiece zero point offset for each axis. For an incremental command (G91), value to be added to the set workpiece zero point offset for each axis (the sum is set as the new offset).

Changing by using G92

G92 X... Z... ;

Explanations

Changing workpiece coordinate systems by using G10

Each workpiece coordinate system can be changed separately by using the G10 command.

Changing workpiece coordinate systems by using G92

A workpiece coordinate system (selected with a code from G54 to G59 and G54 P{1 ...100}) is shifted to set a new workpiece coordinate system by specifying G92 X... Z.... This way, the current tool position is made to match the specified coordinates. If X, Z, is an incremental command value, the work coordinate system is defined so that the current tool position coincides with the result of adding the specified incremental value to the coordinates of the previous tool position (coordinate system shift). Subsequently, the value of the coordinate system shift is added to each individual workpiece zero point offset value. In other words, all of the workpiece coordinate systems are systematically shifted by the same value amount.

Example

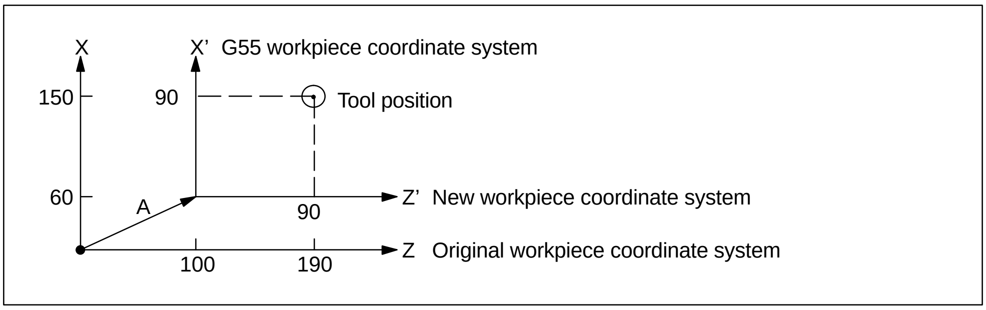


Fig. 3-4 Setting of coordinate system with incremental values (G code system A)

Note

Siemens frames and ISO dialect workpiece coordinate systems are using a common storage area. In other word, changing a frame in Siemens mode will effect the relevant workpiece coordinate system used in ISO dialect mode.

ISO Dialect mode	Siemens mode
G54	G54
G55	G55
G56	G56
G57	G57
G58	G505
G59	G506
G54 P1 ... 48	G507 ... G554
G54 P49 ... !00	G ...
G92	Basic frame

3.2 Determining the coordinate value input modes

This section describes the commands used to input coordinate values.

3.2.1 Absolute/incremental designation

Axis movement data specified following an axis address determines axis movement distance in either incremental or absolute values.

By using addresses X, Z, C, Y, U, W, H, and V, it is possible to use both incremental and absolute values.

Command format

- Absolute commands

  To specify axis movement distance in an absolute value, use addresses X, Z, and C.

  Example: X... Z... C... ;

- Incremental commands

  To specify axis movement distance in an incremental value, use addresses U, W, and H.

  Example: U... W... H... ;

- Use of both incremental and absolute commands in the same block

  It is allowed to use both incremental and absolute values in the same block.

  Example: X... W... ;
  U... Z... ;

  If addresses that represent the same axis are specified in the same block like "X... U... ;", the address specified later becomes valid.

  These G codes specify whether dimension values specified following an axis address are given in an absolute value or incremental value.

Table 3-1 Absolute and incremental commands and meaning

Address	Command value		Meaning (description)
X	Absolute	Diametric value	Position in the X-axis direction
Z		–	Position in the Z-axis direction
C		–	Position in the C-axis direction
Y		–	Position in the Y-axis direction

Table 3-1 Absolute and incremental commands and meaning, continued

Address	Command value		Meaning (description)
U	Incremental value	Diametric value	Movement distance in the X-axis direction
W		–	Movement distance in the Z-axis direction
H		–	Movement distance in the C-axis direction
V		–	Movement distance in the Y-axis direction
I	Incremental value	Radial value	X-axis direction component of the distance to the center of arc viewed from the start point of arc
K		–	Z-axis direction component of the distance to the center of arc viewed from the start point of arc
J		–	Y-axis direction component of the distance to the center of arc viewed from the start point of arc
R	Incremental value	–	Direct designation of arc radius

Since a diametric value is specified for addresses X and U, actual axis movement distance is a half the specified value.

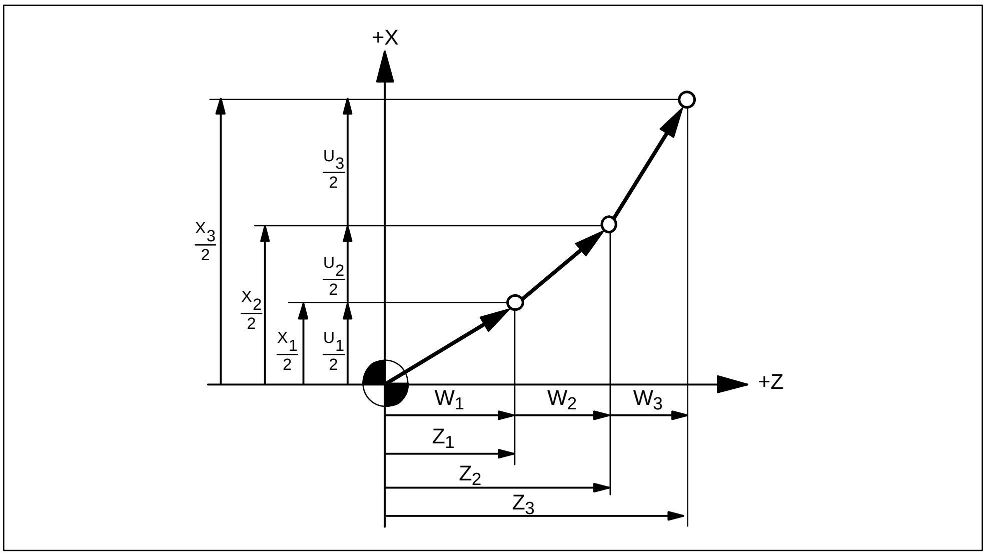


Fig. 3-5 Absolute and incremental coordinate values

Use of G90 and G91 (G code system B and C)

Table 3-2 Function of G90 and G91 commands

G code	Function	Group
G90	Absolute designation	03
G91	Incremental designation	03

Table 3-3 Valid address for G90/G91 designation

Address		G90 command	G91 command
	X, Z, C, Y	Absolute	Incremental
	U, W, H, V	Incremental	Incremental

Example: With the commands of “G91 G00 X40. Z50.;” axis movement commands are executed as incremental commands.

Auxiliary data for circular interpolation

The auxiliary circular interpolation data I, J, K, and R are always interpreted as incremental commands.

Note

It is not allowed to specify G90 and G91 in the same block. If both of these G codes are specified in the same block, the one specified later becomes valid. For example, if the commands of “G01 G90 X80. G91 Z60.;” are specified in a block, G91 specified later becomes valid and all axis movement commands (X80. and Z60.) are interpreted as incremental commands.

3.2.2 Diametric and radial commands for X-axis

To specify X-axis commands, address X or U is used and dimensions are usually specified in diametric values.

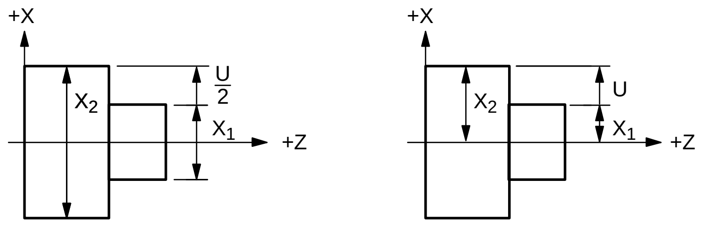


Fig. 3-6 Coordinate values

Table 1.4 Use of Diametric and Radial Designation

Item	Diametric Designation	Radial Designation
Address X command	Diametric value	Radial value
Address U command	Diametric incremental value	Radial incremental value
X-axis position display	Diametric value	
Tool position offset amount	Diametric value	
Tool coordinate data for tool coordinate system	Diametric value	
Nose R amount	Radial value	
Feedrate F in the X-axis direction	Radial value/rev, Radial value/mm	
Radius designation for circular interpolation (I, K, J, R)	Radial value	
G90 to G94, G70 to G76 Chamfering, rounding, multiple chamfering parameters	Radial value	

3.2.3 Inch/metric input designation (G20, G21)

It is possible to select the dimension unit for the input data between “mm” and “inches”. For this selection, the following G codes are used.

Table 3-4 Dimension unit selection G codes

G code	Function	Group
G20 (G70, G code syst. C)	Input in “inch” system	06
G21 (G71, G code syst. C)	Input in “mm” system	06

Command format

G20 (G70) and G21 (G71) should be specified at the beginning of a program in a block without other commands. When the G code which selects the input dimension unit is executed, the following values are processed in the selected dimension unit: subsequent programs, offset amount, a part of parameters, a part of manual operation, and display.

Supplements to the dimension unit designation commands

A parameter is used to select “inch/mm”. Therefore, the state when the power is turned ON is determined by the setting for this parameter.

If the dimension unit system should be switched over during the execution of a program, the tool position offset and nose R offset function must be canceled before the switching over of the dimension unit system.

After switching over the dimension unit system between G20 and G21, the following processing must be accomplished.

- Set the coordinate system before specifying axis move commands.
- If position data are displayed in a workpiece coordinate system, or when an external position data display unit is used, reset the present position data to “0”.

The tool offset amounts stored in memory are treated in a different manner between the G20 and G21 modes.

Table 3-5 Tool offset amounts in G20 (G70) and G21 (G71) modes

Stored offset amount	in the G20 (G70) (inch system) mode	in the G21 (G70) (mm system) mode
150000	1.5000 inch	15.000 mm

3.3 Time-controlling commands

3.3.1 Dwell (G04)

It is possible to suspend the execution of axis move commands specified in the next block for the specified length of time (dwell period).

Format

G04 X...; or G04 P...;

X: Dwell time (decimal point representation)
P: Dwell time (integer representation)

There are two different methods how to execute the programmed dwell time:
MD $MC_EXTERN_FUNCTION_MASK
Bit2 = 0: Dwell always in seconds [s]
Bit2 = 1: Dwell in seconds (G94 mode) or spindle rotations (G95 mode)

The execution of programmed commands is suspended for the length of time in the feed per minute mode (G94) and a number of spindle rotations in the feed per revolution mode (G95) determined by the address X or P by specifying G04 X...; or G04 P...;

The block used to determine dwell is not allowed to contain commands other than G04 commands.

Example

G94 G04 X1000 ;

Standard notation:	1000 * 0.001 = 1s dwell
Pocket calculator notation:	1000s dwell

G95 G04 X1000 ;

Standard notation:	1000 * 0.001 = 1 rev dwell
Pocket calculator notation:	1000 rev dwell

The use of standard notation or pocket calculator notation is decided by MD EXTERN_FLOATINGPOINT_PROG.

3.4 Tool offset functions

The following three kinds of tool offset functions are provided: tool position offset function, nose R offset function, and tool radius offset function.

3.4.1 Tool offset data memory

The memory area where the data of the offset functions and coordinate system setting is called the tool offset data memory.

3.4.2 Tool position offset

The tool position offset function adds the offset amount to the coordinate value specified in a program when a tool offset number is specified and moves the nose R to the position obtained by the addition.

3.4.3 Tool nose radius compensation function (G40, G41/G42)

Since the nose of a cutting tool is rounded, overcuts or undercuts occur in taper cutting or arc cutting since offset simply by the tool position offset function is not satisfactory. How such problems occur is shown in Fig. 3-7. The tool nose radius compensation function called by G41 and G42 compensates for an error to finish the workpiece to the programmed shape.

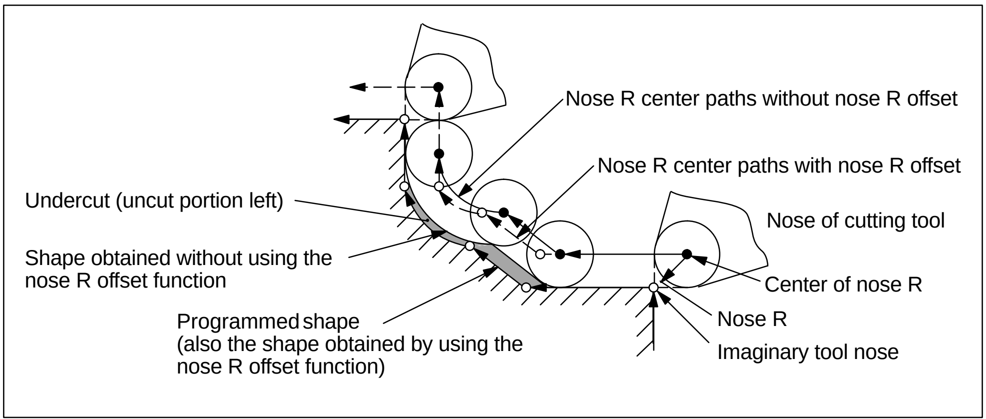


Fig. 3-7 Tool nose radius compensation function

Nose R offset amount

The term “Nose R offset amount” means the distance from the tool nose to the center of nose R.

- Setting the nose R offset amount

  For the nose R offset amount, set the radius of the circle of the tool nose without a sign.

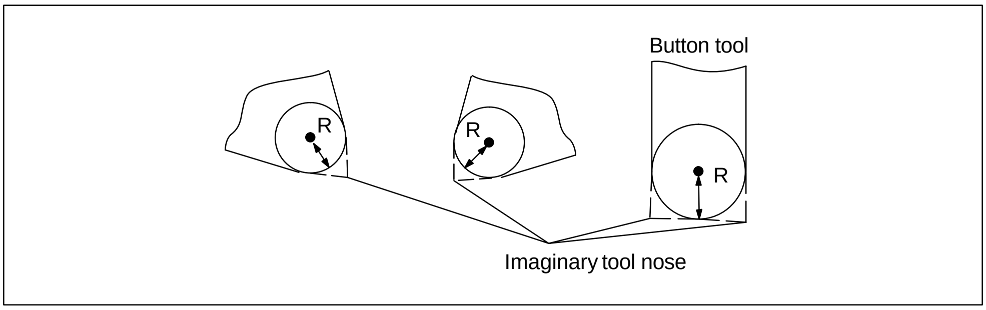


Fig. 3-8 Setting the nose R offset amount and imaginary tool nose

Designation of imaginary tool nose position (control point)

- Control point memory

  The position of the imaginary tool nose viewed from the center of the nose R is expressed using a 1-digit number, 0 to 9. This is called the control point. The control point should be written to the NC memory in advance as with the tool data.

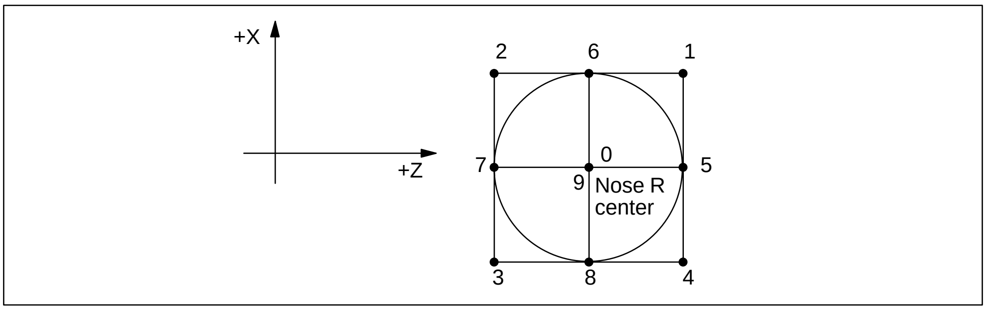


Fig. 3-9 Control point

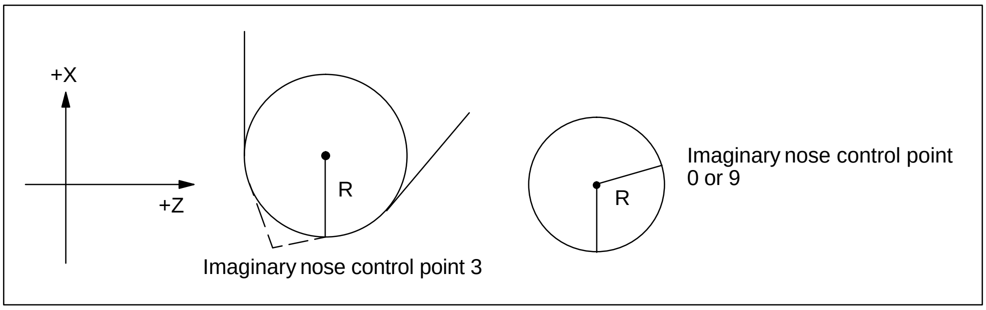
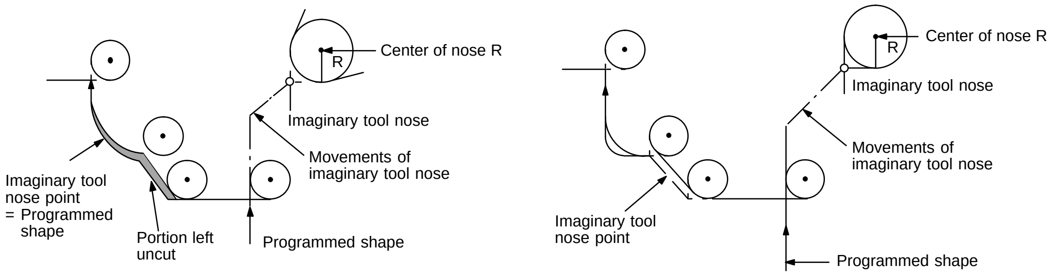


Fig. 3-10 Example of control point setting

Control points and programs

When control points 1 to 8 are used, the imaginary tool nose position should be used as the reference to write a program. Write the program after setting a coordinate system.



(a) Program without nose R offset function

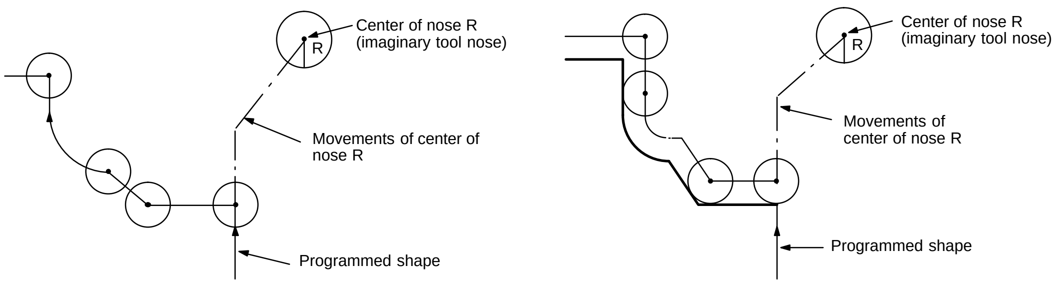
The imaginary tool nose point follows the programmed shape, causing overcuts and undercuts at tapers and arcs.

(b) Program with nose R offset function

The nose R offset function offsets the tool paths from the programmed shape to eliminated overcuts and undercuts.

Fig. 3-11 Program and tool movements for control points 1 to 8

When control points 0 or 9 is used, the center of nose R should be used as the reference to write a program. Write the program after setting a coordinate system. If the nose R offset function is not used, the program shape must not be different from the shape to be machined.



(a) Program without nose R offset function

The center of nose R follows the programmed shape. Therefore, if the coordinate system is set in reference to the center of nose R, the shape to be programmed must be different from the shape to be machined.

(b) Program with nose R offset function

As with the program (b) in Fig. 3-11, appropriate offsetting is made to finish the shape accurately without overcuts and undercuts.

Fig. 3-12 Program and tool movements for control point 0 or 9

Nose R offset commands

- Designation of tool offset amount

  The tool offset amount is called by T command.

- Designation of tool nose radius compensation function ON

  To designate ON/OFF of the tool nose radius compensation function use the following G codes.

Table 3-6 G codes used for turning ON/OFF tool nose radius compensation function

G code	Function	Group
G40	Tool nose radius compensation cancel	07
G41	Tool nose radius compensation, left (nose R center is at the left side)	07
G42	Tool nose radius compensation, right (nose R center is at the right side)	07

G40 and G41/G42 are modal G codes in group 07, and once designated the specified G code mode remains valid until another G code is specified. When the power is turned ON or the CNC is reset, the G40 mode is set.

To enter the tool nose radius compensation mode, specify either G41 or G42 with a T code.

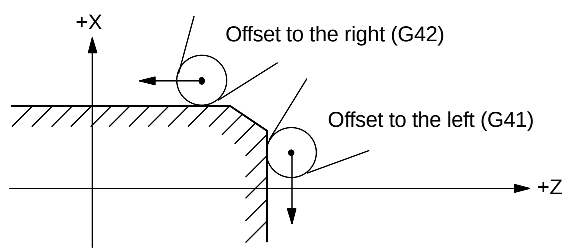


Fig. 3-13 Designation of tool nose radius compensation direction

The tool nose radius compensation direction can be changed over between “to the right” and “to the left” by specifying G41 or G42 during the execution of a program. It is not necessary to cancel the nose R offset mode by specifying G40 or deselecting the tool before changing over direction of offset. To cancel the tool nose radius compensation mode, specify G40.

Outline of tool nose radius compensation movements

Fig. 3-14 shows how the tool nose radius compensation function is executed.

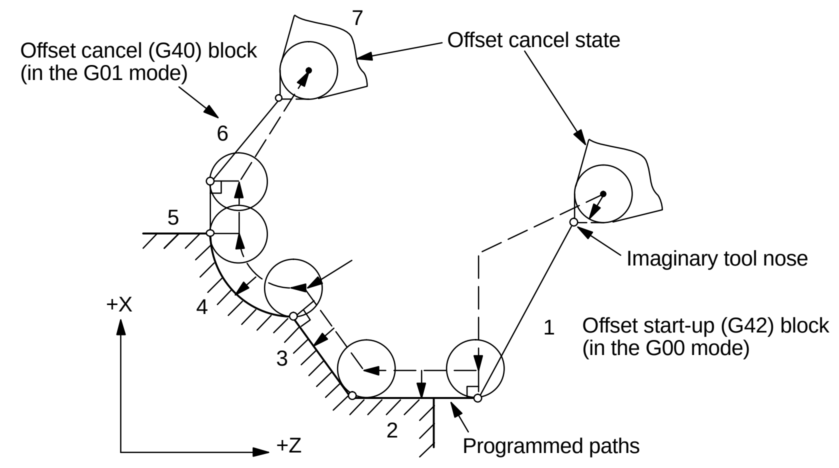


Fig. 3-14 Outline of tool nose radius compensation movements (G42, control point 3)

- In the offset cancel state, the imaginary tool nose position 7 agrees with the point specified in the program 1.
- In the offset mode, the center of nose R is offset by the nose R amount from the programmed paths and it follows the offset paths. Therefore, the imaginary tool nose position does not agree with the programmed point. Note that the present position display shows the position of the imaginary tool nose.
- At the offset start-up block 1 and cancel block 6, the movements to link the compensation mode and compensation cancel mode are inserted. Therefore, special attention must be paid for specifying the offset start-up and cancel blocks.

Note

1. The nose R offset function can be used for circular interpolation specified by radius designation.
2. It is allowed to specify a subprogram (M98, M99) in the offset mode. The nose R offset function is applied to the programmed shape which is offset by the tool position offset function.

Entering the offset mode

The compensation mode is set when both of a tool offset (by a T code) and G41/G42 are specified. More precisely, the compensation mode starts at the time when the AND condition of a T code and a G code is satisfied. There are no differences whichever of these codes is specified first (see Fig. 3.24). The initial movement when the offset mode starts in the offset cancel state is called the start-up motion.

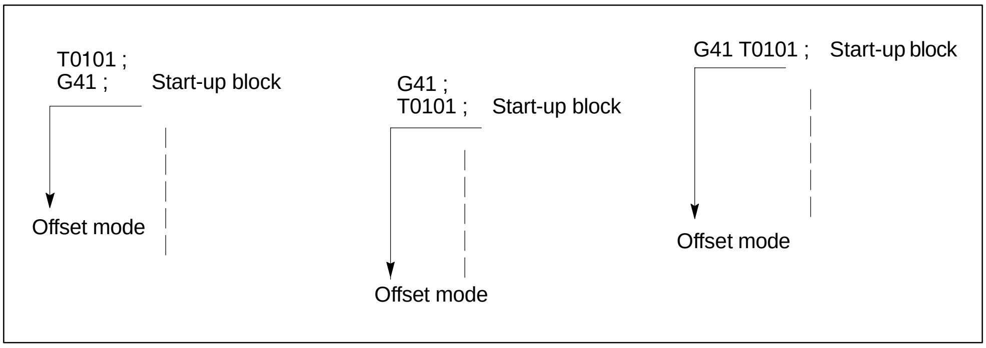


Fig. 3-15 Compensation mode entry methods

Example of programming

```
N1  G92  X140.  Z20. ;
N2  G00  S1700  M03  T0202 ;
N3  (G00)  G42  X0  Z5. ;           ———— Nose R offset start-up block
N4  G01  Z0  F0.2 ;
N5  X20. ;
N6  Z–20. ;
N7  X30.  W–15.  S1100 ;
N8  G1  W–20.  R3. ;                ———— (Rounding)
N9  G1  X50.  K–3.  S700 ;          ———— (Chamfering)
N10  G01  Z–70. ;
N11  G02  X90.  Z–90.  R20.  S360 ; ———— (Arc designated by R command)
N12  G01  X110.  S300 ;
N13  G04  U0 ;                      ———— (Dwell for making sharp edge)
N14  (G01)  Z–110. ;
N15  X120. ;
N16  G00  X140.  Z30. ;
N17  T0  G40 ;                      ———— Nose R offset cancel block
...
```

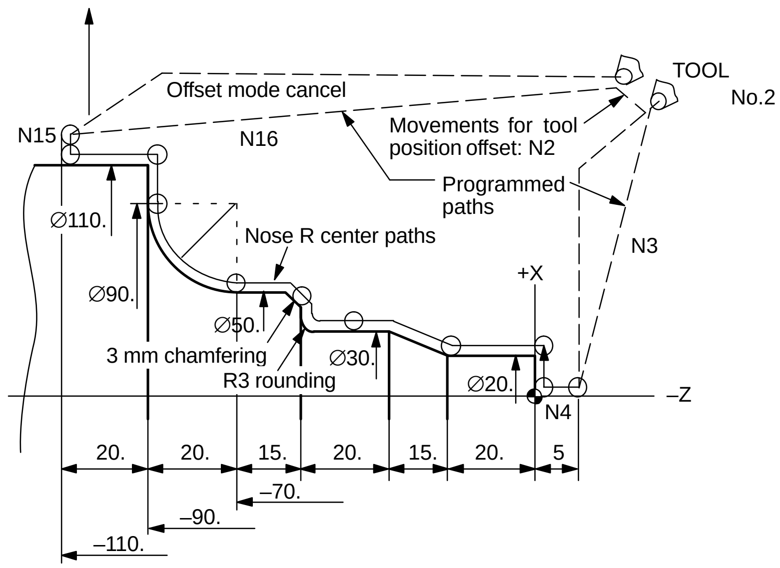


Fig. 3-16 Example of programming

3.5 Spindle function (S function)

3.5.1 Spindle command (S5-digit command)

A spindle speed can be directly specified by entering a 5-digit number following address S. The unit of spindle speed is “r/min”. If an S command is specified with M03 (spindle forward rotation) or M04 (spindle reverse rotation), the program usually advances to the next block only after the spindle has reached the speed specified by the S command. For details, refer to the instruction manuals published by the machine tool builder.

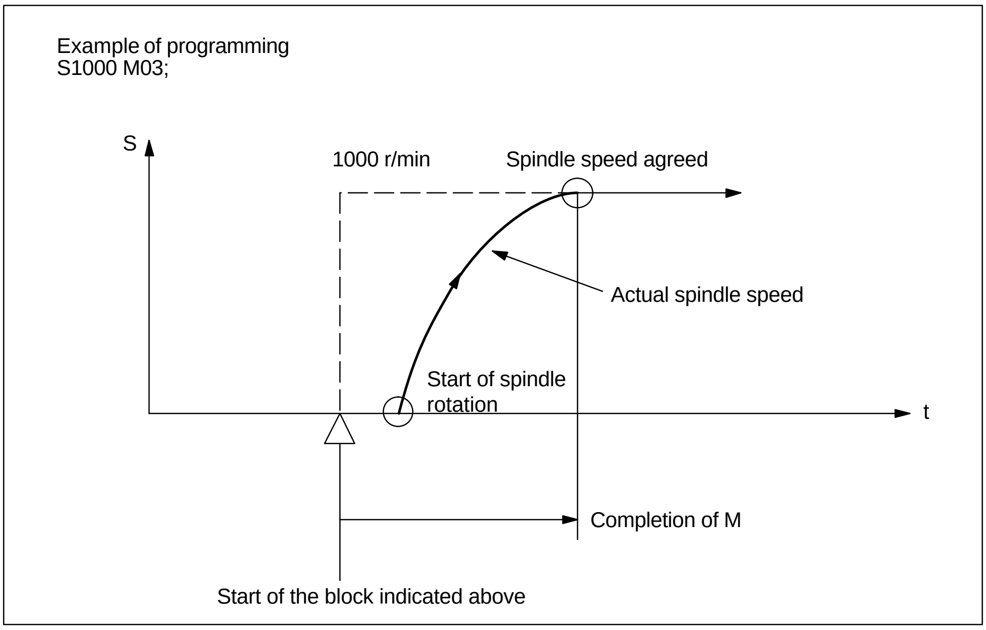


Fig. 3-17 Spindle speed command

- For the output of S5-digit commands, it is possible to add the control function implemented by the PLC can be added by the NC. In this case, it is possible to set the spindle speed in manual operation to the speed that corresponds to the specified S command by using the rotary switch on the machine operation panel. For details, refer to the manuals published by the machine tool builder.
- An S command is modal and, once specified, it remains valid until another S command is given next. If the spindle is stopped by the execution of M05, the S command value is retained. Therefore, if M03 or M04 is specified without an S command in the same block, the spindle can start by using the S command value specified before.
- The lower limit of an S command (S0 or an S command close to S0) is determined by the spindle drive motor and spindle drive system, and it varies with each machine. Do not use a negative value for an S command. For details, refer to the instruction manuals published by the machine tool builder.

