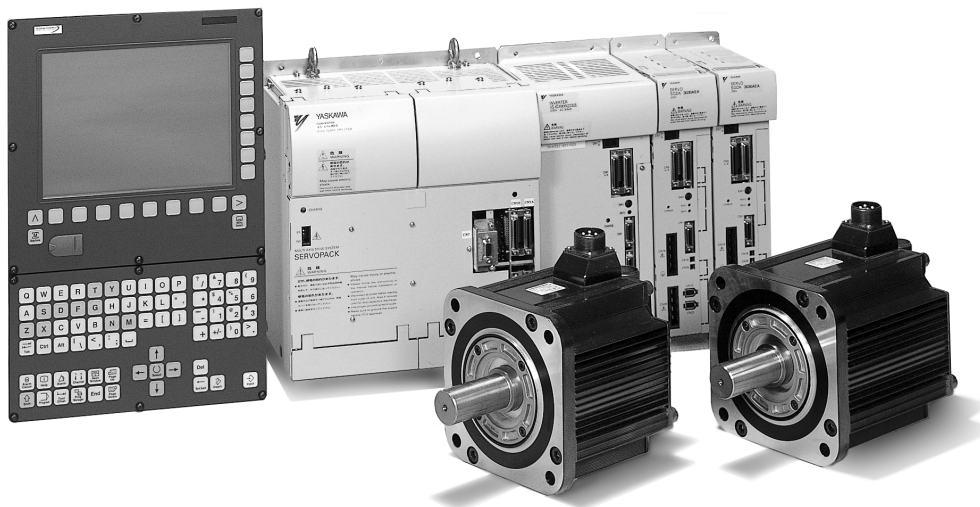


# Yaskawa Siemens CNC Series

## Programming Manual for Machining Center



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 Siemens 840DI

## Programming Manual for Machining Center

### Users Manual

#### Valid for

*Control System*                      *Software Version*  
Yaskawa Siemens 840DI

02.2001 Edition

<b>Programming Basiscs</b>	<b>1</b>
<b>Commands Calling Axis Movements</b>	<b>2</b>
<b>Movement Control Commands</b>	<b>3</b>
<b>Enhanced Level Commands</b>	<b>4</b>
<b>Appendix</b>	
<b>Abbreviations</b>	<b>A</b>
<b>Terms</b>	<b>B</b>
<b>G Code Table</b>	<b>C</b>
<b>MDs and SDs</b>	<b>D</b>
<b>Data Fields, Lists</b>	<b>E</b>
<b>Alarms</b>	<b>F</b>
<b>References</b>	<b>G</b>
<b>Index</b>	
<b>Commands</b>	

# 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.

<b>Edition</b>	<b>MANUAL No.</b>	<b>Remarks</b>
02.01	NCSIE – SP02 – 20	<b>A</b>

## Trademarks

SIMATIC®, SIMATIC HMI®, SIMATIC NET®, SIROTEC®, SINUMERIK® and SIMODRIVE® are Siemens trademarks. 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.

The reproduction, transmission or use of this document or its contents is not permitted without express written authority. Offenders will be liable for damages. All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

© Yaskawa Siemens NC Corp. 2001. All rights reserved.

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.

We have checked that the contents of this document correspond to the hardware and software described. Nonetheless, differences might exist. The information contained in this document is, however, reviewed regularly and any necessary changes will be included in the edition. We welcome suggestions for improvement.

Subject to technical changes without prior notice.

---

# 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. Measuring Cycles...) can be obtained from your local Yaskawa 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:

7. Index of abbreviations

8. 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 Guidelines

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<sup>®</sup> is a registered trademark of the International Business Corporation.  
MS-DOS<sup>®</sup> and WINDOWS<sup>™</sup> 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

<b>1</b>	<b>Programming Basics</b>	<b>1-15</b>
1.1	Introductory explanations	1-16
1.1.1	Siemens mode	1-16
1.1.2	ISO Dialect mode	1-16
1.1.3	Switchover	1-16
1.1.4	G code display	1-17
1.1.5	Maximum number of axes/axis designation	1-17
1.1.6	Block skip (/0 to /7)	1-17
1.2	Basics of feed function	1-19
1.2.1	Rapid traverse	1-19
1.2.2	Cutting feed (F command)	1-19
1.2.3	F1-digit feed	1-22
1.2.4	Feed per minute function (G94)	1-23
1.2.5	Inverse time feed (G93)	1-23
<b>2</b>	<b>Commands Calling Axis Movements</b>	<b>2-25</b>
2.1	Interpolation commands	2-26
2.1.1	Positioning (G00)	2-26
2.1.2	Linear interpolation (G01)	2-28
2.1.3	Circular interpolation (G02, G03)	2-29
2.1.4	Helical interpolation (G02, G03)	2-35
2.2	Reference point return	2-37
2.2.1	Automatic return to reference point (G28)	2-37
2.2.2	Reference point return check (G27)	2-39
2.2.3	Second to fourth reference point return (G30)	2-41
2.2.4	Tool retract and return (G10.6)	2-42
<b>3</b>	<b>Movement Control Commands</b>	<b>3-43</b>
3.1	The coordinate system	3-44
3.1.1	Machine coordinate system (G53)	3-45
3.1.2	Workpiece coordinate system (G92)	3-46
3.1.3	How to select a workpiece coordinate system	3-48
3.1.4	How to change a workpiece coordinate system	3-48
3.1.5	Local coordinate system (G52)	3-52
3.1.6	Plane selection (G17, G18, G19)	3-53
3.1.7	Parallel axes (G17, G18, G19)	3-54
3.1.8	Rotation of coordinate system (G68, G69)	3-55
3.2	Determining the coordinate value input modes	3-57
3.2.1	Absolute/incremental designation (G90, G91)	3-57
3.2.2	Inch/Metric input designation (G20, G21)	3-58
3.2.3	Scaling (G50, G51)	3-60
3.2.4	Programmable mirror image (G50.1, G51.1)	3-64
3.3	Time-controlling commands	3-66
3.3.1	Dwell (G04)	3-66
3.4	Cutting feedrate control	3-67
3.4.1	Exact stop (G09, G61), cutting mode (G64), tapping mode (G63)	3-67
3.5	Tool offset functions	3-68

3.5.1	Tool offset data memory	3-68
3.5.2	Tool length offset (G43, G44, G49)	3-68
3.5.3	Cutter radius compensation (G40, G41, G42)	3-71
3.5.4	Collision monitoring	3-76
3.6	S, T, M, and B functions	3-81
3.6.1	Spindle function (S function)	3-81
3.6.2	Tool function (T function)	3-82
3.6.3	Miscellaneous function (M function)	3-82
3.6.4	Internally processed M codes	3-83
3.6.5	General purpose M codes	3-83
<b>4</b>	<b>Enhanced Level Commands</b>	<b>4-85</b>
4.1	Program support functions (1)	4-86
4.1.1	Canned cycles (G73 to G89)	4-86
4.1.2	High-speed peck drilling cycle (G73)	4-93
4.1.3	Fine boring cycle (G76)	4-94
4.1.4	Drilling cycle, spot drilling (G81)	4-98
4.1.5	Drilling cycle, counter boring cycle (G82)	4-100
4.1.6	Peck drilling cycle (G83)	4-102
4.1.7	Boring cycle (G85)	4-104
4.1.8	Boring cycle (G86)	4-106
4.1.9	Boring cycle, back boring cycle (G87)	4-108
4.1.10	Boring cycle (G89)	4-111
4.1.11	Rigid tapping cycle (G84)	4-113
4.1.12	Left-handed rigid tapping cycle (G74)	4-116
4.1.13	Peck tapping cycle (G84 or G74)	4-119
4.1.14	Canned cycle cancel (G80)	4-122
4.1.15	Program example using tool length offset and canned cycles	4-123
4.2	Programmable data input (G10)	4-125
4.2.1	Changing of tool offset value	4-125
4.2.2	Setting the workpiece coordinate system shift data	4-125
4.3	Subprogram call up function (M98, M99)	4-126
4.4	Polar coordinate command (G15, G16)	4-127
4.5	Polar coordinate interpolation (G12.1, G13.1)	4-128
4.6	Cylindrical interpolation (G07.1)	4-130
4.7	Program support functions (2)	4-135
4.7.1	Working area limitation (G22, G23) (Under development)	4-135
4.7.2	Chamfering and corner rounding commands	4-137
4.8	Automating support functions	4-141
4.8.1	Skip function (G31)	4-141
4.8.2	Multistage skip (G31, P1 – P4)	4-144
4.8.3	Program interrupt function (M96, M97)	4-145
4.8.4	Tool life control function	4-147
4.9	Macroprograms	4-148
4.9.1	Differences from subprograms	4-148
4.9.2	Macroprogram call (G65, G66, G67)	4-148
4.10	Additional functions	4-155

4.10.1	Figure copy (G72.1, G72.2) .....	4-155
<b>A</b>	<b>Abbreviations</b> .....	<b>A-159</b>
<b>B</b>	<b>Terms</b> .....	<b>B-169</b>
<b>C</b>	<b>G Code Table</b> .....	<b>C-199</b>
C.1	G code table .....	C-200
<b>D</b>	<b>Machine and Setting Data</b> .....	<b>D-205</b>
D.1	Machine/Setting data .....	D-205
D.2	Channel-specific machine data .....	D-213
D.3	Axis-specific setting data .....	D-219
D.4	Channel-specific setting data .....	D-219
<b>E</b>	<b>Data Fields, Lists</b> .....	<b>E-221</b>
E.1	Machine data .....	E-221
E.2	Setting data .....	E-223
<b>F</b>	<b>Alarms</b> .....	<b>F-225</b>
<b>G</b>	<b>References</b> .....	<b>G-227</b>
<b>H</b>	<b>Index</b> .....	<b>H-245</b>
<b>I</b>	<b>Commands</b> .....	<b>I-247</b>



# Programming Basics

# 1

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

1.1	Introductory explanations .....	1-16
1.1.1	Siemens mode .....	1-16
1.1.2	ISO Dialect mode .....	1-16
1.1.3	Switchover .....	1-16
1.1.4	G code display .....	1-17
1.1.5	Maximum number of axes/axis designation .....	1-17
1.1.6	Block skip (/0 to /7) .....	1-17
1.2	Basics of feed function .....	1-19
1.2.1	Rapid traverse .....	1-19
1.2.2	Cutting feed (F command) .....	1-19
1.2.3	F1-digit feed .....	1-22
1.2.4	Feed per minute function (G94) .....	1-23
1.2.5	Inverse time feed (G93) .....	1-23

## 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-M the maximum number of axis is 9. Axis designation for the first three axes is fixed to X, Y and Z. Further axes can be designated A, B, C, U, V, W.

### 1.1.6 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.

---

#### Note

Setting units of rapid traverse rate	1 mm/min
	0.1 inch/min
	1 deg./min

Since the most appropriate value is set conforming to the machine capability, refer to the manuals published by the machine tool builder for the rapid traverse rate of your machine.

---

### 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 character F.

With a 6-digit numeral specified following address character F, feedrate of a cutting tool can be designated in units of "mm/min".

Refer to the manuals published by the machine tool builder for programmable range of the F code.

## 1.2 Basics of feed function

The upper limit of feedrates could be restricted by the servo system and the mechanical system. In this case, the allowable upper limit is set by MD and if a feedrate command exceeding this limit value is specified, the feedrate is clamped at the set allowable upper limit.

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.

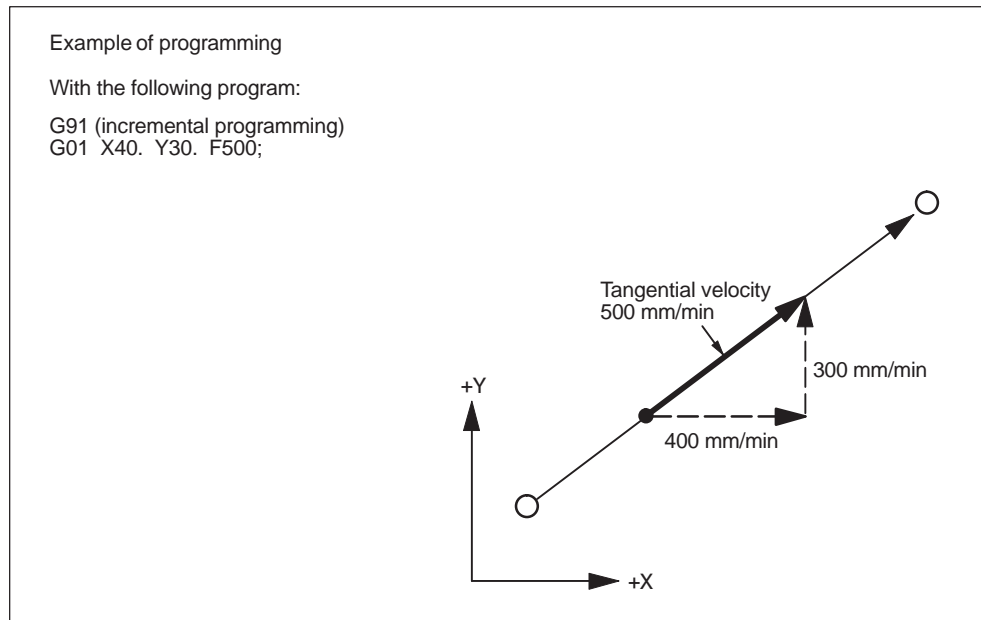


Fig. 1-1 F command in simultaneous 2-axis control linear interpolation

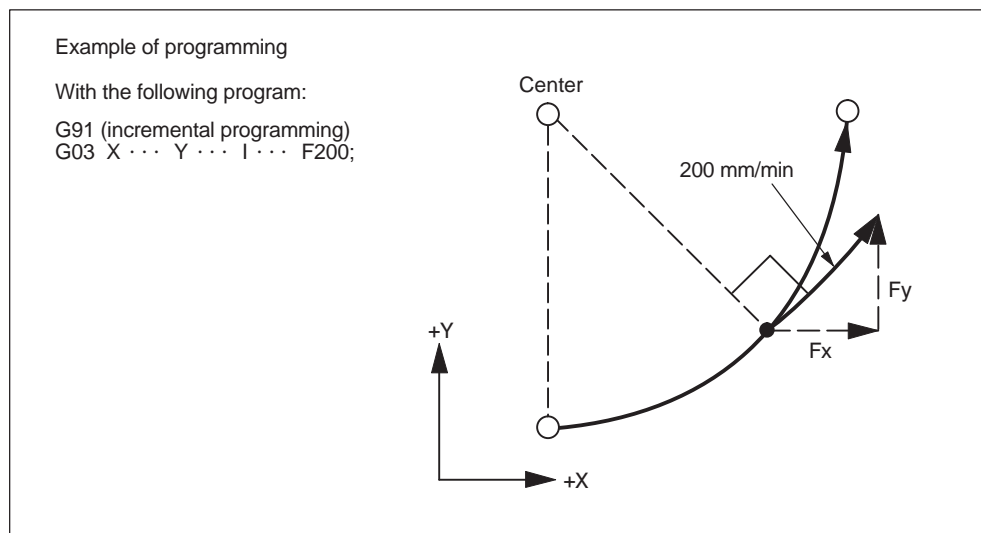


Fig. 1-2 F command in simultaneous 2-axis control circular interpolation

In the simultaneous 3-axis control linear interpolation, an F command indicates the tangential feedrate.

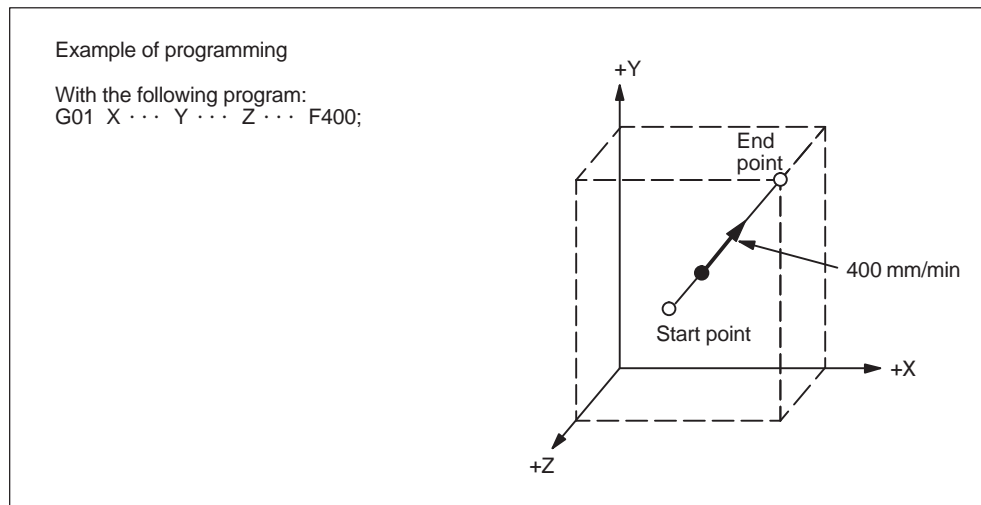


Fig. 1-3 F command in simultaneous 3-axis control linear interpolation

In the simultaneous 4-axis control linear interpolation, an F command indicates the tangential feedrate.

$$F \text{ (mm/min)} = \sqrt{F_x^2 + F_y^2 + F_z^2 + F_\alpha^2}$$

In the simultaneous 5-axis control linear interpolation, an F command indicates the tangential feedrate.

$$F \text{ (mm/min)} = \sqrt{F_x^2 + F_y^2 + F_z^2 + F_\alpha^2 + F_\beta^2}$$

#### Note

1. If "F0" is specified and F 1-digit feed is not used, an alarm occurs.
2. For an F command, a minus value must not be specified. If a minus value is specified for an F command, correct operation cannot be guaranteed.

## 1.2 Basics of feed function

## 1.2.3 F1-digit feed

It is possible to select a feedrate by specifying a 1-digit numeral (1 to 9) following address F. With this manner of designation of an F command, the feedrate preset for the specified numeral is selected.

The F1–Digit Feed function needs to be enabled by MD setting as follows:

`$MC_FIXED_FEEDRATE_F1_F9_ON = TRUE`: F1–Digit Feed enable

`$MC_FIXED_FEEDRATE_F1_F9_ON = FALSE`: F1–Digit Feed disable

With the above mentioned MD set to FALSE, F1 to F9 in a machining program is interpreted as standard feed (F) programming, i.e. F2 = 2 mm/min. With the above mentioned MD set to TRUE, the feedrate to be selected in response to the designation of F1 to F9 should be set for the setting data indicated in Table 1-1. Feedrate 0 is activated if the corresponding value of the setting data is 0.

Table 1-1 Setting data used for presetting F1–digit feedrates

F command	Setting data
F1	<code>\$SC_FIXED_FEEDRATE_F1_F9[0]</code>
F2	<code>\$SC_FIXED_FEEDRATE_F1_F9[1]</code>
F3	<code>\$SC_FIXED_FEEDRATE_F1_F9[2]</code>
F4	<code>\$SC_FIXED_FEEDRATE_F1_F9[3]</code>
F5	<code>\$SC_FIXED_FEEDRATE_F1_F9[4]</code>
F6	<code>\$SC_FIXED_FEEDRATE_F1_F9[5]</code>
F7	<code>\$SC_FIXED_FEEDRATE_F1_F9[6]</code>
F8	<code>\$SC_FIXED_FEEDRATE_F1_F9[7]</code>
F9	<code>\$SC_FIXED_FEEDRATE_F1_F9[8]</code>

Note: Input format=REAL

**Note**

1. If F1–digit command is activated by setting MD \$MC\_FIXED\_FEEDRATE\_F1\_F9\_ON = TRUE and F1 to F9 should not be used, be sure to program the feedrate F as a REAL value. For example, not F1 but F 1.0 for 1 mm/min.
2. If “F0” is specified, it is switched to rapid traverse mode (G00) automatically. Subsequently, G01 needs to be specified in order to use F1–digit command.
3. When the DRY RUN switch is ON, feed commands are all executed at the feedrate set for the dry run operation.
4. The feed override function is invalid for the feedrate selected by the F1-digit command.
5. The feedrate set for setting data is retained in memory if the power is turned OFF.
6. In a macro call using G65/G66, the value commanded with address F is always stored in system variable \$C\_F, meaning that numeral values 1 to 9 will be stored.
7. If F1–digit command is used in a machining program containing a cycle call (G81 to G87), the feedrates are read from the corresponding setting data and stored into variable \$C\_F.

**Example**

```
$SC_FIXED_FEEDRATE_F1_9[0] = 15000
$SC_FIXED_FEEDRATE_F1_9[1] = 5500
```

```
N10 X10 Y10 Z10 F0 G94 ; Positioning, rapid traverse
N20 G01 X150 Y30 F1 ; feedrate 1500 mm/min active
N30 Z0 F2 ; feedrate 550 mm/min active
N40 Z10 F0 ; Positioning, rapid traverse
```

**1.2.4 Feed per minute function (G94)**

When G94 is designated, a feedrate specified following address F is executed in units of “mm (inch)/min”.

**1.2.5 Inverse time feed (G93)**

When G93 is designated, a feedrate specified following address F is executed in units of “1/min”. G93 is a modal G code.

**Example**

```
N10 G93 G1 X100 F2 ;
i.e. the programmed distance will be moved within half a minute.
```

## Notes

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---



# Commands Calling Axis Movements

# 2

Chapter 2 describes the interpolation commands and the reference point return commands.

2.1	Interpolation commands .....	2-26
2.1.1	Positioning (G00) .....	2-26
2.1.2	Linear interpolation (G01) .....	2-28
2.1.3	Circular interpolation (G02, G03) .....	2-29
2.1.4	Helical interpolation (G02, G03) .....	2-35
2.2	Reference point return .....	2-37
2.2.1	Automatic return to reference point (G28) .....	2-37
2.2.2	Reference point return check (G27) .....	2-39
2.2.3	Second to fourth reference point return (G30) .....	2-41
2.2.4	Tool retract and return (G10.6) .....	2-42

## 2.1 Interpolation commands

### 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)

In the absolute programming mode (G90), the axes are moved to the specified point in a workpiece coordinate system, and in the incremental programming mode (G91), the axes move by the specified distance from the present position at a rapid traverse rate.

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

Table 2-1 G codes for positioning

G code	Function	Group
G00	Positioning	01

#### Positioning (G00)

##### Format

G00 X... Y... Z... ;

##### Explanation

When G00 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 3-axis (\*5-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.