- Spindle speed override is possible for the specified S code.
- For the machine that has the gearbox with which gear range can be changed by specifying an M code, specify the M code to select an appropriate gear range before specifying an S code. For the number of gear ranges and the available spindle speed range in the individual gear ranges, refer to the manuals published by the machine tool builder.

3.5.2 Constant surface speed control (G96, G97)

The G codes indicated in table 3–7 are used for the constant surface speed control function. G96 and G97 are modal G code of 02 group.

Table 3-7 G codes for constant surface speed control

G code	Function	Group
G96	Constant surface speed control ON	02
G97	Constant surface speed control cancel	02

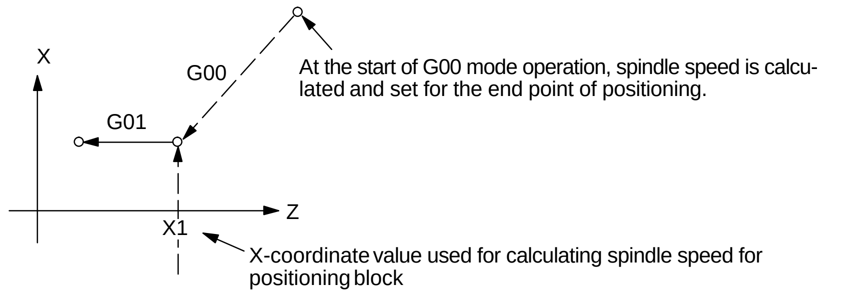
Constant surface speed control ON (G96)

With the commands of “G96 S... (M03) ;”, the workpiece surface speed is designated by a maximum 5-digit number following address S. The unit used for specifying the surface speed is indicated in Table 3-8.

Table 3-8 Units of surface speed designation

	Unit
mm	m/min
inch	ft/min

In the constant surface speed control mode, the NC assumes the present value of the X-axis as the workpiece diameter and calculates the spindle speed every 32 msec so that the specified surface speed is maintained. The specified surface speed can be changed by specifying a required S code in the following blocks.




```
N4  G92  S1500 ;
N6  G96  S150  M03 ;
N7  G00  X40.  Z5. ;
N8  G01  Z0  F0.15 ;
N9  X80.  Z–30. ;
N10  W–10. ;
N11  G2  X120.  W–20.  R20. ;
N12  G01  U10. ;
N13  G97  S500 ;
...
```

Spindle speed clamp value (N4)
Designation of surface speed of 150 m/min (N6)
Constant surface speed control mode (N7–N12)
Cancel of constant surface speed control (N13)

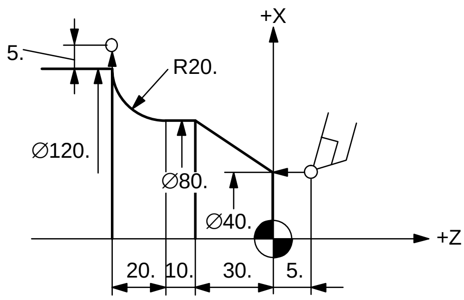


Fig. 3-18 Constant surface speed

Canceling the constant surface speed control (G97)

Specify a spindle speed (r/min) by a maximum of 5-digit number following address S with the commands “G97 S... (M03) ;”. The constant surface speed control mode is canceled, and the spindle rotates at the specified spindle speed.

Spindle gear range selection

For the machine that has the gearbox with which gear range can be changed by specifying an M code, specify the M code to select an appropriate gear range before specifying G96. For details, refer to the manuals published by the machine tool builder.

Example of programming

```
            .
            .
            .
N8  Mxx ;   <------------   M code for selecting gear range
N9  G96  S100  M03 ;         (Example: Gear range No. 4)
            .
            .
            .
```

Fig. 3-19

Supplements to the constant surface speed control commands

- To execute the constant surface speed control, set the G92 coordinate system or a workpiece coordinate system so that the X-coordinate value of the centerline of the spindle will be “0” and program the operation on this coordinate system. With this, X-coordinate values in a program represent the diameter of workpiece accurately.

3.5.3 Rotary tool spindle selection function

*** under construction ***

The constant surface speed control is not valid for the rotary tool spindle.

3.6 Tool function (T function)

The tool function has various command designation types. For details, refer to the instruction manuals published by the machine tool builder.

3.7 Miscellaneous function (M function)

The miscellaneous function is specified by a maximum of a three–digit number following address M. With the exception of specific M codes, the functions of M00 to M97 codes are defined by the machine tool builder. Therefore, for details of the M code functions, refer to the instruction manuals published by the machine tool builder.

The M codes specific to the NC are described below.

3.7.1 M codes relating to stop operation (M00, M01, M02, M30)

When an M code relating to stop is executed, the NC stops buffering. Whether spindle rotation, coolant discharge or another operation stops in response to the execution of such an M code is determined by the machine tool builder. For details, refer to the instruction manuals published by the machine tool builder. For these M codes, a code signal is output independently in addition to M2-digit BIN code.

M00 (program stop)

If M00 is specified during automatic operation, automatic operation is interrupted after the completion of the commands specified with M00 in the same block and the M00R signal is output. The interrupted automatic operation can be restarted by pressing the cycle start switch.

M01 (optional stop)

If M01 is executed with the optional stop switch ON, the same operation as with M00 is executed. If the optional stop switch is OFF, M01 is disregarded.

M02 (end of program)

M02 should be specified at the end of a program. When M02 is executed during automatic operation, automatic operation ends after the commands specified with M02 in the same block have been completed. The NC is reset. The state after the end of a program varies with each machine. For details, refer to the instruction manuals published by the machine tool builder.

M30 (end of tape)

Normally, M30 is specified at the end of tape. When M30 is executed during automatic operation, automatic operation ends after the commands specified with M30 in the same block have been completed. The NC is reset and the tape is rewound. The state after the execution of M30 varies with each machine. For details, refer to the instruction manuals published by the machine tool builder.

Note

When M00, M01, M02, or M30 is specified, the NC stops buffering. For these M codes, the NC output the independent decode signal in addition to the M2-digit BIN code.

Note

Refer to the manuals published by the machine tool builder concerning whether or not the spindle and/or coolant supply is stopped by the M00, M01, M02, and M30.

3.7.2 Internally processed M codes

M codes in the range of M98 and M99 are processed by the NC.

Table 3-9 Internally processed M codes

M code	Function
M98	Subprogram call
M99	End of subprogram

3.7.3 General purpose M codes

Other general M codes

The functions of the M codes other than the specific M codes are determined by the machine tool builder. The representative use of several general M codes is given below. For details, refer to the instruction manuals published by the machine tool builder. If an M code is specified with axis move commands in the same block, whether the M code is executed with the axis move commands simultaneously or it is executed after the completion of the axis move commands is determined by the machine tool builder. For details, refer to the instruction manuals published by the machine tool builder.

Table 3-10 Other general M codes

M code	Function
M03	Spindle start, forward direction
M04	Spindle start, reverse direction
M05	Spindle stop
M08	Coolant ON
M09	Coolant OFF

Designation of multiple M codes in a single block

It is possible to specify up to five M codes in a single block. The specified M codes and sampling output are output at the same time. Concerning the combinations of the M codes that can be specified in the same block, refer to the manuals published by the machine tool builder for restrictions on them.

Notes

Enhanced Level Commands 4

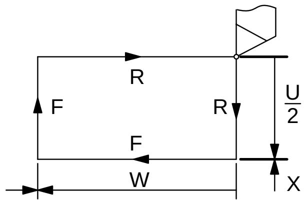
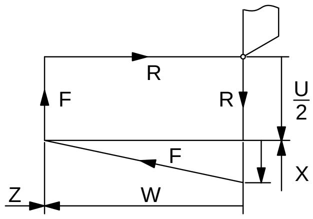
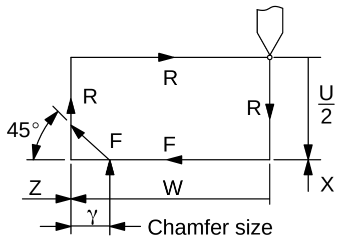
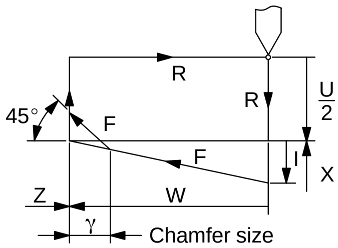
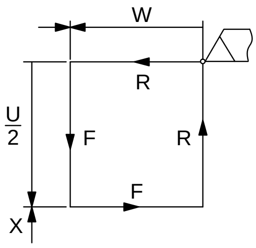
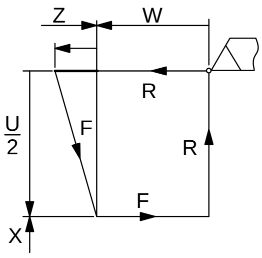
Chapter 4 describes the program support functions, automation support functions, and macroprograms.

4.1 Program support functions (1)

4.1.1 Canned cycles

The canned cycle function defines the four block operations of basic cutting operation, in-feed, cutting (or thread cutting), retraction, and return, in one block (to be called as one cycle).

Table 4-1 Table of canned cycles

G code	Straight cycle	Taper cycle
G90 Cutting cycle (OD cutting)	G90 X (U)· · · Z (W)· · · F · · ·; 	G90 X (U)· · · Z (W)· · · R· · · F · · ·; 
G92 Thread cutting cycle	G92 X (U)· · · Z (W)· · · F · · · ; 	G92 X (U)· · · Z (W)· · · R· · · F · · · ; 
G94 Face cutting cycle	G94 X (U)· · · Z (W)· · · F · · · ; 	G94 X (U)· · · Z (W)· · · R· · · F · · · ; 

Cutting cycle commands

The cutting cycle is used for outside diameter (OD) cutting and has two kinds of cycles – straight cutting cycle and taper cutting cycle.

Straight cutting cycle

Format

G.. X... Z... F... ;

G code system A	G code system B	G code system C
G90	G77	G20

With the commands of “G... X(U)... Z(W)... F... ;”, straight cutting cycle is executed as indicated by sequence 1 to 4 shown in Fig. 4-1.

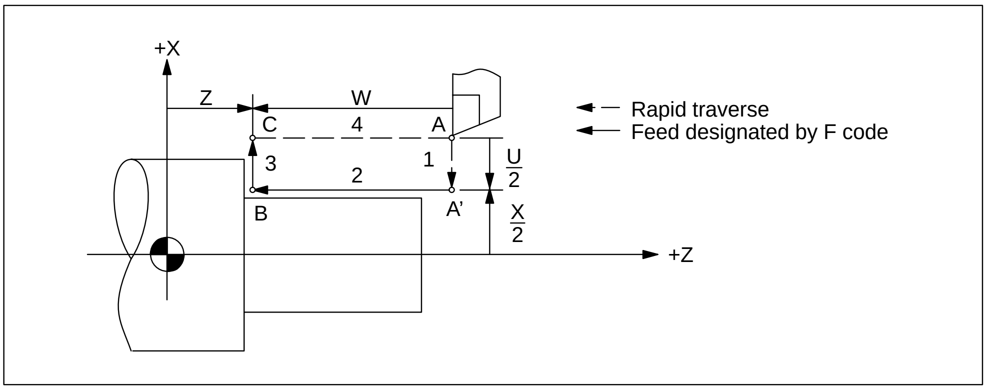


Fig. 4-1 Straight cutting cycle

Since G77 (G90, G20) is a modal G code, cycle operation is executed by simply specifying in-feed movement in the X-axis direction in the succeeding blocks.

Example of programming

```
N10  G00  X94.  Z62. ;
N11  G90  X80.  W–42.  F0.3 ;
N12  X70. ;
N13  X60. ;

N14  G00 · · · ;
```

Start of G77 cycle

Executes G90 cycle by changing the cutting paths.

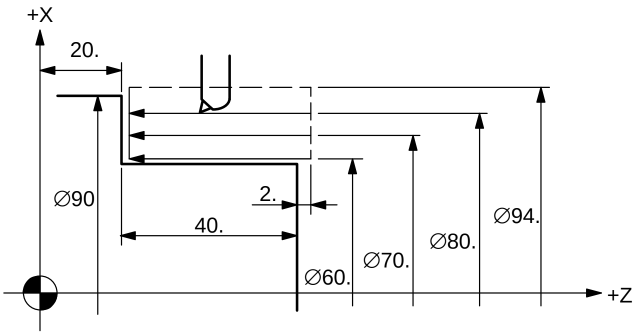


Fig. 4-2 Straight cutting cycle (G code system A)

Taper cutting cycle

Format

G... X... Z... R... F... ;

G code system A	G code system B	G code system C
G90	G77	G20

With the commands of “G... X(U)... Z(W)... R... F... ;” taper cutting cycle is executed as indicated by sequence 1 to 4 shown in Fig. 4-3.

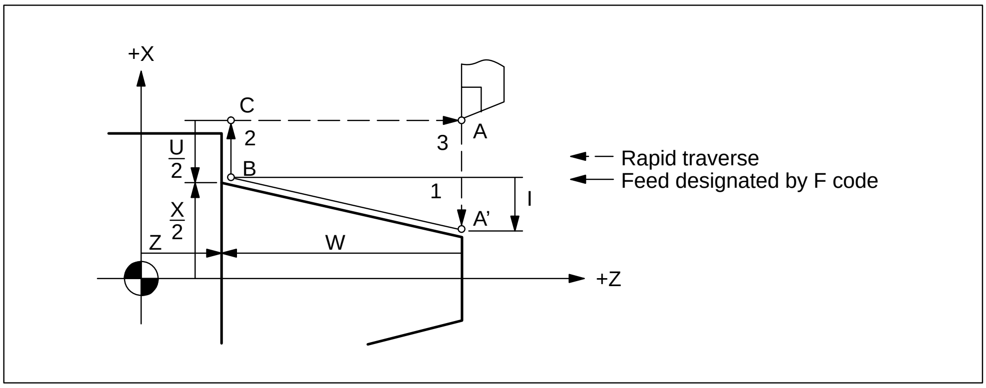


Fig. 4-3 Taper cutting cycle

The sign of address R is determined by the direction viewing point A’ from point B.

Example of programming

```
N20  G00  X87.  Z72. ;

N21  G90  X85.  W–42.  R–10.5  F0.25 ;
N22  X80. ;
N23  X75. ;
N24  X70. ;

N25  G00  · · ·;
```

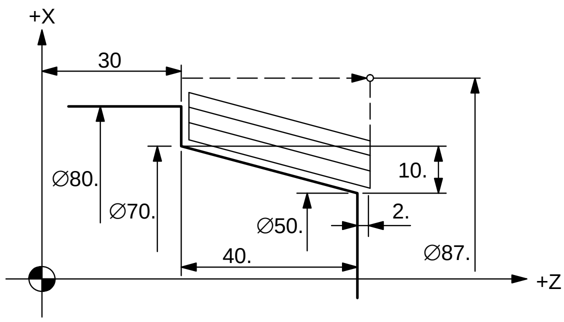


Fig. 4-4 Taper cutting cycle (G code system A)

- If the G77 (G90, G20) cycle is executed with the single block function ON, the cycle is not interrupted halfway but it stops after the completion of the cycle consisting of sequence 1 to 4.

- The S, T, and M functions that are used as the cutting conditions for the execution of the G77 (G90, G20) cycle should be specified in blocks preceding the G77 (G90, G20) block. However, if these functions are specified in a block independently without axis movement commands, such designation is valid if the block is specified in the G77 (G90, G20) mode range.

```
G77   X · · · Z · · · R · · · F · · · ;           ┐
      X · · · ;                                   │
      X · · · ;                                   │ G77 valid range
      X · · · T0505 M05 ;  ← Error                ┘
G00   X · · · Z · · · ;

G77   X · · · Z · · · R · · · F · · · ;           ┐
      X · · · ;                                   │ G77 valid range
      X · · · ;                                   │
G00   X · · · · T0505  M05 ;  ← Correct           ┘
      X · · · Z · · · ;
```

Fig. 4-5

The G77 (G90, G20) mode is valid up to the block immediately before the one in which a G code of 01 group is specified.

Thread cutting cycle command

For thread cutting operations, four kinds of thread cutting cycles are provided – two kinds of straight thread cutting cycles and two kinds of tapered thread cutting cycles.

Format

G... X... Z... F... ;

G code system A	G code system B	G code system C
G92	G78	G21

Straight thread cutting cycle

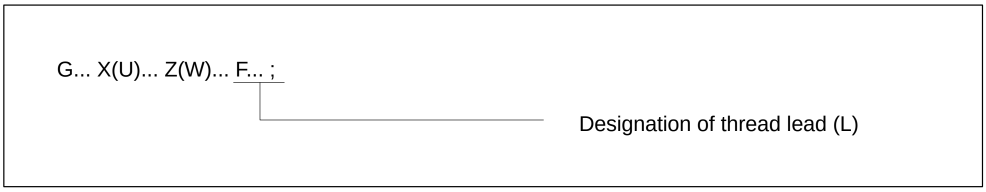


Fig. 4-6

With the commands indicated above, straight thread cutting cycle 1 to 4, shown in Fig. 4-7, is executed.

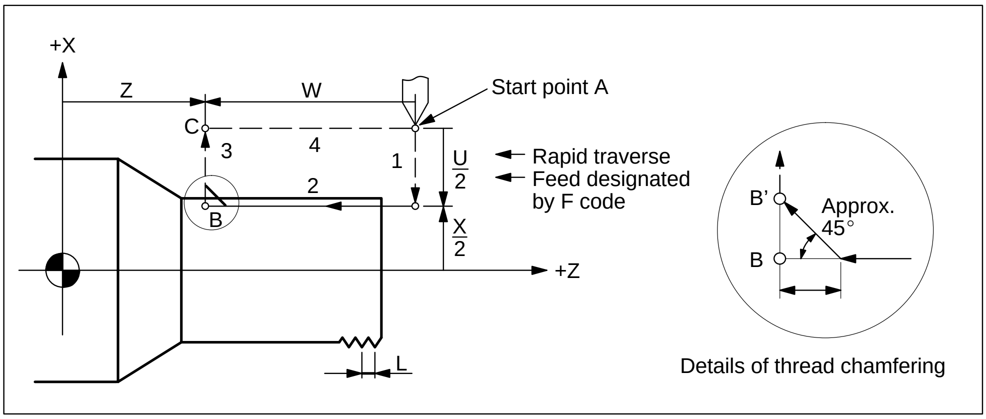


Fig. 4-7 Straight thread cutting cycle

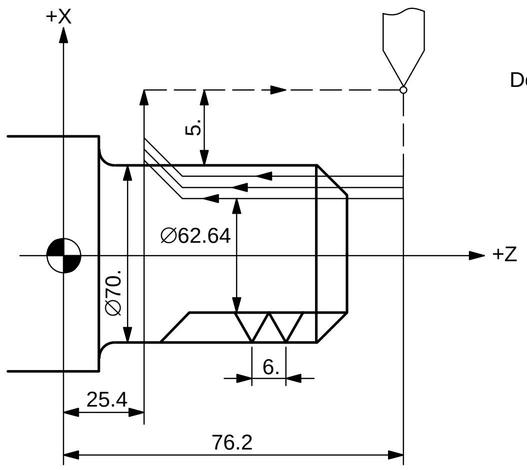
Since G78 (G92, G21) is a modal G code, thread cutting cycle is executed by simply specifying depth of cut in the X-axis direction in the succeeding blocks. It is not necessary to specify G78 (G92, G21) repeatedly in these blocks.

Example of programming

```
N30  G00  X80.  Z76.2 Mxx;          ← Mxx; Thread chamfering ON

N31  G78  X66.4  Z25.4  F6. ;       ┐
N32  X65. ;                         │ Thread cutting cycle, in four in-feeds
N33 X63.8 ;                         │
N34 X62.64 ;                        ┘

N35 G00 X100. Z100. Myy;            ← Myy; Thread chamfering OFF
```



Depth of cut

1st in-feed:	1.8 mm
2nd in-feed:	0.7 mm
3rd in-feed:	0.6 mm
4th in-feed:	0.58 mm

Fig. 4-8 Straight thread cutting cycle (G code system B)

- When the G78 (G92, G21) cycle is executed with the single block function ON, the cycle is not suspended halfway, but it stops after the completion of the cycle consisting of sequence 1 to 4.
- Thread chamfering can be performed in this thread cutting cycle. A signal from the machine tool initiates thread chamfering. Thread chamfering size γ can be set for GUD7 _ZSFI[26] in increments of 0.1L . Here, “L” represents the specified thread lead.

It is recommended to program the sequence that turns ON and OFF the “thread chamfering input” by using appropriate M codes.

Tapered thread cutting cycle

Format

G... X... Z... R... F... ;

G code system A	G code system B	G code system C
G92	G78	G21

With the commands of “G... X(U)... Z(W)... R... F... ;” tapered thread cutting cycle of 1 to 4 as shown in Fig. 4-9 is executed.

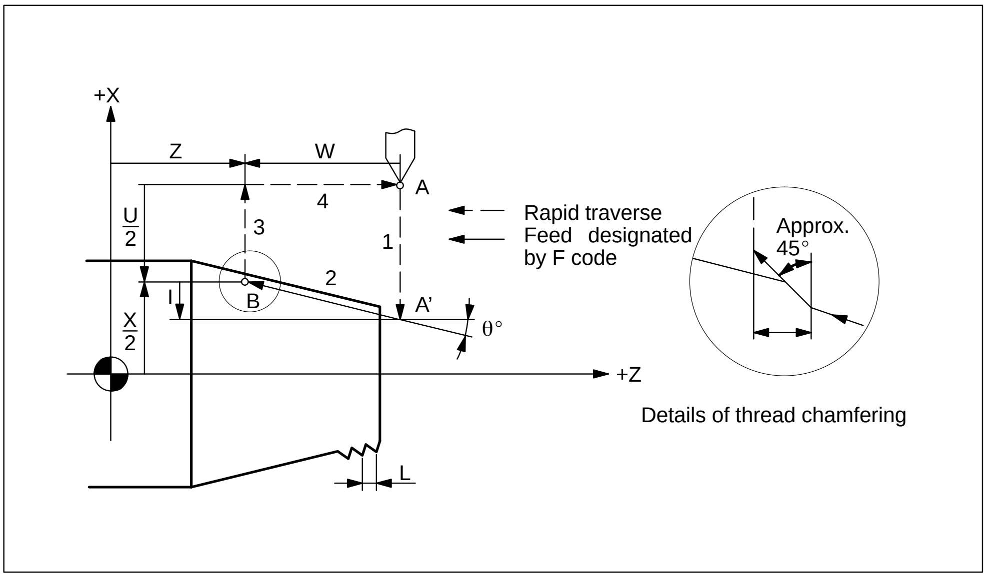


Fig. 4-9 Tapered thread cutting cycle

The sign of address R is determined by the direction viewing point A’ from point B. Since G78 (G92, G21) is a modal G code, thread cutting cycle is executed by simply specifying depth of cut in the X-axis direction in the succeeding blocks. It is not necessary to specify G78 (G92, G21) repeatedly in these blocks.

Example of programming

N50 G00 X80. Z80.8 Mxx ;

N51 G92 X70. W–50.8 I–1.5 F2. ;
N52 X68.8 ;
N53 X67.8 ;

N54 G00 X100. Z100. Myy ;

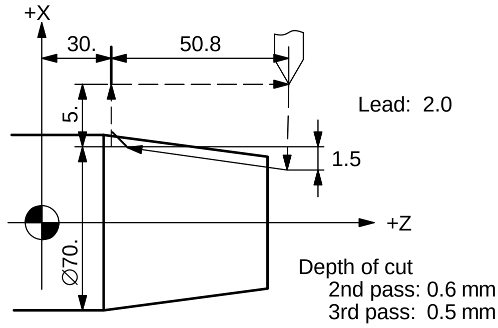


Fig. 4-10 Tapered thread cutting cycle (G code system A)

When the G78 (G92, G21) cycle is executed with the single block function ON, the cycle is not suspended halfway, but it stops after the completion of the cycle consisting of sequence 1 to 4.

The S, T, and M functions that are used as the cutting conditions for the execution of the G78 (G92, G21) cycle should be specified in blocks preceding the G78 (G92, G21) block. However, if these functions are specified in a block independently without axis movement commands, such designation is valid if the block is specified in the G78 (G92, G21) mode range.

When the CYCLE START button is pressed while the cutting tool is at start point A or chamfering completion point B, the suspended cycle is executed again from the beginning.

If the thread cutting feed hold option is not selected, the thread cutting cycle is continued even if the FEED HOLD button is pressed during the execution of thread cutting cycle. In this case, the operation is suspended upon completion of retraction operation after finishing the thread cutting cycle.

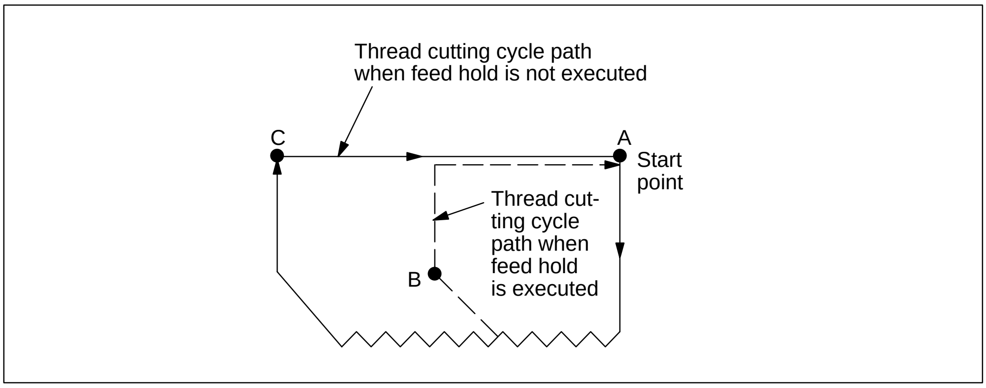


Fig. 4-11 Feed hold during thread cutting cycle

If chamfer size is “0” when the G78 (G92, G21) cycle is executed with chamfering ON, an alarm occurs.

Straight facing cycle

Format

G... X... Z... F... ;

G code system A	G code system B	G code system C
G94	G79	G24

With the commands of “G... X(U)... Z(W)... F... ;”, straight facing cycle of 1 to 4 as shown in Fig. 4-12 is executed.

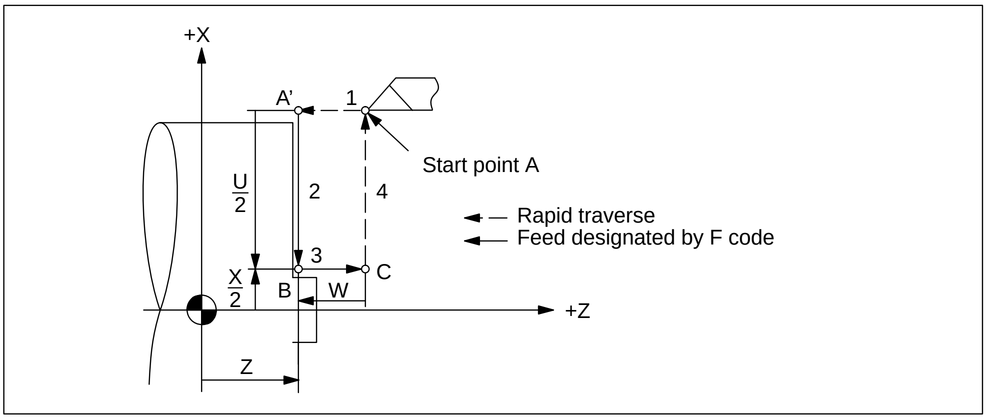


Fig. 4-12 Straight facing cycle

Since G79 (G94, G24) is a modal G code, thread cutting cycle is executed by simply specifying depth of cut in the Z-axis direction in the succeeding blocks. It is not necessary to specify G79 (G94, G24) repeatedly in these blocks.

Example of programming

```
N60  G00  X65.  Z42. ;

N61  G79  X20.  Z38.  F0.35 ;
N62  Z34. ;
N63  Z30. ;

N64  G00 ;
```

Cutting in 3 cycles in the G79 mode

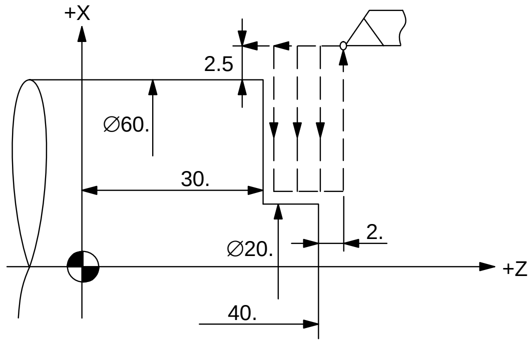


Fig. 4-13 Straight facing cycle (G code system B)

Taper facing cycle

Format

G... X... Z... R... F... ;

G code system A	G code system B	G code system C
G92	G78	G21

With the commands of “G... X(U)... Z(W)... R... F... ;”, taper facing cycle of 1 to 4 as shown in Fig. 4-14 is executed.