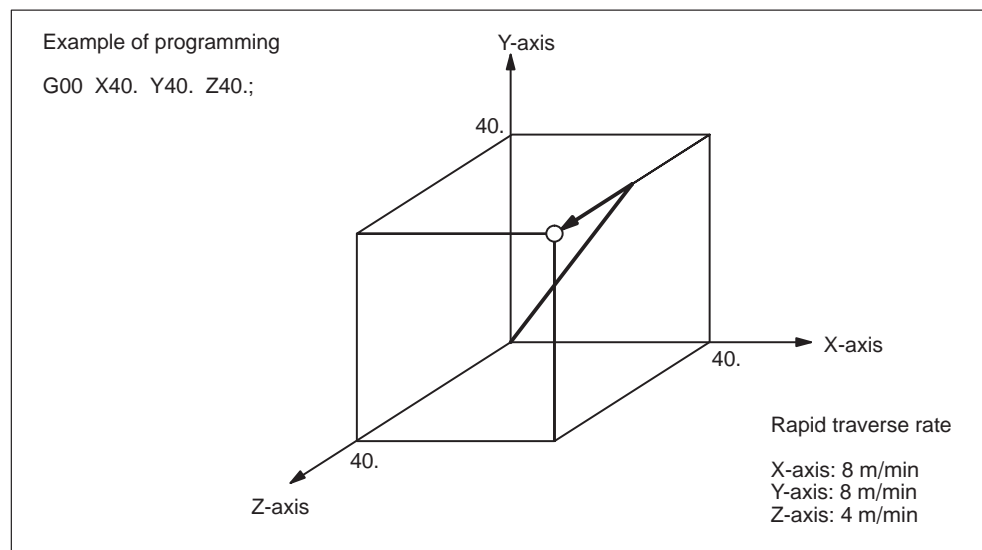


Fig. 2-1 Positioning in simultaneous 3-axis control mode

**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.

**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 Interpolation commands

### 2.1.2 Linear interpolation (G01)

#### Format

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

With the commands of G01, linear interpolation is executed in the simultaneous 3-axis (\*5-axis) control mode. The axes not designated in the G01 block do not move. For the execution of the linear interpolation, the above command 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.

$$F \text{ (mm/min)} = \sqrt{F_x^2 + F_y^2 + F_z^2 + (F_\alpha^2 + F_\beta^2)}$$

(F<sub>x</sub>: feedrate in the X-axis direction)

---

#### Note

If no F code is designated in the block containing G01 or in the preceding blocks, execution of a G01 block causes an alarm.

---

If the optional 4th and 5th axis are rotary axes (A-, B-, or C-axis), feedrates of basic three axes (X-, Y-, and Z-axis) and the optional 4th and 5th axis are determined in the machine data (MD).

## End point

The end point can be specified in either incremental or absolute values. In G code system B and C it is determined corresponding to the designation of G90 or G91 (for details, see 3.2.1, "Absolute/Incremental Programming").

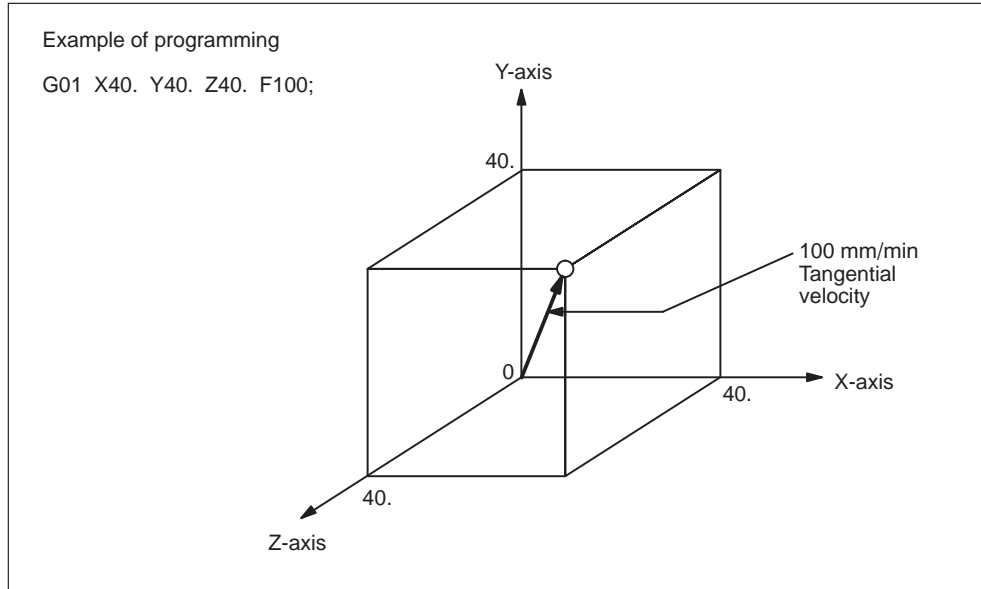


Fig. 2-2 Linear interpolation

## 2.1.3 Circular interpolation (G02, G03)

### Command format

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

Table 2-2 Commands necessary for circular interpolation

Item	Command	Description
Plane designation	G17	Circular arc in the XY plane
	G18	Circular arc in the ZX plane
	G19	Circular arc in the YZ plane
Direction of rotation	G02	Clockwise (CW)
	G03	Counterclockwise (CCW)
Position of end point	Two axes among X, Y, and Z	End point position in a workpiece coordinate system
	Two axes among X, Y, and Z	Signed distance from the start point to the end point
Distance from the start point to the center	Two axes among I, J, and K	Signed distance from the start point to the center
Radius of circular arc	R	Radius of circular arc
Feedrate	F	Velocity along the circular arc

2.1 Interpolation commands

**Plane designation**

With the commands indicated below, a cutting tool moves along the specified circular arc in the XY plane, ZX plane, or YZ plane so that the feedrate specified by the F command will be the tangential velocity of the arc.

- In the XY Plane  
G17 G02 (or G03) X...Y...R... (or I...J...) F...;
- In the ZX Plane  
G18 G02 (or G03) Z...X...R... (or K...I...) F...;
- In the YZ Plane  
G19 G02 (or G03) Y...Z...R... (or J...K...) F...;

To designate the circular interpolation mode (G02, G03), the plane of interpolation should be selected first by specifying the G17, G18, or G19. For the 4th and 5th axis, circular interpolation is allowed only when they are linear axes.

The G code designated to select the plane in which circular interpolation is executed also selects the plane where tool radius offset (G41/G42) is executed. When the power is turned ON, the XY plane (G17) is automatically selected.

G17	XY plane, or X $\alpha$ or X $\beta$ plane
G18	ZX plane, or Z $\alpha$ or Z $\beta$ plane
G19	YZ plane, or Y $\alpha$ or Y $\beta$ plane

If an optional linear 4th-axis is selected, circular interpolation is possible in the X $\alpha$ , Z $\alpha$ , or Y $\alpha$  plane which includes the 4th-axis in addition to the XY, YZ, and ZX planes. ( $\alpha$ =U, V, or W)

- Circular interpolation in X $\alpha$  plane  
G17 G02 (or G03) X... $\alpha$ ...R... (or I...J...) F...;
- Circular interpolation in Y $\alpha$  plane  
G18 G02 (or G03) Z... $\alpha$ ...R... (or K...I...) F...;
- Circular interpolation in Z $\alpha$  plane  
G19 G02 (or G03) Y... $\alpha$ ...R... (or J...K...) F...;

## 2.1 Interpolation commands

If an optional linear 5th-axis is selected, circular interpolation is possible in the  $X\beta$ ,  $Z\beta$ , or  $Y\beta$  plane which includes the 5th-axis in addition to the XY, YZ, and ZX planes. ( $\beta=U, V, \text{ or } W$ )

- Circular interpolation in  $X\beta$  plane  
G17 G02 (or G03) X  $\dots$   $\beta$   $\dots$  R  $\dots$  (or I  $\dots$  J  $\dots$ ) F  $\dots$ ;
- Circular interpolation in  $Z\beta$  plane  
G18 G02 (or G03) Z  $\dots$   $\beta$   $\dots$  R  $\dots$  (or K  $\dots$  L  $\dots$ ) F  $\dots$ ;
- Circular interpolation in  $Y\alpha\beta$  plane  
G19 G02 (or G03) Y  $\dots$   $\beta$   $\dots$  R  $\dots$  (or J  $\dots$  K  $\dots$ ) F  $\dots$ ;
- If address characters which represent the 4th- and 5th-axis are omitted as with the commands of "G17 G02 X  $\dots$  R  $\dots$  (or I  $\dots$  J  $\dots$ ) F  $\dots$ ;" the XY plane is automatically selected for the interpolation plane. Circular interpolation with the 4th or 5th axis is not possible if these additional axes are rotary axes.

### Rotation direction

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

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

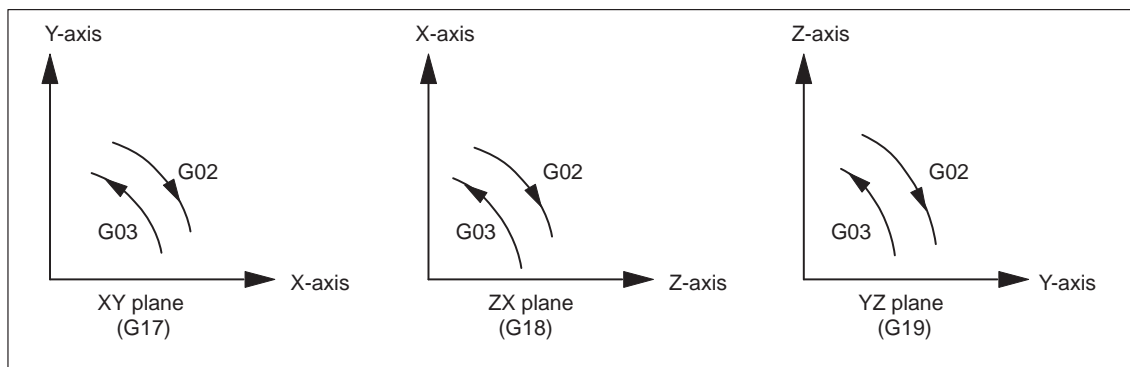


Fig. 2-3 Rotation direction of circular arc

### End point

The end point can be specified in either incremental or absolute values corresponding to the designation of G90 or G91 (not in G code system A).

2.1 Interpolation commands

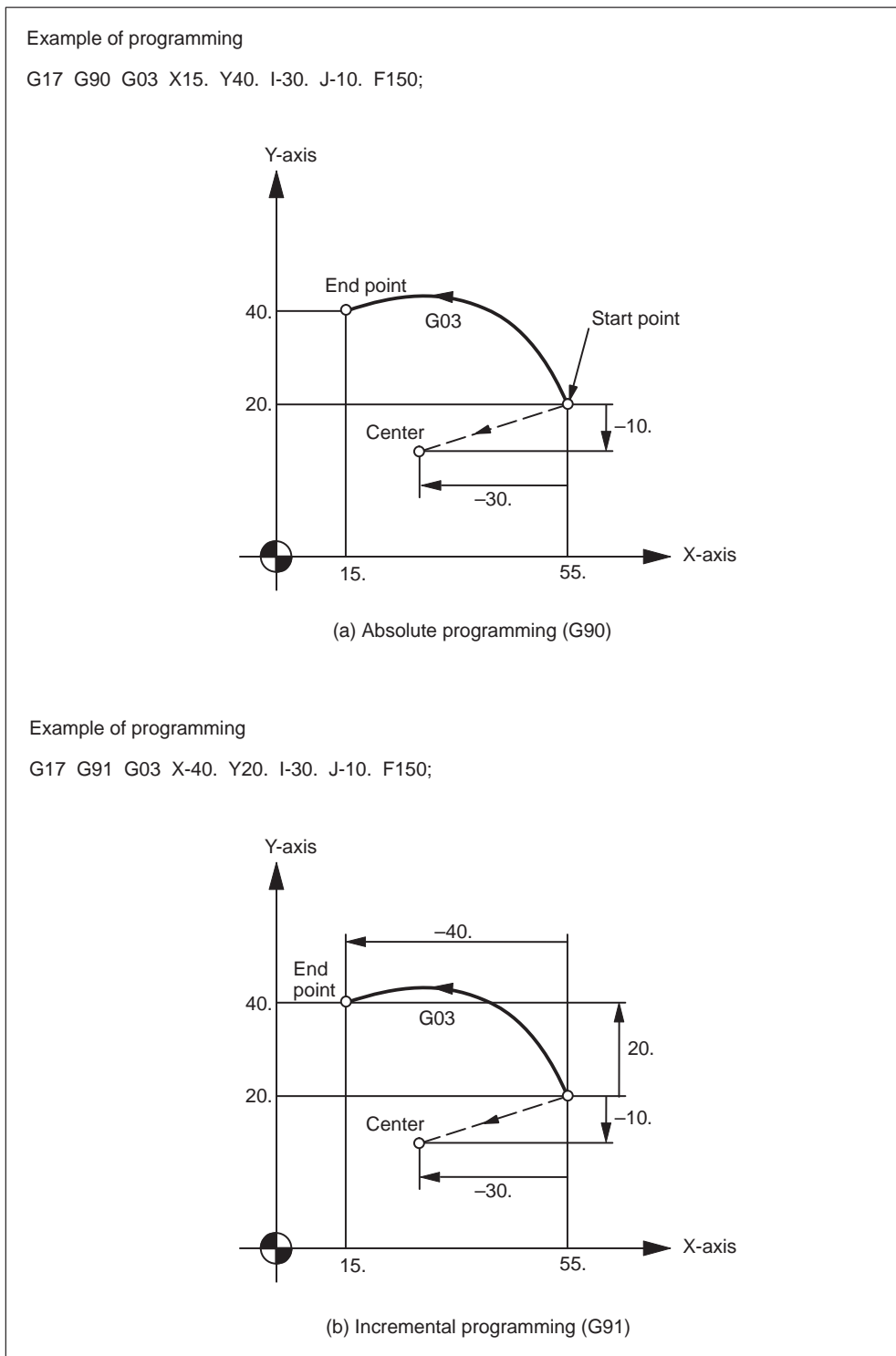


Fig. 2-4 End point of circular arc

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.



## 2.1 Interpolation commands

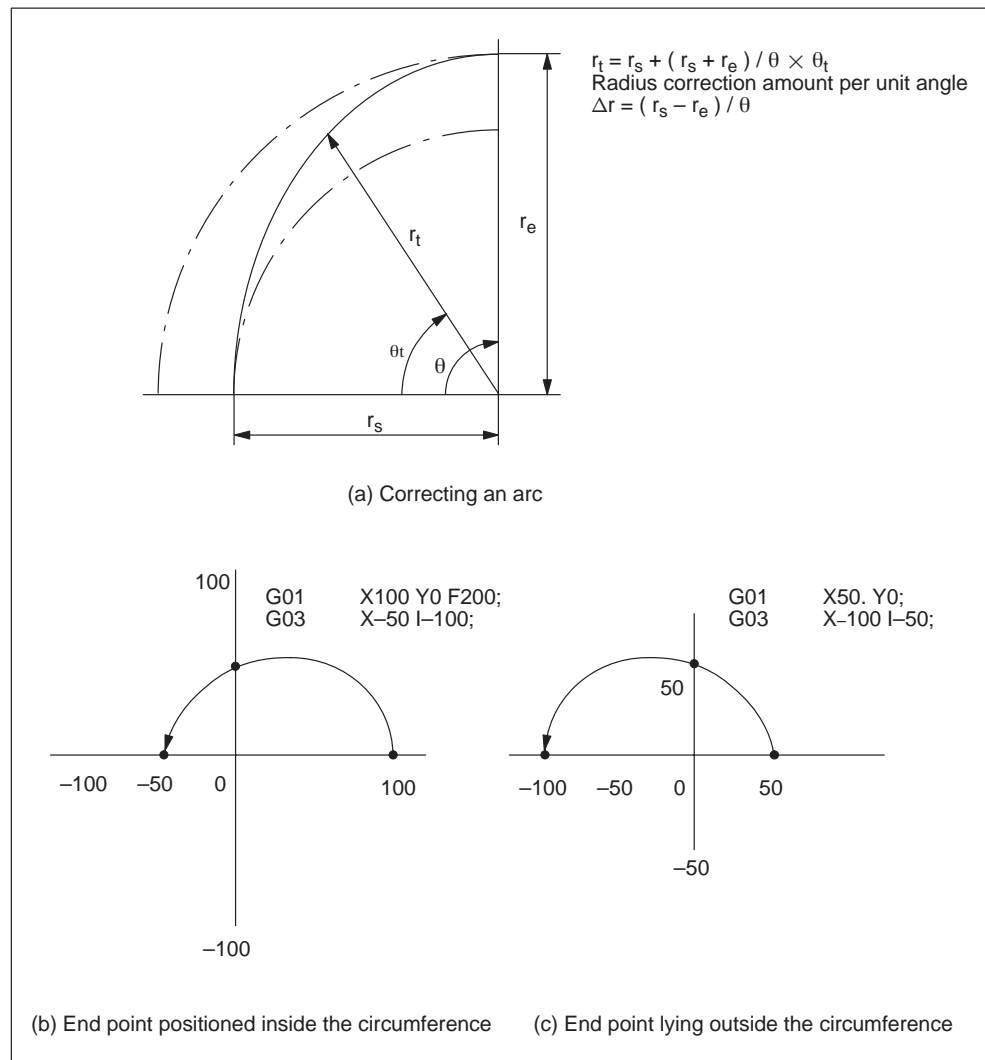


Fig. 2-5 Interpolation with end point of the specified arc

## 2.1 Interpolation commands

**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.

- 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, J, 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”.

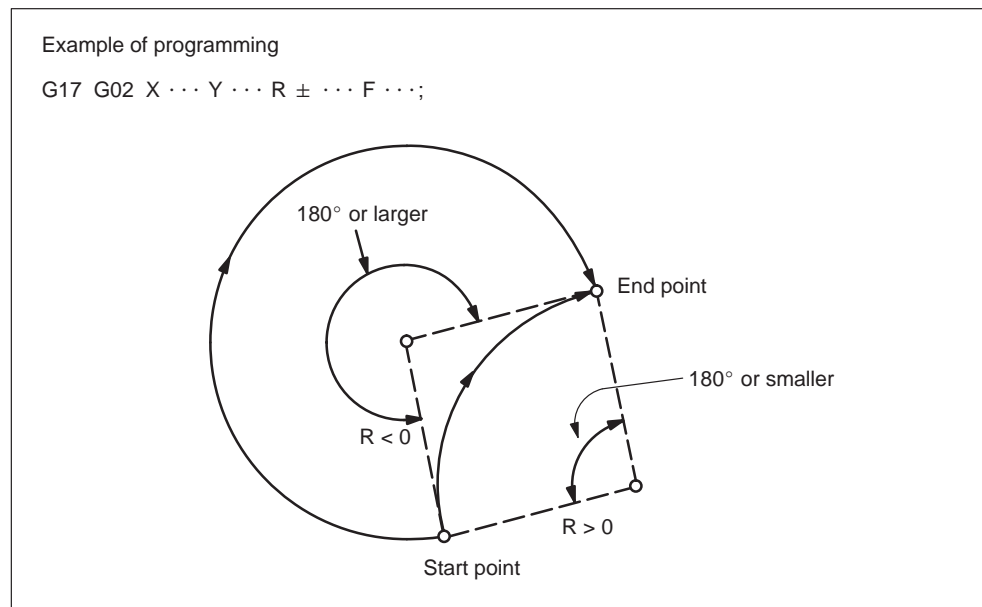


Fig. 2-6 Circular interpolation with radius R designation

**Feedrate**

In the circular interpolation mode, the feedrate can be specified in the same manner as in the linear interpolation mode. Refer to 2.1.2 “Linear interpolation (G01)”.

### Supplements to circular interpolation

A circular arc extending to multiple quadrants can be defined by the commands in a single block. It is also possible to specify a full circle.

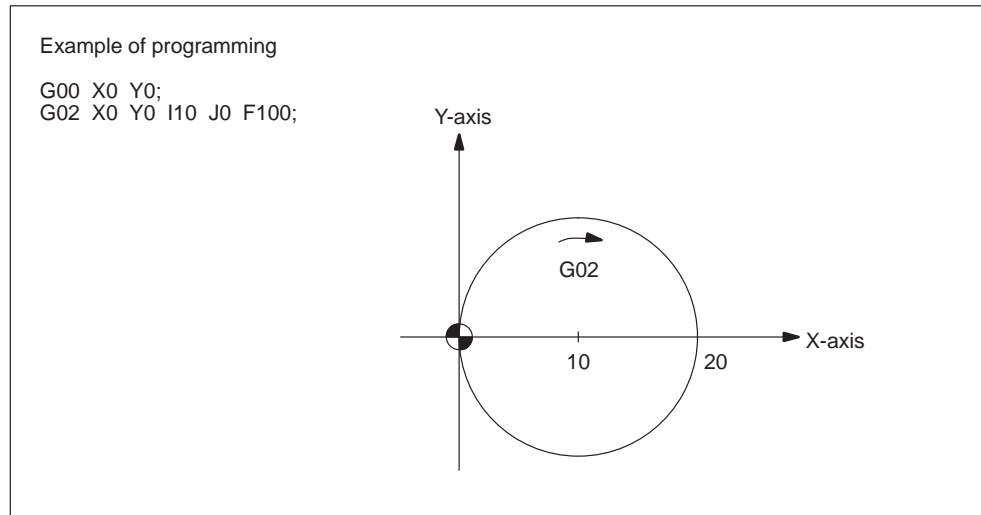


Fig. 2-7 Full circle

With the commands of “G17 G02 (or G03) I ··· J ··· F ··· Ln;”, full-circle interpolation is repeated by n times. If address L is omitted, interpolation is executed once. Execution of the commands with the single-block function ON causes full-circle interpolation to be interrupted after the execution of one full-circle interpolation.

#### 2.1.4 Helical interpolation (G02, G03)

It is possible to execute linear interpolation in synchronization with circular interpolation with the axis which is not included in the circular interpolation plane. This is called helical interpolation. The command format is indicated below.

- In the XY plane  
G17 G02 (or G03) X ··· Y ··· R ··· (or I ··· J ···) Z ( $\alpha$ ,  $\beta$ ) ··· F ···;
- In the ZX plane  
G18 G02 (or G03) Z ··· X ··· R ··· (or K ··· I ···) Y ( $\alpha$ ,  $\beta$ ) ··· F ···;
- In the YZ plane  
G19 G02 (or G03) Y ··· Z ··· R ··· (or J ··· K ···) X ( $\alpha$ ,  $\beta$ ) ··· F ···;
- In the  $X\alpha$  plane  
G17 G02 (or G03) X ···  $\alpha$  ··· R ··· (or I ··· J ···) Z ( $\beta$ ) ··· F ···;
- In the  $Z\alpha$  plane  
G18 G02 (or G03) Z ···  $\alpha$  ··· R ··· (or K ··· I ···) Y ( $\beta$ ) ··· F ···;
- In the  $Y\alpha$  plane  
G19 G02 (or G03) Y ···  $\alpha$  ··· R ··· (or J ··· K ···) X ( $\beta$ ) ··· F ···;

## 2.1 Interpolation commands

- In the  $X\beta$  plane  
G17 G02 (or G03) X  $\cdot\cdot$   $\beta$   $\cdot\cdot$  R  $\cdot\cdot$  (or I  $\cdot\cdot$  J  $\cdot\cdot$ ) Z ( $\alpha$ )  $\cdot\cdot$  F  $\cdot\cdot$ ;
- In the  $Z\beta$  plane  
G18 G02 (or G03) Z  $\cdot\cdot$   $\beta$   $\cdot\cdot$  R  $\cdot\cdot$  (or K  $\cdot\cdot$  L  $\cdot\cdot$ ) Y ( $\alpha$ )  $\cdot\cdot$  F  $\cdot\cdot$ ;
- In the  $Y\beta$  plane  
G19 G02 (or G03) Y  $\cdot\cdot$   $\beta$   $\cdot\cdot$  R  $\cdot\cdot$  (or J  $\cdot\cdot$  K  $\cdot\cdot$ ) X ( $\alpha$ )  $\cdot\cdot$  F  $\cdot\cdot$ ;

Where,  $\alpha$  and  $\beta$  are the linear 4th and 5th axes respectively, each representing any of U-, V-, and W-axis. If no 4th or 5th axis is specified as the end point command of the arc, any of the command format is selected among the commands in the XY plane, ZX plane, and YZ plane.

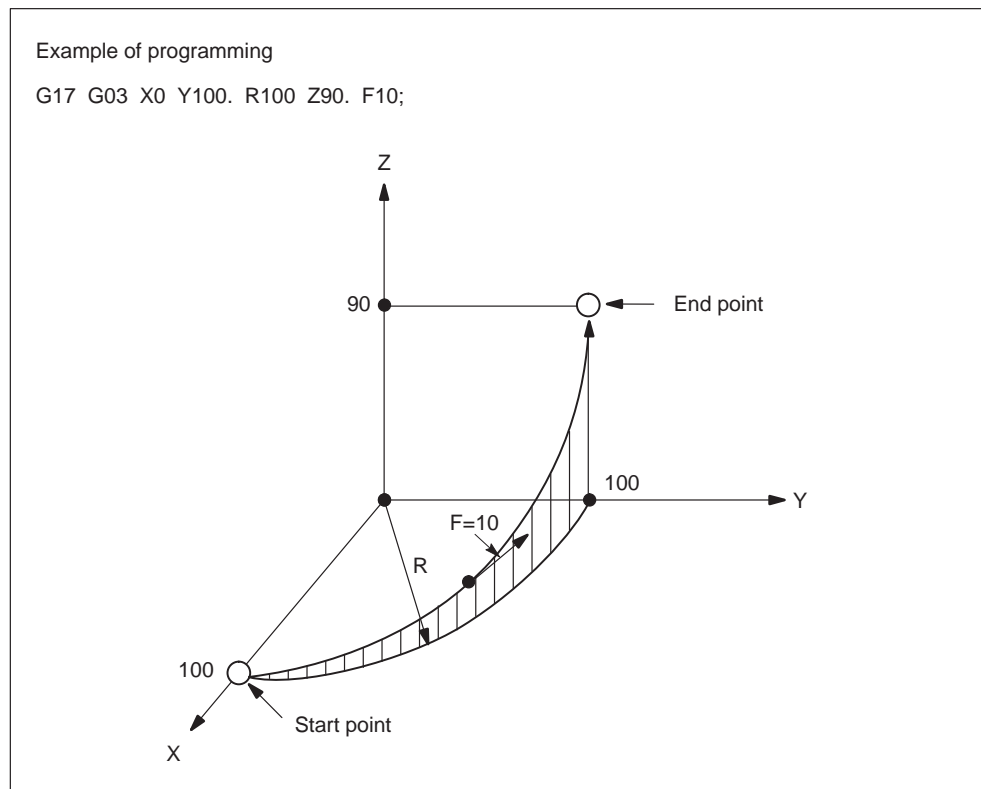


Fig. 2-8 Helical interpolation

**Note**

An arc must be programmed within  $360^\circ$  range.

The feedrate specified with an F command indicates the tangential velocity in the three dimensional space constituted by the circular interpolation plane and the linear axis perpendicular to the interpolation plane.

## 2.2 Reference point return

### 2.2.1 Automatic return to reference point (G28)

#### Format

G28 X... Y... Z... ;

With the commands of “G28 X ··· Y ··· Z ··· ;”, 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. This reference point return operation is possible in up to simultaneous 3-axis control. The axes not designated in the G28 block are not returned to the reference point.

#### Reference position

The reference position refers to a fixed position. The position of the tool can easily be referenced by means of the reference position return function. This could, for instance, be used as the tool change position. A total of four reference positions can be determined by setting the coordinates using MD \$ \_MA\_REFP\_SET\_POS[0] to [3]).

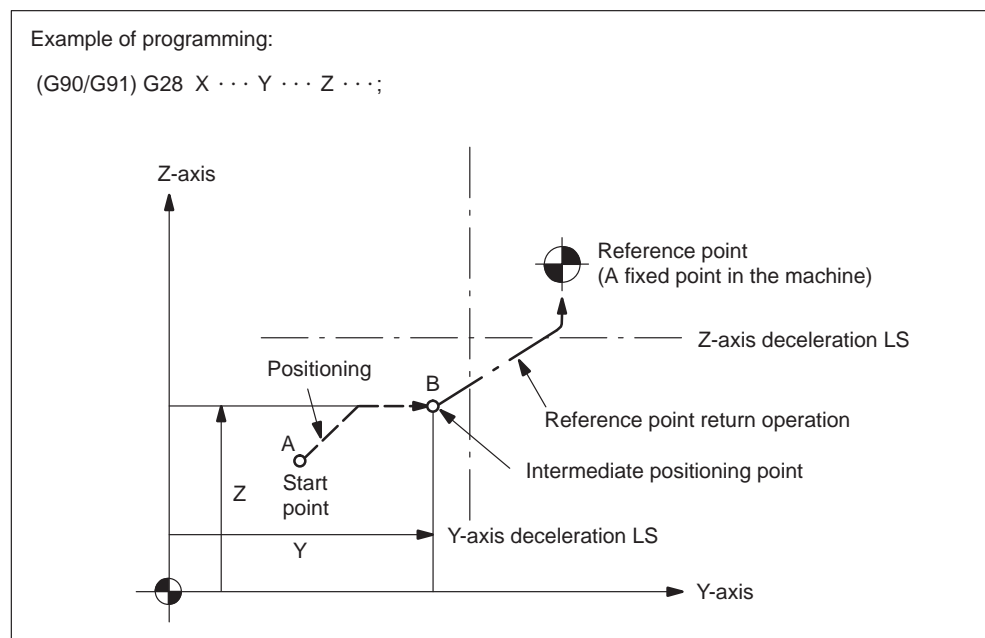


Fig. 2-9 Automatic reference point return

## 2.2 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.

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 high-speed reference point return specification allows the axes to return to the reference point.
- High-speed 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.

### Automatic reference point return for rotary axes

With a rotary axis, it is possible to execute the automatic reference point return the same as with a linear axis. With a rotary axis, if it has been moved by more than  $\pm 360.000^\circ$  from the reference point established first, reference point return is executed to the closest reference point in the preset direction of reference point return. The illustration below shows how the reference point return is executed from points A and B. (The reference point return direction is determined by the setting of MD\_\$MA\_REFP\_CAM\_IS\_MINUS.

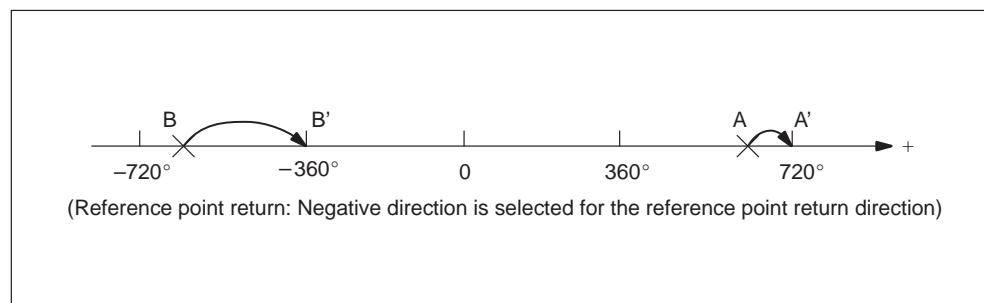


Fig. 2-10

### Supplements to the automatic reference point return commands

#### Tool radius offset and canned cycle

G28 must not be specified in the tool radius offset mode (G41, G42) or in a canned cycle.

**Warning**

Issuing G28 will cancel tool radius offset (G40) followed by axes movement towards the reference point. For that reason, make sure to disable tool radius offset before issuing G28.

**Tool position offset**

If G28 is specified in the tool position offset mode, positioning at the intermediate positioning point is made with the offset data valid. However, for the positioning at the reference point, the offset data are invalid and positioning is made at the absolute reference point.

**Tool length offset**

It is possible to cancel the tool length offset mode by G28 by changing the setting for a parameter. Although cancellation of the tool length offset mode is possible by G28, the tool length offset mode should be canceled before the designation of G28.

**Machine lock intervention**

The lamp for indicating the completion of return does not go on when the machine lock is turned on, even when the tool has automatically returned to the reference position. In this case, it is not checked whether the tool has returned to the reference position even when a G27 command is specified.

**2.2.2 Reference point return check (G27)****Format**

G27 X... Y... 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...Y...Z... (\* α ... β ...);".

In the G27 mode, the function checks whether or not the axes positioned by the execution of these commands in the simultaneous 3-axis (\* 5-axis) control mode are located at the reference point. For the axes not specified in this block, and not moved although the axis command specified, positioning and check are not executed.

---

## 2.2 Reference point return

### 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 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 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 and tool length offset functions are not canceled by the G27 command.
- Check is not made if G27 is executed while the machine lock state is valid even for one axis. For example, if an X-axis movement command is specified in the G27 block while in the Z-axis neglect state, X-axis position is not checked.
- The mirror image function is valid to the direction of axis movement in the reference point return operation called by G27. To avoid a position unmatched error, the mirror image function should be canceled before executing G27.



### 2.2.3 Second to fourth reference point return (G30)

#### Format

G30 Pn X... Y... Z... ;

With the commands of "G30 Pn X ··· Y ··· Z (\* α ··· β ···);", the axes are moved to P2 (second reference point), P3 (third reference point\*), or P4 (fourth reference point\*) in the simultaneous 3-axis (\* 5-axis) control mode after the positioning at the specified intermediate positioning point. If "G30 P3 X30. Y50.;" is specified, the X- and Y-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 2-3 Reference points

Item	MD
3rd reference point	\$_MA_REFP_SET_POS[2]
4th reference point	\$_MA_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 2.2.1, "Automatic Return to Reference Point (G28)".
- If G29 is specified after G30, positioning is made at the point specified with G29 after passing the intermediate positioning point specified with G30. Only the coordinate value of intermediate positioning point of the axis specified with G30 is updated.
- 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, an alarm occurs.

---

## 2.2 Reference point return

### 2.2.4 Tool retract and return (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.

---

# 3

## Movement Control Commands

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	3-44
3.1.1	Machine coordinate system (G53)	3-45
3.1.2	Workpiece coordinate system (G92)	3-46
3.1.3	How to select a workpiece coordinate system	3-48
3.1.4	How to change a workpiece coordinate system	3-48
3.1.5	Local coordinate system (G52)	3-52
3.1.6	Plane selection (G17, G18, G19)	3-53
3.1.7	Parallel axes (G17, G18, G19)	3-54
3.1.8	Rotation of coordinate system (G68, G69)	3-55
3.2	Determining the coordinate value input modes	3-57
3.2.1	Absolute/incremental designation (G90, G91)	3-57
3.2.2	Inch/Metric input designation (G20, G21)	3-58
3.2.3	Scaling (G50, G51)	3-60
3.2.4	Programmable mirror image (G50.1, G51.1)	3-64
3.3	Time-controlling commands	3-66
3.3.1	Dwell (G04)	3-66
3.4	Cutting feedrate control	3-67
3.4.1	Exact stop (G09, G61), cutting mode (G64), tapping mode (G63)	3-67
3.5	Tool offset functions	3-68
3.5.1	Tool offset data memory	3-68
3.5.2	Tool length offset (G43, G44, G49)	3-68
3.5.3	Cutter radius compensation (G40, G41, G42)	3-71
3.5.4	Collision monitoring	3-76
3.6	S, T, M, and B functions	3-81
3.6.1	Spindle function (S function)	3-81
3.6.2	Tool function (T function)	3-82
3.6.3	Miscellaneous function (M function)	3-82
3.6.4	Internally processed M codes	3-83
3.6.5	General purpose M codes	3-83

### 3.1 The coordinate system

## 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... Y... Z...

The above command is called a dimension word.

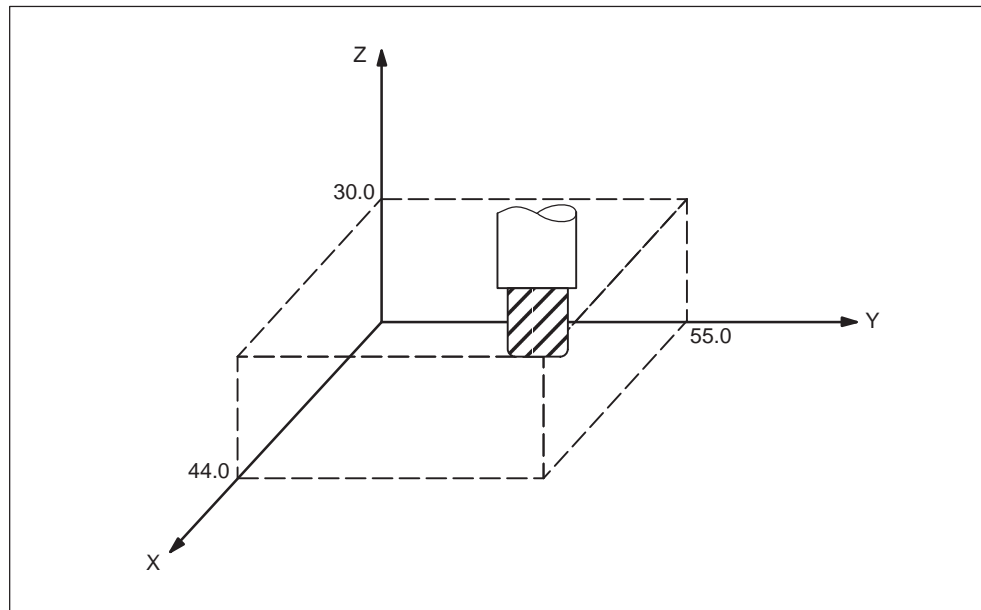


Fig. 3-1 Tool position specified by X... Y... Z...

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

1. Machine coordinate system (G53)
2. Workpiece coordinate system (G92)
3. Local coordinate system (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 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

(G90) G53 X... Y... Z... ;  
X, Y, 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.

#### Cancel of the compensation function

If \$MN\_G53\_TOOLCORR = 0, G53/G153/SUPA is non-modal suppression of zero offsets, tool length compensation and tool radius compensation, however, remain active.

If \$MN\_G53\_TOOLCORR = 1, G53/G153/SUPA is non-modal suppression of zero offsets, and active tool length and tool radius compensation.

#### 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.

### 3.1 The coordinate system

#### Reference

A machine coordinate system is set so that the reference position is at the coordinate values set using MD \$MC\_CHBFRAME\_POWON\_MASK Bit 0 whenever manual reference position return is applied after power-on.

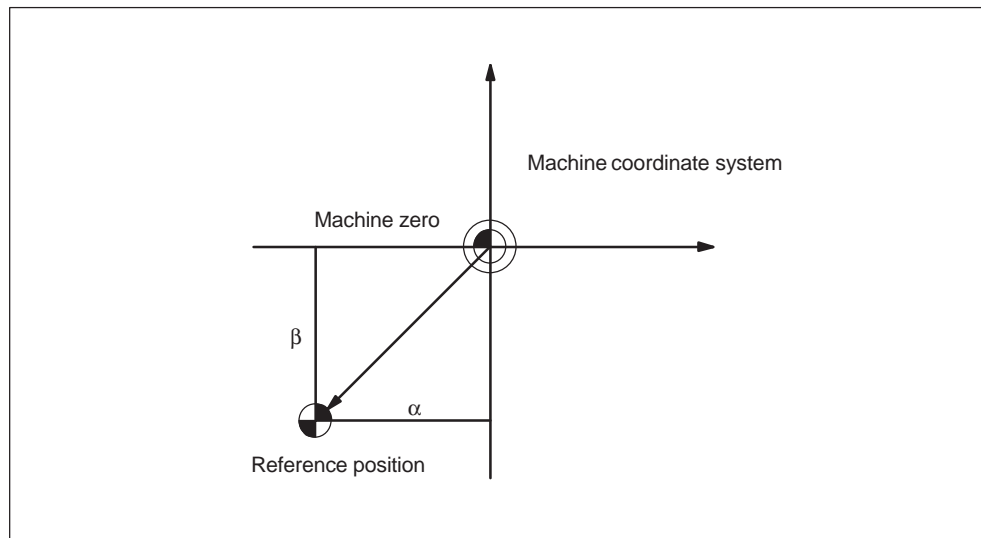


Fig. 3-2 Reference

#### 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 of 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

A workpiece coordinate system is set by determining a value subsequent to G92 within the program.

2. Manually, using the HMI panel

**Format**

(G90) G92 X... Y... Z... ;

**Examples**

Example 1:  
G92X30.5Z27.0;  
(The tip of tool is the start point.)

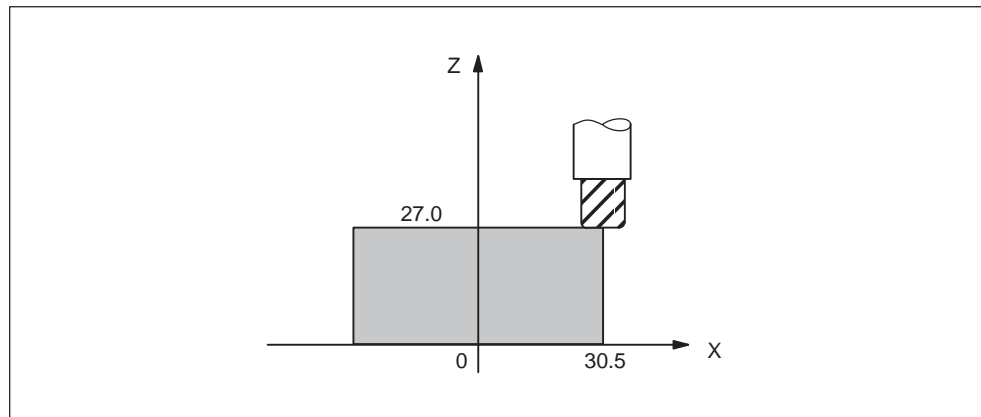


Fig. 3-3

Example 2:  
G92X500.0Z1100.0;  
(The base point on the tool holder is the start point.)

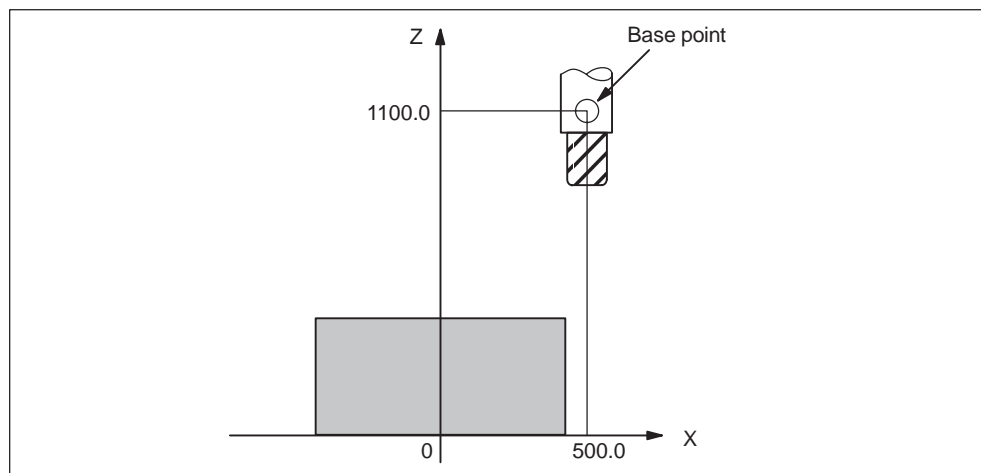


Fig. 3-4

### 3.1 The coordinate system

Whenever an absolute command is issued, the base point moves to the targeted position. The difference in position between the tool tip and the base point is compensated by the tool length offset in order to move the tool tip to the targeted position.

#### 3.1.3 How to select a workpiece coordinate system

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

1. G92

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

A workpiece coordinate systems can be selected by determining a G code from G54 to G59, and G54 P{1...100}, .

Workpiece coordinate systems are set up subsequent to reference position return after power-on. The default coordinate system after power-on is G54.

#### Examples

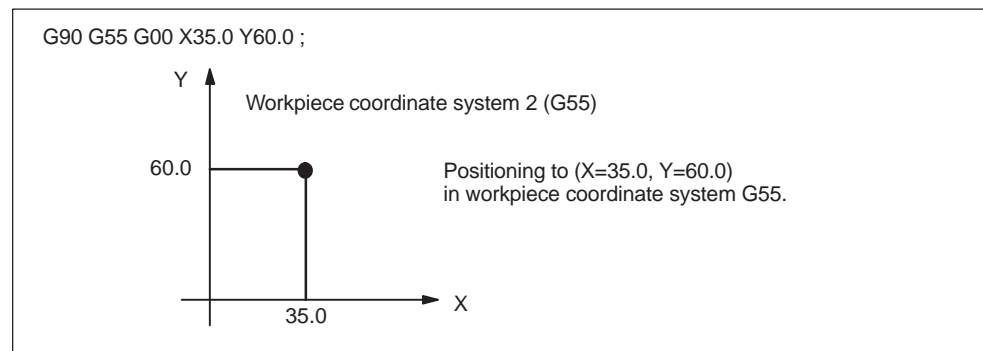


Fig. 3-5 Workpiece coordinate system G55

#### 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. Entering data using the HMI panel
2. By program command G10 or G92



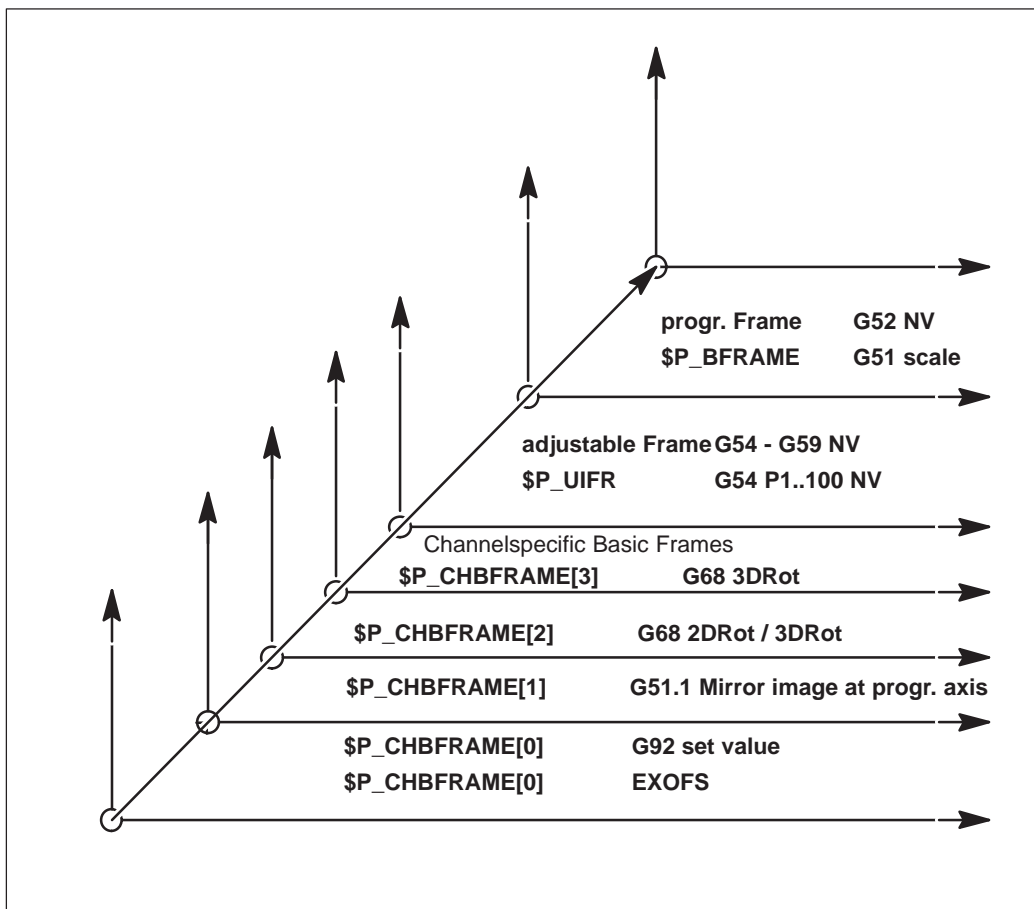


Fig. 3-6 ISO-dialect coordinate systems

G54P1...P100 (changes at Siemens Mode G505–G554 )

G58 (changes at Siemens Mode G505 )

G59 (changes at Siemens Mode G506 )

### 3.1 The coordinate system

#### Format

Changing by G10:

G10 L2 Pp X... Y... 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, Y, Z: Workpiece zero point offset for each axis in case of absolute command (G90).