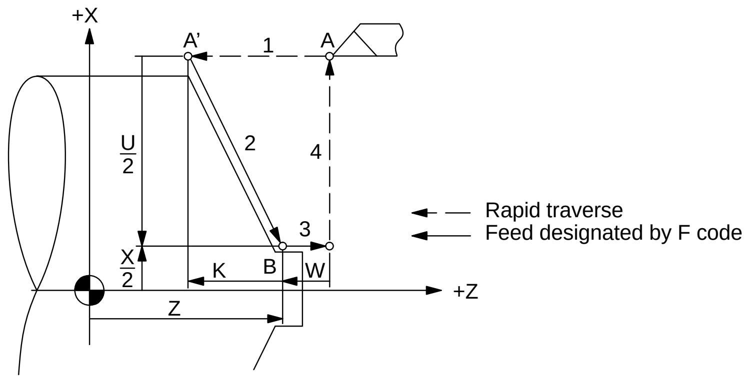


Fig. 4-14 Taper facing cycle

The sign of address R is determined by the direction viewing point A’ from point B.

Example of programming

```
N70N    G00  X74.  Z32. ;

N71     G79  X20.  Z30.  R–5.29  F0.3 ;
N72     Z25. ;
N73     Z20. ;

N74     G00 ;
```

Taper cutting in 3 cycles in the G79 mode

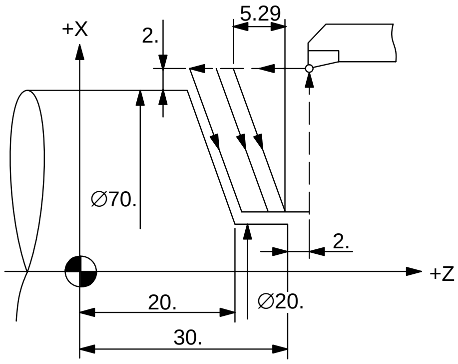


Fig. 4-15 Taper facing cycle (G code system B)

The S, T, and M functions that are used as the cutting conditions for the execution of the G79 (G94, G24) cycle should be specified in blocks preceding the G79 (G94, G24) block. However, if these functions are specified in a block independently without axis movement commands, such designation is valid if the block is specified in the G79 (G94, G24) mode range.

If the G79 (G94, G24) cycle is executed with the single block function ON, the cycle is not interrupted halfway but it stops after the completion of the cycle consisting of sequence 1 to 4.

4.1.2 Multiple repetitive cycles

By using the multiple repetitive cycles, programming steps can be considerably reduced due to the features that both rough and finish cutting cycles can be executed by simply defining the finishing shape, and the like.

For the multiple repetitive cycles, seven kinds of cycles (G70 to G76) are provided In G code systems A and B as indicated in Table 4-2. Note that these G codes are all non-modal G code.

Table 4-2 Cycles called by G70 to G76 (G code system A and B)

<table>
<tr><th>G code</th><th>Cycle name</th><th colspan="2">Remark</th></tr>
<tr><td>G70</td><td>Finishing cycle</td><td></td><td rowspan="4">Nose R offset possible</td></tr>
<tr><td>G71</td><td>Stock removal cycle, longitudinal axis</td><td rowspan="3">G70 cycle can be used for finishing</td></tr>
<tr><td>G72</td><td>Stock removal cycle transverse axis</td></tr>
<tr><td>G73</td><td>Contour repetition</td></tr>
<tr><td>G74</td><td>Deep hole drilling and recessing in longitudinal axis</td><td colspan="2" rowspan="3"></td></tr>
<tr><td>G75</td><td>Deep hole drilling and recessing in transverse axis</td></tr>
<tr><td>G76</td><td>Multiple thread cutting cycle</td></tr>
</table>

The same cycles are provided in G code system C. However, different G codes are used as indicated below.

Table 4-3 Cycles called by G72 to G78 (G code system C)

<table>
<tr><th>G code</th><th>Cycle name</th><th colspan="2">Remark</th></tr>
<tr><td>G72</td><td>Finishing cycle</td><td></td><td rowspan="4">Nose R offset possible</td></tr>
<tr><td>G73</td><td>Stock removal cycle, longitudinal axis</td><td rowspan="3">G72 cycle can be used for finishing</td></tr>
<tr><td>G74</td><td>Stock removal cycle transverse axis</td></tr>
<tr><td>G75</td><td>Contour repetition</td></tr>
<tr><td>G76</td><td>Deep hole drilling and recessing in longitudinal axis</td><td colspan="2" rowspan="3"></td></tr>
<tr><td>G77</td><td>Deep hole drilling and recessing in transverse axis</td></tr>
<tr><td>G78</td><td>Multiple thread cutting cycle</td></tr>
</table>

Note

The following cycle description of the a.m. cycles refers to G code system A and B.

Stock removal cycle, longitudinal axis (G71)

By using the multiple repetitive cycles, programming steps can be considerably reduced due to the features that both rough and finish cutting cycles can be executed by simply defining the finishing shape, and the like.
There are two types of stock removal cycles.

Type I

The specified area is removed by ⊿d (infeed depth for stock removal) with finishing allowances ⊿u/2 and ⊿w left over, whenever a contour of A to A' to B is described by an NC program.

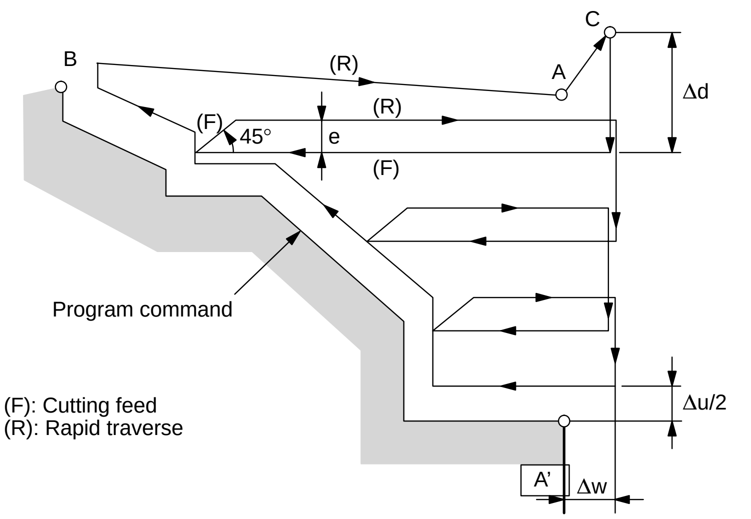


Fig. 4-16 Cutting path in stock removal in turning (type I)

Format

G71 U... R... ;

U: Infeed depth for stock removal **(⊿d)**, radius designation
This value is modal and can also be preset using GUD7, _ZSFI[30]. The value set here can be overwritten by the NC program command.
R: Retraction amount **(e)**
This value is modal and can also be preset using GUD7, _ZSFI[31]. The value set here can be overwritten by the NC program command.

G71 P... Q... U... W... F... S... T...

P: Starting block of contour definition
Q: Ending block of contour definition
U: Finishing allowance in X direction **(⊿u)** (diameter/radius designation)
W: Finishing allowance in Z direction **(⊿w)**
F: Machining feed
S: Spindle speed
T: Tool selection

F , S, or T functions issued within the NC program block range specified by address P and Q will be ignored. The relevant F, S, or T functions specified in the G71 block are effective.

Note

1. Both ⊿d and ⊿u are specified by means of the address U. If the addresses P and Q are present, then ⊿u is the case.
2. Four cutting sectors are possible. The relevant signs of ⊿u and ⊿w vary as shown in the figure below:

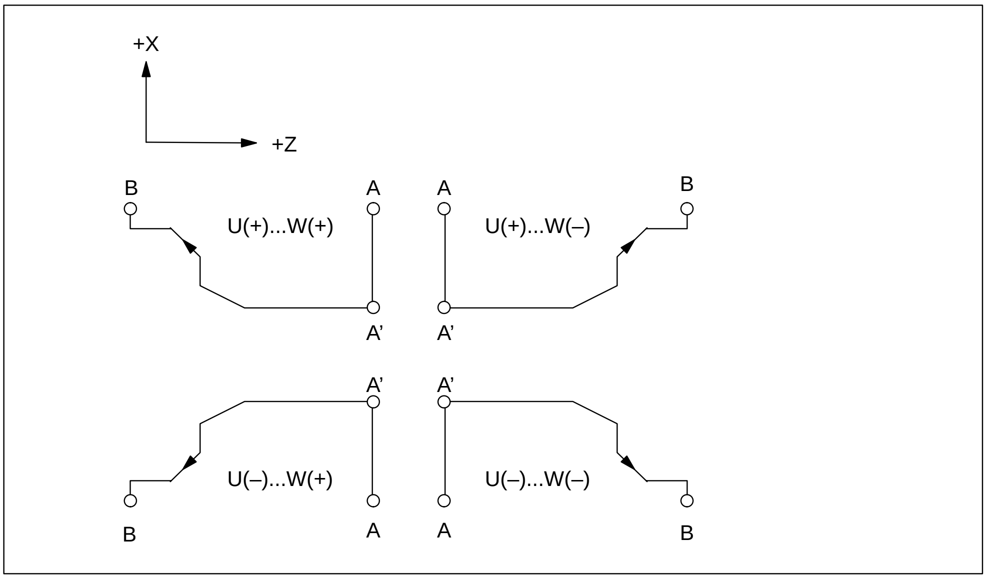


Fig. 4-17

Within the block specified by address P, the contour between points A and A' is determined (G00 or G01). A move command in the Z axis cannot be specified in this block.

The contour defined between A' and B must represent a steadily increasing or decreasing pattern in both X and Z axis.

3. Within the range of NC blocks specified by address P and Q, subprograms cannot be called.

Type II

In contrast to type I, type II does not necessarily have to show a steady increase or decrease along the X axis. In other words, it can also contain concaves (pockets).

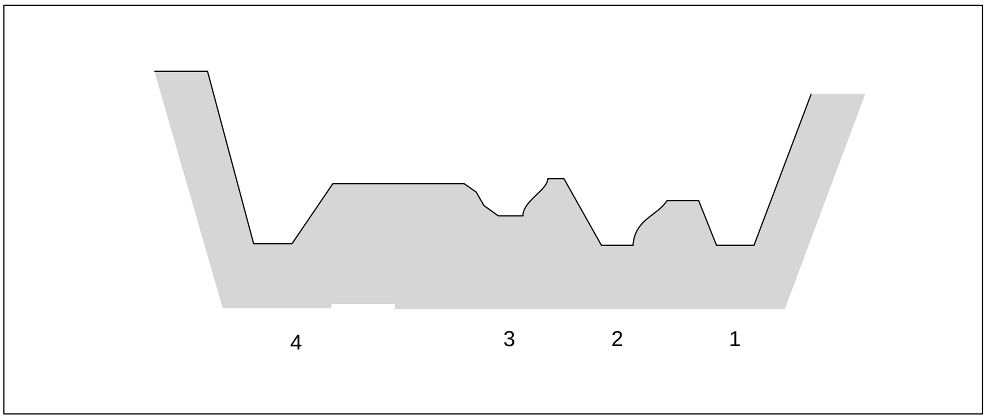


Fig. 4-18 Pockets in stock removal cycle (type II)

However, the Z axis profile must represent a monotone decrease or increase. For example, the following profile cannot be machined:

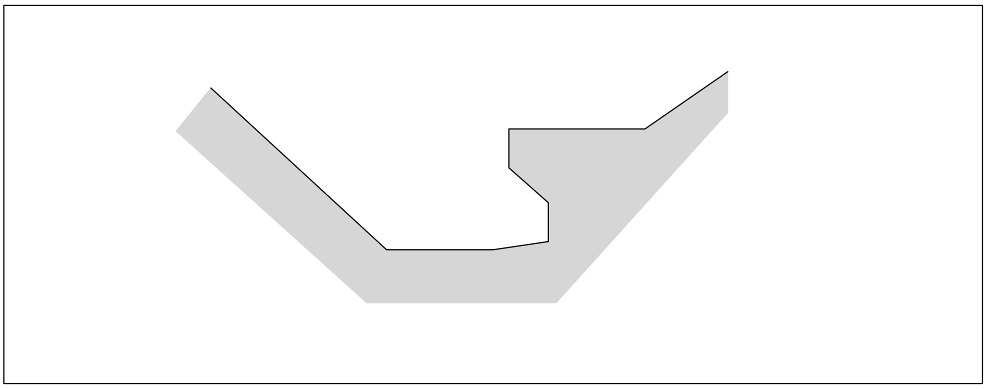


Fig. 4-19 Contour which cannot be machined in G71 cycle

How to distinguish between type I and type II

Type I: Only one axis is specified in the first block of the contour description
Type II: Two axes are specified in the first block of the contour description

Whenever the first block does not contain a Z axis movement command and type II should be used, W0 has to be specified.

Example

Type I	Type II
G71 V10.0 R4.0 ; G71 P50 Q100 .... ; N50 X(U)... ; :: :: N100.............. ;	G71 V10.0 R4.0 ; G71 P50 Q100 .... ; N50 X(U)... Z(W)... ; :: :: N100........... ;

Stock removal cycle, transverse axis (G72)

With the G72 command, stock removal cycle and rough finishing cycle in which finishing allowance is left on face can be specified. In comparison to the cycle called by G71, which carries out cutting by the movement in parallel to the Z–axis, the G72 cycle carries out cutting by the movements parallel to the X–axis. Therefore, the cycle called by G72 executes the same operation as with the cycle called by G71 in a different direction.

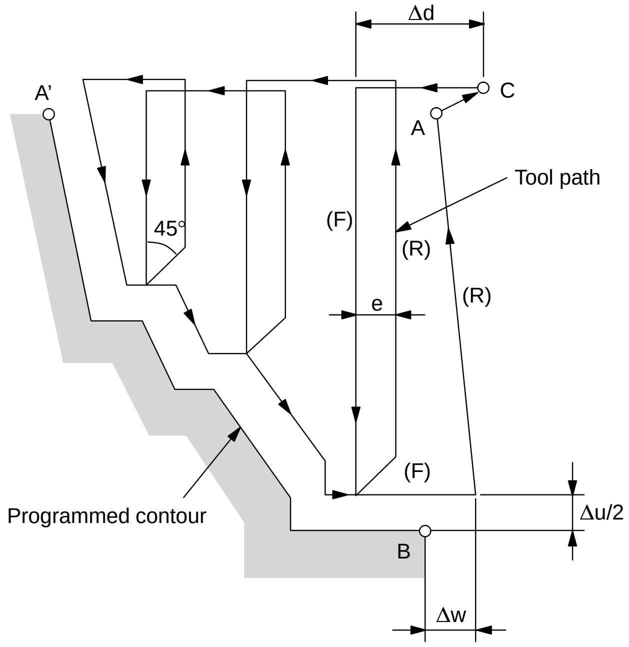


Fig. 4-20 Cutting path of a stock removal cycle, transverse axis

Format

G72 W... R... ;

The meaning of addresses W **(Δd)** and R **(e)** are basicly the same as U and R in the G71 cycle.

G72 P... Q... U... W... F... S... T... ;

The meaning of addresses P, Q, U **(Δu)**, W **(Δw)**, F, S, and T are the same as those in the G71 cycle.

Signs of specified numbers

Four cutting sectors are possible. The relevant signs of Δu and Δw vary as shown in the figure below:

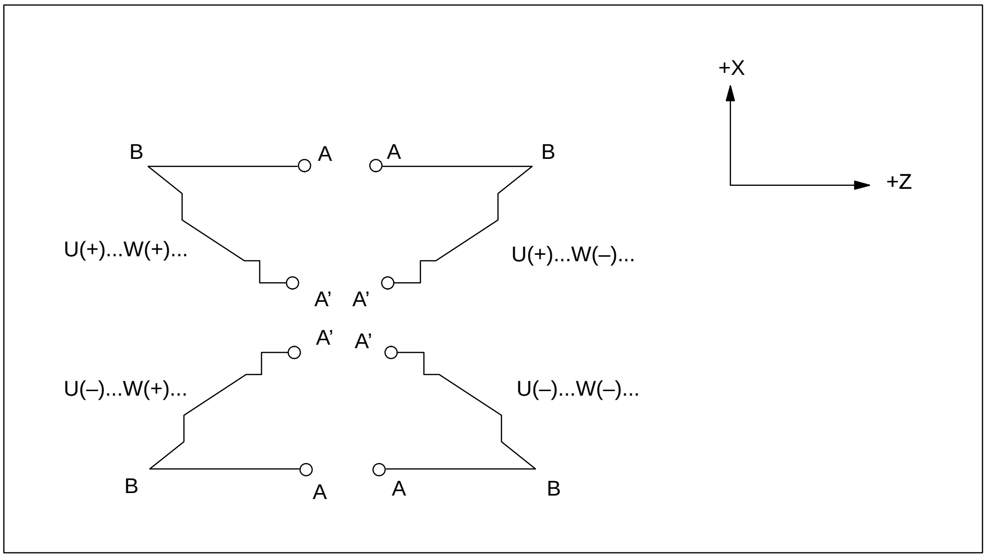


Fig. 4-21 Signs of numbers specified with U and W in stock removal in facing

The contour between A and A' is determined in the block specified by address P (G00 or G01). A move command in the X axis cannot be specified in this block. The contour between A' and B has to show a steadily increasing and decreasing pattern in both X and Z axes.

Contour repetition (G73)

The G73 contour repetition cycle is more effective when machining a workpiece that has a similar shape to the finishing shape, like a cast and forged workpieces.

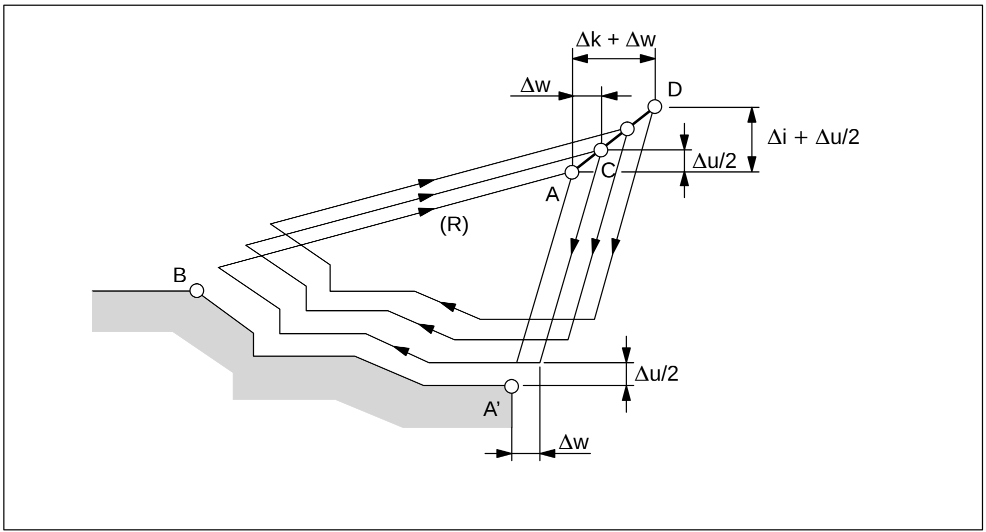


Fig. 4-22 Cutting path in contour repetition

Programmed contour: A → A' → B

Format

G73 U... W... R... ;

U: Distance **(Δi)** in the X axis direction from the start point to the current tool position (radius designation). This value is modal and can also be preset using GUD7, ZSFI[32]. The value set here can be overwritten by the NC program command.

W: Distance **(Δk)** in the Z axis direction from the start point to the current tool position. This value is modal and can also be preset using GUD7, ZSFI[33]. The value set here can be overwritten by the NC program command.

R: Number of cuts parallel to the contour **(d)**.
This value is modal and can also be preset using GUD7, ZSFI[34]. The value set here can be overwritten by the NC program command.

G73 P... Q... U... W F... S... T... ;

P: Starting block of contour definition
Q: Ending block of contour definition
U: Finishing allowance in X axis direction (**Δu)** (diameter/radius designation)
W: Finishing allowance in Z axic direction (**Δw)**
F: Machining feed
S: Spindle speed
T: Tool selection

F , S, or T functions issued within the NC program block range specified by address P and Q will be ignored. The relevant F, S, or T functions specified in the G73 block are effective.

Note

1. The values ⊿i and ⊿k, or ⊿u and ⊿w are determined by address U and W respectively. However, their meanings are specified by the appearance of addresses P and Q present in the G73 block. Addresses U and W refer to ⊿i and ⊿k respectively whenever P and Q are not specified in the same block. Addreses U and W refer to ⊿u and ⊿w respectively whenever P and Q are specified in the same block.
2. Through the G73 command with P and Q specification, the cycle machining is performed. Four cutting sectors are considered here. Note the sign of ⊿u, ⊿w, ⊿k, and ⊿i. The tool returns to point A once the machining cycle has been completed.

Finishing cycle (G70)

While rough cutting is performed by G71, G72 or G73, the finishing is implemented through the following command.

Format

G70 P... Q... ;

P: Starting block of contour definition.
Q: Ending block of contour definition.

Note

1. The functions specified between the blocks determined by addresses P and Q are effective in G70 while those of F, S, and T are specified in the block G71, G72, G73 are not effective.
2. The tool is returned to the start point and the next block is read once the cycle machining through G70 has been completed.
3. Subprograms cannot be called within the blocks determined by the addresses P and Q.

Examples

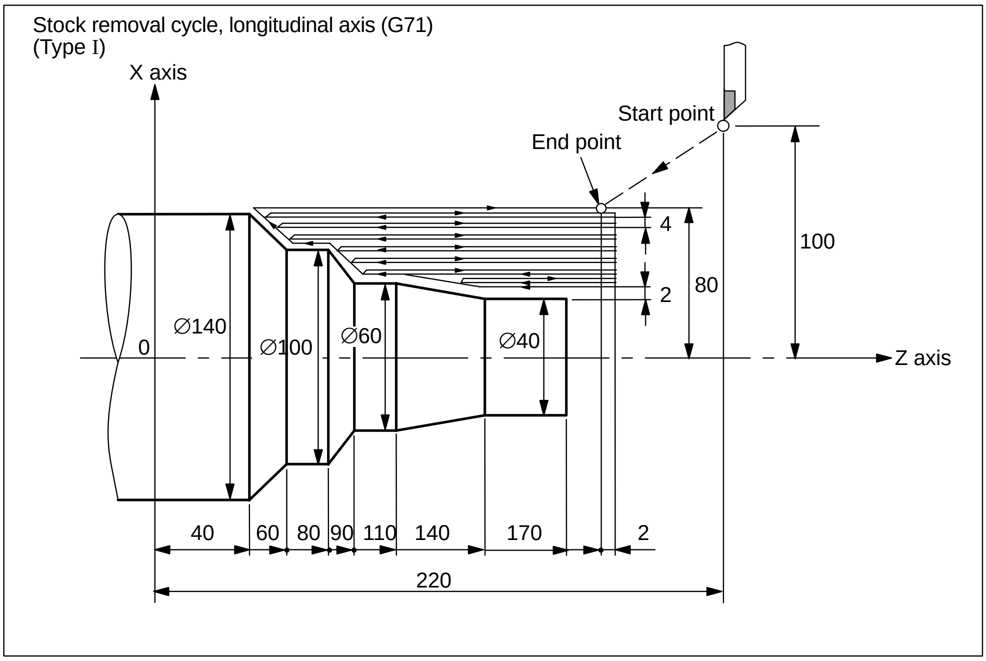


Fig. 4-23

(Diameter designation, metric input)
N010 G00 G90 X200.0 Z220.0 ;
N011 X142.0 Z171.0 ;
N012 G71 U4.0 R1.0 ;
N013 G71 P014 Q020 U4.0 W2.0 F0.3 S550 ;
N014 G00 X40.0 F0.15 S700 ;
N015 G01 Z140.0;
N016 X60.0 Z110.0 ;
N017 Z90.0 ;
N018 X100.0 Z80.0 ;
N019 Z60.0 ;
N020 X140.0 Z40.0 ;
N021 G70 P014 Q020 ;
N022 G00 X200 Z220 ;

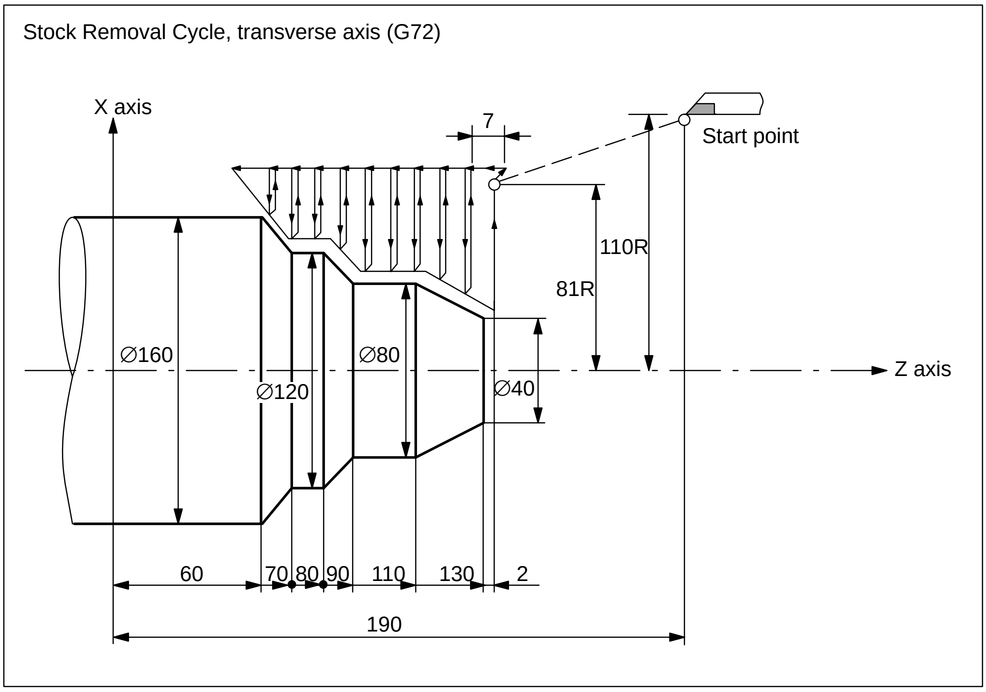


Fig. 4-24

(Diameter designation, metric input)
N010 G00 G90 X220.0 Z190.0 ;
N011 G00 X162.0 Z132.0 ;
N012 G72 W7.0 R1.0 ;
N013 G72 P014 Q019 U4.0 W2.0 F0.3 ;
N014 G00 Z59.5 F0.15 S200;
N015 G01 X120.0 Z70.0 ;
N016 Z80.0 ;
N017 X80.0 Z90.0 ;
N018 Z110.0 ;
N019 X36.0 Z132.0 ;
N020 G70 P014 Q019 ;
N021 X220.0 Z190.0 ;

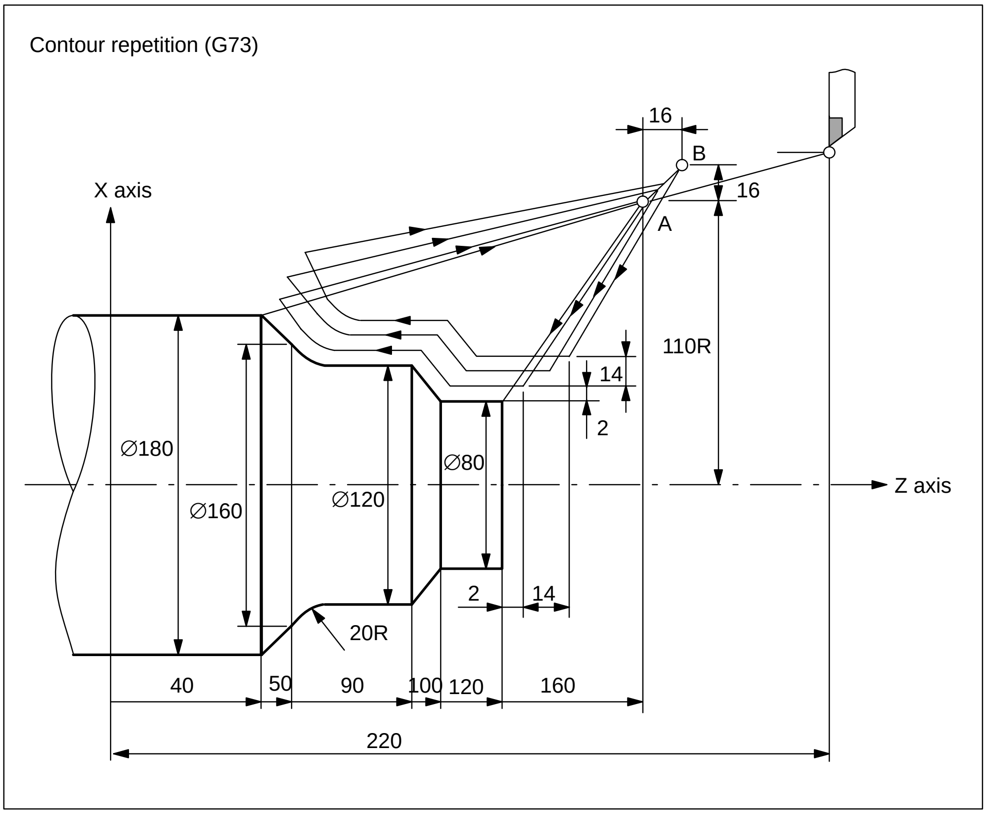


Fig. 4-25

(Diameter designation, metric input)
N010 G00 G90 X260.0 Z220.0 ;
N011 G00 X220.0 Z160.0 ;
N012 G73 U14.0 W14.0 R3 ;
N013 G73 P014 Q020 U4.0 W2.0 F0.3 S0180 ;
N014 G00 X80.0 Z120.0 ;
N015 G01 Z100.0 F0.15 ;
N017 X120 Z90.0 ;
N018 X70 ;
N019 G02 X160.0 Z50.0 R20.0 ;
N020 G01 X180.0 Z40.0 F0.25 ;
N021 G70 P014 Q020 ;
N022 G00 X260.0 Z220.0 ;

Deep hole drilling and recessing in longitudinal axis (G74)

In the cycle called by G74, peck feed operation parallel to the Z–axis is repeated to carry out a face cut–off cycle.