Value to be added to the set workpiece zero point offset for each axis in case of an incremental command (G91).

G10 L20 Pp IP\_ ;

p=1 to 100: Workpiece zero point offset value correspond to workpiece coordinate system G54 P1 ... P100

X, Y, Z: For an absolute command (G90), workpiece zero point offset for each axis.

Value to be added to the set workpiece zero point offset for each axis in case of an incremental command (G91).

Changing by G92:

G92 X... Y... Z... ;

#### Explanations

Changing by using G10

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

Changing 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... Y... Z.... This way, the current tool position is made to match the specified coordinates. If X, Y, 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**

When the tool is positioned at (190, 150) in G54 mode, workpiece coordinate system 1 ( $X' - Y'$ ) shifted by vector A is created whenever G92X90Y90; is commanded.

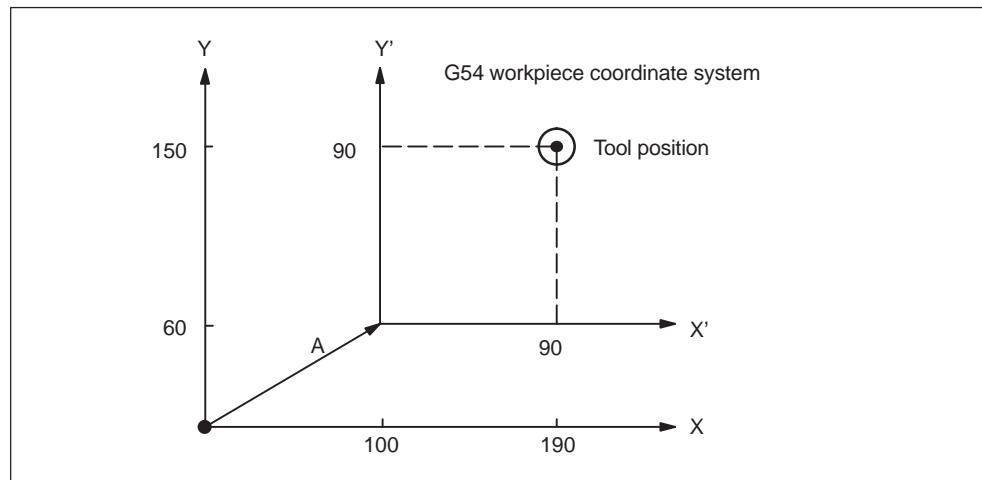


Fig. 3-7 Example of the setting of coordinates

---

### 3.1 The coordinate system

#### 3.1.5 Local coordinate system (G52)

For easier programming, a kind of sub-workpiece coordinate system can be set whenever a program is created in a workpiece coordinate system. Such a sub-coordinate system is called a local coordinate system.

##### Format

G52 X... Y... Z... ; Local coordinate system set

G52 X0 Y0 Z0 ; Local coordinate system cancel

X, Y, Z: Local coordinate system origin

## Explanations

A local coordinate system can be set in all the workpiece coordinate systems (G54 to G59) by specifying G52 X... Y... Z...;. Within the workpiece coordinate system, the origin of each local coordinate system is set to the position determined by X, Y, and Z.

Whenever a local coordinate system is set, the motion commands subsequently commanded in the absolute mode (G90) correspond to the coordinate values within the local coordinate system. By determining the G52 command through the zero point of a new local coordinate system in the workpiece coordinate system, the local coordinate system can be changed.

Match the zero point of the local coordinate system with that of the workpiece coordinate system in order to cancel the local coordinate system and to determine the coordinate value within the workpiece coordinate system.

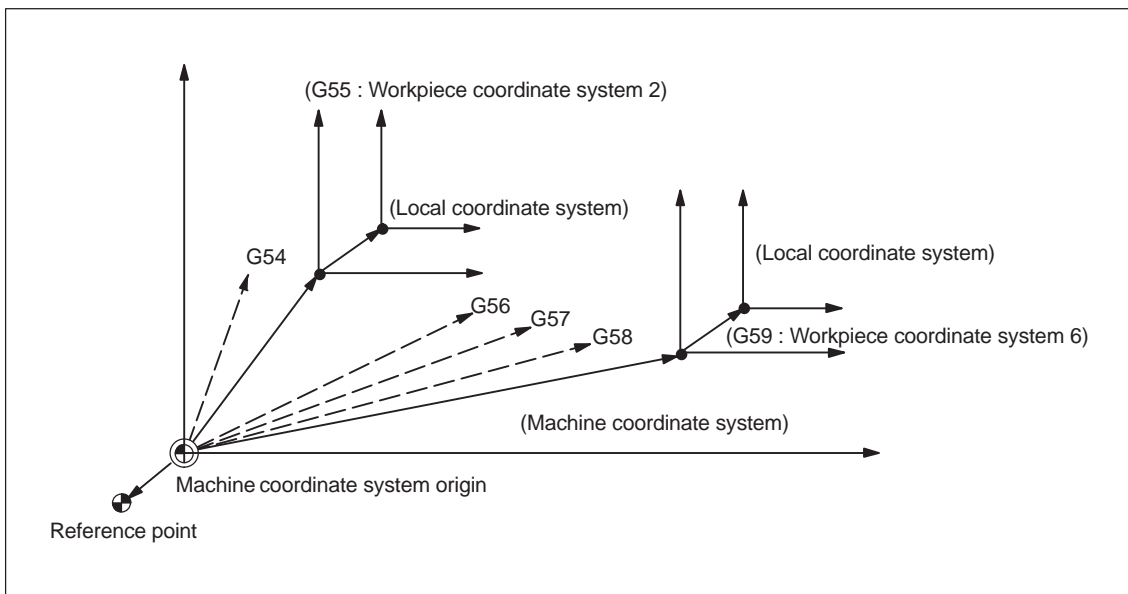


Fig. 3-8 Setting the local coordinate system

### 3.1.6 Plane selection (G17, G18, G19)

The plane where circular interpolation, tool radius offset, and coordinate system rotation are executed is selected by specifying the following G code.

Table 3-1 Plane selection G codes

G code	Function	Group
G17	XY plane	02
G18	ZX plane	02
G19	YZ plane	02

### 3.1 The coordinate system

A plane is defined in the following manner (in the case of XY plane):

The horizontal axis in the first quadrant is “+X-axis” and the vertical axis in the same quadrant “+Y-axis”.

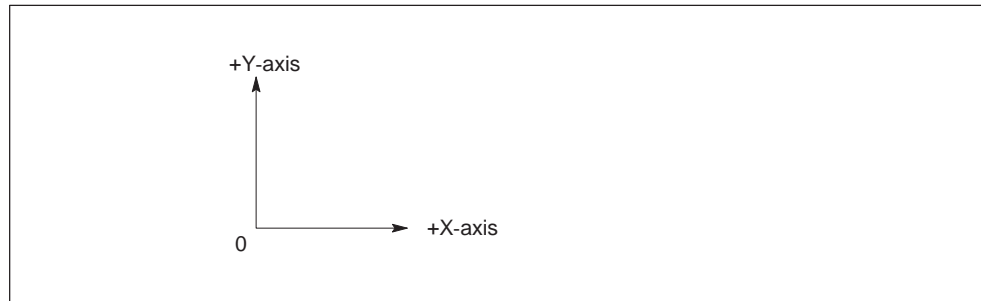


Fig. 3-9

- When the power is turned ON, the XY plane (G17) is selected.
- Axis move command of a single axis can be specified independent of the selection of plane by G17, G18, and G19. For example, the Z-axis can be moved by specifying “G17 Z ....”.
- Execution of a canned cycle is possible only in the G17 plane (hole machining axis: Z-axis).
- The plane on which the tool radius offset is executed by the G41 or G42 command is determined by the designation of G17, G18 or G19; the plane that includes the rotary 4th- or 5th-axis cannot be selected as the offset plane.

#### 3.1.7 Parallel axes (G17, G18, G19)

Using the function G17 (G18, G19) <axis name>, an axis parallel to one of the three basic axes of the coordinate system can be activated.

The three basic axes are, for example, X, Y, and Z.

#### Example

```
G17 U0 Y0
```

Parallel axis U is activated, replacing the X axis within the G17 plane.

## Explanations

- The parallel axes command is emulated using the Siemens function GEOAX(...). With the help of this function, a geometrical axis can be exchanged by any available channel axis.
- For each of the geometrical axes, a related parallel axis can be determined using machine data \$MC\_EX-TERN\_PARALLEL\_GEOAX[].
- Only axes related to the programmed plane (G17, G18, G19) can be exchanged.
- Usually, when exchanging axes, all offsets (frames) except for handwheel and external offsets, work area limitation and protection zones are cleared. Be sure to set the following machine data to prevent from clearing such values:
  - Offsets (frames)  
\$MN\_FRAME\_GEOAX\_CHANGE\_MODE
  - Protection zones  
\$MC\_PROTA-REA\_GEOAX\_CHANGE\_MODE
  - Work area limitation  
\$MN\_WALIM\_GEOAX\_CHANGE\_MODE
- Refer to machine data description for detail.
- Alarm 12726 is issued, if a basic axis is programmed together with its parallel axis in a plane selection command.

### 3.1.8 Rotation of coordinate system (G68, G69)

#### Using the G68 and G69 commands

#### Features of G68 and G69

For the rotation of a coordinate system, the following G codes are used.

Table 3-2 Coordinate system rotation G codes

G code	Function	Group
G68	Coordinate system rotation	16
G69	Cancel of coordinate system rotation	16

G68 and G69 are modal G codes belonging to 16-group. When the power is turned ON and when the NC is reset, G69 is automatically selected.

The G68 and G69 blocks must not include other G codes.

The coordinate system rotation which is called by G68 must be canceled by G69.

## 3.1 The coordinate system

**Command format**

G68 X\_ Y\_ R\_ ;

X\_, Y\_ :

Absolute coordinate values of the center of rotation. If omitted, the actual position is regarded as center of rotation.

R\_ :

Rotation angle, absolute or incremental depending on G90/G91. If omitted, the value of the channel specific setting \$SC\_DEFAULT\_ROT\_FACTOR\_R is used as rotation angle.

- By specifying "G17 (or G18, G19) G68 X...Y...R...";, the commands specified in the following blocks are rotated by the angle specified with R around the point (X, Y). Rotation angle can be specified in units of 0.001 degree.

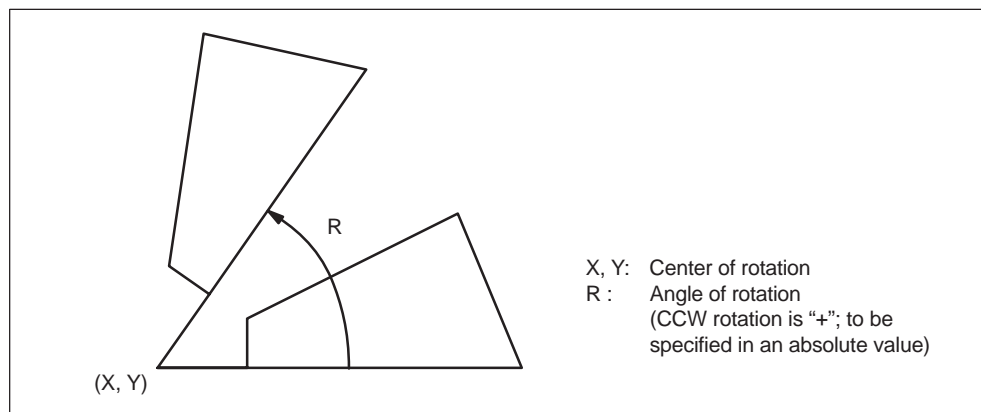


Fig. 3-10 Rotation of coordinate system

- By specifying "G69;", the coordinate system rotation mode is canceled.
- The G68 command is executed in the plane that has been selected when the G68 command is specified. The 4th and 5th axis must be linear axes.

G17 : XY plane or X $\alpha$ , X $\beta$  plane

G18 : ZX plane or Z $\alpha$ , Z $\beta$  plane

G19 : YZ plane or Y $\alpha$ , Y $\beta$  plane

**Supplements to the coordinate system rotation commands**

- MD \$MC\_MM\_NUM\_BASE\_FRAMES must be set to a value  $\geq 3$  if coordinate system rotation is used.
- If "X" and "Y" are omitted, the present position when the G68 block is executed is taken as the center of rotation.
- When the coordinate system is rotated, position data are given in the rotated coordinate system.
- Usually, the coordinate system rotation is turned ON before the start of approach motion and turned OFF after the completion of machining. The workpiece cannot be machined correctly if it is turned ON during machining.



## 3.2 Determining the coordinate value input modes

This section describes the commands used to input coordinate values.

### 3.2.1 Absolute/incremental designation (G90, G91)

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

#### Using the G90/G91 command

#### Features of G90 and G91

Table 3-3 Absolute/incremental designation G codes

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

- G90 and G91 are modal G code belonging to 03-group. If G90 and G91 are specified in the same block, the one specified later is valid.
- The power-ON state, that is the G90 or G91 mode that is valid when the power is turned ON, can be set to MD 20154:  
EXTERN\_GCODE\_RESET\_VALUES[2].

3.2 Determining the coordinate value input modes

**Command format**

- For the commands specified in and after the G90 block, the dimension values specified following an address of X, Y, Z, 4th are treated as absolute values.
- For the commands specified in and after the G91 block, the dimension values are treated as incremental values.

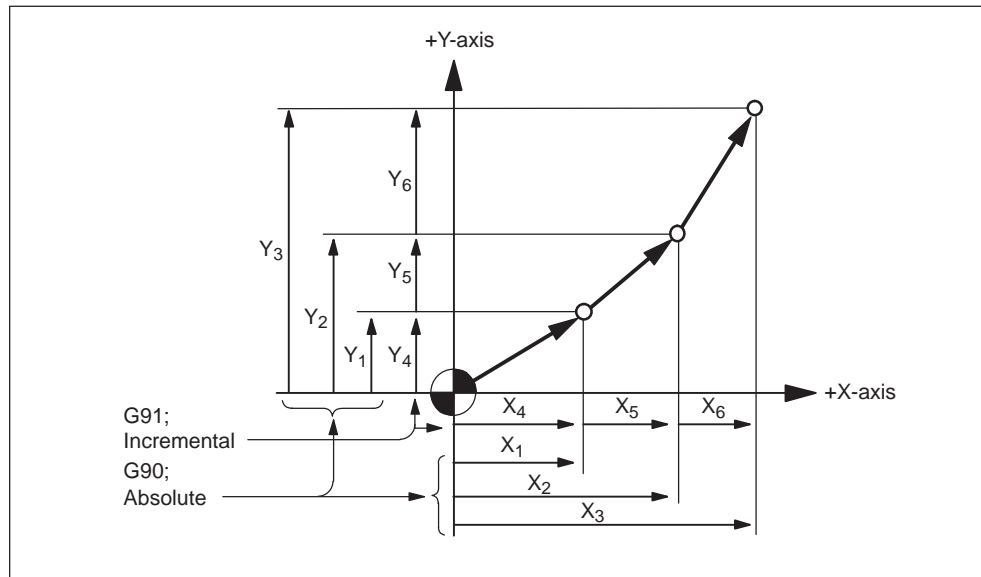


Fig. 3-11 Absolute/incremental commands (G90, G91)

**3.2.2 Inch/Metric input designation (G20, G21)**

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

Table 3-4 Dimension unit selection G codes

G code	Function	Group
G20	Input in “inch” system	06
G21	Input in “mm” system	06

### Command format

G20 and G21 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.

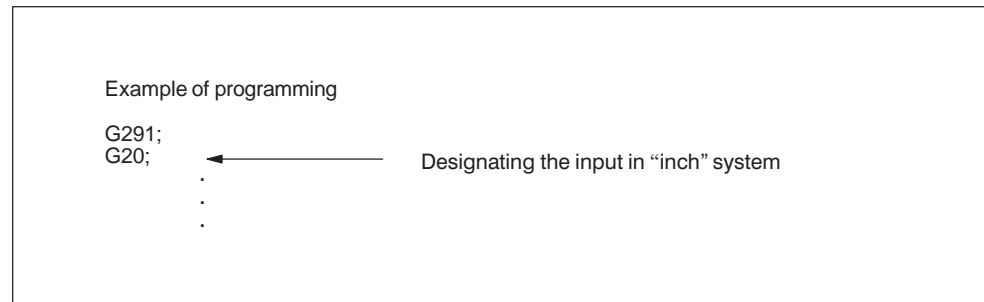


Fig. 3-12

### Supplements to the dimension unit designation commands

- The state when the power is turned ON is determined by MD \$MC\_EXTERN\_GCODE\_RESET\_VALUES[5].
- On switchover, the zero offset values are converted completely.
- If the dimension unit system should be switched over during the execution of a program, the following processing must be accomplished in advance.
  - If a workpiece coordinate system (G54 to G59) is being used, return it to the base coordinate system.
  - Cancel all tool offsets (G41 to G48).
- After switching over the dimension unit system between G20 and G21, the following processing must be accomplished.
  - Execute G92 (coordinate system setting) for all axes before specifying axis move commands.
- The handwheel and increment weighting are not switched over with G20 and G21. This switchover is initiated by PLC program in this case. The relevant MD is \$MA\_JOG\_INCR\_WEIGHT.

### 3.2 Determining the coordinate value input modes

#### 3.2.3 Scaling (G50, G51)

The shape defined by a part program can be enlarged or reduced according to a required scale. For the scaling processing, the following G codes are used.

Table 3-5 Scaling G codes

G code	Function	Group
G50	Scaling OFF	11
G51	Scaling ON	11

The G50 and G51 blocks must be specified in the manner as indicated above without other commands entered in these blocks. The scaling function which is called by G51 must be canceled by G50. If G51 is specified in the scaling mode, it is disregarded.

#### Format

Two different kinds of scaling can be applied.

#### Scaling along all axes at same magnification rate

G51 X... Y... Z... P... ; Scaling start

G50; Scaling cancel

X, Y, Z: Center coordinate value of scaling (absolute command)

P: Scaling magnification

#### Scaling along each individual axis at a different magnification rate (mirror image)

G51 X... Y... Z... I... J... K... ; Scaling start

G50; Scaling cancel

X, Y, Z: Center coordinate value of scaling (absolute command)

I, J, K: X-, Y-, and Z axis scaling magnification

## Explanations

### Scaling along all axes at same magnification rate

Least input increment of scaling magnification is: 0.001 or 0.00001 depending on the setting of MD \$MN\_EXTERNINCREMENT\_SYSTEM. If P is not specified in the block of scaling (G51X... Y... Z... P... ;), the scaling magnification set to MD \$MC\_WEIGHTING\_FACTOR\_FOR\_SCALE is applied.

The reference point during scaling is always the workpiece zero. It is not possible to program a reference point.

### Programmable mirror image (negative magnification)

Applying a negative magnification value will generate a mirror image.

Each axis scaling (mirror image) needs to be enabled by setting MD \$MC\_AXES\_SCALE\_ENABLE = 1.

Omitting I, J, K within the G51 block activates the default values from the setting data.

3.2 Determining the coordinate value input modes

**Example**

```

_N_0512_MPF;          (part program)
N01 G291;
N10 G17 G90 G00 X0 Y0;  Approach start position
N30 G90 G01 G94 F6000;
N32 M98 P0513;         1) Contour as programmed in subprogram
N34 G51 X0. Y0. I-1000 J1000;  2) Mirror contour around X
N36 M98 P0513;
N38 G51 X0. Y0. I-1000 J-1000; 3) Mirror contour around X and Y
N40 M98 P0513;
N42 G51 X0. Y0. I1000 J-1000; 4) Mirror contour around Y
N44 M98 P0513;
N46 G50;              Deselect scaling and mirroring
N50 G00 X0 Y0
N60 M30
    
```

```

_N_0513_MPF;          (subprogram for 00512)
N01 G291
N10 G90 X10. Y10.;
N20 X50;
N30 Y50;
N40 X10. Y10.;
N50 M99;
    
```

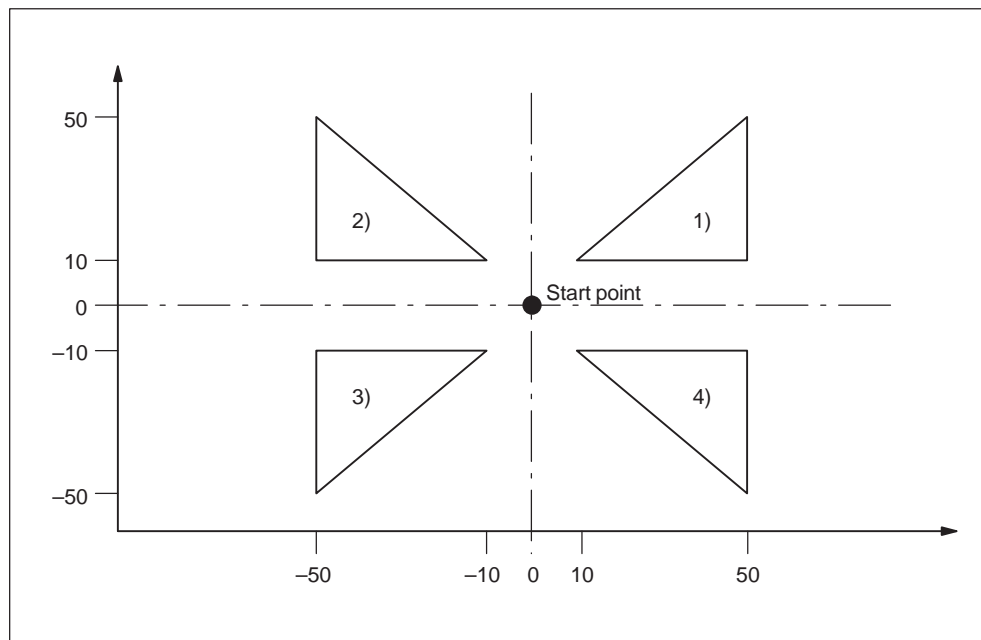


Fig. 3-13 Scaling of each axis, programmable mirror image

**Tool compensation**

This scaling is not applicable to cutter compensation values, tool length offset values, and tool offset values.

**Commands with respect to reference position return and coordinate system**

Do not use G27,G28,G30, or commands related to the coordinate system (G52 to G59,G92) in scaling mode.

3.2 Determining the coordinate value input modes

3.2.4 Programmable mirror image (G50.1, G51.1)

With respect to a programmed axis of symmetry, a mirror image of a programmed contour can be created.