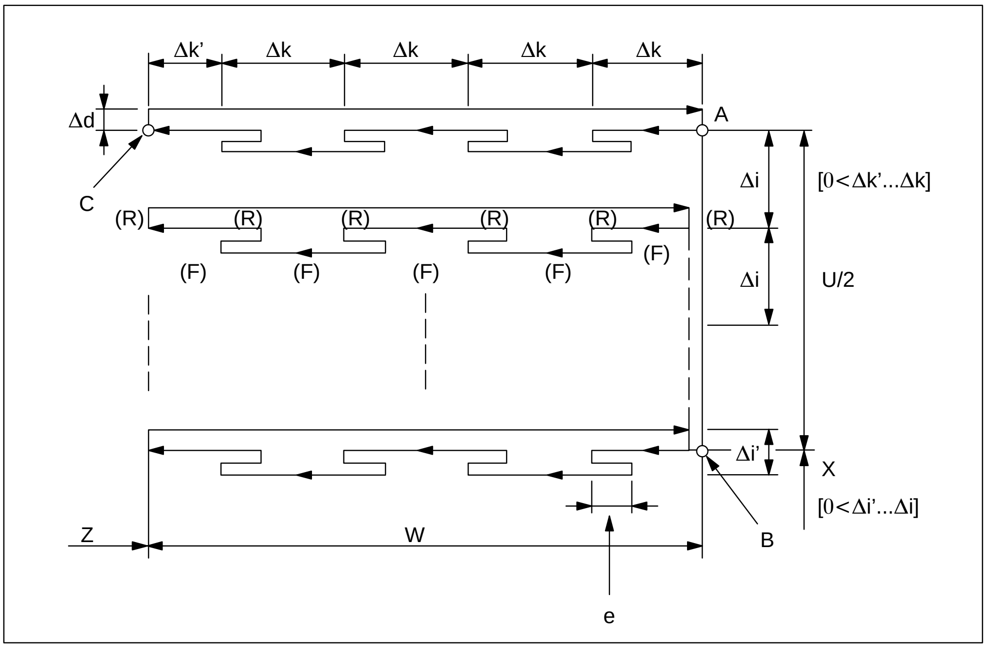


Fig. 4-26 Cutting path in deep hole drilling cycle

Format

G74 R... ;

R: Retraction amount **(e)**
This value is modal and can be preset using GUD7, ZSFI[29]. The value set here can be overwritten by the NC program command.

G74 X(U)... Z(W)... P... Q... R... F...(f) ;

X: Starting point X (absolute position)
U: Starting point X (incremental)
Z: Starting point Z (absolute position)
W: Starting point Z (incremental)
P: Infeed amount **(Δi)** in X axis direction (without sign)
Q: Infeed amount **(Δk)** in Z axis direction (without sign)
R: Retraction amount **(Δd)** at recess base
F: Feed rate

Note

1. While both e and ⊿d are determined by address R their meanings are specified by the appearance of address X (U). ⊿d is used when X(U) is specified.
2. Through the G74 command with an X (U) specification, cycle machining is performed.

Deep hole drilling and recessing in transverse axis (G75)

The G75 cycle executes an OD cut–off cycle while carrying out peck feed operation parallel to the X–axis. In comparison to the G74 cycle in which the OD cut–off cycle is executed in parallel to the X–axis, the G75 cycle executes virtually the same operation excluding that the cycle is executed in parallel to the X–axis.

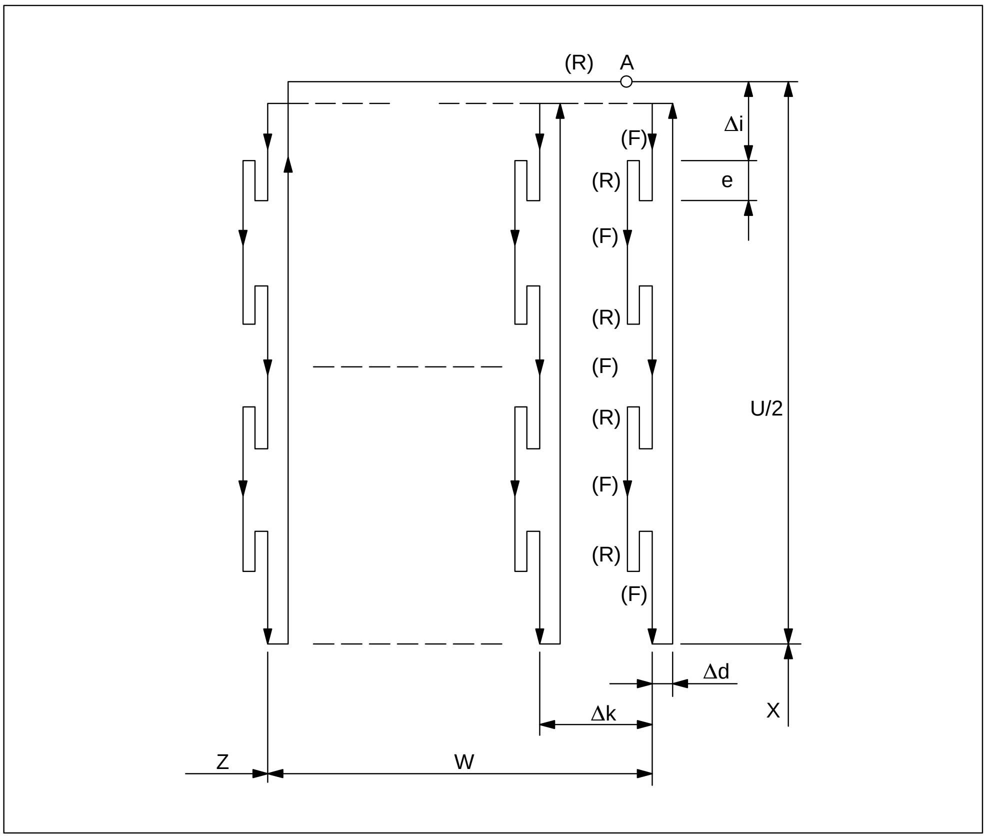


Fig. 4-27 Fig. 4-28 Cutting path in deep hole drilling and recessing in tranverse axis (G75)

Format

G75 R... ;

G75 X(U)... Z(W)... P... Q... R... F... ;

The meaning of the addresses are the same as those of G74 cycle.

Four cutting sectors are possible.

Multiple thread cutting cycle (G76)

G76 calls an automatic thread cutting cycle for cutting straight or taper thread in which in–feed is carried out along a thread angle.

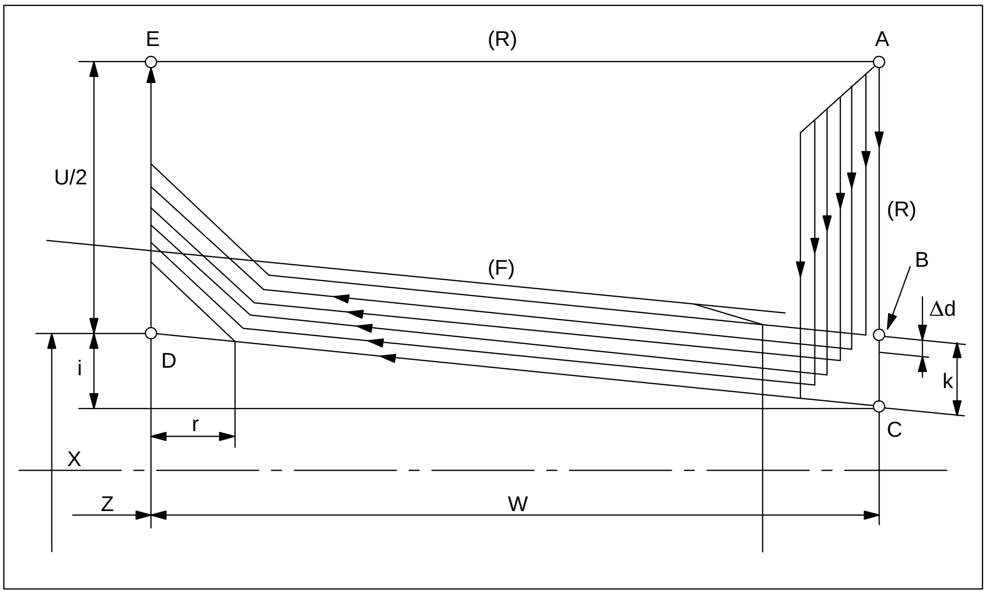


Fig. 4-29 Cutting path of a multiple thread cutting cycle

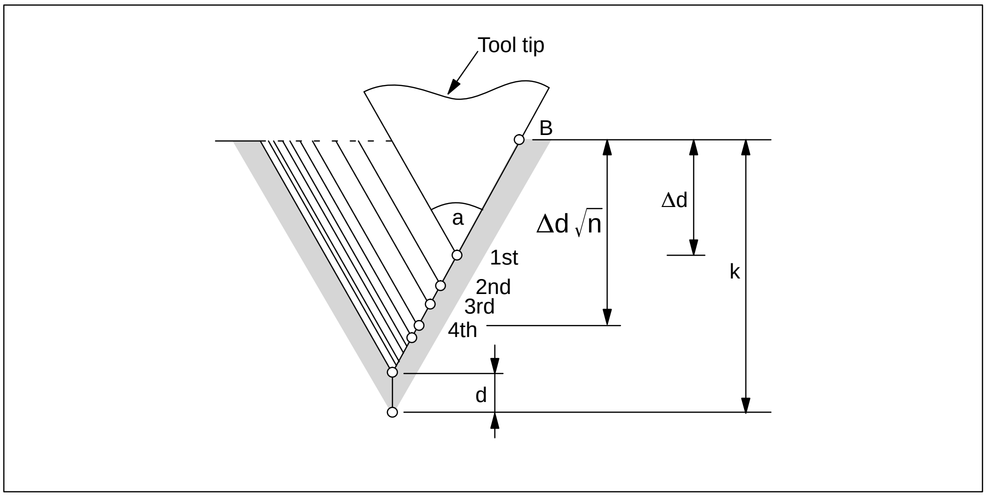


Fig. 4-30 In–feed in thread cutting

Format

G76 P... (m, r, a) Q... R... ;

P:
m: Number of finishing cuts
This value is modal and can also be preset using GUD7, ZSFI[24]. The value set here can be overwritten by the NC program command.
r: Size of chamfer at the end of the thread (1/10 * thread pitch)
This value is modal and can also be preset using GUD7, ZSFI[26]. The value set here can be overwritten by the NC program command.
a: Angle of tool cutting edge
This value is modal and can also be preset using GUD7, ZSFI[25]. The value set here can be overwritten by the NC program command.

All the above parameters are specified by address P at the same time.

Example for address P:

G76 P012055 Q4 R0.5

P = 012055
Angle of tool cutting edge = 55 degree
Chamfer at the end of the thread=2,0 x pitch
1 finishing cut

Q: Minimum infeed depth (⊿**dmin)**, radius value
The cutting depth is clamped at the value specified at address Q whenever the cutting depth of one cycle operation (⊿d – ⊿d–1) becomes less than this limit. This value is modal and can also be preset using GUD7, ZSFI[27]. The value set here can be overwritten by the NC program command.

R: Finishing allowance **(d)**
This value is modal and can also be preset using GUD7, ZSFI[28]. The value set here can be overwritten by the NC program command.

G76 X(U)... Z(W)... R... P... Q... F... ;

X, U: Endpoint of thread in X axis direction (absolute position (X), incremental (U))

Z, W: Endpoint of thread in Z axis direction

R: Radius difference for tapered thread **(i).** i = 0 for ordinary straight thread

P: Thread depth **(k)**, radius value

Q: Infeed amount for the 1st cut **(Δd)**, radius value

F: Lead **(L)**

Note

1. The appearance of X (U) and X (W) determine the meaning of the data specified by address P, Q, and R.
2. Through the G76 command with X (U) and Z (W) specification cycle machining is performed. One edge cutting is performed and the load on the tool tip is reduced when this cycle is applied.

   The amount of cutting per cycle is kept constant by assigning the cutting depth ⊿d to the first path, and ⊿dn to the nth path. Corresponding to the sign of each address, four symmetrical sectors are considered here.
3. The notes on thread cutting are equivalent to those on G32 for thread cutting and G92 for the thread cutting cycle.

Examples

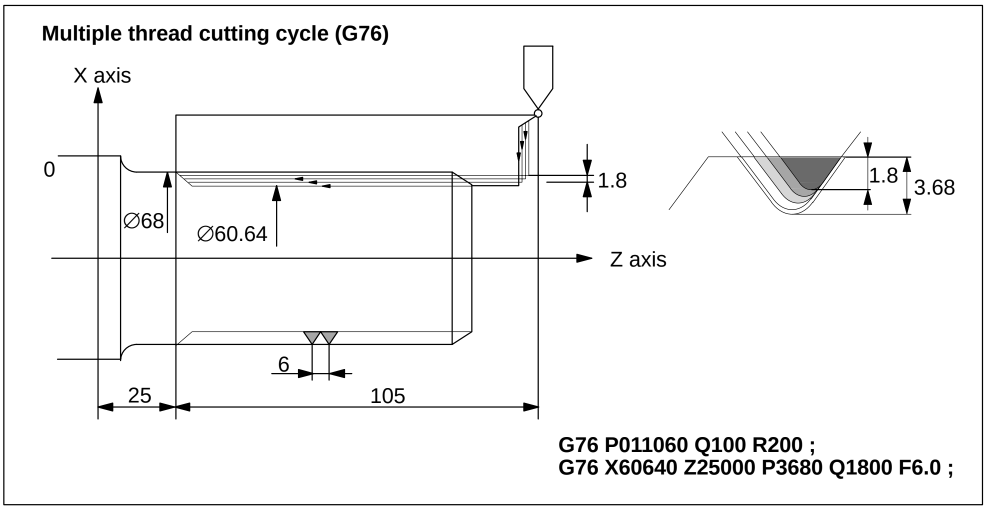


Fig. 4-31 Multiple thread cutting cycle (G76)

Notes on multiple repetitive cycle (G70–G76)

1. G70, G71, G72, or G73 cannot be commanded in MDA mode. If it is commanded, alarm 14011 is generated. However, G74, G75, and G76 can be commanded in MDA mode.
2. M98 (subprogram call) and M99 (subprogram end) cannot be commanded in the blocks in containg G70, G71, G72, or G73 and between the sequence numbers specified by addresses P and Q.
3. The following commands cannot be specified in the blocks between the sequence numbers specified by addresses P and Q:
   - One shot G codes with the exception of G04 (dwell)
   - 01 group G codes with the exception of G00, G01, G02, and G03
   - 06 group G codes
   - M98 / M99
4. Do not program in such a way that the final movement command of the contour definition for G70, G71, G72, and G73 finishes off with chamfering or corner rounding. An alarm is issued whenever the above is specified.
5. In the G74, G75, and G76 cycles, addresses P and Q use the least input increments to specify the amount of travel and depth of cut.
6. No tool nose radius compensation can be carried out within G71, G72, G73, G74, G75, G76, or G78 cycles.

4.1.3 Hole-machining canned cycles (G80 to G89)

Hole-machining canned cycles (G80 to G89) can define specific movements for machining holes that usually require several blocks of commands by single-block commands. G80 cancels the called out canned cycle program.

G codes that call out canned cycles G80 to G89 are common for all G code systems.

G codes calling canned cycles and axis movement patterns of canned cycles

G codes that call out a canned cycle and the axis movement pattern of the called canned cycle are indicated in Table 4-4.

Table 4-4 Hole-machining canned cyles

G code	Hole machining operation (direction)	Processing at bottom hole	Retraction (+ direction)	Applications
G80	–	–	–	Cancel
G83	Cutting feed/intermittent	Dwell	Rapid traverse	Front drilling cycle
G84	Cutting feed	Dwell –> spindle CCW	Cutting feed	Front tapping cycle
G85	Cutting feed	Dwell	Cutting feed	Front boring cycle
G87	Cutting feed/intermittent	Dwell	Rapid traverse	Side drilling cycle
G88	Cutting feed	Dwell –> spindle CCW	Cutting feed	Side tapping cycle
G89	Cutting feed	Dwell	Cutting feed	Side boring cycle
G88	Cutting feed	Spindle forward rotation after dwell	Manual return → Spindle forward rotation	Boring
G89	Cutting feed	Dwell	Cutting feed	Boring

When using canned cycles the sequence of operations is generally carried out as described below:

Operation 1 – Positioning of X (Z) and C axis
Operation 2 – Rapid traverse movement to level R
Operation 3 – Hole machining
Operation 4 – Operation at hole bottom
Operation 5 – Retraction to R level
Operation 6 – Rapid retraction to the initial point

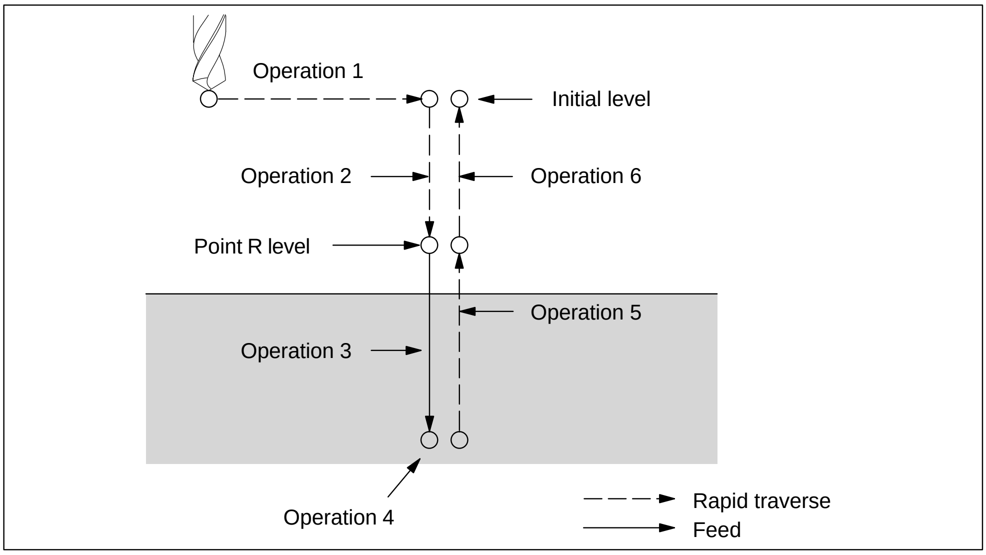


Fig. 4-32 Sequence of drilling cycle operation

Explanations

Positioning axis and drilling axis

As shown below, a drilling G code determines the positioning axes as well as the drilling axis. The C-axis and X or Z-axis correspond to the positioning axes. The drilling axis is represented by the X or Z-axis: These axes are not used as positioning axes.

Table 4-5 Positioning plane and its respective drilling axis

G code	Positioning plane	Drilling axis
G83, G84, G85	X axis, C axis	Z axis
G87, G88, G89	Z axis, C axis	X axis

G83 and G87, G84 and G88, and G85 and G89 have the same sequence except for the drilling axis.

Drilling mode

The G codes (G83–G85 / G87–89) are modal, and remain active until they are canceled. The current state is the drilling mode whenever they are active. The data is retained until modified or canceled once drilling data is specified in the drilling mode.
All necessary drilling data have to be specified at the beginning of the canned cycles. Only data modifications are allowed to be specified while canned cycles are being carried out.

Return point level (G98/G99)

When the G code system A is active, the tool traverse away from the bottom of a hole and goes back to the initial level. When specifying G98 while the G code system B or C is active, the tool, coming from the bottom of a hole, returns to the initial level. When specifying G99, the tool returns to the R level from the bottom of a hole.
The figure below describes the movement of the tool when G98 or G99 is specified. G99 is generally applied for the first drilling operation, while G98 is applied for the last drilling operation. Even when drilling is performed in the G99 mode, the initial level does not change.

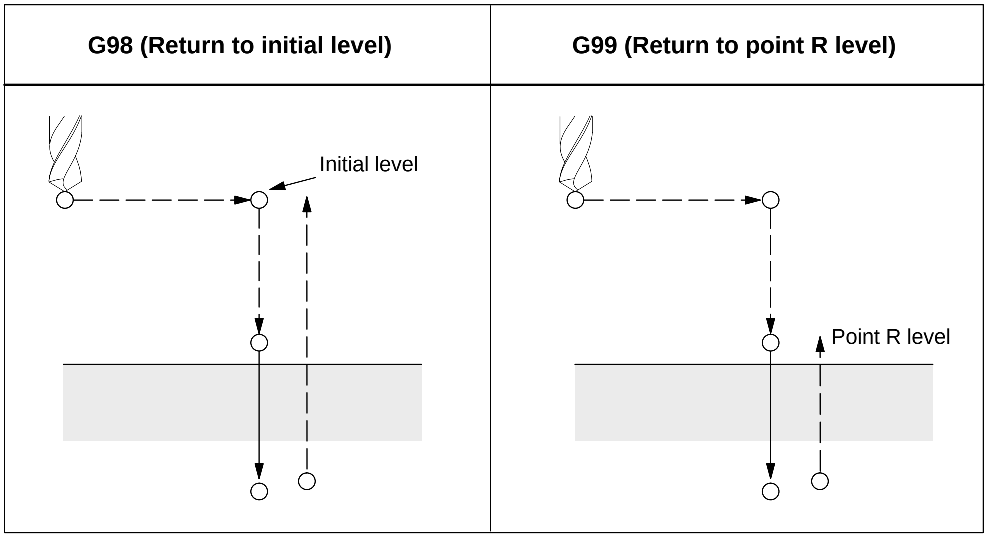


Fig. 4-33 Return point level (G98/G99)

Repetition

Specify the number of repeats in K in order to repeat the drilling for equally spaced holes. K only becomes effective in the block where it is specified. Specifying the first hole in absolute mode (G90) results in drilling at the same position. Therefore, specify K in incremental mode (G91).

Drilling data is stored, but drilling is not performed whenever K0 is specified.

Cancel

Use G80 or a group 01 G code (G00, G01, G02, G03) to cancel a canned cycle.

Symbols and figures

The individual canned cycle are explained in the following sections. The following symbols are used in the figures below:

- - - →	Positioning (rapid traverse G00)
——→	Cutting feed (linear interpolation G01)
∼∼→	Manual feed
P1	Dwell time
Mα	M code for C-axis clamp
M (α+I)	M code for C-axis unclamp

Fig. 4-34

Caution

In each canned cycle, the address R (distance between initial level and point R) is always treated as a radius.
However, Z or X (distance between point R and hole bottom) is treated either as a diameter or radius, depending on the specification.

Face deep hole drilling cycle (G83) / side deep hole drilling cycle (G87)

The setting of GUD7, _ZSFI[20] decides whether The deep hole drilling cycle or high-speed deep hole drilling cycle is applied. The normal drilling cycle is applied whenever depth of cut for each drilling is not specified.

High-speed deep hole drilling cycle (G83, G87) (GUD7, _ZSFI[20]=0)

When using high-speed deep hole drilling cycle, the drill repeats the cycle of drilling at the cutting feedrate. It intermittently retracts by a specified distance until the tool reaches the bottom of the hole.

Format

G83 X(U)... C(H)... Z(W)... R... Q... P... F... M... ;
or
G87 Z(W)... C(H)... X(U)... R... Q... P... F... M... ;

X, C or Z, C: Hole position

Z or X: Distance from point R to the bottom of hole

R_: Distance from the initial level to R level

Q_: Infeed

P_: Dwell time at bottom of hole

F_: Cutting feedrate

K_: Number of repetitions (if required)

M_: M code for clamping C-axis (if required)

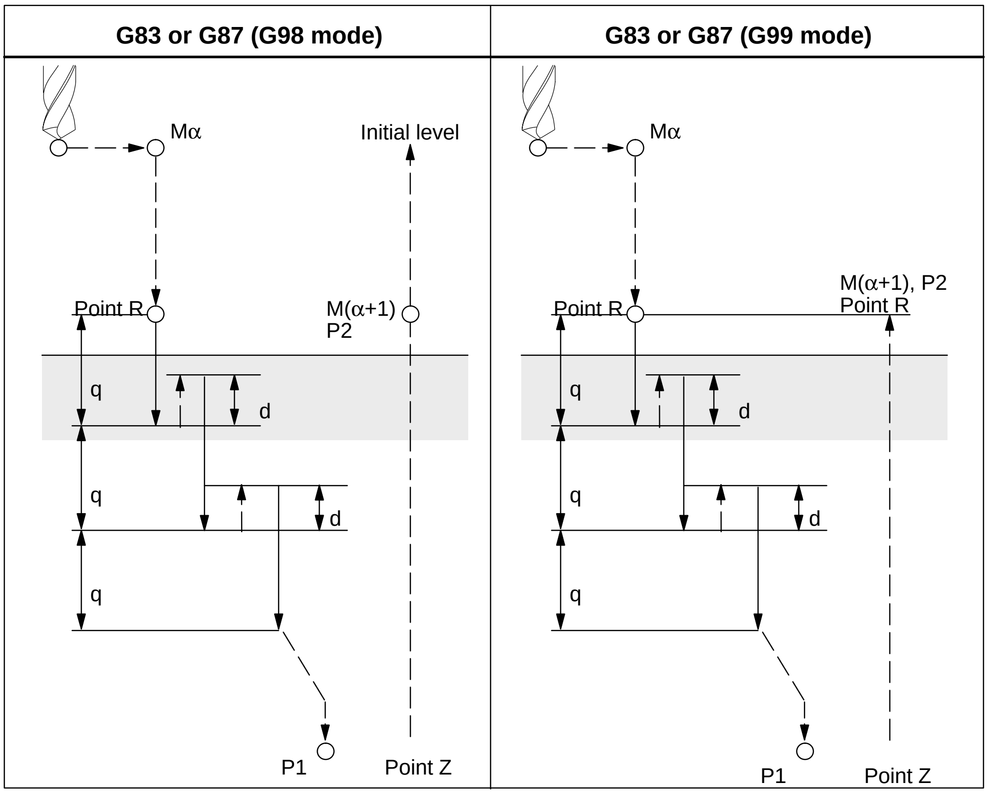


Fig. 4-35

Mα:	M code for clamping C-axis
M(α+1):	M code for unclamping C-axis
P1:	Dwell time (program)
P2:	Dwell time specified in GUD7, _ZSFR[22]
d:	Retraction amount specified in GUD7, _ZSFR[21]

Deep hole drilling cycle (G83, G87)
(GUD7, _ZSFI[20]=1)

Format

G83 X(U)... C(H)... Z(W)... R... Q... P... F... M... K... ;
or
G87 Z(W)... C(H)... X(U)... R... Q... P... F... M... K... ;

X, C or Z, C: Hole position

Z or X: Distance from point R to the bottom of hole

R_: Distance from the initial level to R level

Q_: Infeed

P_: Dwell time at bottom of hole

F_: Cutting feedrate

K_: Number of repetitions (if required)

M_: M code for clamping C-axis (if required)

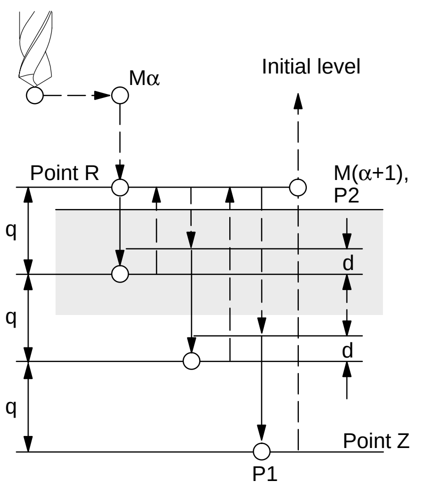
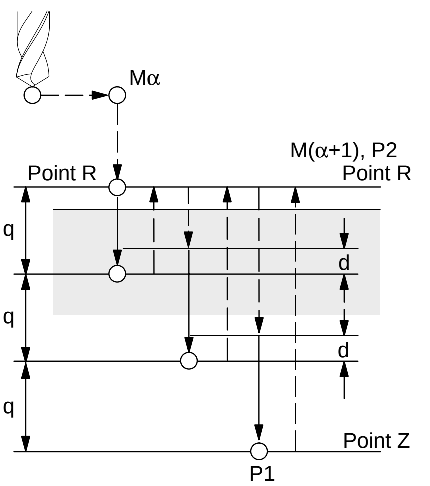
G83 or G87 (G98 mode)	G83 or G87 (G99 mode)
	

Fig. 4-36

Mα:	M code for clamping C-axis
M(α+1):	M code for unclamping C-axis
P1:	Dwell time (program)
P2:	Dwell time specified in GUD7, _ZSFR[22]
d:	Retraction amount specified in GUD7, _ZSFR[21]

Example

M3 S2500 ;	Rotate the drilling tool
G00 X100.0 C0.0 ;	Positioning of X and C axis
G83 Z–35.0 R–5.0 Q5000 F5.0 ;	Maching hole 1
C90.0 ;	Maching hole 2
C180.0 ;	Maching hole 3
C270.0 ;	Maching hole 4
G80 M05 ;	Cycle cancel and drilling tool stop

Drilling cycle (G83 or G87)

The normal drilling cycle is applied whenever the depth of cut for each drilling is not specified. In this case, the tool is retracted from the bottom of the hole in rapid traverse.