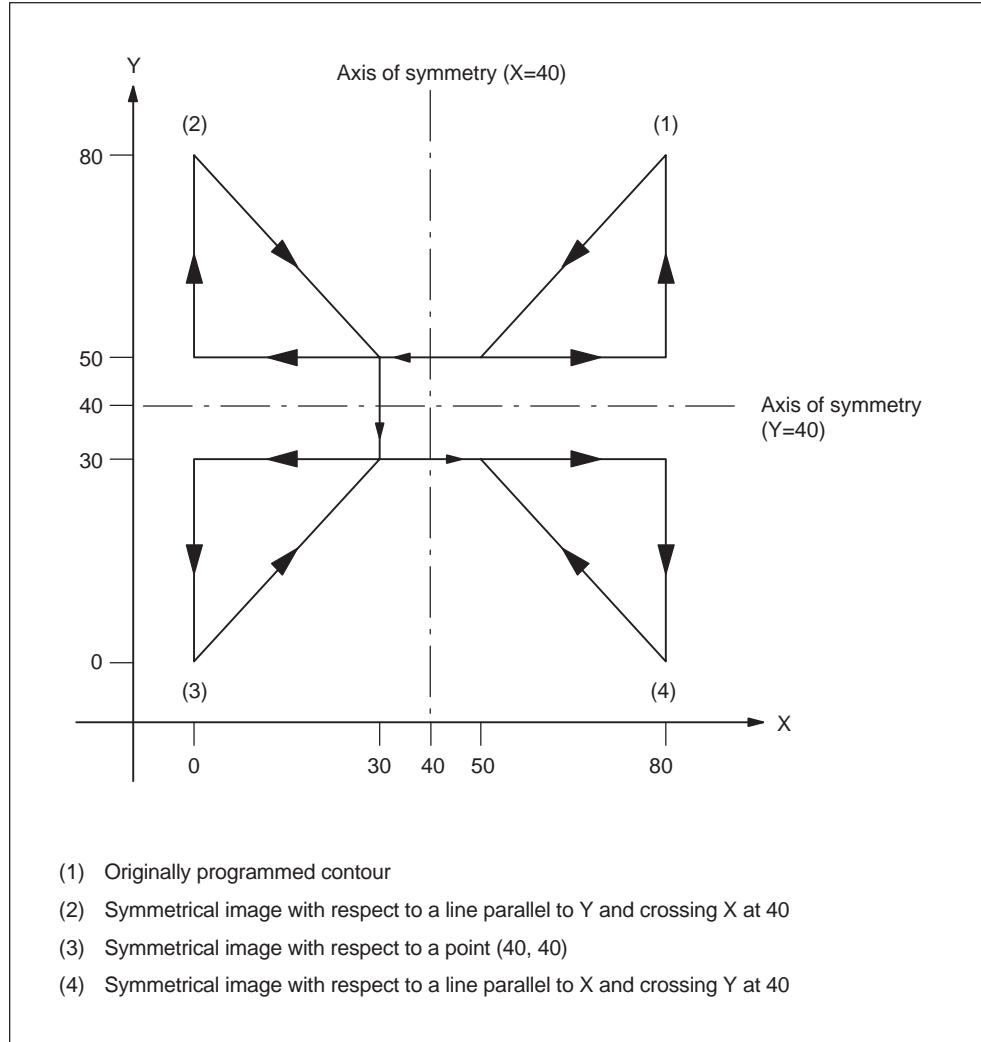


Fig. 3-14 Programmable mirror image



**Format**

```

G51.1 X... Y... Z... ; Creating a programmable image
...           ;
...           ; These blocks describe the contour through which a mirror image
...           ; is created with respect to the axis of symmetry
...           ; specified by G51.1 X... Y... Z... ;
...           ;
G50.1 X... Y... Z... ; Programmable mirror image cancel

```

X, Y, Z :

Position and axis of symmetry for creating a mirror image when specified through G51.1.

**Explanations****Related machine data**

G51.1 uses the channel specific basic frame[1]. Therefore, set MD  
\$MC\_MM\_NUM\_BASE\_FRAMES > = 2.

**Mirror image with respect to single axis in a specified plane**

The following commands are subject to be changed when applying mirror image to one of the axes on a preset plane as described below:

Table 3-6

Command	Explanation
Circular interpolation	G02 and G03 are interchanged
Cutter compensation	G41 and G42 are interchanged
Coordinate rotation	CW and CCW (directions of rotation) are interchanged

**Limitations****Scaling/coordinate system rotation**

Processing proceeds from program mirror image to scaling and coordinate rotation in the stated order. The commands should be specified in this order, and, for cancellation, in the reverse order.

Do not specify G50.1 or G51.1 during scaling or coordinate rotation mode.

**Commands related to reference position return and coordinate system**

Do not use G codes related to reference position return (G27,G28,G30), or commands related to the coordinate system (G52 to G59,G92, etc.) in programmable mirror image mode.

## 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) or a number of spindle revolutions.

In the feed per minute mode (G94) the dwell time unit is seconds [s], while in the feed per revolution mode (G95) the dwell time unit is spindle revolution [rev].

#### Format

G04 X\_; or G04 P\_;

X\_:Specify a time (decimal point permitted)

P\_:Specify a time (decimal point not permitted)

By specifying G04 X\_; or G04 P\_; execution of programmed commands is suspended for the length of time or number of spindle revolutions specified by address X or P.

- The block used to specify dwell must not include commands other than G04 commands.
- The maximum programmable value with address X or P is indicated in the table below.

Table 3-7 Command value range of dwell time (command by X)

Increment system	Command value range	Dwell time unit
IS-B	0.001 to 99999.999	s or rev
IS-C	0.0001 to 9999.9999	s or rev

Table 3-8 Command value range of dwell time (command by P)

Increment system	Command value range	Dwell time unit
IS-B	1 to 99999999	0.001 s or rev
IS-C	1 to 99999999	0.001 s or rev

## 3.4 Cutting feedrate control

### 3.4.1 Exact stop (G09, G61), cutting mode (G64), tapping mode (G63)

Cutting feedrate can be controlled, as indicated in the table below.

Table 3-9

Designation	G code	Validity of G code	Description
Exact stop	G09	Valid only in the block where specified	Deceleration and stop at the end point of the block, position check prior to proceeding to the next block
Exact stop mode	G61	Modal G code, remains effective until G62, G63, or G64 is specified.	Deceleration and stop at the end point of the block, position check prior to proceeding to the next block
Cutting mode	G64	Modal G code, remains effective until G61, G62, or G63 is specified.	No deceleration to the end point of the block prior to proceeding to the next block
Tapping mode	G63	Modal G code, remains effective until G61, G62, or G64 is specified.	No deceleration to the end point of the block prior to proceeding to the next block, feedrate override and feed hold are not effective

#### Format

G09 X... Y... Z... ; Exact stop  
 G61 ; Exact stop mode  
 G64 ; Cutting mode  
 G63 ; Tapping mode

## 3.5 Tool offset functions

### 3.5.1 Tool offset data memory

Since Siemens and ISO Dialect programs are to run alternately on the control, the implementation must use the Siemens tool data memory. The length, geometry and wear are therefore available in each offset memory. In Siemens mode, the offset memory is addressed by T (tool number) and D (tool edge number), abbreviated to T/D number.

In ISO Dialect programs, the offset number is addressed by D (radius) or H (length), referred to below as D/H number.

In order to establish a unique assignment between the D or H number and a T/D number, an element \$TC\_DPH[t,d] has been added to the offset data set. The D/H number of the ISO Dialect is entered in this element.

Table 3-10 Example: tool offset data set

T	D/cutting edge	ISO_H \$TC_DPH	Radius	Length
1	1	10		
1	2	11		
1	3	12		
2	1	13		
2	2	14		
2	3	15		

Setting data \$SSC\_TOOL\_LENGTH\_CONST must contain the value 17 for the assignment of tool length offsets to geometry axes to be independent of the plane selection. Length 1 is then always assigned to the Z axis.

### 3.5.2 Tool length offset (G43, G44, G49)

The tool length offset function adds or subtracts the amount stored in the tool offset data memory to or from the Z coordinate values specified in a program to offset the programmed paths according to the length of a cutting tool.

#### Commands

In the execution of the tool length offset function, addition or subtraction of the offset data is determined by the specified G code and the direction of offset by the H code.

#### G Codes used for tool length offset

The tool length offset function is called by the following G codes.

Table 3-11 G codes used for tool length offset

G code	Function	Group
G43	add	08
G44	Substract	08
G45	Cancel	08

- G43 and G44 are modal and, once executed, they remain valid until canceled by G49. G49 cancels the tool length offset mode. H00 also cancels the tool length offset mode.
- By specifying "G43 (or G44) Z ··· H ··· ;", tool offset amount specified by the H code is added to or subtracted from the specified Z-axis position, and the Z-axis moves to this offset target position. That is, the target position of Z-axis movement specified in the program is offset by the tool offset amount.
- By specifying "(G01) Z ··· ; G43 (or G44) H ··· ;", the Z-axis moves by the distance corresponding to the tool offset amount which is specified by the H code.
- By specifying "G43 (or G44) Z ··· H ··· ; H ··· ;", the Z-axis moves by the distance which is equivalent to the difference between the previous tool offset amount and the new tool offset amount.

---

**Note**

G43, G44, and G49 can be specified only in the mode called by the G code (G00, G01) in 01 group. If they are specified in other modes such as G02 or G03 mode, an error occurs.

---

3.5 Tool offset functions

**H code for designating the direction of offset**

The direction of offset is determined by the sign of tool offset amount, specified by an H code, and a G code.

Table 3-12 Sign of tool offset amount and direction of offset

	Sign of tool offset amount (H code)	
	Positive	Negative
G43	Offset in the positive direction	Offset in the negative direction
G44	Offset in the negative direction	Offset in the positive direction

Example of programming

```

H10 ..... Offset amount -3.0
H11 ..... Offset amount 4.0

                Position data display
                including offset amount
                (Z-axis only)

N101 G92 Z0;           0.000
N102 G90 G00 X1.0 Y2.0; 0.000
N103 G43 Z-20. H10;    -23.000
N104 G01 Z-30. F1000;  -33.000
N105 G00 Z0 H00;      0.000
.
.
.

N201 G00 X-2.0 Y-2.0;
N202 G44 Z-30 H11;    -34.000
N203 G01 Z-40 F1000; -44.000
N204 G00 Z0 H00;      0.000
    
```

Fig. 3-15 Tool position offset function

- Related Machine data:  
\$MC\_TOOL\_CORR\_MOVE\_MODE defines whether the compensation is applied in the block containing the selection or the next time the axis is programmed.

\$MC\_CUTTING\_EDGE\_DEFAULT = 0 defines that no tool length compensation is active initially on a tool change.

\$MC\_AUXFU\_T\_SYNC\_TYPE is used to define whether the output to PLC takes place during or after the movement.

\$MC\_RESET\_MODE\_MASK, bit 6 can be used to activate tool length compensation beyond a reset.

- It is possible to call up the cutter compensation function in the tool length offset mode.
- It is not allowed to specify G43, G44, or G49 in a canned cycle mode.
- G43, G44, and G49 can be specified only in the G00 or G01 mode. Designation of these G codes in the G02 or G03 mode is not allowed.

### Tool length compensation in multiple axes

Tool length offsets can be activated on multiple axes. However, it is not possible to display the resulting tool length compensation.

### 3.5.3 Cutter radius compensation (G40, G41, G42)

The cutter radius compensation function automatically offsets the programmed tool paths by specifying the radius of the cutting tool to be used. The distance to be offset (radius of cutting tool) can be stored to the tool offset data memory by using the NC operation panel. Existing tool offsets can be overwritten using a G10 command, however, new tool offsets cannot be created by G10.

In a program, the offset data are called up by specifying the number of the tool offset data memory using a D code.

### Commands

To call up the cutter radius compensation function, the following G codes are used.

Table 3-13 G codes used to call up the cutter radius compensation function

G code	Function	Group
G40	Tool radius offset C mode cancel	07
G41	Tool radius offset C (offset to the left)	07
G42	Tool radius offset C (offset to the right)	07

3.5 Tool offset functions

The cutter radius compensation function is called up by the execution of G41 or G42 and canceled by G40. Direction of offset is determined by the designated G code (G41, G42) and the offset amount is selected by the D code which is specified with the G code designated to call up the tool radius offset mode. When the power is turned ON, the G40 mode is set.

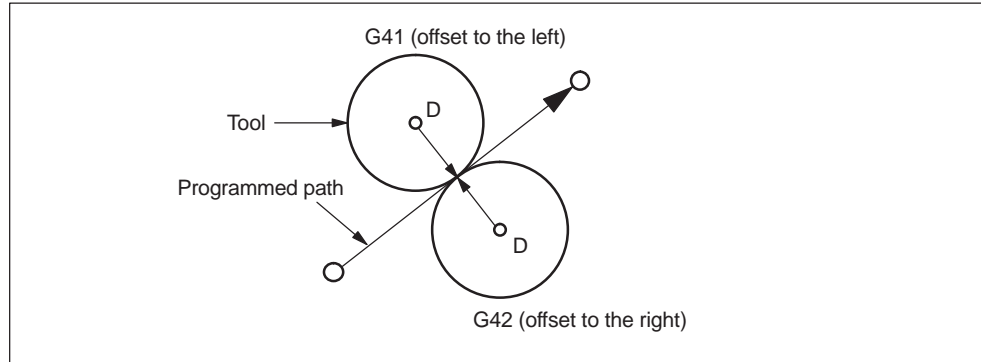


Fig. 3-16 Cutter radius compensation

- If a negative value is set in the tool offset data memory specified by the D code, the offset direction is reversed. The D code must be specified with G41 or G42 in the same block or in a preceding block. If D00 is specified, it specifies the tool radius of "0".
- The tool radius offset plane is selected by the designation of G17, G18, or G19. The G code used to select the plane must be specified with G41 or G42 in the same block or in a block preceding the G41 or G42 block.

Table 3-14 Plane selection G codes

G code	Function	Group
G17	XY plane selection	02
G18	ZX plane selection	02
G19	YZ plane selection	02

- It is not allowed to change the selected plane in the offset mode. If a plane selection G code is specified in the offset mode, an alarm occurs.

**Start-up of cutter compensation**

Since the offset start-up is executed with the offset taken into account, the G code in 01-group must be either G00 or G01. If a G code other than G00 or G01 is specified, an alarm occurs. If the offset starts in the G00 mode, the axes move to the offset point at their individual rapid traverse rates. Therefore, be aware of possible interference of a cutting tool with the workpiece.

There are two types of start-up such as start-up at inside corner and start-up at outside corner.



### Blocks not including axis move commands in the offset mode

In the tool radius offset mode, the NC generates the tool paths by buffering the data of two blocks. If a block not including axis move commands is read, the NC reads one more block to generate the offset tool paths. Designation of such a block which does not include axis move commands is allowed in the tool radius offset mode for up to two consecutive blocks.

After the designation of G41, there must not be three or more consecutive blocks that do not include the movement commands of the axes in the offset plane.

### Consecutive three or more blocks not including axis move commands

If three or more blocks not containing axis move commands in the offset plane are given consecutively, the cutting tool is moved to the position offset normally by the specified offset amount at the end point of the block immediately preceding such blocks.

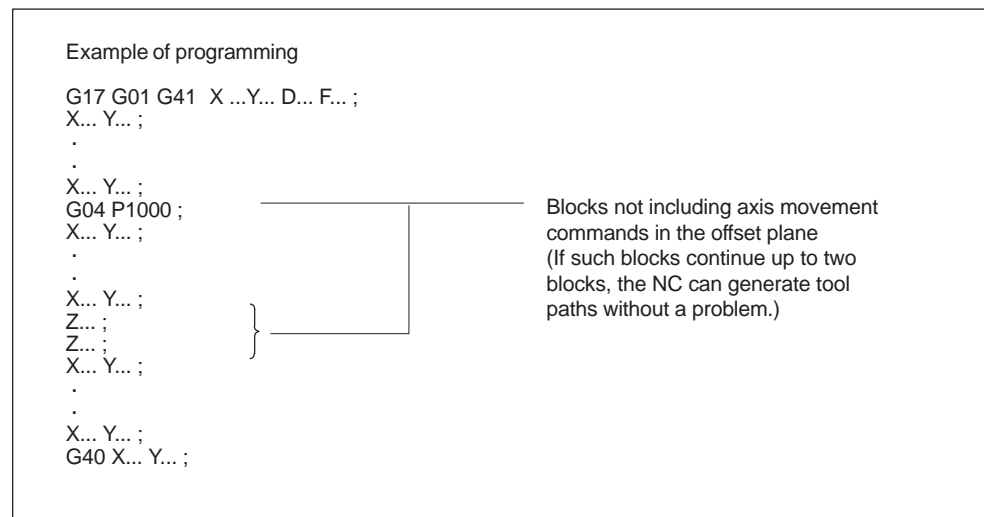


Fig. 3-17

3.5 Tool offset functions

**Switching the G41 and G42 in the cutter compensation mode**

The direction of offset (left side and right side) can be directly switched without canceling the offset mode.

The offset direction is switched at the start and end of the block in which the switching of the offset direction is specified.

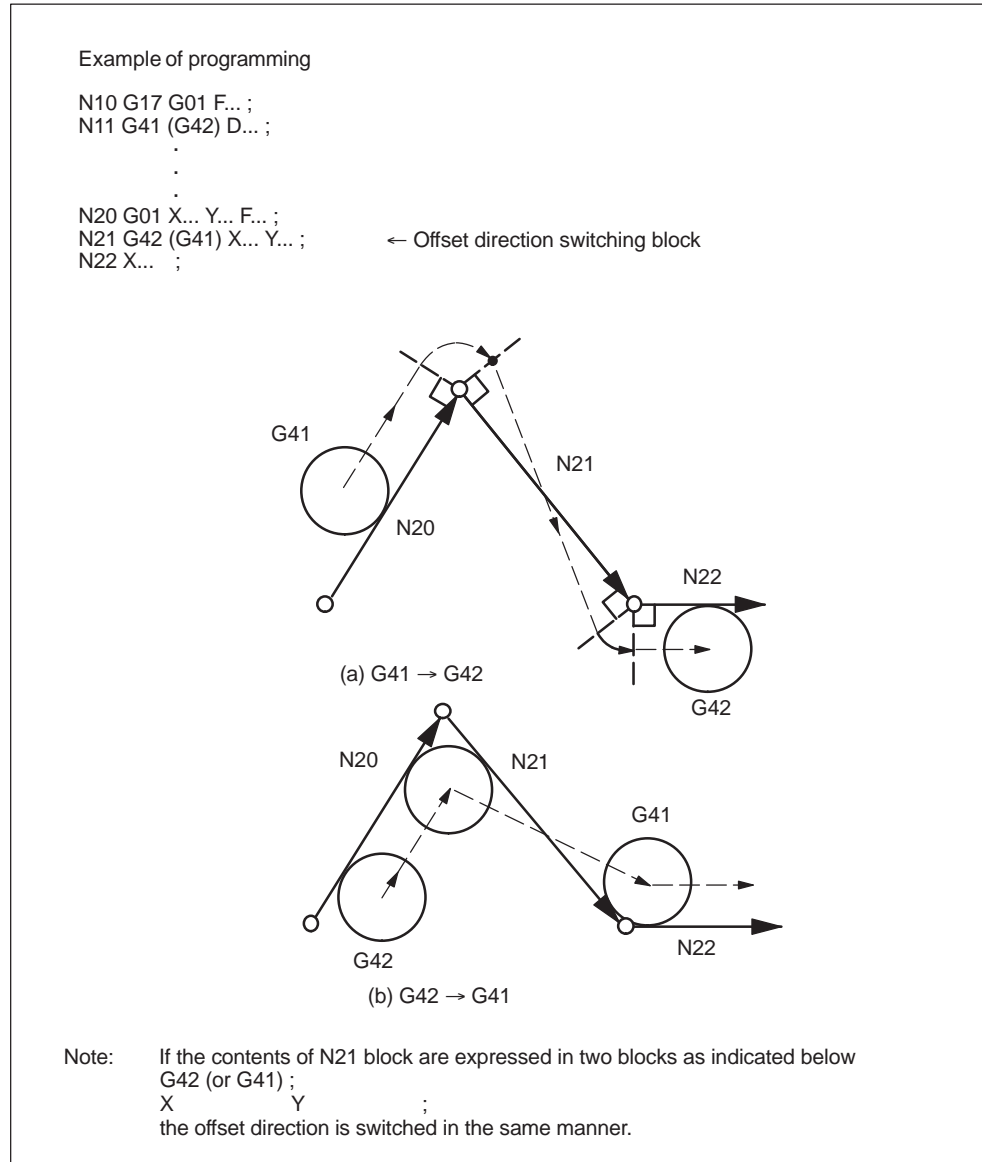


Fig. 3-18 Switching the offset direction at the start and end of the block

### Canceling the offset mode

There are two types of offset mode cancellation methods which can be selected by the MD setting.

#### 1. Type A:

Offset mode cancellation movement is not executed in the G40 block if no axis move commands are given. The offset mode is canceled by the first axis move command given in a block specified following the G40 block. Axis move commands should be specified with G40 in the same block.

#### 2. Type B:

Offset mode cancellation movement is executed in the G40 block even if there are no axis move commands given. The cutting tool moves normally to the offset position at the end point of the block immediately before the G40 block. Since G40 calls up offset mode cancellation axis movements, it must be specified in the G00 or G01 mode. If it is specified in a mode other than G00 or G01, an alarm occurs.

### Canceling the offset mode at inside corner (smaller than 180 )

#### Straight-line to straight-line

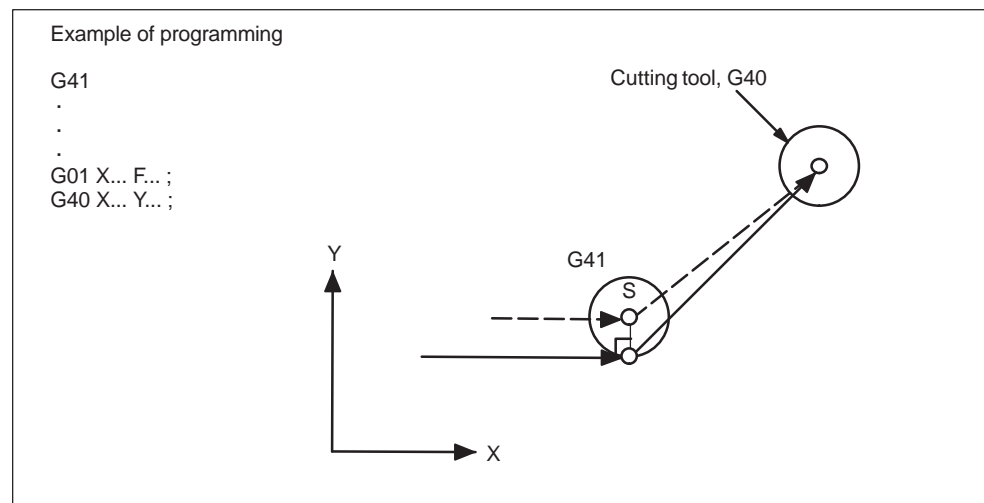


Fig. 3-19 Canceling the offset mode at inside corner (straight-line to straight-line)

3.5 Tool offset functions

**Arc to straight-line**

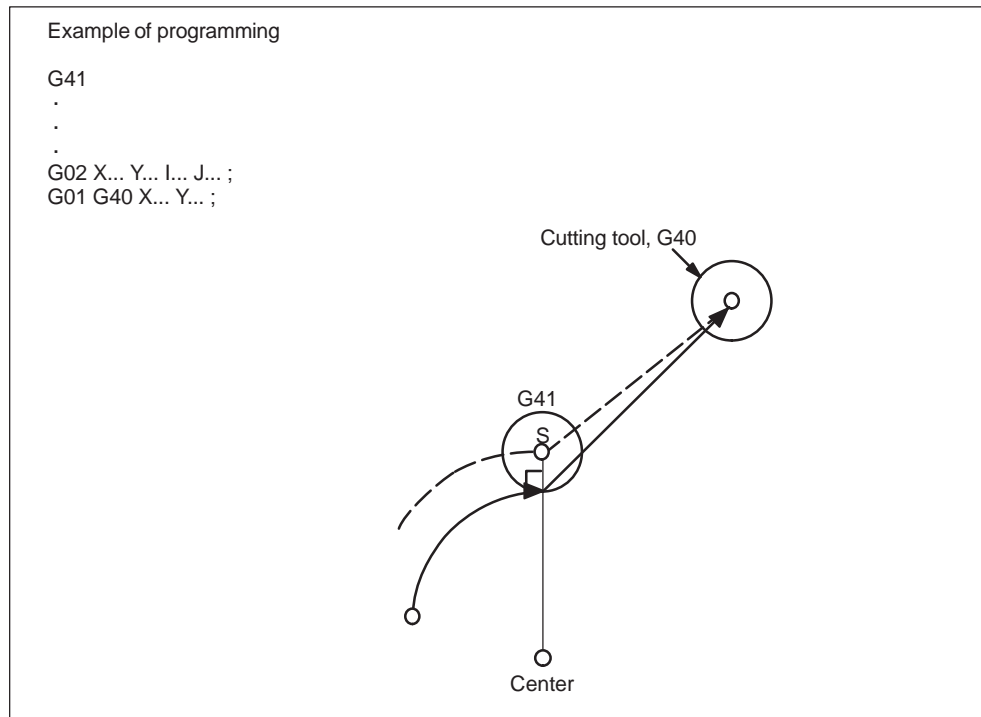


Fig. 3-20 Canceling the offset mode at inside corner (arc to straight-line)

**3.5.4 Collision monitoring**

**Activation by NC program**

Although the collision monitoring function is available only in Siemens mode, it can also be applied within the ISO dialect mode. However, activation and deactivation needs to be carried out in Siemens mode.

```
G290;      Activate Siemens mode
CDON;     Activate bottleneck detection
G291;     Activate ISO dialect mode
...
...
G290;     Activate Siemens mode
CDOF;     Deactivate bottleneck detection
G291;     Activate ISO dialect mode
```

**Activation by MD setting**

```
MD 20150 $MC_GCODE_RESET_VALUES[22] = 2: CDON (modal active)
MD 20150 $MC_GCODE_RESET_VALUES[22] = 1: CDOF (modal inactive)
```

## Function

When CDON (Collision Detection ON) and tool radius compensation are active, the control monitors the tool paths with Look Ahead contour calculation. This Look Ahead function allows possible collisions to be detected in advance and permits the control to actively avoid them.