Format

G83 X(U)... C(H)... Z(W)... R... P... F... M... K... ;
or
G87 Z(W)... C(H)... X(U)... R... P... F... M... K... ;

X, C or **Z, C**: Hole position

Z or X: The distance from point R to the bottom of the hole

R_: Distance from the initial level to R level

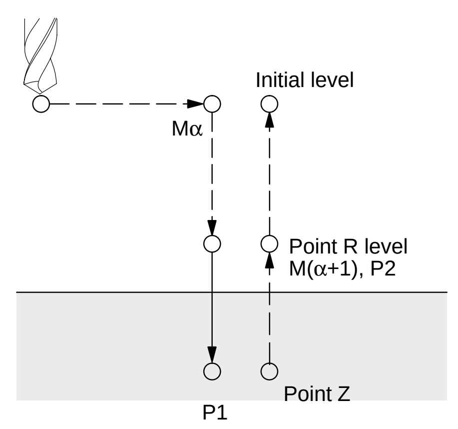
P_: Dwell time at bottom of hole

F_: Cutting feedrate

K_: Number of repetitions (if required)

M_: M code for clamping C-axis (if required)

G83 or G87 (G98 mode)	G83 or G87 (G99 mode)



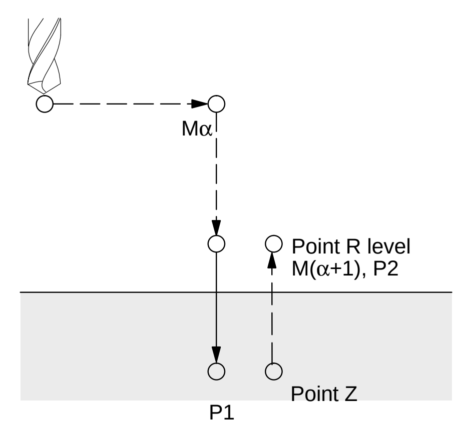


Fig. 4-37

Mα:	M code for clamping C-axis
M(α+1):	M code for unclamping C-axis
P1:	Dwell time (program)
P2:	Dwell time specified in GUD7, _ZSFR[22]

Example

M3 S2500 ;	Rotate the drilling tool
G00 X100.0 C0.0 ;	Positioning of X and C axis
G83 Z–35.0 R–5.0 P500 F5.0 ;	Machining hole 1
C90.0 ;	Machining hole 2
C180.0 ;	Machining hole 3
C270.0 ;	Machining hole 4
G80 M05 ;	Cycle cancel and drilling tool stop

Face tapping cycle (G84)
Side tapping cycle (G88)

In this cycle, the rotation direction of the spindle is reversed at the bottom of the hole.

Format

G84 X(U)... C(H)... Z(W)... R... P... F... M... K... ;
or
G88 Z(W)... C(H)... X(U)... R... P... F... M... K... ;

X, C or **Z, C:** Hole position

Z or X: The distance from point R to the bottom of the hole

R_: Distance from the initial level to R level

P_: Dwell time at bottom of hole

F_: Cutting feedrate

K_: Number of repetitions (if required)

M_: M code for clamping C-axis (if required)

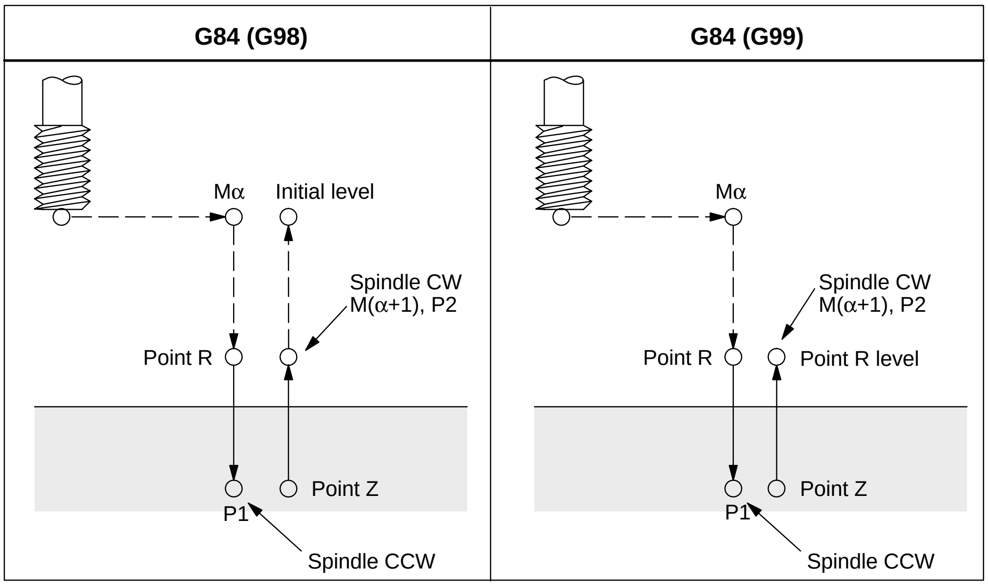


Fig. 4-38

P2: Dwell specified in GUD7, _ZSFR[22]

Explanations

In tapping operation, the spindle is rotated clockwise towards the bottom of the hole and reversed for retraction. The cycle is not stopped until the return operation in completed.

Example

M3 S2500 ;	Rotate the tapping tool
G00 X50.0 C0.0 ;	Positioning X and C axis
G84 Z–35.0 R–5.0 P500 F5.0 ;	Tapping hole 1
C90.0 ;	Tapping hole 2
C180.0 ;	Tapping hole 3
C270.0 ;	Tapping hole 4
G80 M05 ;	Cycle cancel and tapping tool stop

Face drilling cycle (G85)
Side drilling cycle (G89)

Format

G85 X(U)... C(H)... Z(W)... R... P... F... K... M... ;
or
G89 Z(W)... C(H)... X(U)... R... P... F... K... M... ;

X, C or **Z, C:** Hole position

Z or X: The distance from point R to the bottom of the hole

R: Distance from the initial level to R level

P: Dwell time at bottom of hole

F: Cutting feedrate

K: Number of repetitions (if required)

M: M code for clamping C-axis (if required)

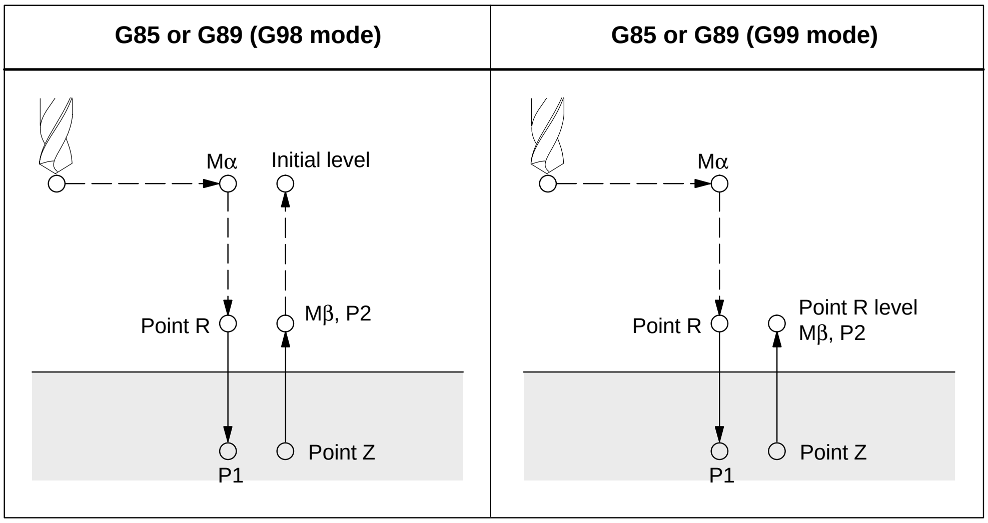


Fig. 4-39

P2: Dwell specified in GUD7, _ZSFR[22]

Explanations

Rapid traverse is performed to point R after positioning at the hole position. Drilling is then carried out from point R to point Z and subsequently returned to point R.

Example

M3 S2500 ;	Rotate the drilling tool
G00 X50.0 C0.0 ;	Positioning X and C-axis
G85 Z–40.0 R–5.0 P500 F5.0 M31 ;	Machining hole 1
C90.0 M31 ;	Machining hole 2
C180.0 M31 ;	Machining hole 3
C270.0 M31 ;	Machining hole 4
G80 M05 ;	Cycle canceling and drilling tool stop

Canned cycle for drilling cancel (G80)

G80 cancels canned cycle.

Format

G80 ;

Explanations

Canned cycle for drilling is canceled and normal operation is continued.

4.2 Program support functions (2)

4.2.1 Changing of tool offset value Programmable data input (G10)

By using the commands of “G10 P· · · X(U)· · · Y(V)· · · Z(W)· · · R(C)· · · Q ;”, it is possible to write and update the tool offset amount using a part program. If an address is omitted in the designation of data input block, the offset amounts for the omitted addresses remains unchanged.

Table 4-6 Description of addresses

Address	Description
P	Offset number (see explanation below)
X Y Z	Offset value on X axis (absolute, incremental) Offset value on Y axis (absolute, incremental) Offset value on Z axis (absolute, incremental)
U V W	Offset value on X axis (incremental) Offset value on Y axis (incremental) Offset value on Z axis (incremental)
R	Tool nose radius offset value (absolute)
C	Tool nose radius offset value (incremental)
Q	Imaginary tool nose number

Address P

Address P specifies the tool offset number and, at the same time, whether tool geometry offset or tool wear offset is to be changed. The value to be specified with address P depends on the setting of MD $MC_EXTERN_FUNCTION_MASK, Bit1 as follows:

$MC_EXTERN_FUNCTION_MASK, Bit1 = 0
P1 to P99: Writing tool wear offset
P100 + (1 to 1500): Writing tool geometry offset

$MC_EXTERN_FUNCTION_MASK, Bit1 = 1
P1 to P9999: Writing tool wear offset
P10000 + (1 to 1500): Writing tool geometry offset

Note

Use of this command in a program allows the tool to advance little by little. This command can also be used to input offset values one at a time from a program by specifying this command successively instead of inputting these values one at a time from the MDI unit.

Example of programming

G10 P16 X32.5 W0.05 ;

- W0.05: Adds 0.05 mm to the offset amount of Z-axis.
- X32.5: Updates the present offset amount of X-axis to 32.5 mm.
- P16: Declares that the following data are reflected to tool offset number “16”.

Fig. 4-40

Setting the workpiece coordinate system shift data

With the commands of “G10 P00 X (U) · · · Z (W) · · · C (H) · · · ;”, it is possible to write and update the workpiece coordinate system shift data using a part program. If an address is omitted in the designation of data input block, the offset amounts for the omitted addresses remain unchanged.

X, Z, C	:	Absolute or incremental setting data of the workpiece coordinate system shift amount
U, W, H	:	Incremental setting data of the workpiece coordinate system shift amount

4.2.2 Subprogram call up function (M98, M99)

This function can be used when subprograms are stored in the part program memory. Subprograms registered to the memory with program numbers assigned can be called up and executed as many times as required.

The created subprograms should be stored in the part program memory before they are called up.

Commands

The M codes indicated in Table 4-7 are used.

Table 4-7 Subprogram call M code

M code	Function
M98	Subprogram call up
M99	End of subprogram

Subprogram call (M98)

- M98 P xxxx yyyy
  y: Program number (max. 4 digits)
  x: Number of repetitions (max. 4 digits)
- The program syntax M98 Pxxxxyyyy is used to call a subprogram with the number yyyy and repeat it xxxx times. If the xxxx is not programmed, the sub–program is executed only once. The subprogram name always consists of 4 digits or is extended to 4 digits by adding 0's.
  For example, if M98 P21 is programmed, the part program memory is searched for program name 0021.spf and the subprogram is executed once. To execute the subprogram 3 times, program M98 P30021.
- As an alternative, the number of subprogram executions can also be pro–grammed at address 'L'. The number of the subprogram is still programmed as Pxxxx. If the number of executions is programmed at both addresses, the number of executions programmed at address 'L' is valid. A valid range for address 'L' is 1 to 9999.
- Nesting of subprograms is possible - the allowable nesting level is four. If the nesting level exceeds this limit, an alarm occurs.

Example:

N20 M98 P20123 ;	Subprogram 1023.spf will be executed twice
N40 M98 P55 L4 ;	Subprogram 0055.spf will be executed four times
N60 M98 P30077 L2 ;	Subprogram 0077.spf will be executed twice The number of executions programmed at address 'P' = 3 is ignored

End of subprogram code (M99)

M99 terminates the subprogram.
If M99 Pxxxx is programmed, execution resumes at block number Nxxxx on the return jump to the main program. The block number must always begin with ’N’. The system initially searches forwards for the block number (from the subprogram call towards the end of the program). If a matching block number is not found, the part program is then scanned backwards (towards the start of the program). If M99 appears without a block number (Pxxxx) in a subprogram, the subprogram is terminated and the processor jumps back to the main program to the block following the subprogram call.
If M99 appears without a block number (Pxxxx) in a main program, it is jumped back back to the head of the main program and the program is executed again.
These M commands are not output to the PLC.

Subprogram return jump with ’RET’

In the Siemens shell cycles for stock removal (as in ISO Dialect), it is necessary after roughing to resume program execution in the main program after the contour definition. To achieve this, the shell cycle must contain a subprogram return jump to the block after the end of the contour definition. The RET command has been extended with two optional parameters for skipping the blocks with the contour definition in the stock removal cycles after the subprogram call (with G71–G73).
The command RET (STRING: <sequence no./label>) is used to resume program execution in the calling program (main program) at the block with <sequence no./label>.
If program execution is to be resumed at the next block after <sequence no./label>, the 2nd parameter in the RET command must be > 0; RET (<sequence no./label>, 1). If a value > 1 is programmed for the 2nd parameter, the subprogram also jumps back to the block after the block with <sequence no./label>.
In G70–G73 cycles, the contour to be machined is stored in the main program. The extended RET command is required in order to resume execution after the contour definition in the main program at the end of G70 (finish cut via contour with stock removal cycle). To jump to the next NC block after the contour definition at the end of the shell cycle for G70, the shell cycle must be terminated with the following return syntax:
RET (’N’ << $C_Q, 1)
Search direction:
The search direction for <sequence no./label> is always forwards first (towards the end of the program) and then backwards (towards the head of program).

Example

```
N10 X10. Y20.
N20 G71 P30 Q60 U1 W1 F1000 S1500
            N10 ... ;                    Shell cycle for stock removal cycle
            N20 DEF STRING[6]BACK
            N30 ...
            N90
            N100 RET ('N'<<$C_Q, 1) ;    Return jump to block after
                                         ;Contour def. –> N70
N30 X50. Z20.
N40 X60.
N50 Z55.
N60 X100. Z70.
N70 G70 P30 Q60
N80 G0 X150. Z200.
N90 M30
```

Note

M30 in Siemens mode: is interpreted as a return jump in a subprogram.
M30 in ISO Dialect mode: is also interpreted as the end of the part program in a subprogram.

4.3 Automating support functions

4.3.1 Skip function (G31)

By specifying “G31 X(U)... Z(W)... F... ;”, special linear interpolation is executed. If a skip signal is input during the execution of linear interpolation, linear interpolation is interrupted and the program advances to the next block without executing the remaining linear interpolation.

The skip function is used when the end of machining is not programmed but specified with a signal from the machine. It is used also for measuring the dimensions of a workpiece. For details of how to use this function, refer to the manual supplied by the machine tool builder.

Format

G31 X... Z... F_;
G31: One–shot G code (It is effective only in the block in which it is specified)

If skip signal is turned ON

When the skip signal is input, the coordinate values of the point where the skip signal is input are automatically saved to the parameters. Therefore, the coordinate values of the skip point can be used as the coordinate data in macro programs.

$AA_IM[X]	Saving the X-axis coordinate value
$AA_IM[Z]	Saving the Z-axis coordinate value

If skip signal is not turned ON

If the skip signal is not turned ON during the execution of the commands specified in the G31 block, the operation stops upon completion of these commands and an alarm occurs. Note that G31 is a non-modal G code.

If G31 is issued while the skip signal input is ON, alarm 21700 is issued.

Operation after skip signal ON

How the axes move after the turning ON of the skip signal varies depending on the commands specified in the block to be executed next.

When axis move commands in the next block are incremental commands

The position where the skip signal is turned ON is taken as the reference point to execute the incremental commands in the next block.

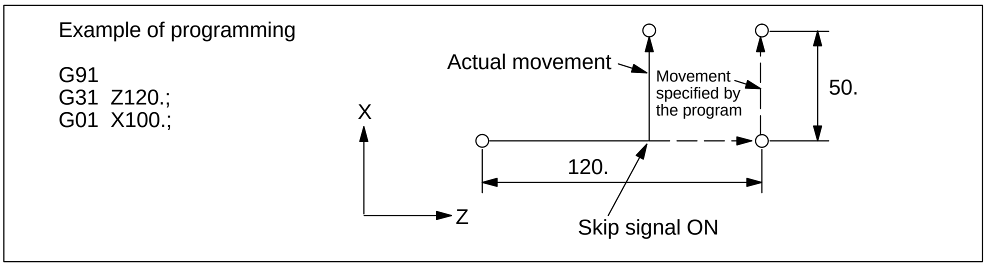


Fig. 4-41

When axis move command in the next block is absolute command (one axis)

The axis specified in the next block moves to the specified position and the other axis remains at the position where the skip signal has turned ON.

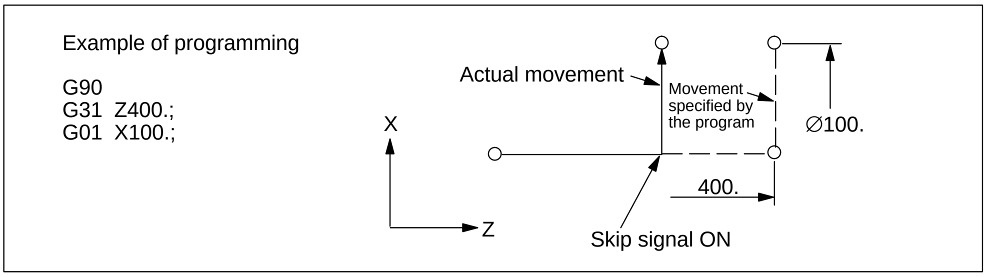


Fig. 4-42

When axis move commands in the next block are absolute command (two axes)

The axes move to the specified position when the skip signal is turned ON.

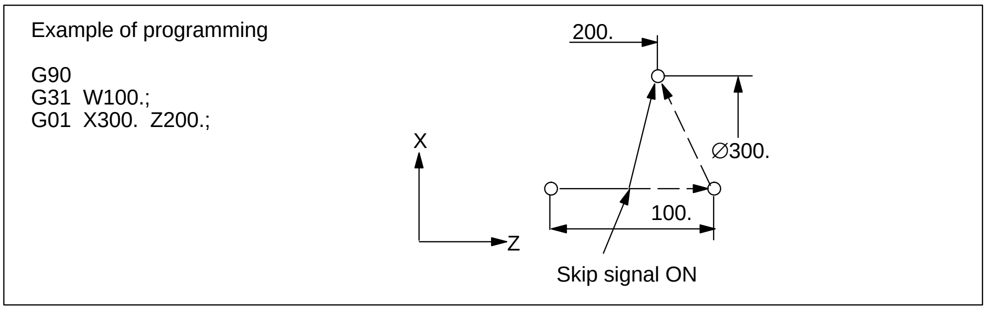


Fig. 4-43

Note

Before specifying G31, cancel the nose R offset mode by specifying G40. If G31 is specified without canceling the nose R offset mode, an alarm occurs.

4.3.2 Multistage skip (G31, P1–P2)

The multistage skip function stores coordinates in a macro variable within a block specifying P1 to P2 after G31 whenever a skip signal is turned on. In order to match multiple Pn (n=1,2) as well as to match a Pn on a one–to–one basis, one skip signal can be set at a time.

Format

Move command
G31 X... Z... F... P ... ;
X, Z: End point
F: Feedrate
P: P1–P2

Explanation

Multistage skip is activated by specifying P1 or P2 in a G31 block. The digital inputs are assigned to addresses P1 and P2 through machine data as follows:
P1: $MN_EXTERN_MEAS_G31_P_SIGNAL[0]
P2: $MN_EXTERN_MEAS_G31_P_SIGNAL[1]

For an explanation of selecting (P1 or P2), refer to the manual supplied by the machine tool builder.

4.4 Macroprograms

The NC has a set of instructions that can be used by the machine tool builders and the users to implement the original functions. The program created by using these instructions is called a macroprogram, which can be called and executed by the commands specified in a block with G65 or G66.

A macroprogram provides the following:

- Variables can be used.
- Arithmetic and logical operations using variables and constants are possible.
- Control commands for branch and repeat can be used.
- Commands to output messages and data can be used.
- Arguments can be specified.

This makes it possible to create a program in which complicated operations and operations requiring conditional judgment are included.

4.4.1 Differences from subprograms

Differences between macroprograms and subprograms are indicated below.

- With macroprogram call up commands (G65, G66), arguments can be specified. However, with subprogram call up command (M98), it is not possible to use arguments.
- If commands other than P, Q, and L are specified in the M98 block, the program jumps to the specified subprogram after executing these commands. With G65 and G66, commands other than P and L are regarded as argument specification and the program jumps to the specified macroprogram immediately. In this case, however, the commands specified preceding G65 and G66 are executed normally.

4.4.2 Macroprogram call (G65, G66, G67)

Macroprograms are usually executed after being called up.

The procedure used for calling up a macroprogram is indicated in Table 4-8.

Table 4-8 Macroprogram calling format

Calling up method	Command code	Remarks
Simple call up	G65	
Modal call up (a)	G66	Canceled by G67

Simple call up (G65)

Format

G65 P... L... ;

By specifying “G65 P... L... <argument specification>; ”, the macroprogram which is assigned the program number specified with P is called up and executed L times.

If it is necessary to pass arguments to the called up macroprogram, these arguments can be specified in this block.

Table 4-9 P and L commands

Address	Description	Number of digits
P	Program number	5 digits
L	Number of repetitions	9 digits

Modal call up (G66, G67)

The modal call up commands set the mode for calling up a macroprogram. The specified macroprogram is called up and executed when the specified conditions are satisfied.

- By specifying “G66 P... L ... <argument-specification>; ”, the mode for calling up the macroprogram is set. Once this block is executed, the macroprogram which is assigned the program number specified with P is called up and executed L times after the completion of move commands.

  If an argument is specified, the argument is passed to the macroprogram each time it is called up as with the simple call up of a macroprogram. The correspondence between the address of argument and local variables is the same as in the case of simple call up (G65).
- G67 cancels the G66 mode. When arguments are specified, G66 must be specified before all arguments. If G66 is specified, G67 must be specified in the same program corresponding to it.

Table 4-10 Modal call up conditions

Call up conditions	Mode setting code	Mode cancel code
After the execution of move command	G66	G67

Modal call up (G66)

In the modal call up mode, the specified macroprogram is called up and executed at each execution of a move command. If more than one G66 is specified in the same program, the prior G66 command specified is valid during the execution of a macroprogram called up by the G66 command given later. Therefore, after the execution of a move command given in the macroprogram called up by G66 specified later, the macroprogram specified with the previous G66 is also executed. In other words, the macroprograms are executed sequentially starting with the one specified last.

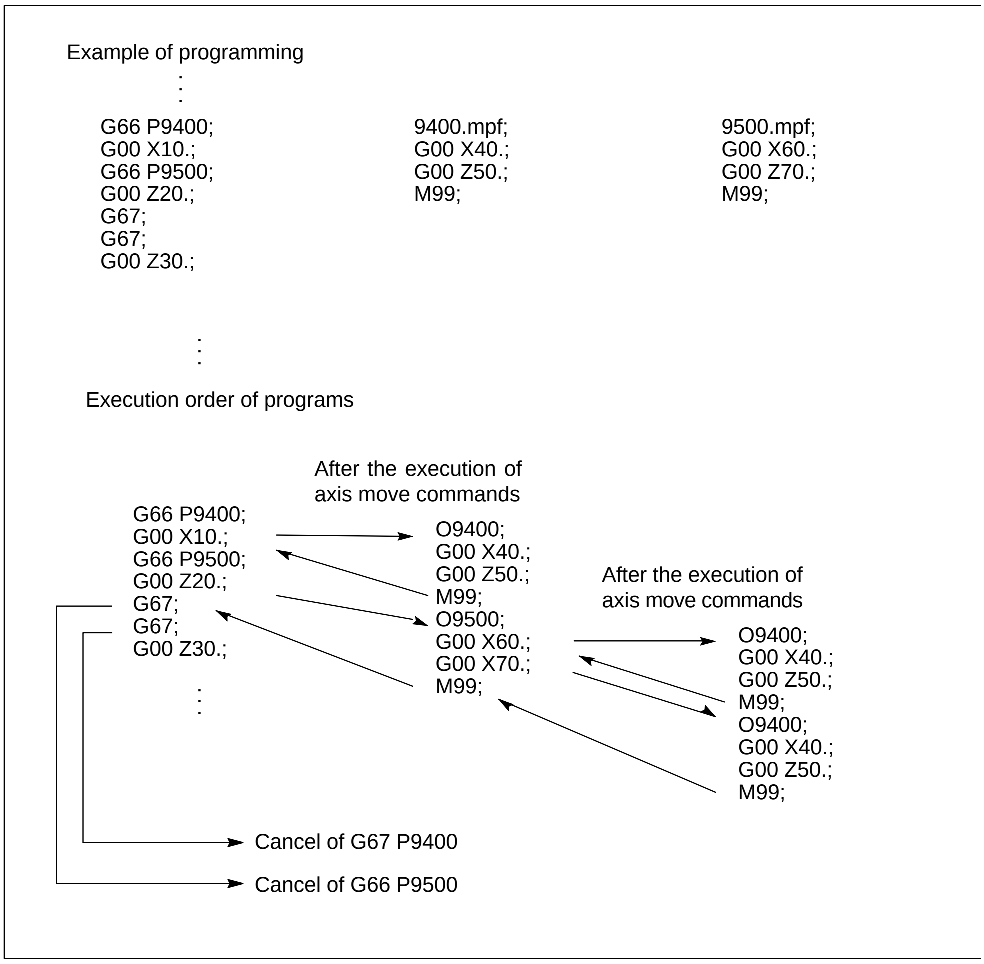


Fig. 4-44 Nesting of macroprogram call

Note

If macroprogram call up is nested by specifying more than one G66, cancel code G67, cancels G66 sequentially beginning with the one specified last. It is not allowed to specify G66 in the macroprogram which is called up by G66.

Specifying argument

The term “to specify argument” means “assigning a real number” for local variables used in a macroprogram. There are two types of argument specifications: type I and type II. These types can be used as required, including a combination of the two types.

Correspondence between addresses and system variables (Type I)

Table 4-11 Address – variable correspondence and usable addresses for call up commands (type I)

Address – variable correspondence		Address – variable correspondence	
Address in Type I	**System variable**	**Address in Type I**	**System variable**
A	$C_A	Q	$C_Q
B	$C_B	R	$C_R
C	$C_C	S	$C_S
D	$C_D	T	$C_T
E	$C_E	U	$C_U
F	$C_F	V	$C_V
H	$C_H	W	$C_W
I	$C_I[0]	X	$C_X
J	$C_J[0]	Y	$C_Y
K	$C_K[0]	Z	$C_Z
M	$C_M		

Correspondence between addresses and system variables (Type II)

To use I, J, and K, they must be specified in the order of I, J, and K. Suffixes 1 to 10 specified in the table below indicate the order they are used in a set, and the suffix is not written in actual commands.

Since addresses I, J, K can be programmed up to ten times in a block with macro call, an array index must be used to access the system variables within the macro program for these addresses. The syntax for these three system variables is then $C_I[..], $C_j[..], $C_K[..]. The values are stored in the array in the order programmed. The number of addresses I, J, K programmed in the block is stored in variables $C_I_NUM, $C_J_NUM and $C_K_NUM.

Unlike the rest of the system variables, an array index must always be specified for these three variables. Array index 0 must always be used for cycle calls (e.g. G81); e.g. N100 R10 = $C_I[0]

Table 4-12 Address – variable correspondence and usable addresses for call up commands (type II)

Address – variable correspondence		Address – variable correspondence	
Address in Type II	**System variable**	**Address in Type II**	**System variable**
A	$C_A	K5	$C_K[4]
B	$C_B	I6	$C_I[5]
C	$C_C	J6	$C_J[5]
I1	$C_I[0]	K6	$C_K[5]
J1	$C_J[0]	I7	$C_I[6]
K1	$C_K[0]	J7	$C_J[6]
I2	$C_I[1]	K7	$C_K[6]
J2	$C_J[1]	I8	$C_I[7]
K2	$C_K[1]	J8	$C_J[7]
I3	$C_I[2]	K8	$C_K[7]
J3	$C_J[2]	I9	$C_I[8]
K3	$C_K[2]	J9	$C_J[8]
I4	$C_I[3]	K9	$C_K[8]
J4	$C_J[3]	I10	$C_I[9]
K4	$C_K[3]	J10	$C_J[9]
I5	$C_I[4]	K10	$C_K[9]
J5	$C_J[4]		

Note: If more than one set of I, J, or K is specified, the order of sets is determined for each I/J/K set, so that variable numbers are determined corresponding to that order.

Example of argument specification

When arguments are specified, the macroprogram call up code must always be specified before the specification of arguments. If argument specification is given before the macroprogram call up code, an alarm occurs. The value of argument specification can include a sign and decimal point independent of the address.

If no decimal point is used, the value is saved to the variable as the value with a decimal point according to the normal number of digits of that address.

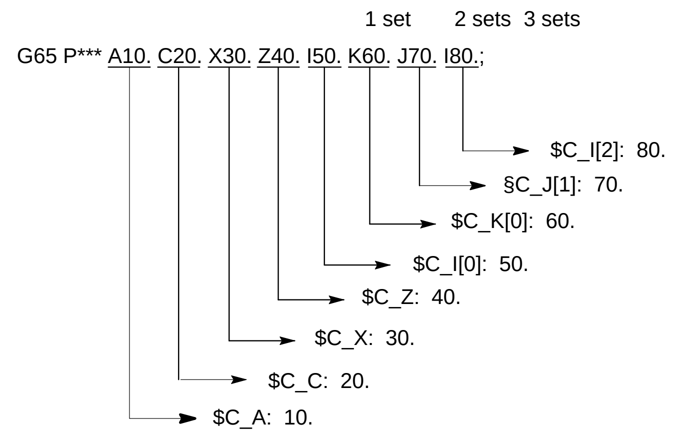


Fig. 4-45 Example of argument specification

Siemens mode/ISO mode macro program execution

The called macro program can either be executed in Siemens mode or ISO mode. The execution mode is decided in the first block of the macro program.
If a PROC <program name> instruction is included in the first block of the macro program, it is automatically switched to Siemens mode. If no such instruction is included, ISO mode is retained.

By executing a macro program in Siemens mode, transfer parameters can be stored into local variables using the DEF instruction. In ISO mode, however, transfer parameters cannot be stored into local variables.

In order to read transfer parameters within the macroprogram executed in ISO mode, switch to Siemens mode by G290 command.

Examples

Main program containing the macro call:

```
_N_M10_MPF:
N10 M3 S1000 F1000
N20 X100 Y50 Z33
N30 G65 P10 F55 X150 Y100 S2000
N40 X50
N50 ....
N200 M30
```

Macro program in Siemens mode:

```
_N_0010_SPF:
PROC 0010 ; Switching into Siemens mode
N10 DEF REAL X_AXIS ,Y_AXIS, S_SPEED, FEED
N15 X_AXIS = $C_X Y_AXIS = $C_Y S_SPEED = $C_S FEED = $C_F
N20 G01 F=FEED G95 S=S_SPEED
...
N80 M17
```

Macro program in ISO mode:

```
_N_0010_SPF:
N10 G290 ; Switching into Siemens mode,
          ; if transfer parameters need to be read
N20 G01 F=$C_F G95 S=$C_S
N30 G1 X=$C_X Y=$C_Y
N40 G291 ; Switching into ISO mode
N50 M3 G54 T1
N60
...
N80 M99
```

4.5 Advanced functions

4.5.1 High–speed cycle cutting (G05)

The G05 command is used to call any subprogram simular to a M98 P_ subprogram call. The subprogram to be called can be a pre–compiled partprogram deriving from Siemens code.

Format

G05 Pxxxxx Lxxx ;

Pxxxxx program number to be called
Lxxx number of repetitions
(L1 applies when this parameter is omitted)

Example

G05 P10123 L3 ;

This block calls program 10123.mpf and executes it 3 times.

Limitations

- Only Siemens code part programs can be pre–compiled.
- When calling a subprogram by G05, it is not switched into Siemens mode.
  The G05 command behaves like a M98 P_ subprogram call.
- A block containing a G05 command without address P is ignored without alarm.
- A block containing a G05.1 command with or without address P as well as G05 P0 or G05 P01 is ignored without alarm.

4.5.2 Polygonal turning

When rotating the workpiece and a tool at a certain ratio, a polygonal figure can be machined.

For example, by changing conditions such as rotation ratio of workpiece and tool as well as the number of cutters, a square or hexagon can be machined. Under certain circumstances, the machining time can be reduced compared to machining using C and X axis in polar coordinate interpolation.
Due to the nature of such kind of machining however, the machined figure is not exactly polygonal. Typical applications are the heads of square and/or hexagon bolts or nuts.

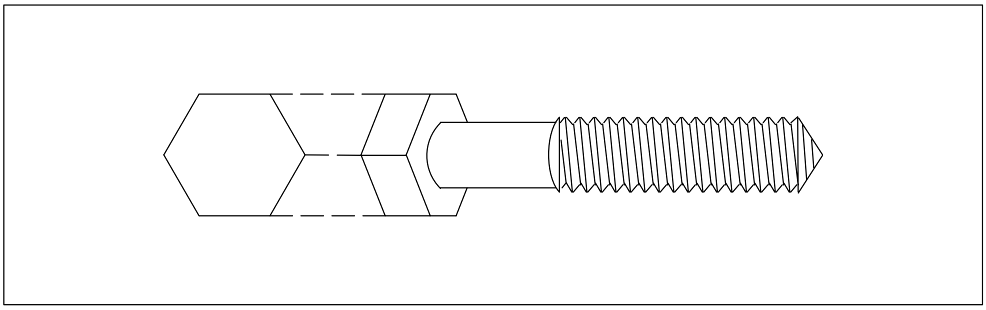


Fig. 4-46 Hexagon bolt

Format

G51.2 P...Q...;

P, Q: Rotation ratio (spindle / Y axis)
Setting range: Integer 1 to 9 for both P and Q

The sign of address Q is used to specify the Y axis rotation direction.

Example

G00 X120.0 Z30.0 S1200.0 M03; set workpiece rotation speed to 1200 rmp
G51.2 P1 Q2 ; start tool rotation (2400 rpm)
G01 X80.0 F10.0 ; X axis infeed
G04 X2. ;
G00 X120.0 ; X axis retract
G50.2 ; stop tool rotation
M05 ; Spindle stop

G50.2 and G51.2 need to be specified in seperate blocks.

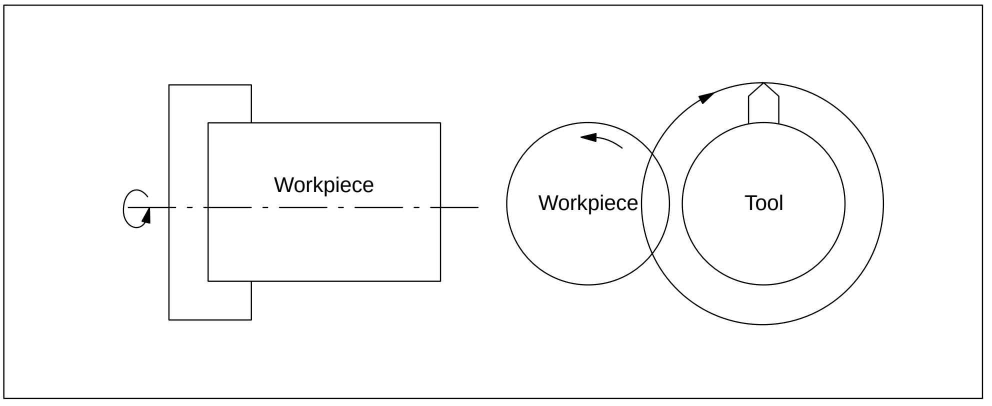


Fig. 4-47 Polygonal turning

Abbreviations

A

ASCII	American Standard Code for Information Interchange
ASUB	Asynchronous Subroutine
BA	Mode of operation
BAG	Mode Group
BCD	Binary Coded Decimals
BCS	Basic Coordinate System
BIN	Binary Files
BP	Basic Program
C1 .. C4	Channel 1 to channel 4
CAD	Computer–Aided Design
CAM	Computer–Aided Manufacturing
CNC	Computerized Numerical Control
COM	Communication
COR	Coordinate Rotation
CPU	Central Processing Unit
CR	Carriage Return

CRC	Cutter Radius Compensation
CSF	Control System Flowchart (PLC programming method)
CTS	Clear To Send (serial data interfaces)
CUTOM	Cutter Radius Compensation (Tool radius compensation)
DB	Data Block in the PLC
DBB	Data Block Byte in the PLC
DBW	Data Block Word in the PLC
DBX	Data Block Bit in the PLC
DC	Direct Control: The rotary axis is moved along the shortest path to the absolute position within one revolution.
DCE	Data Communications Equipment
DDE	Dynamic Data Exchange
DIO	Data Input/Output: Data transfer display
DIR	Directory
DLL	Dynamic Link Library: Module which can be accessed by a running program. Often contains program sections that are required by different programs.
DOS	Disk Operating System
DPM	Dual–Port Memory
DPR	Dual–Port RAM
DRAM	Dynamic Random Access Memory

DRF	Differential Resolver Function
DRY	Dry Run
DSB	Decoding Single Block
DTE	Data Terminal Equipment
DW	Data Word
EIA Code	Special punchtape code, number of punched holes per character always odd
ENC	Encoder
EPROM	Erasable Programmable Read Only Memory
FB	Function Block
FC	Function Call: Function block in the PLC
FDB	Product Database
FDD	Floppy Disk Drive
FDD	Feed Drive
FEPROM	Flash–EPROM
FIFO	First In First Out: Memory which operates without address specification from which data are read in the same order as they are stored.
FM	Function Module
FM–NC	Function Module – Numerical Control
FPU	Floating Point Unit

FRA	Frame Block
FRAME	Data Record (frame)
FST	Feed Stop
GUD	Global User Data
HD	Hard Disk
HEX	Abbreviation for hexadecimal
HHU	Handheld Unit
HMI	Human Machine Interface: SINUMERIK operating functions for operator control, programming and simulation. MMC and HMI are identical in meaning.
HW	Hardware
I	Input
I/O	Input/Output
I/RF	Infeed/Regenerative Feedback Unit (power supply) of SIMODRIVE 611(D)
IK (GD)	Implicit Communication (Global Data)
IKA	Interpolative Compensation
IM	Interface Module
IMR	Interface Module Receive
IMS	Interface Module Send
INC	Increment

INI	Initializing Data
IPO	Interpolator
IS	Interface Signal
ISO Code	Special punchtape code, number of punched holes per character always even
JOG	Jog mode
K Bus	Communication Bus
$K_Ü$	Transmission Ratio
K_V	Servo Gain Factor
LAD	Ladder Diagram (PLC programming method)
LEC	Leadscrew Error Compensation
LF	Line Feed
LUD	Local User Data
MB	Megabyte
MC	Measuring Circuit
MCP	Machine Control Panel
MCS	Machine Coordinate System
MD	Machine Data
MDA	Manual Data Automatic

MMC	Human Machine Communication: User interface on numerical control systems for operator control, programming and simulation. MMC and HMI are identical in meaning.
MPF	Main Program File: NC part program (main program)
MPI	Multi Port Interface
MSD	Main Spindle Drive
NC	Numerical Control
NCK	Numerical Control Kernel (with block preparation, traversing range, etc.)
NCU	Numerical Control Unit: Hardware unit of the NCK
NURBS	Non Uniform Rational B–Spline
O	Output
OB	Organization Block in the PLC
OEM	Original Equipment Manufacturer: The manufacturer of equipment that is marketed by another vendor, typically under a different name.
OI	Operator Interface
OP	Operator Panel
OPI	Operator Panel Interface
P Bus	I/O (Peripherals) Bus
PC	Personal Computer
PCIN	Name of SW for exchanging data with the control system

PCMCIA	Personal Computer Memory Card International Association
PG	Programming Device
PLC	Programmable Logic Control
PP	Production Planning
RAM	Random Access Memory (read–write memory)
REF	Reference Point Approach Function
REPOS	Reposition Function
ROV	Rapid Override
RPA	R Parameter Active: Memory area in the NCK for R–NCK for R parameter numbers
RPY	Roll Pitch Yaw: Type of coordinate system rotation
RTS	Request To Send (serial data interfaces)
SBL	Single Block
SD	Setting Data
SDB	System Data Block
SEA	Setting Data Active: Identification (file type) for setting data
SFB	System Function Block
SFC	System Function Call
SK	Softkey

SKP	Skip Block
SM	Stepper Motor
SOP	Shopfloor–Oriented Programming
SPF	Sub Program File (subroutine file)
SR	Subroutine
SRAM	Static RAM (battery–backed)
STL	Statement List
SSI	Serial Synchronous Interface
SW	Software
SYF	System Files
T	Tool
TC	Tool Change
TEA	Testing Data Active: Identifier for machine data
TLC	Tool length compensation
TNRC	Tool Nose Radius Compensation
TO	Tool Offset
TOA	Tool Offset Active: Identification (file type) for tool offsets
TRANSMIT	Transform Milling into Turning: Coordinate conversion on turning machines for milling operations
TRC	Tool Radius Compensation

UFR	User Frame: Zero offset
V.24	Serial Interface (definition of interchange lines between DTE and DCE)
WCS	Workpiece Coordinate System
WPD	Work Piece Directory
ZO	Zero Offset
ZOA	Zero Offset Active: Identification (file type) for zero offset data

Notes

A

Terms

B

Important terms are listed below in alphabetical order, accompanied by explanations. Cross–references to other entries in this glossary are indicated by the symbol ”–>”.

A

A spline

The A spline runs tangentially through the programmed interpolation points (3rd degree polynomial).

Absolute dimension

A destination for an axis movement is defined by a dimension that refers to the origin of the currently active coordinate system. See also –> incremental dimension.

AC control

(Adaptive Control)

A process variable (e.g. path–specific or axial feedrate) can be controlled as a function of another, measured process variable (e.g. spindle current). Typical application: To maintain a constant chip removal volume during grinding.

Acceleration with jerk limitation

In order to obtain the optimum acceleration gradient for the machine while providing effective protection for the mechanical components, the machining program offers a choice between instantaneous acceleration and continuous (smooth) acceleration.

Access rights

The CNC program blocks and data are protected by a 7–level system of access restrictions:

- Three password levels for system manufacturers, machine manufacturers and users and
- Four keyswitch settings which can be evaluated via the PLC.

Activate/deactivate

Working area limitation is a means of restricting the axis movement over and above the restrictions imposed by the limit switches. A pair of values delimiting the protected zone area can be specified for each axis.

Address

Addresses are fixed or variable identifiers for axes (X, Y, ...), spindle speed (S), feedrate (F), circle radius (CR), etc.

Alarms

All –> messages and alarms are displayed in plain text on the operator panel. Alarm text also includes the date, time and corresponding symbol for the reset criterion.

Alarms and messages are displayed separately.

3. Alarms and messages in the part program
   Alarms and messages can be displayed directly from the part program in plaintext.
4. Alarms and messages from PLC
   Alarms and messages relating to the machine can be displayed from the PLC program in plaintext. No additional function block packages are required for this purpose.

Analog input/output module

Analog input/output modules are signal transducers for analog process signals.

Analog input modules convert analog measured values into digital values that can be processed in the CPU.

Analog output modules convert digital values into manipulated variables.

Approach fixed machine point

Approach motion towards one of the predefined –> fixed machine points.

Archiving

Exporting files and/or directories to an **external** storage device.

Asynchronous subroutine

- A part program that can be started asynchronously (or independently) by means of an interrupt signal (e.g. ”High–speed NC input” signal) while the part program is active (SW package 3 and earlier).

- A part program that can be started asynchronously (or independently) of the current program status by means of an interrupt signal (e.g. ”High–speed NC input” signal) (SW package 4 and later).

Automatic

Control system operating mode (block–sequential to DIN): Mode in NC systems in which a –> part program is selected and continuously executed.

Auxiliary functions

Auxiliary functions can be used to pass –> parameters to the –> PLC in –> part programs, triggering reactions there which are defined by the machine manufacturer.

Axes

CNC axes are classified according to their functional scope as:

- Axes: Interpolative path axes
- Positioning axes: Non–interpolative infeed and positioning axes with axis–specific feedrates; axes can move across block limits. Positioning axes need not be involved in workpiece machining as such and include tool feeders, tool magazines, etc.

Axis address

See –> axis identifier

Axis identifier

In compliance with DIN 66217, axes are identified as X, Y and Z for a right–handed rectangular –> coordinate system.

–> Rotary axes rotating around X, Y, Z are assigned the identifiers A, B, C. Additional axes, which are parallel to those specified, can be identified with other letters.

Axis name

See –> axis identifier

Axis/spindle replacement

An axis/spindle is permanently assigned to a particular channel via machine data. This MD assignment can be ”undone” by program commands and the axis/spindle then assigned to another channel.

B

B spline The programmed positions for the B spline are not interpolation points, but merely ”check points”. The curve generated does not pass directly through these check points, but only in their vicinity (1st, 2nd or 3rd degree polynomial).

Back up A copy of the memory contents (hard disk) stored on an external device for data backup and/or archiving..

Backlash compensation Compensation of a mechanical machine backlash, e.g. backlash due to reversal of leadscrews. The backlash compensation can be entered separately for each axis.

Backup battery The backup battery provides non–volatile storage for the –> user program in the –> CPU and ensures that defined data areas and flags, timers and counters are retentive.

Base axis Axis whose setpoint or actual value is employed in calculating a compensatory value.

Basic coordinate system Cartesian coordinate system, is mapped onto machine coordinate system by means of transformation.

In the –> part program, the programmer uses the axis names of the basic coordinate system. The basic coordinate system exists in parallel to the –> machine coordinate system when no –> transformation is active. The difference between the systems relates only to the axis identifiers.

Baud rate Rate at which data transmission takes place (bit/s).

Blank The unmachined workpiece.

Block A section of a –> part program terminated with a line feed. A distinction is made between –> main blocks and –> subblocks.

Block All files required for programming and program execution are known as blocks.

Block search The block search function allows selection of any point in the part program at which machining must start or be continued. The function is provided for the purpose of testing part programs or continuing machining after an interruption.

Booting Loading the system program after Power ON.

Bus connector A bus connector is an S7–300 accessory that is supplied with the –> I/O modules. The bus connector extends the –> S7–300 bus from the –> CPU or an I/O module to the next adjacent I/O module.

C

C axis Axis about which the tool spindle describes a controlled rotational and positioning movement.

C spline The C spline is the best known and the most widely used spline. The spline passes through each of the interpolation points at a tangent and along the axis of curvature. 3rd–degree polynomials are used.

Channel structure The channel structure makes it possible to process the –> programs of individual channels simultaneously and asynchronously.

Circular interpolation The –> tool is required to travel in a circle between defined points on the contour at a specified feed while machining the workpiece.

Clearance control (3D), sensor–driven A position offset for a specific axis can be controlled as a function of a measured process variable (e.g. analog input, spindle current...). This function can automatically maintain a constant clearance to meet the technological requirements of the machining operation.

CNC –> NC

CNC high–level language The high–level language offers: –> user variables, –> predefined user variables, –> system variables, –> indirect programming, –> arithmetic and angular functions, –> relational and logic operations, –> program jumps and branches, –> program coordination (SINUMERIK 840D), –> macros.

CNC programming language

The CNC programming language is based on DIN 66025 with high–level language expansions. The –> CNC programming language and –> high–level language expansions support the definition of macros (sequenced statements).

COM

Numerical control component for the implementation and coordination of communication.

Command axis

Command axes are started from synchronized actions in response to an event (command). They can be positioned, started and stopped fully asynchronous to the part program.

Compensation axis

Axis having a setpoint or actual value modified by the compensation value.

Compensation table

Table of interpolation points. It supplies the compensation values of the compensation axis for selected positions of the base axis.

Compensation value

Difference between the axis position measured by the position sensor and the desired, programmed axis position.

Connecting cables

Connecting cables are pre–assembled or user–assembled 2–wire cables with a connector at each end. They are used to connect the –> CPU via the –> multipoint interface (MPI) to a –> programming device or to other CPUs.

Continuous–path mode

The purpose of continuous–path control mode is to prevent excessive deceleration of the –> path axes at the part program block limits that could endanger the operator or the control, machine or other assets of the plant and to effect the transition to the next block at as uniform a path speed as possible.

Contour

Outline of a –> workpiece.

Contour monitoring

The following error is monitored within a definable tolerance band as a measure of contour accuracy. Overloading of the drive, for example, may result in an unacceptably large following error. In such cases, an alarm is output and the axes stopped.

Coordinate system See –> machine coordinate system, –> workpiece coordinate system

CPU Central Processor Unit –> programmable controller

Cycle Protected subroutine for executing a recurring machining operation on the –> workpiece.

Cycles support The available cycles are listed in menu ”Cycle support” in the ”Program” operating area. Once the desired machining cycle has been selected, the parameters required for assigning values are displayed in plaintext.

D

Data block

3. Data unit of the –> PLC which can be accessed by –> HIGHSTEP programs.
4. Data unit of the –> NC: Data blocks contain data definitions for global user data. These data can be initialized directly when they are defined.

Data transfer program PCIN PCIN is a routine for transmitting and receiving CNC user data, e.g. part programs, tool offsets, etc. via the serial interface. The PCIN program can run under MS–DOS on standard industrial PCs.

Data word A data unit, two bytes in size, within a –> PLC data block.

Deletion of distance–to–go Command in part program which stops machining and clears the remaining path distance to go.

Design

- The SINUMERIK FM–NC is installed in the CPU tier of the SIMATIC S7–300. The 200 mm wide, fully encapsulated module has the same external design as the SIMATIC S7–300 modules.
- The SINUMERIK 840D is installed as a compact module in the SIMODRIVE 611D converter system. It has the same dimensions as a 50 mm wide SIMODRIVE 611D module. The SINUMERIK 840D comprises the NCU module and the NCU box.
- The SINUMERIK 810D has the same design as the SIMODRIVE 611D with a width of 150mm. The following components are integrated: SIMATIC S7–CPU, 5 digital servo drive controls and 3 SIMODRIVE 611D power modules.

Diagnosis

3. Control operating area
4. The control incorporates a self–diagnosis program and test routines for servicing: Status, alarm and service displays.

Digital input/output module

Digital modules are signal transducers for binary process signals.

Dimensions in metric and inch systems

Position and lead/pitch values can be programmed in inches in the machining program. The control is set to a basic system regardless of the programmable unit of measure (G70/G71).

DRF

Differential Resolver Function NC function which generates an incremental zero offset in AUTOMATIC mode in conjunction with an electronic handwheel.

B

Drift compensation

When the CNC axes are in the constant motion phase, automatic drift compensation is implemented in the analog speed control. (SINUMERIK FM–NC).

Drive

- SINUMERIK FM–NC has an analog ±10V interface to the SIMODRIVE 611A converter system.
- The SINUMERIK 840D control system is linked to the SIMODRIVE 611D converter system via a high–speed digital parallel bus.

E

Editor	The editor makes it possible to create, modify, extend, join and insert programs/texts/program blocks.
Electronic handwheel	Electronic handwheels can be used to traverse the selected axes simultaneously in manual mode. The handwheel clicks are analyzed by the increment analyzer.
Exact stop	When an exact stop is programmed, a position specified in the block is approached accurately and, where appropriate, very slowly. In order to reduce the approach time, –> exact stop limits are defined for rapid traverse and feed.
Exact stop limit	When all path axes reach their exact stop limits, the control responds as if it had reached its destination point precisely. The –> part program continues execution at the next block.
External zero offset	A zero offset specified by the –> PLC.

F

Fast retraction from contour	When an interrupt is received, it is possible to initiate a motion via the CNC machining program which allows the tool to be retracted quickly from the workpiece contour currently being machined. The retraction angle and the distance retracted can also be parameterized. An interrupt routine can be executed after the rapid retraction. (SINUMERIK FM–NC, 810D, 840D).
Feedforward control, dynamic	Contour inaccuracies resulting from following errors can be almost completely eliminated by the dynamic, acceleration–dependent feedforward control function. Feedforward control ensures an excellent degree of machining accuracy even at high tool path velocities. Feedforward control can only be selected or deselected for all axes together via the part program.

Feedrate override

The current feedrate setting entered via the control panel or by the PLC is overlaid on the programmed feedrate (0–200 %). The feedrate can also be corrected by a programmable percentage factor (1–200 %) in the machining program.

An offset can also be applied via motion–synchronous actions independently of the running program.

Finished–part contour

Contour of the finished workpiece. See also –> blank.

Fixed machine point

A point defined uniquely by the machine tool, such as the reference point.

Fixed–point approach

Machine tools can execute defined approaches to fixed points such as tool–change points, loading points, pallet–change points, etc. The coordinates of these points are stored on the control. Where possible, the control moves these axes in –> rapid traverse.

Frame

A frame is a calculation rule that translates one Cartesian coordinate system into another Cartesian coordinate system. A frame contains the components –> zero offset, –> rotation, –> scaling and –> mirroring.

G

General reset

The following memories of the –> CPU are erased by a general reset operation:

- –> Working memory
- Read/write area of the –> load memory
- –> System memory
- –> Backup memory

Geometry

Description of a –> workpiece in the –> workpiece coordinate system.

Geometry axis

Geometry axes are used to describe a 2 or 3–dimensional area in the workpiece coordinate system.

Global main run/subroutine
Each global main run/subroutine can be stored only once under its name in the directory. However, the same name can be used in different directories.

Ground
”Ground” is the term applied to all the electrically inactive, interconnected parts of a piece of equipment which cannot carry any hazardous contact voltage even in the event of a fault.

H

Helical interpolation
The helical interpolation function is ideal for machining internal and external threads using form milling cutters and for milling lubrication grooves. The helix comprises two movements:

3. Circular movement in one plane
4. Linear movement perpendicular to this plane.

High–speed digital inputs/outputs
As an example, high–speed CNC program routines (interrupt routines) can be started via the digital inputs. High–speed, program–driven switching functions can be initiated via the digital CNC outputs (SINUMERIK 840D). (SINUMERIK 840D).

HIGHSTEP
Combination of the programming features for the –> PLC in the S7–300/400 range.

I

Identifier
In accordance with DIN 66025, identifiers (names) for variables (arithmetic variables, system variables, user variables), for subroutines, for vocabulary words and for words can contain several address letters. These letters have the same meaning as the words in the block syntax. Identifiers must be unique. Identical identifiers must not be used for different objects.

B

Inch system of measurement
System of measurement that defines distances in ”inches” and fractions thereof.

Inclined axis
Fixed angular interpolation with allowance for an inclined infeed axis or grinding wheel through specification of the angle. The axes are programmed and displayed in the Cartesian coordinate system.

Increment A destination for axis traversal is defined by a distance to be covered and a direction referenced to a point already reached. See also –> absolute dimension.

Increment Travel path length specification based on number of increments. The number of increments can be stored as a –> setting data or selected with keys labeled with 10, 100, 1000, 10 000.

Initialization block Initialization blocks are special –> program blocks. They contain values which must be assigned before the program is executed. Initialization blocks are used primarily for initializing predefined data or global user data.

Initialization file An initialization file can be created for each –> workpiece. In it, the various variable value instructions which apply exclusively to one workpiece can be stored.

Intermediate blocks Movements with selected tool offset (G41/G42) can be interrupted by a limited number of intermediate blocks (blocks without axis motions in the offset plane). When such blocks are used, the tool offset can still be calculated correctly. The permissible number of intermediate blocks read in advance by the control can be set via system parameters.

Interpolation cycle The interpolation cycle is a multiple of the basic system cycle. It specifies the cycle time for updating the setpoint interface to the position controllers. The interpolation cycle determines the resolution of the velocity profiles.

Interpolative compensation Interpolative compensation provides a means of compensating for leadscrew errors (LEC) and measuring–system errors (MSEC) resulting from the production process.

Interpolator Logical unit of the –> NCK which determines intermediate values for the movements to be traversed on the individual axes on the basis of destination positions specified in the part program.

Interrupt routine

Interrupt routines are special –> subroutines which can be started by events (external signals) in the machining process. The part program block being processed is aborted and the axis position at the instant of interruption is stored automatically.

See –> ASUB

Inverse–time feedrate

On SINUMERIK FM–NC and 840D controls, it is possible to program the time required to traverse the path of a block instead of the feedrate speed for the axis movement (G93).

I/O module

I/O modules create the link between the CPU and the process. I/O modules are:

- –>Digital input/output modules
- –>Analog input/output modules
- –>Simulator modules

J

Jog

Control system operating mode (setup): The machine can be set up in Jog mode. Individual axes and spindles can be jogged by means of direction keys. Other functions in Jog mode are –> reference point approach, –> Repos and –> Preset –> (set actual value).

K

Keyswitch

3. **S7–300**: The keyswitch is the mode selector switch on the –> CPU. The keyswitch is operated by means of a removable key.
4. **840D/FM–NC**: The keyswitch on the –> machine control panel has 4 positions which are assigned functions by the operating system of the control. There are also three keys of different colors belonging to the keyswitch that can be removed in the specified positions.

$K_Ü$

Transmission Ratio

K_V

Servo gain factor, control variable of a control loop

B

L

Languages

The user interface texts, system messages and alarms are available in five system languages (floppy disk):
German, English, French, Italian and **Spanish**.
The user can select **two** of the listed languages at a time in the control.

Leadscrew error compensation

Compensation of mechanical inaccuracies in a leadscrew involved in the feed motion. Errors are compensated by the control based on stored deviation measurements.

Limit speed

Minimum/maximum (spindle) speed: The maximum speed of a spindle can be limited by values defined in the machine data, the –> PLC or –> setting data.

Linear axis

The linear axis is an axis which, in contrast to a rotary axis, describes a straight line.

Linear interpolation

The tool travels along a straight line to the destination point while machining the workpiece.

Look Ahead

The **Look Ahead** function is a means of optimizing the machining velocity by looking ahead over a parameterizable number of traversing blocks.

Look Ahead for contour violations

The control detects and reports the following types of collision:

3. Path is shorter than tool radius.
4. Width of inside corner is less than the tool diameter.

M

Machine

Control operating area

Machine axes

Axes which exist physically on the machine tool.

Machine control panel

An operator panel on a machine tool with operating elements such as keys, rotary switches, etc. and simple indicators such as LEDs. It is used for direct control of the machine tool via the PLC.

Machine coordinate system

System of coordinates based on the axes of the machine tool.

Machine zero

A fixed point on the machine tool which can be referenced by all (derived) measurement systems.

Machining channel

A channel structure makes it possible to reduce downtimes by allowing sequences of motions to be executed in parallel. For example, a loading gantry can execute its movements during a machining operation. In this case, a CNC channel ranks as an autonomous CNC control complete with decoding, block preparation and interpolation.

Macros

Multiple programming language instructions can be combined in a single statement. This abbreviated sequence of instructions is called in the CNC program under a user–defined name. The macro executes the instructions sequentially.

Main block

A block prefixed by ":" containing all the parameters required to start execution of a –> part program.

Main program

–> Part program identified by a number or name in which other main programs, subroutines or –> cycles may be called.

Main run

Part program blocks which have been decoded and prepared by the preprocessor are executed during the "main run".

MDA

Control system operating mode: Manual Data Automatic. In the MDA mode, individual program blocks or block sequences with no reference to a main program or subroutine can be input and executed immediately afterwards through actuation of the NC Start key.

Measuring circuits

- SINUMERIK FM–NC: The requisite control circuits for axes and spindles are integrated in the control module as standard. A maximum total of 4 axes and spindles can be implemented, with no more than 2 spindles.
- SINUMERIK 840D: The signals from the sensors are analyzed in the SIMODRIVE 611D drive modules. The maximum total configuration is 8 axes and spindles, with no more than 5 spindles.

Messages

All messages programmed in the part program and –> alarms detected by the system are displayed in plain text on the operator panel. Alarms and messages are displayed separately.

Metric system

Standardized system of units for lengths in millimeters (mm), meters (m), etc.

Mirroring

Mirroring exchanges the leading signs of the coordinate values of a contour in relation to an axis. Mirroring can be performed simultaneously in relation to several axes.

Mode

An operating concept on a SINUMERIK control. The modes –> Jog, –> MDA, –> Automatic are defined.