When collision detection is off (CDOF), a search is made at inside corners in the previous traversing block (and if necessary in blocks further back) for a common intersection point for the current block. If no intersection is found with this method, an error is generated.

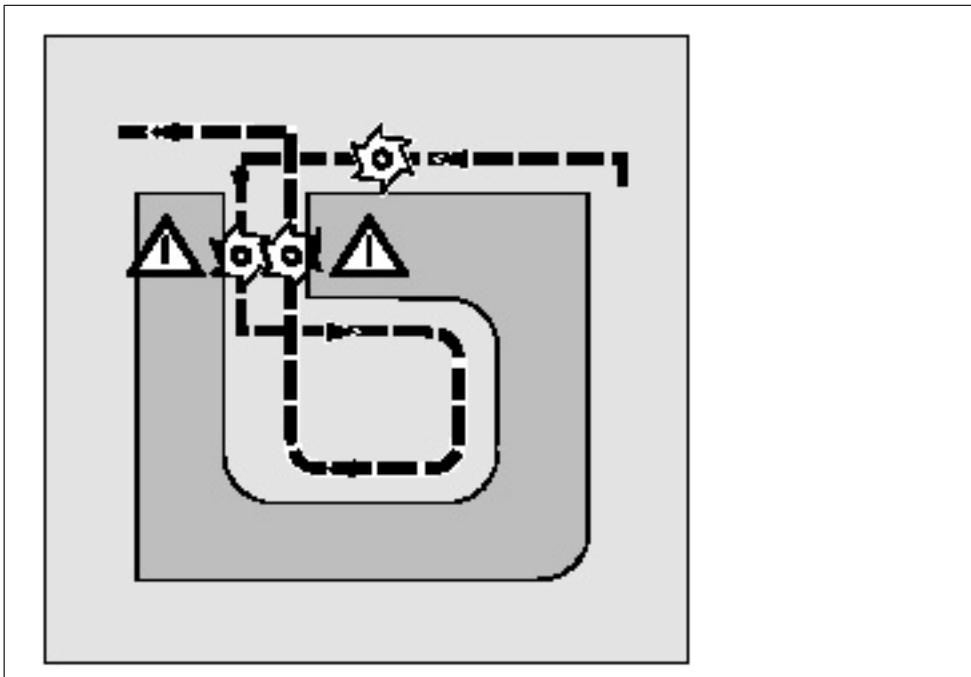


Fig. 3-21

CDOF helps prevent the incorrect detection of bottlenecks, e.g. due to missing information which is not available in the NC program.

The number of NC blocks monitored can be defined in the machine data (see machine manufacturer).

## Examples

The following are some examples of critical machining situations which can be detected by the control and compensated for by modifying the tool paths.

In order to prevent program stops, you should always select the tool with the widest radius from all of the tools used when testing the program.

In each of the following examples a tool with too wide a radius was selected for machining the contour.

## 3.5 Tool offset functions

**Bottleneck detection**

Since the tool radius selected is too wide to machine this inside contour, the "bottleneck" is bypassed. An alarm is output.

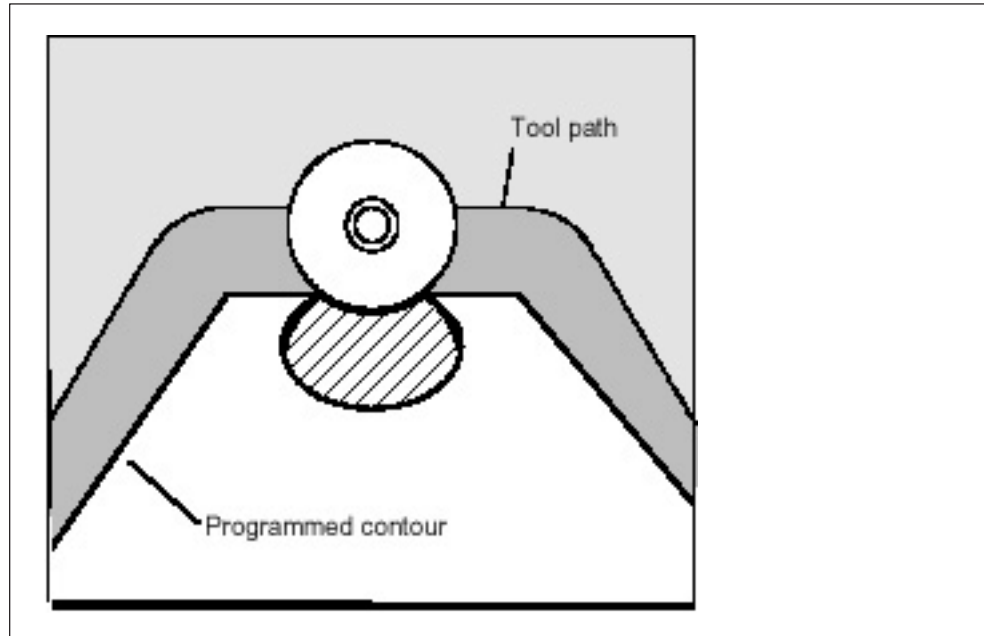


Fig. 3-22

**Contour path shorter than tool radius**

The tool travels round the workpiece corner on a transition circle and then continues to follow the programmed contour exactly.

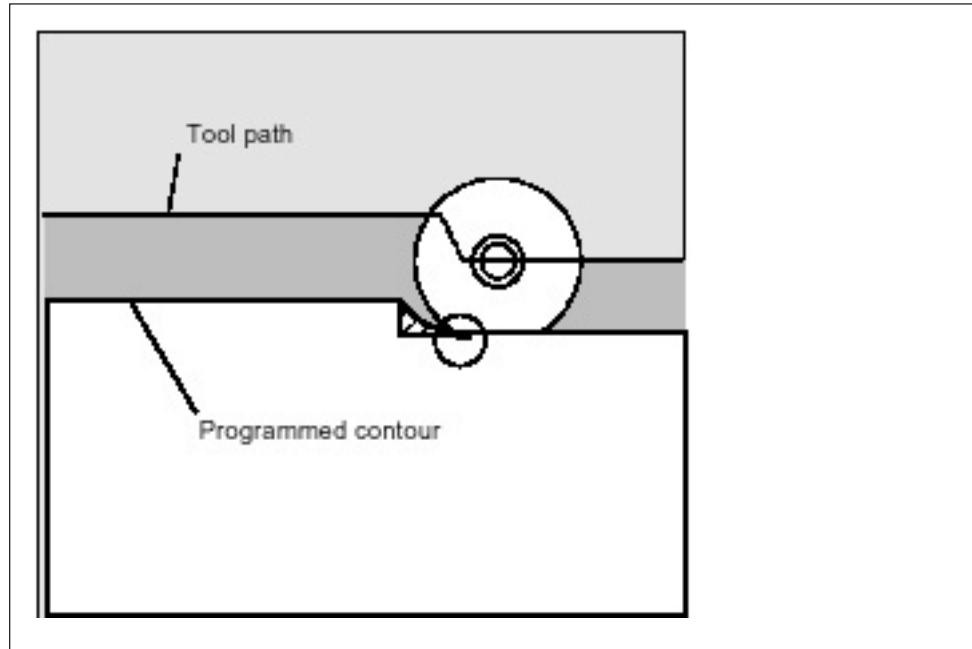


Fig. 3-23

## 3.5 Tool offset functions

**Tool radius too wide for inside machining**

In such cases, machining of the contours is performed only as far as is possible without causing damage to the contour.

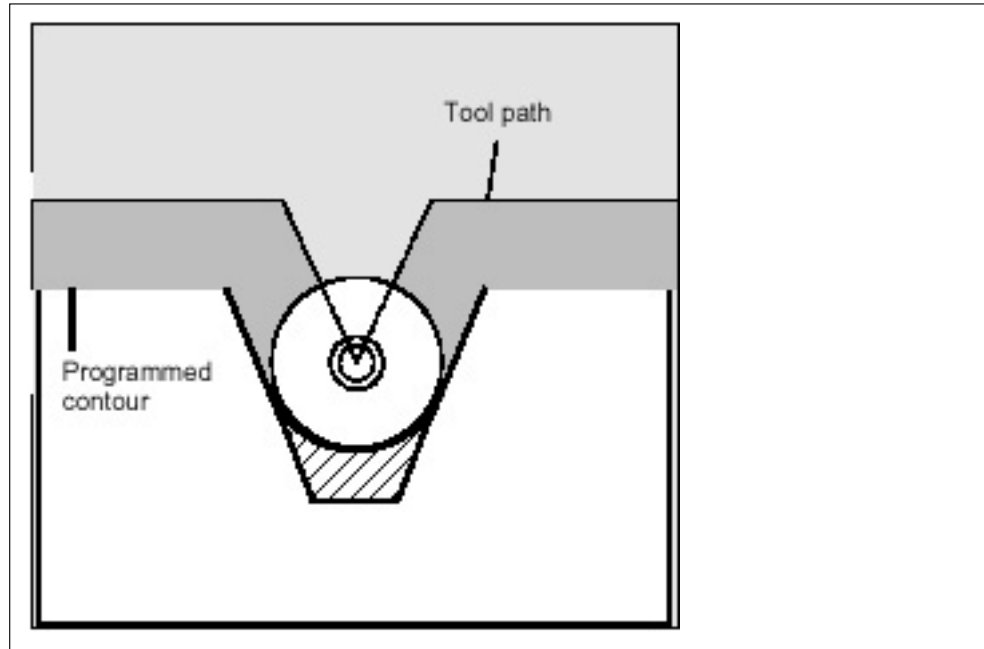


Fig. 3-24



## 3.6 S, T, M, and B functions

### 3.6.1 Spindle function (S function)

A spindle speed can be directly specified by entering a 5-digit number following address S (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.

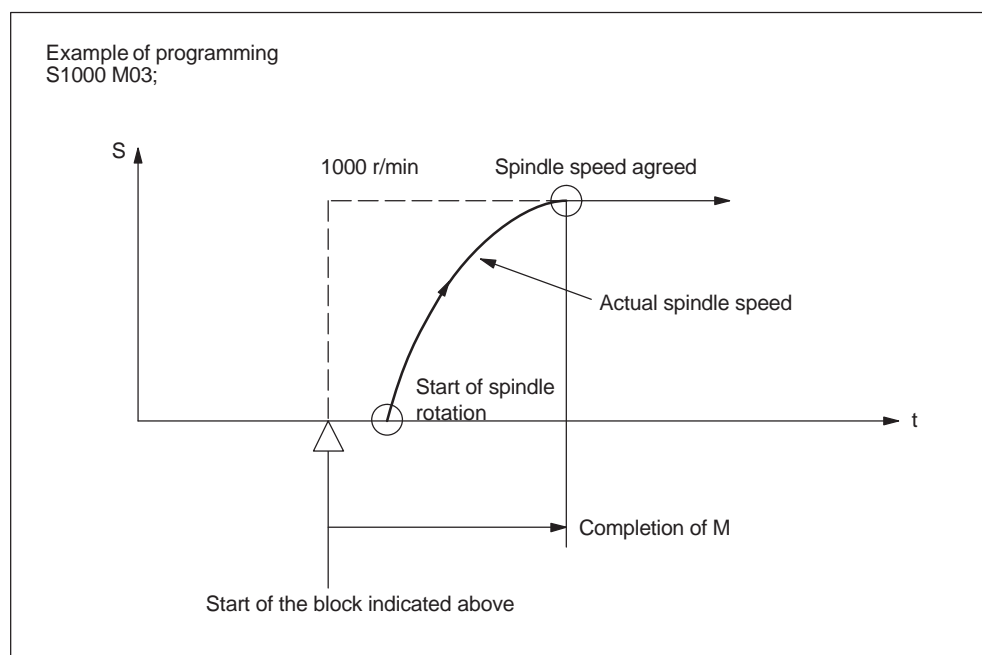


Fig. 3-25 Spindle speed command

- 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.
- If a spindle speed is changed while the spindle is rotating by the execution of M03 or M04, pay attention to the selected spindle speed gear range. For details, refer to the instruction manuals published by the machine tool builder.
- 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.

---

### 3.6 S, T, M, and B functions

#### 3.6.2 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.

#### 3.6.3 Miscellaneous function (M function)

The miscellaneous function is specified by a maximum of a three-digit number (M□□□) following address M. With the exception of specific M codes, the functions of M00 to M89 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.

#### 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 outputs 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.

**3.6.4 Internally processed M codes**

M codes in the range of M90 to M99 are processed by the NC.

Table 3-15 Internally processed M codes

M code	Function
M98	Subprogram call
M99	End of subprogram

**3.6.5 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.

## 3.6 S, T, M, and B functions

Table 3-16 Other general M codes

M code	Function	Remarks
M03	Spindle start, forward direction	Generally, M state between M03 and M04 cannot be switched directly. To change the M code state, execute M05 once.
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.

**Second miscellaneous function (B function)**

B functions are output to the PLC as H auxiliary functions with address extension H1=.

Example: B1234 is output as H1=1234.

# 4

## Enhanced Level Commands

Chapter 4 describes the program support functions, automation support functions, and macro programs.

4.1	Program support functions (1)	4-86
4.1.1	Canned cycles (G73 to G89)	4-86
4.1.2	High-speed peck drilling cycle (G73)	4-93
4.1.3	Fine boring cycle (G76)	4-94
4.1.4	Drilling cycle, spot drilling (G81)	4-98
4.1.5	Drilling cycle, counter boring cycle (G82)	4-100
4.1.6	Peck drilling cycle (G83)	4-102
4.1.7	Boring cycle (G85)	4-104
4.1.8	Boring cycle (G86)	4-106
4.1.9	Boring cycle, back boring cycle (G87)	4-108
4.1.10	Boring cycle (G89)	4-111
4.1.11	Rigid tapping cycle (G84)	4-113
4.1.12	Left-handed rigid tapping cycle (G74)	4-116
4.1.13	Peck tapping cycle (G84 or G74)	4-119
4.1.14	Canned cycle cancel (G80)	4-122
4.1.15	Program example using tool length offset and canned cycles	4-123
4.2	Programmable data input (G10)	4-125
4.2.1	Changing of tool offset value	4-125
4.2.2	Setting the workpiece coordinate system shift data	4-125
4.3	Subprogram call up function (M98, M99)	4-126
4.4	Polar coordinate command (G15, G16)	4-127
4.5	Polar coordinate interpolation (G12.1, G13.1)	4-128
4.6	Cylindrical interpolation (G07.1)	4-130
4.7	Program support functions (2)	4-135
4.7.1	Working area limitation (G22, G23) (Under development)	4-135
4.7.2	Chamfering and corner rounding commands	4-137
4.8	Automating support functions	4-141
4.8.1	Skip function (G31)	4-141
4.8.2	Multistage skip (G31, P1 – P4)	4-144
4.8.3	Program interrupt function (M96, M97)	4-145
4.8.4	Tool life control function	4-147
4.9	Macroprograms	4-148
4.9.1	Differences from subprograms	4-148
4.9.2	Macroprogram call (G65, G66, G67)	4-148
4.10	Additional functions	4-155
4.10.1	Figure copy (G72.1, G72.2)	4-155

## 4.1 Program support functions (1)

### 4.1.1 Canned cycles (G73 to G89)

By using canned cycles, it is made easier for the programmer to create programs. By means of canned cycles, machining operations frequently used can be determined in a single block through a G function. Normally more than one block is required when operating without canned cycles. Using canned cycles can also shorten the program in order to save memory.

The functionality of the ISO Dialect cycles is implemented in the standard Siemens cycles. A shell cycle is called from the ISO Dialect program. All addresses programmed in the block are passed to this shell cycle in the form of system variables. The shell cycle matches the data to the standard Siemens cycle and calls it by name.

#### Procedure for cycle call via G command

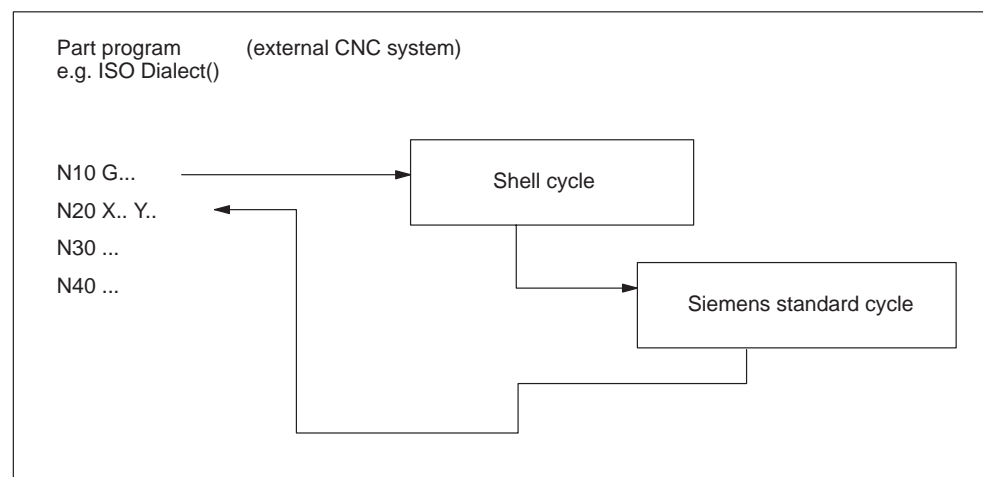


Fig. 4-1 General cycle call in ISO Dialect mode

#### Cycle parameters

Various cycle parameters in channel-specific GUD (Global User Data) must be initialized for the machining cycles. The names and meanings of the GUD are listed in the tables below.

## 4.1 Program support functions (1)

Table 4-1 GUD7 for programmed cycle values (ISO dialect program data)

GUD	Description/use	Cycle
<b>Real values</b>		
_ZFPR[0]	Initial plane (current position on 1st call with G.), retraction position active on G98	381M, 383M, 384M, 387M
_ZFPR[1]	Reference plane, retraction position active on G99 (retraction is only possible to initial position with G87).	381M, 383M, 384M, 387M
_ZFPR[2]	Final drilling depth	381M, 383M, 384M, 387M
_ZFPR[3]	Retraction position, depending on G98/G99 (initial plane/R plane)	381M, 383M, 384M, 387M
_ZFPR[4]	Feed rate for drilling	381M, 383M, 384M, 387M
_ZFPR[5]	Dwell time at final depth (G82/G89/G76/G87)	381M, 384M, 387M
_ZFPR[6]	1st drilling depth (single drilling depth), incremental (G73/G83)	383M
_ZFPR[7]	1st drilling depth, absolute (G73/G83)	383M
_ZFPR[8]	Lift-off/in-feed distance (G76)	387M
_ZFPR[9]	Speed for tapping (G74/G84)	384M
<b>Integer values</b>		
_ZFPI[0]	Current G code of ISO Dialect0 drilling cycle	381M, 383M, 384M
_ZFPI[1]	M function for spindle start (M3, M4) after spindle stop	381M, 384M

Table 4-2 GUD7 for cycle setting data (ISO dialect setting data)

GUD	Description/use	Cycle
<b>Real values</b>		
_ZSFR[0]	Safety clearance to reference plane	381M, 383M
_ZSFR[1]	Retraction amount for chip-breaking (G73)	383M
_ZSFR[2]	Angle offset for oriented spindle stop, tool must be oriented in +X direction (G76) Retraction direction: -X G17 plane XY -Z G18 plane ZX -Y G19 plane YZ	387M
<b>Integer values</b>		
_ZSFI[0]	0=Drilling axis is the axis perpendicular to the selected plane (default) 1=Drilling axis always "Z"	381M, 383M, 384M, 387M
_ZSFI[1]	0= Rigid tapping 1= Tapping with compensating chuck 2= Deep hole tapping with chip-breaking 3= Deep hole tapping with swarf removal	384M, 387M
_ZSFI[2]	Retraction speed factor (1-200%) for tapping (G74/G84)	384M
_ZSFI[3]	Polar coordinates 0 = OFF 1 = ON	381M, 383M, 384M, 387M

## 4.1 Program support functions (1)

For calling normal hole-machining canned cycles, the following G codes are used.