Mode group

All axes/spindles are assigned to one and only one channel at any given time. Each channel is assigned to a mode group. The same –> mode is always assigned to the channels of a mode group.

Motion synchronization

This function can be used to initiate actions that are synchronized with the machining operation. The starting point of the actions is defined by a condition (e.g. status of a PLC input, time elapsed since beginning of a block). The start of motion–synchronous actions is not tied to block boundaries. Examples of typical motion–synchronous actions are:
Transfer M and H auxiliary functions to the PLC or deletion of distance–to–go for specific axes.

Multipoint interface

The multipoint interface (MPI) is a 9–pin sub–D port. A parameterizable number of devices can be connected to an MPI for the purpose of communicating with one another:

- Programming devices
- MMI (HMI) systems
- Other automation systems

The ”Multipoint Interface MPI” parameter block of the CPU contains the –> parameters which define the properties of the multipoint interface.

N

NC

Numerical Control It incorporates all the components of the machine tool control system: –> NCK, –> PLC, –> MMC, –> COM.
Note: CNC (computerized numerical control) would be a more appropriate description for the SINUMERIK 840D or FM–NC controls. computerized numerical control.

NCK

Numerical Control Kernel: Component of the NC control which executes –> part programs and essentially coordinates the movements on the machine tool.

Network

A network is the interconnection of several S7–300s and other terminal devices such as a programming device, for example, interlinked by means of –> connecting cables. The networked devices interchange data via the network.

Node number

The node number is the ”contact address” of a –> CPU or the –> programming device or another intelligent I/O module if these devices are exchanging data with one another via a –> network. The node number is assigned to the CPU or the programming device by the S7 tool –> ”S7 Configuration”.

NRK

Numeric Robotic Kernel (operating system of the –> NCK)

NURBS

Motion control and path interpolation are implemented internally in the control on the basis of NURBS (Non–Uniform Rational B Splines). A standard procedure is thus available (SINUMERIK 840D) as an internal control function for all modes of interpolation.

O

Oblique–plane machining
Drilling and milling operations on workpiece surfaces which are oblique to the coordinate planes of the machine are supported by the ”Oblique surface machining” function. The position of the oblique plane can be defined by inclining the coordinate system (see FRAME programming).

OEM
The scope for implementing individual solutions (OEM applications) for the SINUMERIK 840D has been provided for machine manufacturers who wish to create their own operator interface or integrate process–oriented functions in the control.

Offset memory
Data area in the control in which tool offset data are stored.

Online tool offset
This function can be used for grinding tools only.

The reduction in size of the grinding wheel resulting from dressing is transferred as a tool offset to the currently active tool and immediately applied.

Operator interface
The operator interface (OI) is the human–machine interface of a CNC. It takes the form of a screen and has eight horizontal and eight vertical softkeys.

Oriented spindle stop
Stops the workpiece spindle at a specified orientation angle, e.g. to perform an additional machining operation at a specific position.

Oriented tool retraction
RETTOOL: If machining is interrupted (e.g. when a tool breaks), a program command can be used to retract the tool in a user–specified orientation by a defined distance.

Override
Manual or programmable control feature which enables the user to override programmed feedrates or speeds in order to adapt them to a specific workpiece or material.

P

Parameters

3. **S7–300**: The S7–300 uses two types of parameter:
   – Parameter of a STEP 7 statement
     A parameter of a STEP 7 statement is the address of the operand to be processed or a constant.
   – Parameter of a –> parameter block
     A parameter of a parameter block determines the behavior of a module.
4. **840D/810D/FM–NC**:
   – Control operating area
   – Computation parameter, can be set any number of times or queried by the programmer for any purpose in the part program.

Part program

A sequence of instructions to the NC control which combine to produce a specific –> workpiece by performing certain machining operations on a given –> blank.

Part program management

The part program management function can be organized according to –> workpieces. The quantity of programs and data to be managed is dependent on the control memory capacity and can also be configured via MD settings. Each file (programs and data) can be given a name consisting of a maximum of 16 alphanumeric characters.

Path axis

Path axes are all the machining axes in the –> channel which are controlled by the –> interpolator such that they start, accelerate, stop and reach their end positions simultaneously.

Path feed

The path feed acts on –> path axes. It represents the geometrical sum of the feeds on the participating –> path axes.

Path velocity

The maximum programmable path velocity depends on the input resolution. With a resolution of 0.1 mm, for example, the maximum programmable path velocity is 1000 m/min.

PG

Programming Device

PLC	Programmable Logic Control –> Speicherprogrammierbare Steuerung. Component of the –> NC: Programmable controller for processing the control logic on the machine tool.
PLC program memory	• SINUMERIK FM–NC: The PLC user program, the user data and the basic PLC program are stored together in the PLC user memory of the CPU 314. S7–CPU314 has a user memory of 24 KB for this purpose. • SINUMERIK 840D: The PLC user program, the user data and the basic PLC program are stored together in the PLC user memory. The PLC user memory can be expanded up to 128 KB. • SINUMERIK 810D: The PLC user program, the user data and the basic PLC program are stored together in the PLC user memory of the CPU 314. The basic version of the S7–CPU314 has a user memory of 64 KB which can be optionally expanded up to 128 KB.
PLC programming	The PLC is programmed with the **STEP 7** software. The STEP 7 programming software is based on the standard **WINDOWS** operating system and incorporates the functionality of STEP 5 programming with innovative expansions and developments.
Polar coordinates	A coordinate system which defines the position of a point on a plane in terms of its distance from the origin and the angle formed by the radius vector with a defined axis.
Polynomial interpolation	Polynomial interpolation provides a means of generating a very wide range of curves, including **straight–line, parabolic and exponential functions** (SINUMERIK 840D/810D).
Positioning axis	An axis which performs an auxiliary movement on a machine tool (e.g. tool magazine, pallet transport). Positioning axes are axes that do not interpolate with the –> path axes.
Power ON	The action of switching the control off and then on again.

B

Preprocessing memory, dynamic

The traversing blocks are preprocessed prior to execution and stored in a ”preprocessing memory”. Block sequences can be executed at a very fast rate from the memory. Blocks are uploaded continuously to the preprocessing memory during machining.

Preprocessing stop

Program command. The next block in a part program is not executed until all other blocks which have already been preprocessed and stored in the preprocessing memory have been executed.

See also ”Preprocessing memory”.

Preset

The control zero point can be redefined in the machine coordinate system by means of the Preset function. Preset does not cause the axes to move; instead, a new position value is entered for the current axis positions.

Program

3. Control operating area
4. Sequence of instructions to the control system.

Programmable frames

Programmable –> frames can be used to define new coordinate system starting points dynamically while the part program is running. A distinction is made between absolute definition using a new frame and additive definition with reference to an existing starting point.

Programmable logic controller

Programmable logic controllers (PLC) are electronic controllers whose functions are stored as a program in the control unit. The design and wiring of the unit are not, therefore, dependent on the control functions. Programmable logic controllers have the same structure as a computer, i.e. they consist of a CPU with memory, input/output modules and an internal bus system. The I/Os and programming language are selected according to the requirements of the control technology involved.

Programmable working area limitation

Limitation of the movement area of the tool to within defined, programmable limits.

Programming key
Characters and character sequences which have a defined meaning in the programming language for –> part programs (see Programming Guide).

Protection zone
Three–dimensional area within a –> working area which the tool tip is not permitted to enter (programmable via MD).

Q

Quadrant error compensation
Contour errors on quadrant transitions caused by frictional fluctuations on guideways can be largely eliminated by means of quadrant error compensation. A circularity test is performed to parameterize the quadrant error compensation function.

R

R parameter
Calculation parameter. The programmer can assign or request the values of the R parameter in the –> part program as required.

Rail
This rail is used to mount the modules of the S7–300 system.

Rapid traverse
The highest traversing speed of an axis used, for example, to bring the tool from an idle position to the –> workpiece contour or retract it from the workpiece contour.

Reference point
Point on the machine tool with which the measuring system of the –> machine axes is referenced.

Reference point approach
If the position measuring system used is not an absolute–value encoder, then a reference point approach operation is required to ensure that the actual values supplied by the measuring system are in accordance with the machine coordinate values.

REPOS

3. Reapproach contour, triggered by operator
   REPOS allows the tool to be returned to the interrupt position by means of the direction keys.

4. Programmed contour reapproach
   A selection of approach strategies are available in the form of program commands: Approach point of interruption, approach start of block, approach end of block, approach a point on the path between start of block and interruption.

Revolutional feedrate

The axis feedrate is adjusted as a function of the speed of the master spindle in the channel (programmed with G95).

Rigid tapping

This function is used to tap holes without the use of a compensating chuck. The spindle is controlled as an interpolative rotary axis and drill axis, with the result that threads are tapped precisely to the final drilling depth, for example, in blind tapped holes (precondition: Spindle axis mode).

Rotary axis

Rotary axes cause the tool or workpiece to rotate to a specified angle position.

Rotary axis, continuously turning

The range of motion of a rotary axis can be set to a modulo value (in machine data) or defined as continuous in both directions, depending on the application. Continuously turning rotary axes are used, for example, for eccentric machining, grinding and winding.

Rotation

Component of a –> frame which defines a rotation of the coordinate system through a specific angle.

Rounding axis

Rounding axes cause the workpiece or tool to rotate to an angle position described on a graduated grid. When the grid position has been reached, the axis is ”in position”.

S

S7 Configuration

S7 Configuration is a tool for parameterizing modules. S7 Configuration is used to set a variety of
–> parameter blocks of the –> CPU and the I/O modules on the

–> programming device. These parameters are uploaded to the CPU.

S7–300 bus

The S7–300 bus is a serial data bus which supplies modules with the appropriate voltage and via which they exchange data with one another. The connection between the modules is made by means of –> bus connectors.

Safety functions

The control includes continuously active monitoring functions which detect faults in the –> CNC, the programmable controller (–> PLC) and the machine so early that damage to the workpiece, tool or machine rarely occurs. In the event of a fault, the machining operation is interrupted and the drives stopped. The cause of the malfunction is logged and an alarm issued. At the same time, the PLC is notified that a CNC alarm is pending.

Safety Integrated

Effective personnel and machine protection integrated in the control in conformance with EC Directive >>89/392/EEC<< in >>Safety Category 3<< to EN–954–1 (Categories B. 1–4 are defined in this standard) for safe setup and testing.

Discrete fail–safety is assured. If an individual fault occurs, the safety function is still effective.

Scaling

Component of a –> frame which causes axis–specific scale alterations.

B

Serial interface V.24

For the purpose of data input and output, the

- MMC module MMC 100 has a serial V.24 interface (RS–232) and the
- MMC modules MMC 101 and MMC 102 have two V.24 interfaces.

Machining programs and manufacturer and user data can be imported and exported via these interfaces.

Services

Control operating area

Setting data

Data which provide the control with information about properties of the machine tool in a way defined by the system software.

Unlike –> machine data, setting data can be modified by the user.

Softkey

A key whose name appears on an area of the screen. The choice of softkeys displayed is adapted dynamically to the operating situation. The freely assignable function keys (softkeys) are assigned to functions defined in the software.

Software limit switches

Software limit switches define the limits of the travel range of an axis and prevent the slide contacting the hardware limit switches. Two pairs of values can be assigned per axis and activated separately via the –> PLC.

Spindles

The spindle functionality is a two–level construct:

3. Spindles: Speed–controlled or position–controlled spindle drives, analog
   $\pm$10V (SINUMERIK FM–NC)
   digital (SINUMERIK 840D)
4. Auxiliary spindles: Speed–controlled spindle drives without actual position sensor, e.g. for power tools. ”Auxiliary spindle” function package, e.g. for power tools.

Spline interpolation

Using the spline interpolation function, the control is able to generate a smooth curve from just a small number of specified interpolation points along a setpoint contour.

Standard cycles

Standard cycles are used to program machining operations which repeat frequently:

- For drilling/milling
- For measuring tools and workpieces
- For turning (SINUMERIK FM–NC)

The available cycles are listed in menu ”Cycle support” in the ”Program” operating area. Once the desired machining cycle has been selected, the parameters required for assigning values are displayed in plaintext.

Subblock

Block prefixed by ”N” containing information for a machining step such as a position parameter.

Subroutine

A sequence of instructions of a –> part program which can be called repeatedly with different initial parameters. A subroutine is called from within a main program. Every subroutine can be locked against unauthorized export and viewing (with MMC 102/103). –> Cycles are a type of subroutine.

Synchronization

Instructions in –> part programs for coordination of the operations in different –> channels at specific machining points.

Synchronized actions

3. Auxiliary function output
   While a workpiece is being machined, technological functions (–> auxiliary functions) can be output from the CNC program to the PLC. These auxiliary functions control, for example, ancillary equipment on the machine tool such as the sleeve, gripper, chuck, etc.

4. High–speed auxiliary function output
   The acknowledgement times for the –> auxiliary functions can be minimized and unnecessary halts in the machining process avoided for time–critical switching functions.

Synchronized actions can be combined to form programs (technology cycles). Axis programs can be started in the same IPO cycle, for example, by scanning digital inputs.

Synchronized axes

Synchronized axes require the same amount of time to traverse their path as –> geometry axes for their path.

Synchronous spindle

Accurate angular synchronism between one master spindle and one or more slave spindles. Enables flying transfer of a workpiece from spindle 1 to spindle 2 on turning machines.

In addition to speed synchronism, it is also possible to program the relative angular positions of the spindles, e.g. on–the–fly, position–oriented transfer of inclined workpieces.

Several pairs of synchronous spindles can be implemented.

System variable A variable which exists although it has not been programmed by the –> part program programmer. It is defined by the data type and the variable name, which is prefixed with **$**. See also –> User–defined variable.

T

Teach In **Teach In** is a means of creating or correcting part programs. The individual program blocks can be input via the keyboard and executed immediately. Positions approached via the direction keys or handwheel can also be stored. Additional information such as G functions, feedrates or M functions can be entered in the same block.

Text editor –> Editor

Tool A tool employed to shape the workpiece, for example, a turning tool, milling cutter, drill, laser beam, grinding wheel, etc.

Tool A tool for machining workpieces (e.g. drill, cutter, etc.).

Tool nose radius compensation A contour is programmed on the assumption that a pointed tool will be used. Since this is not always the case in practice, the curvature radius of the tool being used is specified so that the control can make allowance for it. The curvature centre point is guided equidistantly to the contour at an offset corresponding to the curvature radius.

Tool offset A tool is selected by programming a **T function** (5 decades, integer) in the block. Up to nine tool edges (D addresses) can be assigned to each T number. The number of tools to be managed in the control is set in parameterization.

Tool length compensation is selected by programming D numbers.

Tool radius compensation In order to program a desired –> workpiece contour directly, the control must traverse a path equidistant to the programmed contour, taking into account the radius of the tool used (G41/G42). (G41/G42).

Transformation

Programming in a Cartesian coordinate system, execution in a non–Cartesian coordinate system (e.g. with machine axes as rotary axes).

Employed in conjunction with Transmit, Inclined Axis, 5–Axis Transformation.

Transmit

This function is used to mill the outside contours on turned parts, e.g. four–sided parts (linear axis with rotary axis).

3D interpolation with two linear axes and one rotary axis is also possible.

The benefits afforded by Transmit are simplified programming and improved machine efficiency through complete machining: Turning and milling on the same machine without reclamping.

Travel to fixed stop

This function allows axes (tailstocks, sleeves) to be traversed to a fixed stop position in order, for example, to clamp workpieces. The contact pressure can be defined in the part program.

Traversing range

The maximum permissible travel range for linear axes is ± 9 decades. The absolute value depends on the selected input and position control resolution and the unit of measurement (inch or metric).

U

User–defined variable

Users can define variables in the –> part program or data block (global user data) for their own use. A definition contains a data type specification and the variable name. See also –> system variable.

User memory

All programs and data such as part programs, subroutines, comments, tool offsets, zero offsets/frames and channel and program user data can be stored in the common CNC user memory.

User program

–> Part program

V

Variable definition

A variable is defined through the specification of a data type and a variable name. The variable name can be used to address the value of the variable.

Velocity control	In order to achieve an acceptable travel velocity in movements which call for very small adjustments of position in a block, the control can –> look ahead.
Vocabulary words	Words with a specific notation which have a defined meaning in the programming language for –> part programs.

W

Working memory	The working storage is a Random Access Memory in the –> CPU which the processor accesses as it executes the application program.
Working space	Three–dimensional zone into which the tool tip can be moved on account of the physical design of the machine tool. See also –> protection zone.
Workpiece	Part to be produced/machined by the machine tool.
Workpiece contour	Setpoint contour of the –> workpiece to be produced/machined.
Workpiece coordinate system	The origin of the workpiece coordinate system is the –>workpiece zero. In machining operations programmed in the workpiece coordinate system, the dimensions and directions refer to this system.
Workpiece zero	The workpiece zero is the origin for the –> workpiece coordinate system. It is defined by its distance from the machine zero.

X

Y

Z

Zero offset

Specification of a new reference point for a coordinate system through reference to an existing zero and a –> frame.

4. Settable
   SINUMERIK FM–NC: Four independent zero offsets can be selected per CNC axis.
   SINUMERIK 840D: A parameterizable number of settable zero offsets is available for each CNC axis. Each of the zero offsets can be selected by G functions and selection is exclusive.
5. External
   All offsets which define the position of the workpiece zero can be overlaid with an external zero offset
   – defined by handwheel (DRF offset) or
   – defined by the PLC.
6. Programmable
   Zero offsets can be programmed for all path and positioning axes by means of the TRANS instruction.

G Code Table

C

Appendix 1 describes the G code and the functions.

C

C.1 G code table

Table C-1 G code table

G code		Description
Group 1		
G00 1)	1	Rapid traverse
G01	2	Linear motion
G02	3	Circle/helix, clockwise
G03	4	Circle/helix, counterclockwise
G33	5	Thread cutting with constant lead
G34	9	Thread cutting with variable lead
G77	6	Longitudinal turning cycle
G78	7	Thread cutting cycle
G79	8	Face turning cycle
Group 2		
G96	1	Constant cutting rate ON
G97 1)	2	Constant cutting rate OFF
Group 3		
G90 1)	1	Absolute programming
G91	2	Incremental programming
Group 4		
Group 5		
G94	1	Feed in [mm/min, inch/min]
G95 1)	2	Feed in [mm/rev, inch/rev]
Group 6		
G20 1)	1	Input system inch
G21	2	Input system metric
Group 7		
G40 1)	1	Deselect cutter radius compensation
G41	2	Compensation to left of contour
G42	3	Compensation to right of contour
Group 8		
Group 9		
G22	1	Working area limitation, protection zone 3 ON
G23	2	Working area limitation, protection zone 3 OFF

C

Table C-1 G code table

G code		Description
Group 10		
G80 1)	1	Drilling cycle off
G83	2	Face deep hole drilling
G84	3	Face tapping
G86	4	Face drilling
G87	5	Side deep hole drilling
G88	6	Side tapping
G89	7	Side drilling
Group 11		
G98 1)	1	Return to starting point for drilling cycles
G99	2	Return to point R for drilling cycles
Group 12		
G66	1	Modal macro call
G67 1)	2	Delete modal macro call
Group 13		
Group 14		
G54 1)	1	Select zero offset
G55	2	Select zero offset
G56	3	Select zero offset
G57	4	Select zero offset
G58	5	Select zero offset
G59	6	Select zero offset
Group 15		
Group 16		
G17	1	XY plane
G18 1)	2	ZX plane
G19	3	YZ plane
Group 17		
Group 18 (non–modal)		
G04	1	Dwell
G05	20	High–speed cycle cutting
G07.1	18	Cylindrical interpolation
G10	2	Write zero offset / tool offset

C

Table C-1 G code table

G code		Description
G10.6	19	Rapid lift ON/OFF
G27	16	Referencing check (available soon)
G28	3	Approach 1st reference point
G30	4	Approach 2nd/3rd/4th reference point
G30.1	21	Floating reference position
G31	5	Measurement with touch–trigger probe
G52	6	Additive zero offset
G53	17	Approach position in machine coordinate system
G65	7	Call macro
G70	8	Finishing cycle
G71	9	Stock removal cycle longitudinal axis
G72	10	Stock removal cycle transverse axis
G73	11	Repeat contour
G74	12	Deep hole drilling and recessing in longitudinal axis (Z)
G75	13	Deep hole drilling and recessing in facing axis (X)
G76	14	Multiple thread cutting cycle
G92	15	Preset actual value memory, spindle speed limitation
Group 20		
G50.2	1	Synchronous spindle OFF
G51.2	2	Synchronous spindle ON
Group 21		
G13.1	1	TRANSMIT OFF
G12.1	2	TRANSMIT ON
Group 22		
Group 25		
Group 31		
G290 1)	1	Select Siemens mode
G291	2	select ISO dialect mode

Note: The NC establishes the G code modes, identified by 1), when the power is turned ON or when the NC is reset.

C

Machine and Setting Data

D

D.1 Machine/Setting Data

10604	**WALIM_GEOAX_CHANGE_MODE**				
MD number	Work area limitation when switching geometrical axes				
Default setting: 0		Minimum input limit: 0		Maximum input limit: 1	
Changes effective after Power On			Protection level: 2/7		Unit: –
Data type: BYTE				Applies with effect from SW version: 6.2	
Meaning:	Retain or deactivate work area limitation when switching geometrical axes. The MD is bit–coded and has the following meaning: Bit = =0: Deactivate work area limitation when switching geometrical axes =1: Retain work area limitation when switching geometrical axes				

10615	**NCFRAME_POWERON_MASK**				
MD number	Delete global base frames on Power On				
Default setting: 0		Minimum input limit: 0		Maximum input limit: 0	
Changes effective after Power On			Protection level: 2/7		Unit: –
Data type: DWORD				Applies with effect from SW version: 5.2	
Meaning:	This machine data defines whether global base frames are deleted on a Power On reset. The selection can be made separately for the individual base frames. Bit 0 corresponds to base frame 0, bit 1 to base frame 1, etc. 0: Base frame is retained on Power On 1: Base frame is deleted on Power On.				

D

10652	**CONTOUR_DEF_ANGLE_NAME**	
MD number	Definable name for angle in the contour short description	
Default setting: ”ANG”	Minimum input limit: –	Maximum input limit: [illegible]
Changes effective after Power On	Protection level: 2/7	Unitt: –
Datentype: STRING	Applies with effect from SW version: 5	
Meaning:	The setting is effective for Siemens G code programming only, i.e. G290. The name used to program the angle in the contour short description is definable. This allows, for example, identical programming in different language modes: If the angle is named “A”, it is programmed in the same way with Siemens and ISO Dia–lect0. The name must be unique, i.e. axes, variables, macros, etc. must not exist with the same name.	

10654	**RADIUS_NAME**	
MD number	Definable name for radius non–modally in the contour short description	
Default setting: “RND“	Minimum input limit: –	Maximum input limit: –
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: STRING	Applies with effect from SW version: 5	
Meaning:	The name used to program the radius in the contour short description is definable. This allows, for example, identical programming in different language modes: If the radius is named “R”, it is programmed in the same way with Siemens and ISO Dia–lect0. The name must be unique, i.e. axes, variables, macros, etc. must not exist with the same name. The setting is effective for Siemens G code programming, i.e. G290.	

10656	**CHAMFER_NAME**	
MD number	Definable name for chamfer in the contour short description	
Default setting: ”CHR”	Minimum input limit: –	Maximum input limit: –
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: STRING	Applies with effect from SW version: 5	
Meaning:	The name used to program the chamfer in the contour short description is definable. This allows, for example, identical programming in different language modes: If the chamfer is named “C”, it is programmed in the same way with Siemens and ISO Dia–lect0. The name must be unique, i.e. axes, variables, macros, etc. must not exist with the same name. The setting is effective for Siemens G code programming, i.e. G290. The chamfer in the original direction of movement. Alternatively, the chamfer length can be programmed with the name CHF.	

D

10715	**M_NO_FCT_CYCLE[0]**				
MD number	M function number for cycle call				
Default setting: –1		Minimum input limit: –1		Maximum input limit: –	
Changes effective after Power On			Protection level: 2/7		Unit: –
Data type: DWORD				Applies with effect from SW version: 5.2	
Meaning:	M number with which a subprogram is called. The name of the subprogram is stored in $MN_M_NO_FCT_CYCLE_NAME. If the M function defined by $MN_M_NO_FCT_CYCLE is programmed in a part program, the subprogram defined in M_NO_FCT_CYCLE_NAME is started at the end of the block. If the M function is programmed again in the subprogram, the substitution no longer takes place by means of a subprogram call. $MN_M_NO_FCT_CYCLE is effective both in Siemens mode G290 and in external language mode G291. A subprogram call may not be superimposed on M functions with fixed meanings. In the event of a conflict, alarm 4150 is output: – M0 to M5, – M17, M30, – M40 to M45, – M function for spindle/axis mode switchover according to $MC_SPIND_RIGID_TAPPING_M_NR (default M70) – M functions for nibbling/punching according to configuration via $MC_NIBBLE_PUNCH_CODE if activated via $MC_PUNCHNIB_ACTIVATION. – With applied external language ($MN_MM_EXTERN_LANGUAGE) M19, M96–M99. Exception: The M functions defined for the tool change with $MC_TOOL_CHANGE_M_CODE. $MN_M_NO_FCT_CYCLE_NAME and $MN_T_NO_FCT_CYCLE_NAME may not be active in the same block (part program line), i.e. only one M/T function substitution can be active per block. Neither an M98 call nor a modal subprogram call can be programmed in the block with the M function substitution. A subprogram return jump or end of part program is not allowed. Alarm 14016 is output in the event of a conflict.					

D

10716	M_NO_FCT_CYCLE_NAME[0]				
MD number	Name of tool–changing cycle for M functions from MD $MN_MFCT_CYCLE				
Default setting: –		Minimum input limit: –		Maximum input limit: –	
Changes effective after Power On			Protection level: 2/7		Unit: –
Data type: STRING				Applies with effect from SW version: 5.2	
Meaning:	The name of the cycle is stored in the machine data. This cycle is called when the M function from machine data $MN_M_NO_FCT_CYCLE is programmed. If the M function is programmed in a motion block, the cycle is executed after the move–ment. $MN_M_NO_FCT_CYCLE is effective both in Siemens mode G290 and in external language mode G291. If a T number is programmed in the calling block, the programmed T number can be scanned in the cycle in variable $P_TOOL. $MN_M_NO_FCT_CYCLE_NAME and $MN_T_NO_FCT_CYCLE_NAME may not be active in the same block, i.e. only one M/T function substitution can be active per block. Neither an M98 call nor a modal subprogram call can be programmed in the block with the T function substitution. A subprogram return jump or end of part program is not allowed. Alarm 14016 is output in the event of a conflict.				

10717	T_NO_FCT_CYCLE_NAME				
MD number	Name for tool–changing cycle with T number				
Default setting: –		Minimum input limit: –		Maximum input limit: –	
Changes effective after Power On			Protection level: 2/7		Unit: –
Data type: STRING				Applies with effect from SW version: 5.2	
Meaning:	If a T function is programmed in a part program block, the subprogram defined in T_NO_FCT_CYCLE_NAME is called at the end of the block. System variable $C_T / $C_T_PROG can be used in the cycle to scan the programmed T no. as a decimal value, and $C_TS / $C_TS_PROG as a string (only with tool management). If a T number is programmed with the D number, it can be scanned in the cycle in system variable $C_D/$C_D_PROG. System variable $C_T_PROG or $C_D_PROG can be used in the subprogram to check whether the T or D command was programmed. The values can be read out with system variable $C_T or $C_D. If another T command is programmed in the subprogram, no sub–stitution takes place, but the T word is output to the PLC. $MN_T_NO_FCT_CYCLE_NAME and system variables $C_T / $C_TS_PROG are effec–tive both in Siemens mode G290 and in external language mode G291. $MN_M_NO_FCT_CYCLE_NAME and $MN_T_NO_FCT_CYCLE_NAME may not be active in the same block i.e. only one M/T function substitution can be active per block. Neither an M98 call nor a modal subprogram call can be programmed in the block with the T function substitution. A subprogram return jump or end of part program is not allowed. Alarm 14016 is output in the event of a conflict.				

D

10760	G53_TOOLCORR	
MD number	Mode of action when G53, G153 and SUPA is specified	
Default setting: 2	Minimum input limit: 2	Maximum input limit: 4
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: BYTE	Applies with effect from SW version: 5.2	
Meaning:	The MD is effective in both Siemens mode and in external language mode. This machine data defines whether tool length compensation and tool radius compensation are suppressed with language commands G53, G153 and SUPA. 0 = G53/G153/SUPA is non–modal suppression of zero offsets, tool length compensation and tool radius compensation remain active. 1= G53/G153/SUPA is non–modal suppression of zero offsets, and active tool length and tool radius compensation.	

10800	EXTERN_CHAN_SYNC_M_NO_MIN	
MD number	First M code for channel synchronization	
Default setting: –1	Minimum input limit: 100	Maximum input limit:
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: DWORD	Applies with effect from SW version: 6.2	
Meaning:	Lowest number M code out of an M code number area which is reserved for channel synchronization.	

10802	EXTERN_CHAN_SYNC_M_NO_MAX	
MD number	Last M code for channel synchronization	
Default setting: –1	Minimum input limit: 100	Maximum input limit:
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: DWORD	Applies with effect from SW version: 6.2	
Meaning:	Highest number M code out of an M code number area which is reserved for channel sysnchronization. The number of M codes must not exceed a number of 10 times the number of channels (for example: 2 channels results in max. 20 M codes). Alarm 4170 is issued if an excessive M code area is specified.	

D

10804	EXTERN_M_NO_SET_INT	
MD number	ASUP activating M code	
Default setting: 96	Minimum input limit: 0	Maximum input limit:
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: DWORD	Applies with effect from SW version: 6.2	
Meaning:	M code to activate interruption type subprogram call in ISO dialect T/M mode (ASUP).	

10806 MD number	EXTERN_M_NO_DISABLE_INT ASUP deactivating M code	
Default setting: 97	Minimum input limit: 0	Maximum input limit:
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: DWORD	Applies with effect from SW version: 6.2	
Meaning:	M code to activate interruption type subprogram call in ISO dialect T/M mode (ASUP).	

10808 MD number	EXTERN_INTERRUPT_BITS_M96 Interrupt program – Execution (M96)	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 8
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: WORD	Applies with effect from SW version: 6.2	
Meaning:	Using the data bits described below, the behaviour of the interruption type subprogram activated by M96 P .. can be specified. Bit 0: =0, Disable interruption type subprogram; M96/M97 are treated as standard M codes =1, Enable activation/deactivation of interruption type subprogram using M96/M97 Bit 1: =0, Execution of the part program is continued at the target position of the NC block subsequent to the NC block where the interruption took place =1, Execution of the part program is continued at the interruption position Bit 2: =0, The current NC block is interrupted immediatly and the subprogram is called if the interrupt signal is detected. =1, The subprogram is called after completion of the current NC block Bit 3: =0, When detecting an interrupt signal during execution of a machining cycle, the machining cycle is interrupted. =1, Interrupt after machining cycle completion	

10810 MD number	EXTERN_MEAS_G31_P_SIGNAL Measuring signal input assignment for G31 P..	
Default setting: 1	Minimum input limit: 0	Maximum input limit: 3
Changes effective after Power On	Protection level: 2/7	Unit: –
Datentype: BYTE	Applies with effect from SW version: 6.2	
Meaning:	Measuring inputs 1 and 2 are assigned to the arguments P .. of G31 P1 to P4 command. It is a bit coded MD. Only bit 0 and bit 1 are evaluated. For example: $MN_EXTERN_MEAS_G31_P_SIGNAL[1], Bit 0=1, the 1st measuring input is activated by G31 P2. $MN_EXTERN_MEAS_G31_P_SIGNAL[3] = 2, the 2nd measuring input is activated by G31 P4. Bit 0: =0: Deactivate measuring input 1 for G31 P1 (–P4) =1 Activate measuring input 1 for G31 P1 (–P4) Bit 1: =0 Deactivate measuring input 2 for G31 P1 (–P4) =1 Activate measuring input 2 for G31 P1 (–P4)	

D

10880	**EXTERN_CNC_SYSTEM**	
MD number	External control system whose programs are executed	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: WORD	Applies with effect from SW version: 5	
Meaning:	Selection of the external language 1 = ISO–2: System Fanuc0 Milling (from 5.1) 2 = ISO–3: System Fanuc0 Turning (from 5.2) The functional scope defined in the current Siemens documentation is valid. This data is only evaluated if machine data $MN_MM_EXTERN_LANGUAGE is set.	

10881	**EXTERN_GCODE_SYSTEM**	
SD number	ISO mode T: G code system	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 2
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: DWORD	Applies with effect from SW version: 6.2	
Meaning:	This MD deternines the G code system used for ISO dialect T mode: setting value = 0: ISO_T: G code system B setting value = 1: ISO_T: G code system A setting value = 2: ISO_T: G code system C	

10882	**NC_USER_EXTERN_GCODES_TAB [n]:0...59**	
MD number	List of user–specific G codes of an external NC language	
Default setting: –	Minimum input limit: –	Maximum input limit: –
Changes effective after Power On	Protection level: 2/2	Unit: –
Datentype: STRING	Applies with effect from SW version: 5	
Meaning:	Code B is implemented by default for external programming language ISO Dialect0–T. Code A and Code C have different G function names. $MN_NC_USER_EXTERN_GCODES_TAB can be used to rename the G functions. The G command codes can be changed for external NC languages. The G group and the position within the G group remain the same. Only the G command codes can be changed. Up to 30 code changes are possible. Example: $MN_NC_USER_EXTERN_GCODES_TAB[0]="G20" $MN_NC_USER_EXTERN_GCODES_TAB[1]="G70" —> G20 is reassigned to G70; If G70 already exists, an error message appears on NCK reset.	

D

10884 MD number	**EXTERN_FLOATINGPOINT_PROG** Valuation of programmed values not containing a decimal point	
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: BOOLEAN	Applies with effect from SW version: 5.2	
Meaning:	This machine data is effective for external programming languages, i.e. if MD 18800: MM_EXTERN_LANGUAGE = 1. The machine data defines how programmed values without decimal points are evaluated. 0: Standard Notation: Values without decimal points are interpreted in internal units IS–B, IS–C (see MD EXTERN_INCREMENT_SYSTEM). Values without decimal points are interpreted in internal units e. g. X1000 = 1 mm (with 0.001 mm input resolution) X1000.0 = 1000 mm 1: Pocket Calculator Notation: Values without decimal points are interpreted as mm, inch or degrees. Values without decimal points are interpreted as mm, inch or degrees e.g. X1000 = 1000 mm X1000.0 = 1000 mm	

10886 MD number	**EXTERN_INCREMENT_SYSTEM** Increment system	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1
Changes effective after POWER ON	Protection level: 2/7	Unit: –
Data type: BOOLEAN	Applies with effect from SW version: 5.2	
Meaning:	This machine data is effective for external programming languages, i.e. if MD 18800: MM_EXTERN_LANGUAGE = 1. This machine data defines which increment system is active 0: Increment system IS–B= 0.001 mm/degree = 0.0001 inch 1: Increment system IS–C = 0.0001 mm/degree = 0.00001 inch	

10888 MD number	**EXTERN_DIGITS_TOOL_NO** Number of digits for T number in external language mode	
Default setting: 2	Minimum input limit: 2	Maximum input limit: 4
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: BYTE	Applies with effect from SW version: 5.2	
Meaning:	The machine data is only effective with $MN_EXTERN_CNC_SYSTEM = 2. Number of digits for tool number in programmed T value. The number of leading digits specified in $MN_EXTERN_DIGITS_TOOL_NO is interpreted as the tool number from the programmed T value. The trailing digits address the compensation memory.	

10890	EXTERN_TOOLPROG_MODE	
MD number	Tool change programming with external programming language	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1
Changes effective after Power On	Protection level: 2/7	Unit: –
Data type: BYTE	Applies with effect from SW version: 5.2	
Meaning:	Configuration of tool change programming for external programming language: Bit0 = 0: Effective only with $MN_MM_EXTERN_CNC_LANGUAGE =2: The tool number and offset number are programmed in the T value. $MN_DIGITS_TOOLNO determines the number of leading digits representing the tool number. Example: $MN_DIGITS_TOOL_NO = 2 T=1234 ; tool no. 12, ; offset no. 34 Bit0 = 1: Effective only with $MN_MM_EXTERN_CNC_LANGUAGE =2: Only the tool number is programmed in the T value. Offset number = tool number. $MN_DIGITS_TOOL_NO is irrelevant. Example: T=12 ; tool no. 12 ; offset no. 12 Bit1 = 0: Effective only with $MN_MM_EXTERN_CNC_LANGUAGE =2: If the number of digits programmed in the T value is equal to the number in $MN_EXTERN_DIGITS_TOOL_NO, leading zeroes are added. Bit1 = 1: Effective only with $MN_MM_EXTERN_CNC_LANGUAGE =2: If the number of digits programmed in the T value is equal to the number of digits specified in $MN_EXTERN_DIGITS_TOOL_NO, the programmed number is used as the offset number and the tool number	

18800	MM_EXTERN_LANGUAGE	
MD number	External language active in the control	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1
Changes effective after Power On	Protection level: 2/7	Unit: –
Datentype: DWORD	Applies with effect from SW version: 5	
Meaning:	This MD must be set to enable ISO Dialect0–T and ISO Dialect0–M programs to run on the control. Only one external language can be selected at a time. Please refer to the latest documentation for the available command range. Bit 0 (LSB): Execution of part programs in ISO_2 or ISO_3 mode. For coding see $MN_MM_EXTERN_CNC_SYSTEM (10880)	

D

D.2 Channel-specific machine data

20154 MD number	**EXTERN_GCODE_RESET_VALUES[n]: 0, ..., 30** Defines the G codes which are activated on startup if the NC channel is not running in Siemens mode.		
Default setting: –	Minimum input limit: –	Maximum input –	
Changes effective after Power On	Protection level: 2/2	Unit: –	
Data type: BYTE	Applies with effect from SW version: 5		
Meaning:	The following external programming languages are possible: – ISO dialect milling – ISO dialect turning The G group classification to be used is specified in the current SINUMERIK documentation. The following groups can be defined within MD EXTERN_GCODE_RESET_VALUES: **ISO dialect M**: G code group 2: G17/G18/G19 G code group 3: G90/G91 G code group 5: G94/G95 G code group 6: G20/G21 G code group 13: G96/G97 G code group 14: G54–G59 **ISO dialect T**: G code group 2: G96/G97 G code group 3: G90/G91 G code group 5: G94/G95 G code group 6: G20/G21 G code group 16: G17/G18/G19		

D

20380	TOOL_CORR_MODE_G43/G44	
MD number	Processing of prog. length offsets G43/G44	
Default setting: 0	Minimum input limit: 1	Maximum input limit: 2
Changes effective after RESET	Protection level: 2/7	Unit: –
Data type: BYTE	Applies with effect from SW version: 5.2	
Meaning:	The machine data is only effective if MD EXTERN_CNC_LANGUAGE = 1. When G43/G44 is active, it determines how length offsets programmed with H are pro–cessed. 0: mode A The tool length H always acts on the Z axis, independent of the current plane. 1: mode B The tool length H acts on one of the three geometry axes depending on the active plane: G17 on the 3rd geometry axis (usually Z) G18 on the 2nd geometry axis (usually Y) G19 on the 1st geometry axis (usually X) By multiple programming, length offsets can be established in all three geometrical axes in this mode, i.e. by activating an offset, the existing length offset of another axis will not be cancelled. 2: mode C The tool length offset becomes valid in the axis programmed together with the H code regardless of the selected plane. Further, the behaviour is as discribed under mode B.	

20382	TOOL_CORR_MOVE_MODE	
MD number	Traversing the tool length offset	
Default setting: FALSE	Minimum input limit: –	Maximum input limit: –
Changes effective after RESET	Protection level: 2/7	Unit: –
Data type: BOOLEAN	Applies with effect from SW version: 5.2	
Meaning:	The machine data determines how the tool length offsets are applied. FALSE: A tool length offset is only applied if the associated axis was programmed. (Same behaviour as in previous SW versions) TRUE: Tool length offsets are always applied, regardless of whether the associated axes were programmed.	

D

20732	EXTERN_G0_LINEAR_MODE	
MD number	Rapid traverse interpolation selection	
Default setting: 1	Minimum input limit: 0	Maximum input limit: 1
Changes effective after POWER ON	Protection level: 2/4	Unit: –
Data type: BOOLEAN	Applies with effect from SW version:	
Meaning:	This MD determines G00 interpolation behaviour. 0: axes move like positioning axes 1: linear interpolation	

20734	**EXTERN_FUNCTION_MASK**	
MD number	External language function mask	
Default setting:	Minimum input limit: 0	Maximum input limit: 16
Changes effective after RESET	Protection level: 2/7	Unit: –
Data type: DWORD	Applies with effect from SW version: 6.2	
Meaning:	This MD affects functions included in the ISO mode. Bit 0 =0: ISO mode T: "A" and "C" are interpreted as axis designation. When programming a contour, "A" or "C" must be preceded by a comma. =1: "A" and "C" within a part program are always interpreted as contour definition. An A– or C axis must not exist. Bit 1 =0: ISO mode M G10 P<100 tool geometry >100 tool wear =1: G10 P<10 000 tool geometry >10 000 tool wear Bit 2 =0: G04 dwell always [s] or [ms] =1: if G95 active, dwell in rpm Bit 3 =0 errors in the ISO scanner cause an alarm =1: errors in the ISO scanner are not output; the block is passed through to the Siemens translator. Bit 4 =0: G00 is executed according to the exact stop function currently active. Example: In G64 mode, G00 blocks are also moved in G64 =1 G00 blocks are always executed in G09, even if G64 is active.	

22420	**FGROUP_DEFAULT_AXIS[n]: 0, ..., 7**	
MD number	Default value for FGROUP command	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 8
Changes effective after Power On	Protection level: 7/7	Unit: –
Data type: BYTE	Applies with effect from SW version: 5.2	
Meaning:	You can specify up to 8 channel axes whose resulting velocity corresponds to the pro–grammed path feed. If all 8 values are set to zero (default), the geometry axes entered in $MC_AXCONF_GEOAX_ASSIGN_TAB are activated as the default setting for the FGROUP command. Example: The first 4 axes in the channel are relevant for the path feed: $MC_FGROUP_DEFAULT_AXES[0] = 1 $MC_FGROUP_DEFAULT_AXES[2] = 2 $MC_FGROUP_DEFAULT_AXES[3] = 3 $MC_FGROUP_DEFAULT_AXES[4] = 4	

D

22512	**EXTERN_GCODE_GROUPS_TO_PLC[n]: 0, ..., 7**			
MD number	Specifies the G groups which are output to the NCK/PLC interface when an external NC language is active			
Default setting: –		Minimum input limit: –		Maximum input limit: –
Changes effective after POWER ON		Protection level: 2/7		Unit: –
Data type: BYTE			Applies with effect from SW version: 5	
Meaning:	The user can select the G groups of an external NC language with channel MD $MC_EX–TERN_ GCODE_GROUPS_TO_PLC . The active G command is then signaled from the NCK to the PLC for these groups. Default 0: No output The NCK/PLC interface is updated on every block change and after a Reset. It cannot always be assured that a block–synchronous relationship exists between the NC block and the signaled G functions (e.g. if very short blocks are used in continuous–path mode). The same applies to $MC_GCODE_GROUPS_TO_PLC			

22900	**STROKE_CHECK_INSIDE**			
MD number	Determine enternal/external protection zone			
Default setting: 0		Minimum input limit: 0		Maximum input limit: 1
Changes effective after POWER ON		Protection level: 2/7		Unit: –
Data type: BYTE			Applies with effect from SW version: 5.2	
Meaning:	This machine data applies in combination with external programming languages. It is effective with $MN_MM_EXTERN_LANGUAGE = 1. It defines whether protection zone 3 is an internal or external protection zone. Meaning: 0: Protection zone 3 is an internal protection zone 1: Protection zone 3 is an external protection zone			

22910	**WEIGHTING_FACTOR_FOR_SCALE**			
MD number	Input unit for scaling factor			
Default setting: 0		Minimum input limit: 0		Maximum input limit: 1
Changes effective after POWER ON		Protection level: 2/7		Unit: –
Data type: BOOLEAN			Applies with effect from SW version: 5.2	
Meaning:	This machine data applies in combination with external programming languages. It is active with $MN_MM_EXTERN_LANGUAGE = 1. It defines the unit for the scale factor P and the axial scale factors I, J, K Meaning: 0: Scale factor in 0.001 1: Scale factor in 0.00001			

D

22914	**AXES_SCALE_ENABLE**		
MD number	Enable axial scaling (G51)		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 1	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Datentype: BOOLEAN	gültig ab SW–Stand: 5.2		
Meaning:	This MD enables axial scaling. Meaning: 0: Axial scaling not possible 1: Axial scaling possible, (MD DEFAULT_SCALE_FACTOR_AXIS becomes effective)		

22920	**EXTERN_FEEDRATE_F1_F9_ACTIV**		
SD number	Feste Vorschübe mit F0 – F9 erlauben		
Default setting: FALSE	Minimum input limit:	Maximum input limit:	
Changes effective after POWER ON	Protection level: 2/7	Unit:	
Datentype: BOOLEAN	Applies with effect from SW version: 6.2		
Meaning:	FALSE: feedrate values stored in setting data $SC_FEEDRATE_F1_9[ ] cannot be activated by F1 – F9. TRUE When programming F1 – F9, the feedrate values stored in setting data $SC_FEEDRATE_F1_9[ ] become effective. F0 activates rapid traverse.		

22930	**EXTERN_PARALLEL_GEOAX**		
SD number	Assignment of parallel channel geometry axis		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 3	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies with effect from SW version: 6.2		
Meaning:	Assignment of axes parallel to the geometrical axes. Using this table, parallel channel axes can be assigned to geometrical axes. Within the ISO dialect mode, the parallel axes can then be activated as geometrical axes by commanding a G code for plane selection (G17 – G19) together with the axis designation of the relevant parallel axis. Axis interchange is then carried out with the axis defined in $MC_AXCONF_GEOAX_ASSIGN_TAB[ ]. Prerequisite: The channel axes in use must be active. Entering a zero deactivates the relevant parallel geometrical axis.		

D

24004	**CHBFRAME_POWERON_MASK**		
MD number	Delete channel–specific base frame on Power On		
Default setting: 0	Minimum input limit: 0	Maximum input limit: 0xFF	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: DWORD	Applies with effect from SW version: 5.2		
Meaning:	This machine data defines whether channel–specific base frames are deleted on a Power On reset, i.e. work shifts and rotations are reset to 0, scaling is set to 1. Mirroring is switched off. The selection can be made separately for the individual base frames. Bit 0 corresponds to base frame 0, bit 1 to base frame 1, etc. 0: Base frame is retained on Power On 1: Base frame is deleted on Power On.		

D.3 Axis-specific setting data

43120 MD number	**DEFAULT_SCALE_FACTOR_AXIS** Default axial scale factor for G51 active			
Default setting: 1		Minimum input limit: –99999999		Maximum input limit: 99999999
Changes effective IMMEDIATELY			Protection level: 7/7	Unit: –
Data type: DWORD			Applies with effect from SW version: 5.2	
Meaning:	This machine data applies in combination with external programming languages. It is effective with $MN_MM_EXTERN_LANGUAGE = 1. If no axial scale factor I, J or K is programmed in the G51 block, DEFAULT_SCALEFACTOR_AXIS is effective. This MD is valid only if MD AXES_SCALE_ENABLE is set.			

43240 MD number	**M19_SPOS** Position of spindle (degree) when commanding M19			
Default setting: 0		Minimum input limit: –359.999		Maximum input limit: 359.999
Changes effective IMMEDIATELY			Protection level: 7/7	Unit: –
Data type: DOUBLE			Applies with effect from SW version: 5.2	
Meaning:	Das Settingdatum ist auch im Siemens–Mode wirksam.			

D.4 Channel-specific setting data

42110 SD number	**DEFAULT_FEED** Default value for path feed			
Default setting: 0		Minimum input limit: 0		Maximum input limit: –
Changes effective IMMEDIATELY			Protection level: 7/7	Unit: –
Data type: DOUBLE			Applies with effect from SW version: 5.2	
Meaning:	If no path feed is programmed in the part program, the value stored in $SC_DEFAULT_FEED is used. The setting data is evaluated at the start of the part program allowing for the feed type active at the time (see $MC_GCODE_RESET_VALUES and/or $MC_EXTERN_GCODE_RESET_VALUES).			

42140 SD number	**DEFAULT_SCALE_FACTOR_P** Default scale factor for address P			
Default setting: 0		Minimum input limit: –99999999		Maximum input limit: 99999999
Changes effective IMMEDIATELY			Protection level: 7/7	Unit: –

D

42140 SD number	DEFAULT_SCALE_FACTOR_P Default scale factor for address P
Data type: DWORD	Applies with effect from SW version: 5.2
Meaning:	This machine data applies in combination with external programming languages. It is effective with $MN_MM_EXTERN_LANGUAGE = 1. If no scale factor P is programmed in the block, the value in this machine data is applied.

42150 SD number	DEFAULT_ROT_FACTOR_R Default angle of rotation R	
Default setting: 0	Minimum input limit: 0	Maximum input limit: 360
Changes effective IMMEDIATELY	Protection level: 2/7	Unit: degree
Data type: DOUBLE	Applies with effect from SW version:	
Meaning:	When commanding coordinate rotation G68 without specifying an angle of rotation, the value set to this setting data is applied.	

42160 SD number	EXTERN_FIXED_FEEDRATE_F1_F9 F1 digit feed F0 – F9	
Default setting: 0	Minimum input limit: 0	Maximum input limit:
Changes effective IMMEDIATELY	Protection level: 2/7	Unit: mm/min
Data type: DOUBLE	Applies with effect from SW version:	
Meaning:	Pre–defined feedrates which are selected by commanding F0 – F9 when G01 is active.	

D

Data Fields, Lists

E

E.1 Machine data

Number	Identifier	Name	Reference
General ($MN_ ...)			
10604	WALIM_GEOAX_CHANGE_MODE	Work area limitation when switching geometrical axes	
10615	NCFRAME_POWERON_MASK	Delete global base frames on Power On	K2
10715	M_NO_FCT_CYCLE[n]: 0, ..., 0	M function number for tool change cycle call	
10716	M_NO_FCT_CYCLE_NAME[ ]	Name of tool–changing cycle for M functions-from MD $MN_MFCT_CYCLE	
10717	T_NO_FCT_CYCLE_NAME	Name for tool–changing cycle with T no.	
10760	G53_TOOLCORR	Behaviour of G53, G153 and SUPA	
10800	EXTERN_CHAN_SYNC_M_NO_MIN	First M code for channel synchronization	
10802	EXTERN_CHAN_SYNC_M_NO_MAX	Last M code for channel synchronization	
10804	EXTERN_M_NO_SET_INT	ASUP activation M code	
10806	EXTERN_M_NO_DISABLE_INT	ASUP deactivation M code	
10808	EXTERN_INTERRUPT_BITS_M96	Interrupt program execution (M96)	
10810	EXTERN_MEAS_G31_P_SIGNAL	Measuring input assignment for G31 P..	
10880	EXTERN_CNC_SYSTEM	External control system whose programs are to be executed	
10881	EXTERN_GCODE_SYSTEM	ISO mode T: G code system selection	
10882	NC_USER_EXTERN_GCODES_TAB[n]: 0–59	List of user defined G commands for external NC language	
10884	EXTERN_FLOATINGPOINT_PROG	Evaluation for progr. without decimal point	
10886	EXTERN_INCREMENT_SYSTEM	Defines the increment system	
10888	EXTERN_DIGITS_TOOL_NO	Number of digits for T number in external language mode	
10890	EXTERN_TOOLPROG_MODE	Tool change programming with external programming language	
18190	MM_NUM_PROTECT_AREA_NCK	Number of files for machine related protection zones (SRAM)	S7
18800	MM_EXTERN_LANGUAGE	External language active in the control	
Channel–specific ($MC_ ...)			
20050	AXCONF_GEOAX_ASSIGN_TAB[ ]	Assignment geometrical axis / channel axis	K2
20060	AXCONF_GEOAX_NAME_TAB[ ]	Geometrical axis in channel	K2

Channel–specific ($MC_ ...)			
20070	AXCONF_MACHAX_USED[ ]	Valid machine axis number in channel	K2
20080	AXCONF_CHANAX_NAME_TAB[ ]	Channel axis designation in channel	K2
20094	SPIND_RIGID_TAPPING_M_NR	M function number with which the spindle switches to controlled spindle mode	
20095	EXTERN_RIGID_TAPPING_M_NR	M function number in external language mode for spindle switchover to controlled spindle mode	
20100	DIAMETER_AX_DEF	Geometrical axis with cross axis functionality	P1
20150	GCODE_RESET_VALUES[n]: 0 bis max. Anzahl G–Codes	G code groups reset values	K1
20154	EXTERN_GCODE_RESET_VALUES[n]: 0–30	G code groups reset values	
20380	TOOL_CORR_MODE_G43G44	Behaviour of tool length compensation G43/G44	
20382	TOOL_CORR_MOVE_MODE	Traversing tool length compensation	
20732	EXTERN_G0_LINEAR_MODE	Determine traverse movement of G00	
20734	EXTERN_FUNCTION_MASK	External language function mask	
22420	FGROUP_DEFAULT_AXES[ ]	FGROUP command default value	
22512	EXTERN_GCODE_GROUPS_TO_PLC[n]: 0–7	Specifies the G groups which are output to the NCK/PLC interface when an external NC language is active	
22900	STROKE_CHECK_INSIDE	Protection zone direction (inside/outside)	
22910	WEIGHTING_FACTOR_FOR_SCALE	Unit of scale factor	
22914	AXES_SCALE_ENABLE	Enable axial scaling (G51)	
22920	EXTERN_FEEDRATE_F1_F9_ACTIV	Enable F 1–digit feed (F0 – F9)	
22930	EXTERN_PARALLEL_GEOAX	Assign parallel channel geometry axis	
24004	CHBFRAME_POWERON_MASK	Delete channel–specific base frame on Power On	
28080	NUM_USER_FRAMES	Number of zero offsets	
29210	NUM_PROTECT_AREA_ACTIVE	Activate protection zone	
34100	REFP_SET_POS[0]	Reference position / not used when absolute measuring system is applied	
35000	SPIND_ASSIGN_TO_MACHAX	assign spindle / machine axis	

E.2 Setting data

Number	Identifier	Name	Refer–ence
Axis–specific			
43120	$SC_DEFAULT_SCALE_FACTOR_AXIS	Default axial scale factor when G51 active	
43240	$SA_M19_SPOS	Position of spindle when programming M19	
42890	$SA_M19_SPOSMODE	Positioning mode of spindle when commanding M19	
Channel–specific			
42110	$SC_DEFAULT_FEED	Default value for path feed	V1
42140	$SC_DEFAULT_SCALE_FACTOR_P	Default scale factor for address P	
42150	$SC_DEFAULT_ROT_FACTOR_R	Default angle of rotation R	

Notes

E

Alarms

F

If error states are detected in cycles, an alarm is generated and cycle execution is interrupted.
The cycles continue to output messages in the dialog line of the control. These messages do not interrupt execution.
Alarms with numbers between 61000 and 62999 are generated in the cycles This number range is subdivided further according to alarm reactions and cancelation criteria.

Table F-1 Alarm number and alarm description

Alarm no.	Brief description	Source	Explanation/remedy
General alarms			
61001	Pitch of thread not correct	CYCLE376T	Pitch of thread is not specified correctly
61003	No feed programmed in cycle	CYCLE371T, CYCLE374T, CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	No feed F word was programmed in the calling block before the cycle call, see standard Siemens cycles
61004	Configuration of geometry axis not correct	CYCLE328	The order of the geometry axes is incorrect, see standard Siemens cycles
61101	Reference plane improperly de–fined	CYCLE375T, CYCLE81, CYCLE83, CYCLE84, CYCLE87	See standard Siemens cycles
61102	No spindle direction programmed	CYCLE371T, CYCLE374T, CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	Spindle direction M03 or M04 missing, see standard Siemens cycles
61107	First drilling depth incorrectly defined		First drilling depth counter to total drilling depth
61603	Grooving incorrectly defined	CYCLE374T	Grooving depth value 0
61607	Start point incorrect	CYCLE376T	The start point is not outside of the area to be machined
61610	No in–feed programmed	CYCLE374T	In–feed value = 0
ISO alarms			
61800	External CNC system missing	CYCLE300, CYCLE328, CYCLE330, CYCLE371T, CYCLE374T, CYCLE376T, CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	Machine data for external lan–guage MD18800: $MN_MM_EX–TERN_ LAN-GUAGE or option bit 19800 $ON_EXTERN_LAN–GUAGE is not set

F

Table F-1 Alarm number and alarm description, continued

Alarm no.	Brief description	Source	Explanation/remedy
61801	Incorrect G code selected	CYCLE300, CYCLE371T, CYCLE374T, CYCLE376T, CYCLE383T, CYCLE384T, CYCLE385T	An illegal numerical value for the CNC system was programmed in the program call CYCLE300<value> or in the cycle setting data an incorrect value for the G code system was specified.
61802	Incorrect axis type	CYCLE328, CYCLE330	The programmed axis is assigned to a spindle
61803	Programmed axis does not exist	CYCLE328, CYCLE330	The programmed axis does not exist in the system. Check MD20050–20080
61804	Programmed position beyond reference point	CYCLE328, CYCLE330	The programmed intermediate position or current position is located behind the reference point
61805	Value programmed in absolute and incremental dimensions	CYCLE328, CYCLE330, CYCLE371T, CYCLE374T, CYCLE376T, CYCLE383T, CYCLE384T, CYCLE385T	The intermediate position is programmed using both absolute and incremental dimensions
61806	Incorrect axis assignment	CYCLE328	The order of the axis assignment is incorrect
61807	Incorrect spindle direction pro–grammed	CYCLE384M	The programmed spindle direc–tion conflicts with the spindle direction used for the cycle
61808	Final drilling depth or single dril–ling depth missing	CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	Total depth Z or single drilling depth Q missing from G8x block (first call of cycle)
61809	Drilling position not admissible	CYCLE383T, CYCLE384T, CYCLE385T	
61810	ISO G cde not possible	CYCLE383T, CYCLE384T, CYCLE385T	
61811	ISO axis designation not admissible	CYCLE328, CYCLE330. CYCLE371T, CYCLE374T, CYCLE376T, CLE383T, CYCLE384T, CYCLE385T	The calling NC block contains an ISO axis designation which is not admissible
61812	Incorrect numeral value(s) in cycle call	CYCLE371T, CYCLE376T,	The calling NC block contains a numeral value which is not admissible
61813	Incorrect GUD value	CYCLE376T	Not admissable numeral value in cycle setting data
61814	Polar coordinatea not possible	CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	
61815	G40 not active	CYCLE374T, CYCLE376T	G40 was not active prior to the cycle call

Index

R

S

T

V

Yaskawa Siemens CNC Series

In the event that the end user of this product is to be the military and said product is to be employed in any weapons systems or the manufacture thereof, the export will fall under the relevant regulations as stipulated in the Foreign Exchange and Foreign Trade Regulations. Therefore, be sure to follow all procedures and submit all relevant documentation according to any and all rules, regulations and laws that may apply.
Specifications are subject to change without notice for ongoing product modifications and improvements.

Machine Tool OEM Sales Div.

Gate City Osaki West Tower, 1-11-1, Osaki, Shinagawa-ku, Tokyo 141-8644, Japan
PHONE +81-3-3493-7411 FAX +81-3-3493-7422

Siemens Japan K.K.
http://www.siemens.co.jp

Published in Japan

02- WSC02, 02-WSC09