Table 4-3 GUD7 for cycle setting data (ISO dialect setting data)

G code	Drilling (-Z direction)	Processing at hole bottom	Retraction (+Z direction)	Applications
G73	Intermittent feed (dwell at each peck feed possible)	—	Rapid traverse	High-speed deep hole drilling
G74	Cutting feed	Spindle rotation in the reverse direc- tion after dwell	Cutting feed → Dwell → Spindle reverse rotation	Reverse (left- hand) tapping
G76	Cutting feed	Spindle indexing → Shift	Rapid traverse → Shift, Spindle start	Boring
G80	—	—	—	Cancel
G81	Cutting feed	—	Rapid traverse	Drilling, spot drilling
G82	Cutting feed	Dwell	Rapid traverse	Drilling , counter boring
G83	Intermittent feed	—	Rapid traverse	Deep hole drilling (peck drilling)
G84	Cutting feed	Spindle start in the reverse direc- tion after dwell	Cutting feed → Dwell → Spindle reverse rotation	Tapping
G85	Cutting feed	—	Cutting feed	Boring
G86	Cutting feed	Spindle stop	Rapid traverse → Spindle start	Boring
G87	Cutting feed	Spindle CW	Rapid traverse	Back boring
G89	Cutting feed	Dwell	Cutting feed	Boring



## Explanations

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

- Operation 1  
Positioning in the XY plane using cutting feed or rapid traverse rate
- Operation 2  
Rapid traverse movement to level R
- Operation 3  
Machining to drilling depth Z
- Operation 4  
Operation at hole bottom
- Operation 5  
Retraction to R level at cutting feed or rapid traverse rate
- Operation 6  
Rapid retraction to positioning plane XY at rapid traverse rate

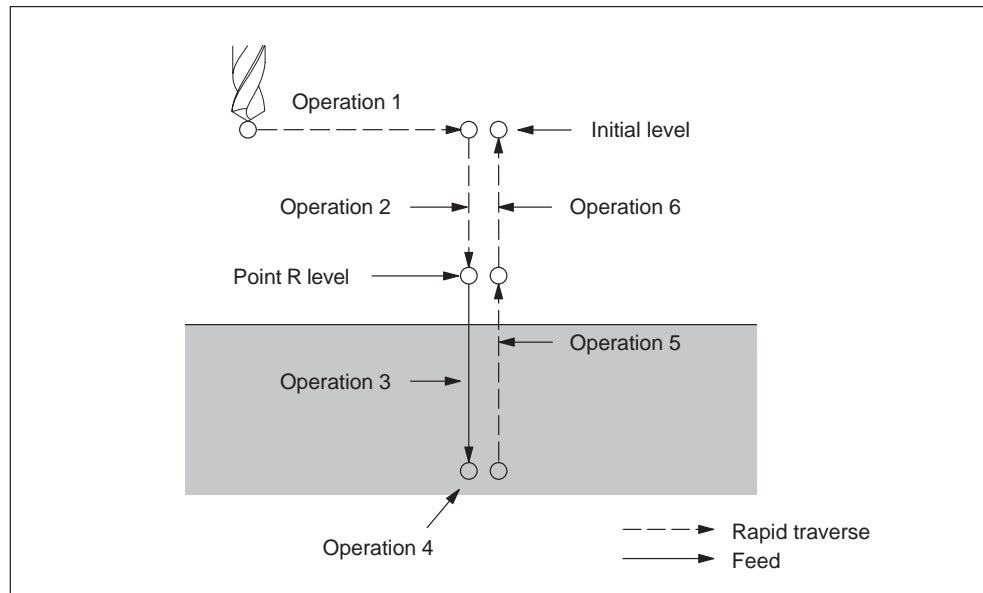


Fig. 4-2 Canned cycle operation sequence

In this chapter the term drilling will only be used to refer to operations implemented with canned cycles, although canned cycles encompass tapping and boring cycles as well as drilling cycles.

## 4.1 Program support functions (1)

**Plane definition**

In the drilling cycles, it is generally assumed that the current workpiece coordinate system in which the machining operation is to be performed is defined by selecting plane G17, G18, or G19 and activating a programmable workpiece offset. The drilling axis is always the applicate of this coordinate system.

A tool length compensation must be selected before the cycle is called. Its effect is always perpendicular to the selected plane. and remains active even after the end of the cycle.

Table 4-4 Positioning plane and drilling axis

G code	Positioning plane	Drilling axis
G17	Xp–Yp plane	Zp
G18	Zp–Xp plane	Yp
G19	Yp–Zp plane	Xp

Xp: X axis or an axis parallel to the X axis

Yp: Y axis or an axis parallel to the Y axis

Zp: Z axis or an axis parallel to the Z axis

**Note**

By applying GUD7 setting data `_ZSFI[0]`, it can be decided whether the Z axis should always be used as the drilling axis. The Z axis always represents the drilling axis whenever `_ZSFI[0]` equals 1.

**Canned cycle execution**

The execution of canned cycles is determined as follows:

1. Cycle call  
G73, 74, 76, 81 through to 89  
depending on the desired machining
2. Data format G90/91

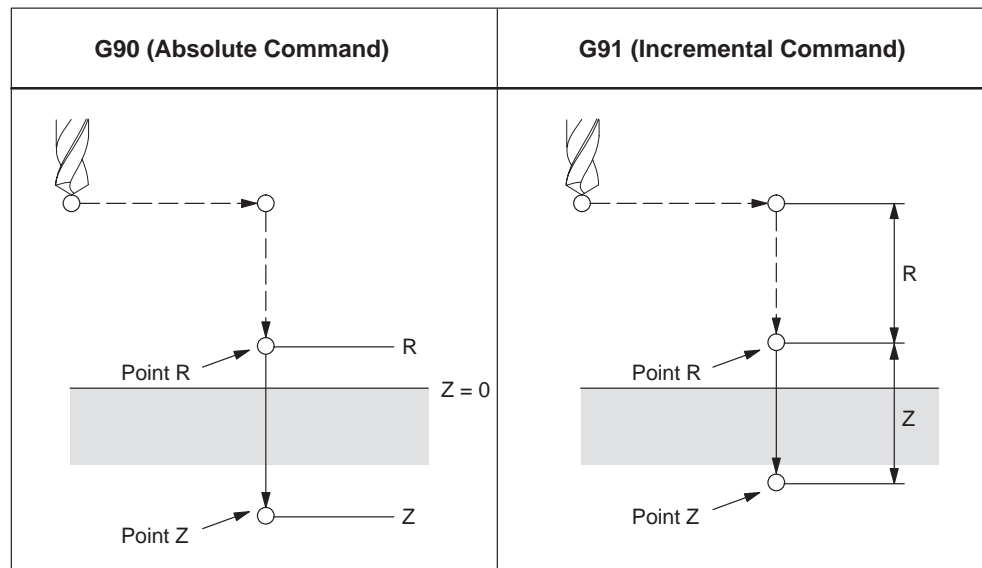


Fig. 4-3 Absolute / incremental command G90/G91

### 3. Drilling mode

G73, G74, G76, and G81 through to G89 are modal G codes and remain effective until canceled. When they are applied, the drilling mode is the current state. The data is retained until modified or canceled, once the drilling data is determined within the drilling mode,

At the beginning of canned cycles, determine all required drilling data. Only determine the data modifications whenever the canned cycles are being carried out.

### 4. Positioning / reference level (G98/G99)

When using canned cycles, the retraction level for the Z axis is determined through G98/99. G98/G99 are modal G codes. G98 is usually set as power-on default.

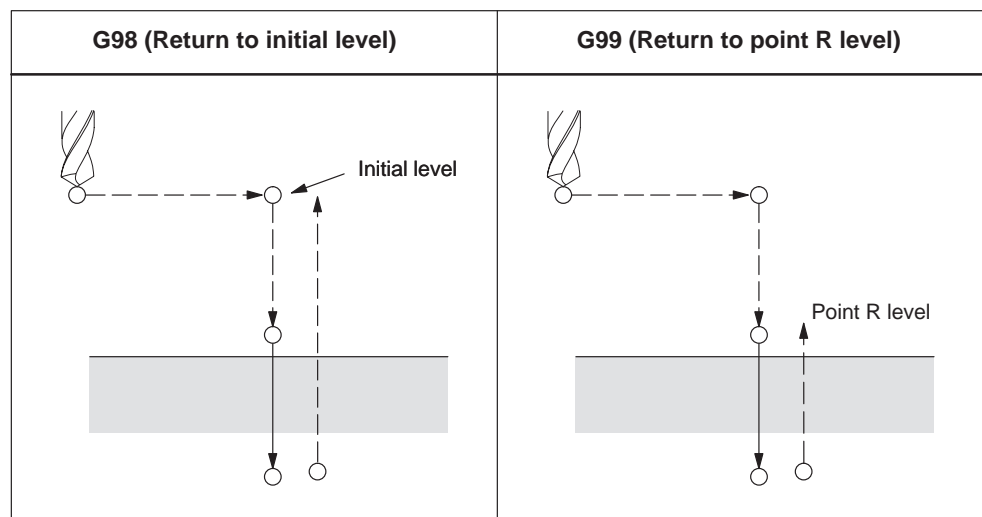


Fig. 4-4 Return point level (G98/G99)

#### 4.1 Program support functions (1)

### 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).

### Comments

A cycle call remains selected until it is cancelled through the G codes G80, G00, G01, G02, or G03, or through another cycle call.

Within the machining cycles the data specified at address Z, R, P, and Q function as self-retaining even after RESET operation. These data can only be changed by reprogramming or are cancelled using the G codes G80, G00, G01, G02, or G03.

### Symbols in figures

Subsequent sections explain the individual canned cycles. Figures in these explanations use the following symbols:





	Positioning (rapid traverse G00)
	Cutting feed (linear interpolation G01)
	Manual feed
M19	Oriented spindle stop (The spindle stops at a fixed rotation position)
	Shift (rapid traverse G00)
P	Dwell

Fig. 4-5 Symbols in figures

### 4.1.2 High-speed peck drilling cycle (G73)

This cycle carries out high-speed peck drilling. It induces intermittent cutting feed to the bottom of a hole. Retract movements enables chip removal.

#### Format

G73 X.. Y... R... Q... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to the bottom of the hole

**R:** Distance from the initial level to R level

**Q:** Depth of cut for each cutting

**F:** Cutting feedrate

**K:** Number of repeats

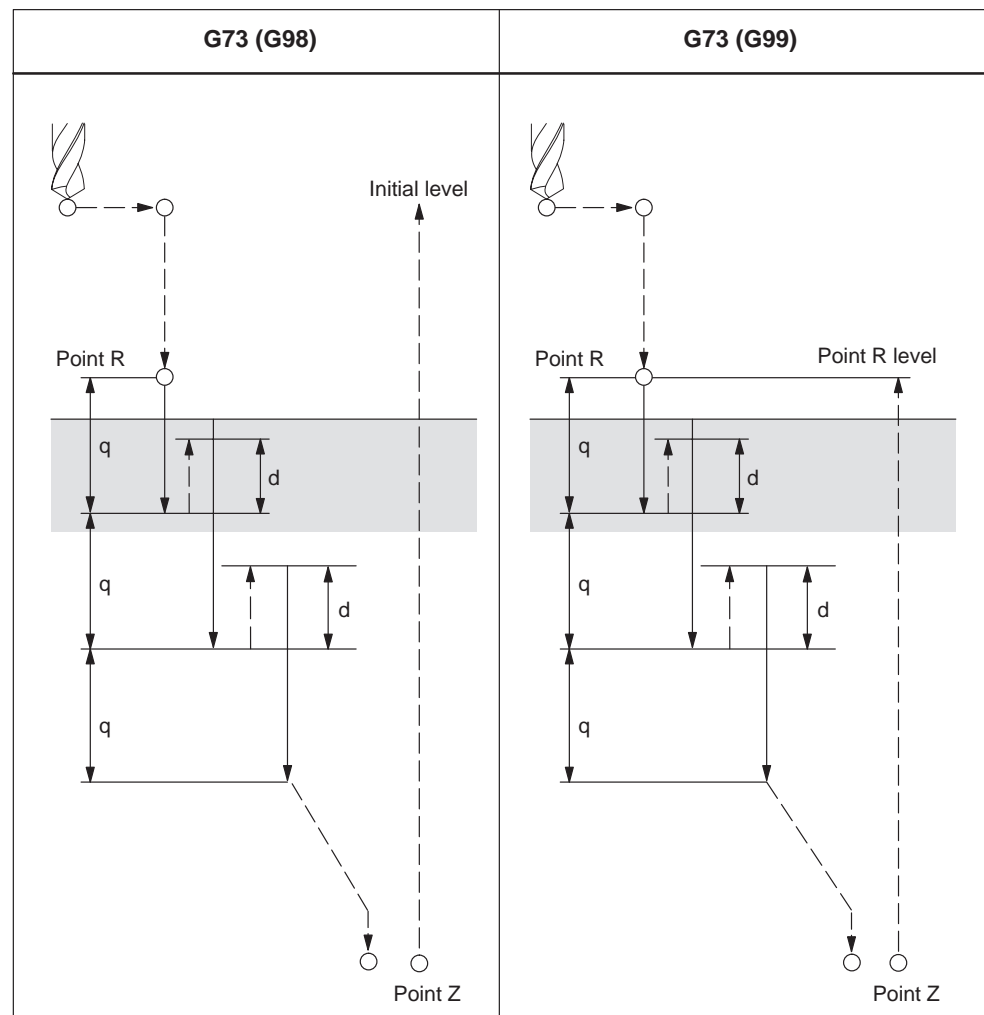


Fig. 4-6 High-speed peck drilling cycle (G73)

## 4.1 Program support functions (1)

**Explanations**

When using cycle G73 retraction movement is performed in rapid traverse after each drilling operation. GUD `_ZSFR[0]` can be used to enter a safety clearance. The retraction amount for chipbreaking (d) is determined through GUD `_ZSFR[1]` as described below:

`_ZSFR[1] > 0` Retraction amount as entered

`_ZSFR[1] 0` Retraction amount is always 1 mm with chipbreaking

Infeed is performed by using depth of cut for each cutting Q, which is incremented by the retraction amount d as of the 2nd infeed.

By means of this drilling cycle a rapid drilling infeed is accomplished. Removal of the drilling chips is facilitated through the retraction movement.

**Example**

```
M3 S1500;      Rotate spindle
G90 G99 G73 X200. Y-150. Z-100. R-100. Q10. F150.;
              Position, drill hole 1, and return to point R.
Y-500.;       Position, drill hole 2, and return to point R.
Y-700.;       Position, drill hole 3, and return to point R.
X950.;        Position, drill hole 4, and return to point R.
Y-500.;       Position, drill hole 5, and return to point R.
G98 Y-700.;   Position, drill hole 6, and return to the initial level.
G80;          Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;           Spindle stop
```

**4.1.3 Fine boring cycle (G76)**

Precise boring of a hole is accomplished through the fine boring cycle.

**Format**

G76 X... Y... R... Q... P... F... K... ;

**X,Y:** Hole position

**Z\_:** Distance from point R to the bottom of the hole

**R\_:** Distance from the initial level to point R level

**Q\_:** Shift amount at the bottom of a hole

**P\_:** Dwell time at the bottom of a hole

**F\_:** Cutting feedrate

**K\_:** Number of repeats

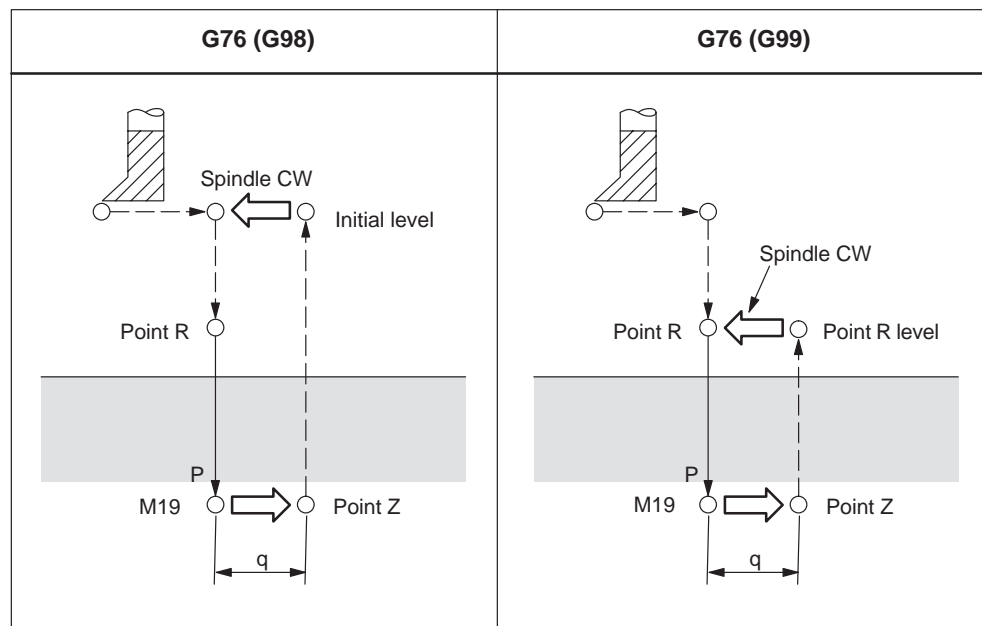


Fig. 4-7 Fine boring cycle (G76)

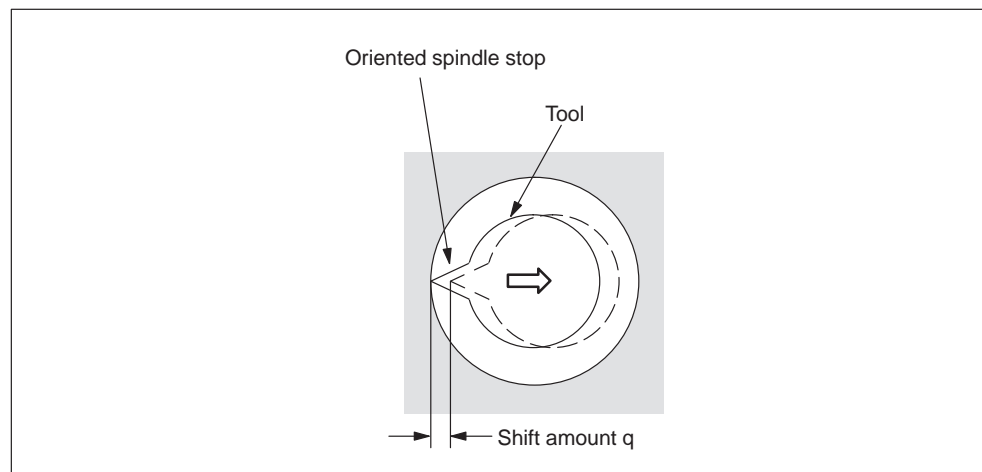


Fig. 4-8

**Warning**

Address Q is a modal value which is retained within canned cycles. Special care has to be taken because it is also used as the depth of cut in cycles G73 and G83.

---

## 4.1 Program support functions (1)

### Explanations

The spindle is stopped at the fixed rotation position when the bottom of the hole has been reached. The tool is then moved in the direction opposite to the tool tip and retracted.

GUD \_ZSFR[0] can be used to enter a safety clearance.

The lift-off path is always with reference to the negative direction of the first geometry axis

G17: lift-off path in -X

G18: lift-off path in -Z

G19: lift-off path in -Y

The angle must be therefore be entered to GUD7 \_ZSFR[2] such that the tool tip points in the positive direction (+) in the defined plane after the spindle stop.

### Limitations

#### Axis switching

The canned cycle must be canceled before the drilling axis can be changed.

#### Boring

Boring is not carried out in a block that does not contain X, Y, Z, R, or any additional axes.

#### Q/R

By all means, specify a positive value at address Q. The sign is ignored if address Q is specified with a negative value. Q equal 0 is set whenever no lift-off amount is programmed. This leads to cycle execution without lift-off.

#### Cancel

G codes of group 01 (G00 to G03) and G76 must not be specified within a single block. Otherwise, G76 is canceled.

#### Tool offset

The tool offsets are ignored in the canned cycle mode.



**Example**

```
M3 S300;      Rotate spindle
G90 G99 G76 X200. Y-150. Z-100. R-100. Q10. P1000 F120.;
              Position, bore hole 1, then return to point R,
              Stop at the bottom of the hole for 1 s.
Y-500.;      Position, drill hole 2, and return to point R.
Y-700.;      Position, drill hole 3, and return to point R.
X950.;       Position, drill hole 4, and return to point R.
Y-500.;      Position, drill hole 5, and return to point R.
G98 Y-700.;  Position, drill hole 6, and return to the initial level.
G80;         Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;         Spindle stop
```

## 4.1 Program support functions (1)

## 4.1.4 Drilling cycle, spot drilling (G81)

Center drilling and spot drilling can be carried out by means of this cycle. After reaching drilling depth Z retraction movement is immediately performed in rapid traverse rate.

## Format

G81 X... Y... R... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to the bottom of the hole

**R:** Distance from the initial level to R level

**F:** Cutting feedrate

**K:** Number of repeats

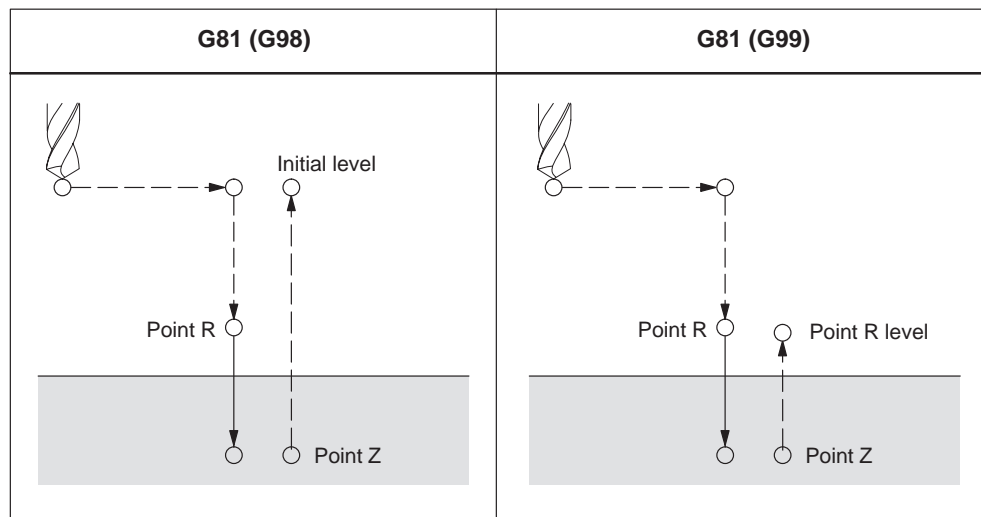


Fig. 4-9 Drilling cycle, spot drilling (G81)

## Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

## Drilling

Drilling is not carried out in a block that does not contain X, Y, Z, R, or any additional axes.

## Cancel

G codes of group 01 (G00 to G03) and G76 must not be specified within a single block. Otherwise, G76 is canceled.

## Tool offset

Tool offsets are ignored in the canned cycle mode.

## Example

```
M3 S1500;      Rotate spindle
G90 G99 G81 X200. Y-150. Z-100. R-100. F120.;
               Position, drill hole 1, and return to point R.
Y-500.;       Position, drill hole 2, and return to point R.
Y-700.;       Position, drill hole 3, and return to point R.
X950.;        Position, drill hole 4, and return to point R.
Y-500.;       Position, drill hole 5, and return to point R.
G98 Y-700.;   Position, drill hole 6, and return to the initial level.
G80;          Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;          Spindle stop
```

## 4.1 Program support functions (1)

## 4.1.5 Drilling cycle, counter boring cycle (G82)

Normal drilling can be carried out by means of this cycle. Upon reaching the drilling depth Z, a programmed dwell time is carried out after which the retraction movement is performed in rapid traverse.

## Format

G82 X... Y... R... P... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to the bottom of the hole

**R:** The distance from the initial level to R level

**P:** Dwell time at the bottom of a hole

**F:** Cutting feed rate

**K:** Number of repeats

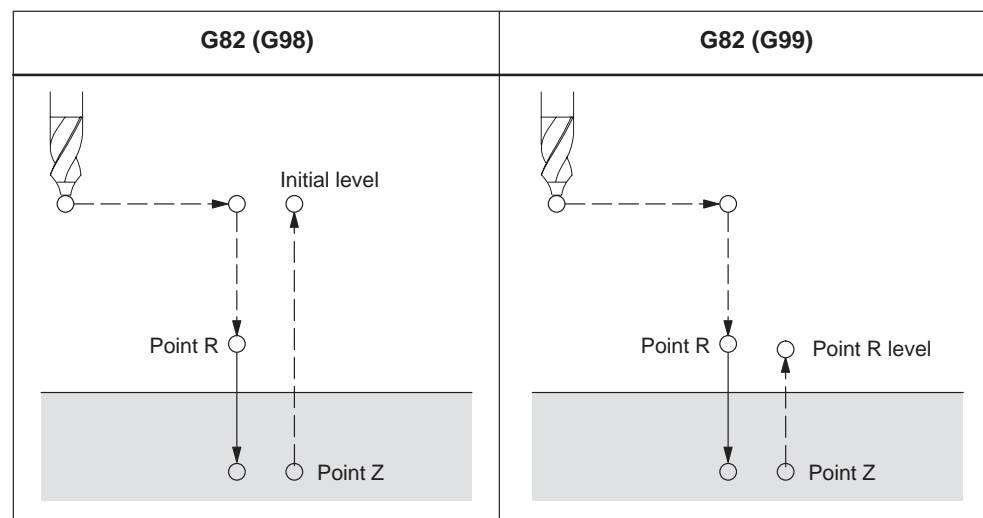


Fig. 4-10 Drilling cycle, counter boring cycle (G82)

## Axis switching

The canned cycle must be canceled before the drilling axis can be changed.

## Drilling

Drilling is not carried out in a block that does not contain X, Y, Z, R, or any other axes.

## Cancel

G codes of group 01 (G00 to G03) and G81 must not be specified in a single block. Otherwise, G81 is canceled.

## Tool offset

Tool offsets are ignored in the canned cycle mode.

## Example

```
M3 S2000;      Rotate spindle
G90 G99 G82 X200. Y-150. Z-100. R-100. P1000 F150.;
              Position, drill hole 1, dwell for 1 s at the bottom of the hole,
              and return to point R.
Y-500.;      Position, drill hole 2, and return to point R.
Y-700.;      Position, drill hole 3, and return to point R.
X950.;      Position, drill hole 4, and return to point R.
Y-500.;      Position, drill hole 5, and return to point R.
G98 Y-700.;  Position, drill hole 6, and return to the initial level.
G80;        Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;        Spindle stop
```

## 4.1 Program support functions (1)

## 4.1.6 Peck drilling cycle (G83)

By means of this cycle peck drilling is performed. It is used for deep hole drilling with shaving extraction.

## Format

G83 X... Y... R... Q... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to the bottom of the hole

**R:** Distance from the initial level to R level

**Q:** Depth of cut for each cutting feed

**F:** Cutting feedrate

**K:** Number of repeats

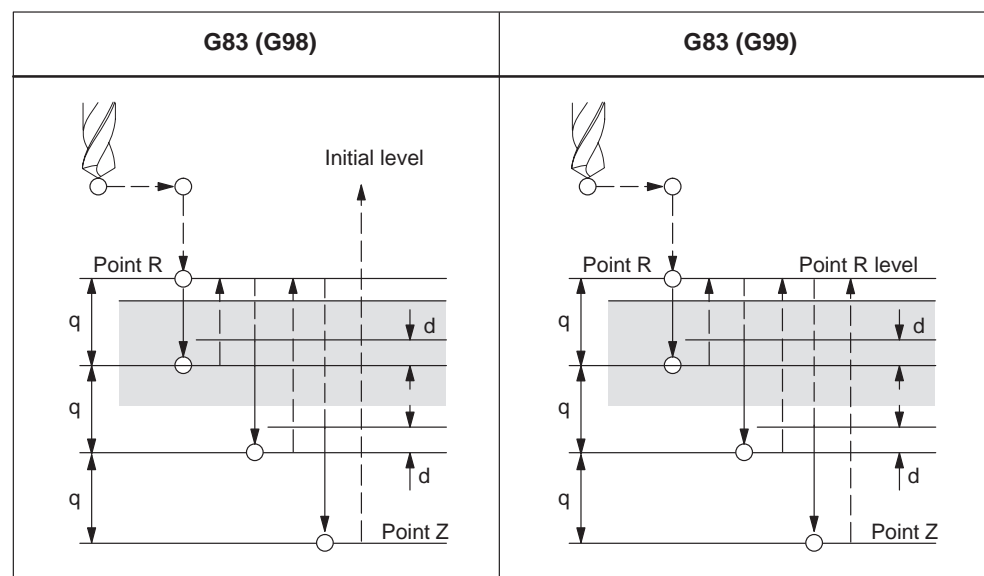


Fig. 4-11 Peck drilling cycle (G83)

## Explanations

After reaching the programmed depth of cut for each cutting feed  $Q$ , retraction to reference level  $R$  is performed in rapid traverse. Approach movement for a renewed cut is again carried out in rapid traverse up to a distance ( $d$ ) which is set to  $GUD7\_ZSFR[1]$ . Distance  $d$  and the depth of cut for each cutting feed  $Q$  are traversed with cutting feed. Specify  $Q$  incrementally implemented without sign.

## Axis switching

The canned cycle must be canceled before the drilling axis can be changed.

## Drilling

Drilling is not performed in a block that does not contain X, Y, Z, R, or any other axes.

## Cancel

G codes of group 01 (G00 to G03) and G83 must not be specified in a single block. Otherwise, G83 is canceled.

## Tool offset

Tool offsets are ignored in the canned cycle mode.

## Example

```
M3 S2000;      Rotate spindle.
G90 G99 G83 X200. Y-150. Z-100. R-100. Q10. F150.;
              Position, drill hole 1, and return to point R.
Y-500.;      Position, drill hole 2, and return to point R.
Y-700.;      Position, drill hole 3, and return to point R.
X950.;      Position, drill hole 4, and return to point R.
Y-500.;      Position, drill hole 5, and return to point R.
G98 Y-700.;  Position, drill hole 6, and return to the initial level.
G80;        Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;        Spindle stop
```

## 4.1 Program support functions (1)

## 4.1.7 Boring cycle (G85)

## Format

G85 X... Y... R... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to the bottom of the hole

**R:** Distance from the initial level to R level

**F:** Cutting feed rate

**K:** Number of repeats

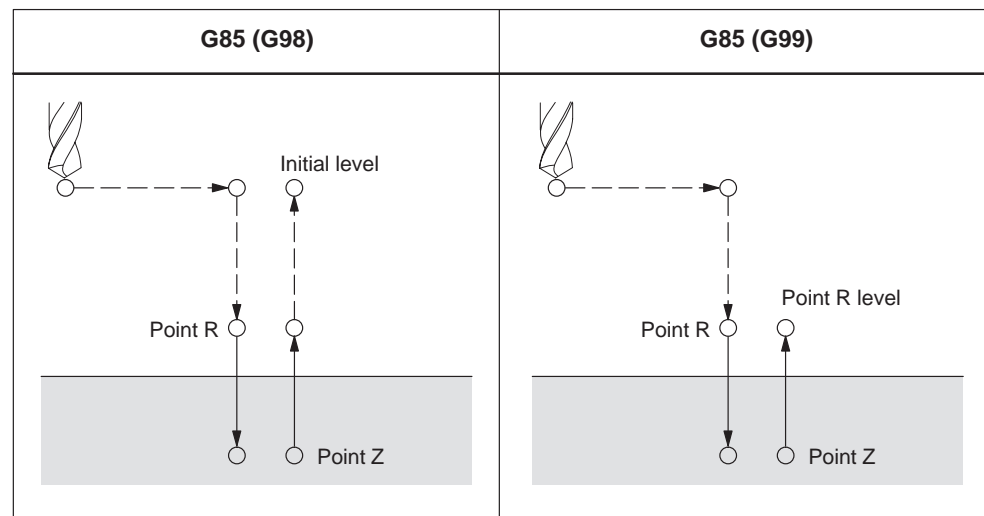


Fig. 4-12 Boring cycle (G85)

## Explanations

Rapid traverse is carried out to point R after positioning along the X- and Y- axis. Drilling is performed from point R to point Z. After reaching point Z, cutting feed is performed back to point R.

## Axis switching

Before the drilling axis can be changed the canned cycle must be canceled.

## Drilling

Drilling is not performed in a block that does not contain X, Y, Z, R, or any other axes.



**Cancel**

G codes of group 01 (G00 to G03) and G85 must not be specified in a single block. Otherwise, G85 is canceled.

**Tool offset**

Tool offsets are ignored in the canned cycle mode.

**Example**

```
M3 S150;      Rotate spindle
G90 G99 G85 X200. Y-150. Z-100. R-100. F150.;
              Position, drill hole 1, and return to point R.
Y-500.;      Position, drill hole 2, and return to point R.
Y-700.;      Position, drill hole 3, and return to point R.
X950.;       Position, drill hole 4, and return to point R.
Y-500.;      Position, drill hole 5, and return to point R.
G98 Y-700.;  Position, drill hole 6, and return to the initial level.
G80;         Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;         Spindle stop
```

## 4.1 Program support functions (1)

## 4.1.8 Boring cycle (G86)

## Format

G86 X... Y... R... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to the bottom of the hole

**R:** Distance from the initial level to point R

**F:** Cutting feed rate

**K:** Number of repeats

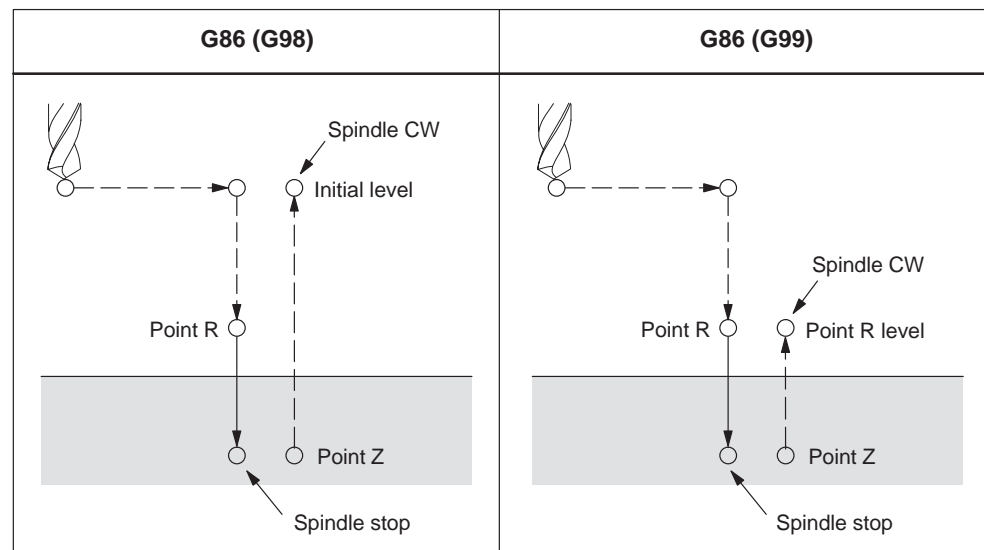


Fig. 4-13 Boring cycle (G86)

## Explanations

Rapid traverse is performed to point R after positioning along the X and Y axes. Drilling is performed from point R to point Z. After the spindle is stopped at the bottom of the hole, the tool is retracted in rapid traverse.

## Axis switching

The canned cycle must be canceled before the drilling axis can be changed.

## Drilling

Drilling is not performed in a block that does not contain X, Y, Z, R, or any other axes.

## Cancel

G codes of group 01 (G00 to G03) and G86 must not be specified in a single block. Otherwise, G86 is canceled.

## Tool offset

Tool offsets are ignored in the canned cycle mode.

## Example

```
M3 S1500;      Rotate spindle
G90 G99 G86 X200. Y-150. Z-100. R-100. F150.;
               Position, drill hole 1, and return to point R.
Y-500.;       Position, drill hole 2, and return to point R.
Y-700.;       Position, drill hole 3, and return to point R.
X950.;        Position, drill hole 4, and return to point R.
Y-500.;       Position, drill hole 5, and return to point R.
G98 Y-700.;   Position, drill hole 6, and return to the initial level.
G80;          Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;          Spindle stop
```

4.1 Program support functions (1)

**4.1.9 Boring cycle, back boring cycle (G87)**

Accurate boring is performed by this cycle.

**Format**

G87 X... Y... R... Q... P... F... K... ;

**X,Y:** Hole position

**Z:** Distance from the bottom of the hole to point Z

**R:** Distance from the initial level to point R (the bottom of the hole)

**Q:** Tool shift amount

**P:** Dwell time

**F:** Cutting feed rate

**K:** Number of repeats

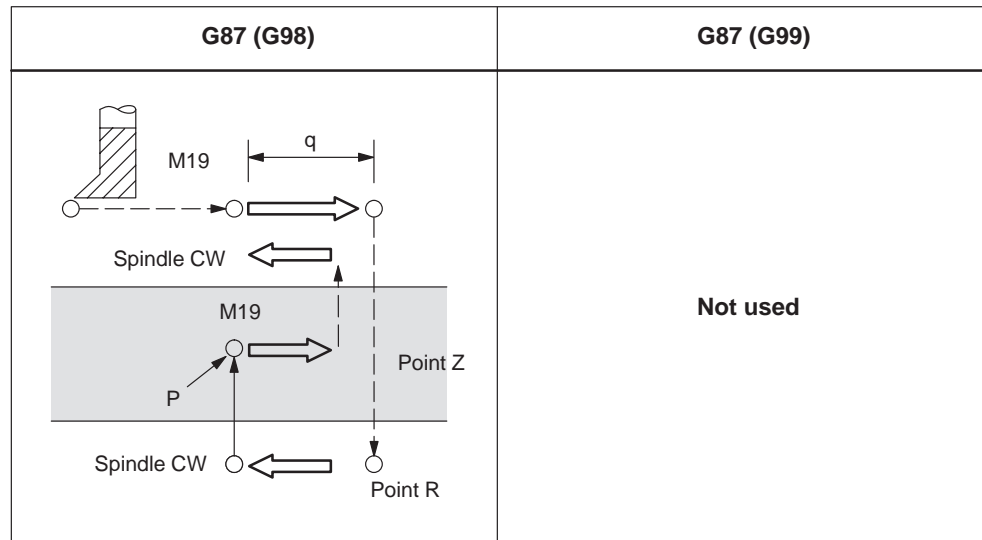


Fig. 4-14 Boring cycle, back boring cycle (G87)

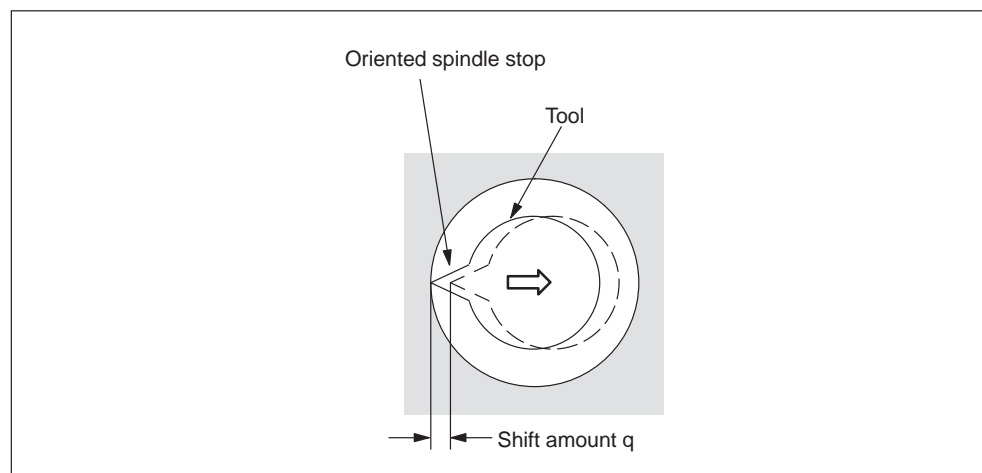


Fig. 4-15



### Warning

Address Q (shift at the bottom of a hole) is a modal value which is retained within canned cycles. Special care has to be taken because it is also used as the depth of cut in cycles G73 and G83.

### Explanations

The spindle is stopped at the fixed rotation position after positioning along the X and Y axes. The tool is moved in the direction opposite the tip of the tool. Positioning (rapid traverse) is carried out to the bottom of the hole (point R).

Then the tool is shifted into the direction of the tool tip, and the spindle is rotated clockwise. Boring is carried out in the positive direction along the Z axis until point Z.

The spindle is stopped at the fixed rotation position again at point Z. The tool is then shifted into the direction opposite the tool tip, and the tool is shifted back to the initial level. Subsequently, the tool is shifted into the direction of the tool tip, and the spindle is rotated clockwise in order to proceed to the next block operation. To enter a safety clearance, GUD \_ZSFR[0] can be applied.

The lift-off path is always specified with reference to the negative direction of the first geometry axis

G17, lift-off path in -X

G18, lift-off path in -Z

G19, lift-off path in -Y

Therefore, the angle has to be entered to GUD7 \_ZSFR[2] in such a way that the tool tip points into the positive direction (+) within the defined plane after the spindle has stopped.

Example:

If plane G17 is activated, the tool tip has to point into the +X direction.

### Axis switching

The canned cycle must be canceled before the drilling axis can be changed.

### Boring

Boring is not performed within a block that does not contain X, Y, Z, R, or any additional axes.

### Q/R

By all means, specify a positive value at address Q. The sign is ignored if address Q is specified with a negative value. Q equal 0 is set whenever no lift-off amount is programmed. This leads to cycle execution without lift-off.

---

#### 4.1 Program support functions (1)

### Cancel

G codes of group 01 (G00 to G03) and G87 must not be specified in a single block. Otherwise, G87 is canceled.

### Tool offset

Tool offsets are ignored in the canned cycle mode.

### Example

```
M3 S400;      Rotate spindle
G90 G87 X200. Y-150. Z-100. R-100. Q3. P1000 F150.;
              Position, bore hole 1,
              orient at the initial level, then shift by 3 mm,
              stop at point Z for 1 s
Y-500.;      Position, drill hole 2.
Y-700.;      Position, drill hole 3.
X950.;       Position, drill hole 4.
Y-500.;      Position, drill hole 5.
Y-700.;      Position, drill hole 6
G80;         Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;          Spindle stop
```

### 4.1.10 Boring cycle (G89)

#### Format

G89 X... Y... R... P... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to the bottom of the hole

**R:** Distance from the initial level to point R

**P:** Dwell time at the bottom of a hole

**F:** Cutting feed rate

**K:** Number of repeats

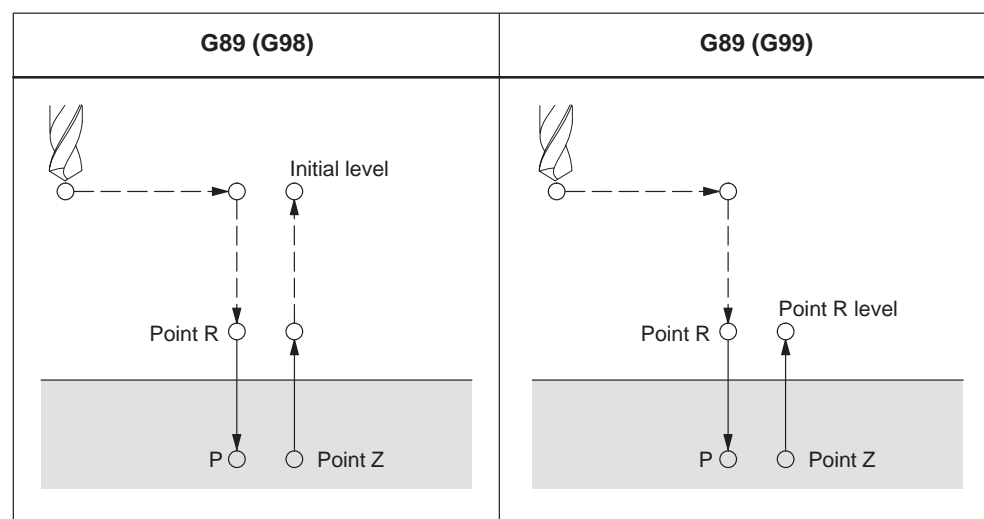


Fig. 4-16 Boring cycle (G89)

#### Explanations

This cycle is almost the same as G85 except that this cycle performs a dwell at the bottom of the hole.

Use a miscellaneous function (M code) to rotate the spindle before specifying G89.

#### Axis switching

The canned cycle must be canceled before the drilling axis can be changed.

#### Drilling

Drilling is not performed in a block that does not contain X, Y, Z, R, or any other axes.

---

#### 4.1 Program support functions (1)

### Cancel

G codes of group 01 (G00 to G03) and G89 must not be specified in a single block. Otherwise, G89 is canceled.

### Tool offset

Tool offsets are ignored in the canned cycle mode.

### Example

```
M3 S150;      Rotate spindle
G90 G99 G89 X200. Y-150. Z-100. R-100. P1000 F150.;
              Position, drill hole 1, return to point R
              then stop at the bottom of the hole for 1 s.
Y-500.;      Position, drill hole 2, and return to point R.
Y-700.;      Position, drill hole 3, and return to point R.
X950.;       Position, drill hole 4, and return to point R.
Y-500.;      Position, drill hole 5, and return to point R.
G98 Y-700.;  Position, drill hole 6, and return to the initial level.
G80;         Cancel canned cycle
G28 G91 X0 Y0 Z0; Return to the reference position return
M5;          Spindle stop
```



### 4.1.11 Rigid tapping cycle (G84)

When the spindle motor is controlled in rigid mode as if it were a servo motor, a tapping cycle can be sped up.

#### Format

G84 X... Y... Z... R... P... F... K... ;

**X,Y:** Hole position

**Z:** Distance from point R to bottom of the hole

**R:** Distance from the initial level to R level

**P:** Dwell time at bottom of the hole and at point R when a return is made

**F:** Cutting feedrate

**K:** Number of repeats (if required)

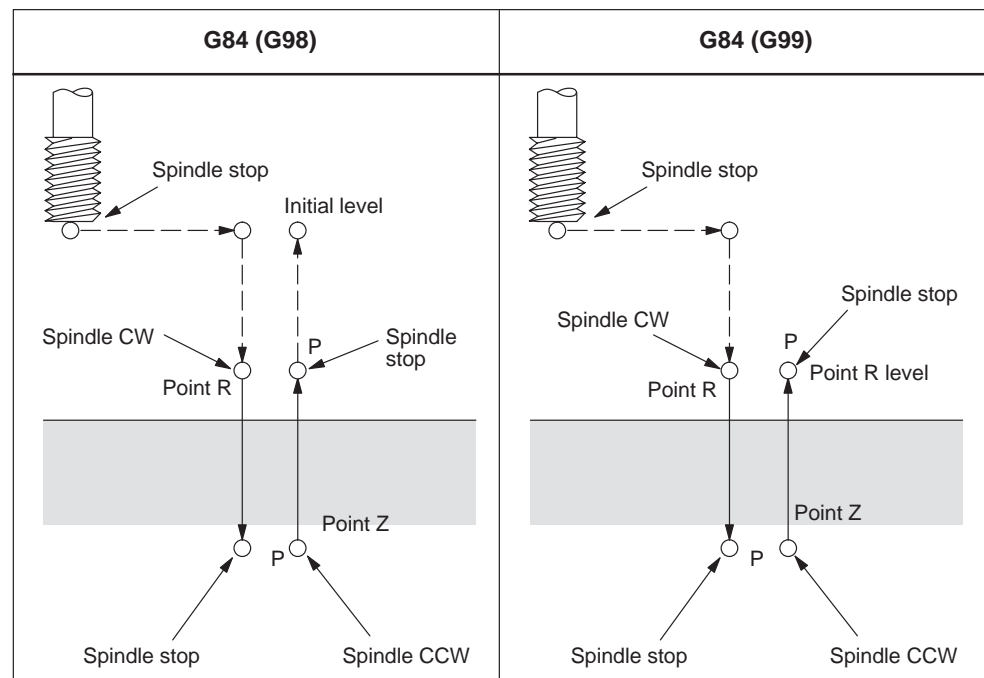


Fig. 4-17 Rigid tapping (G84)

#### Explanations

Rapid traverse is carried out to point R after positioning along the X and Y axes. Tapping is carried out from point R to point Z. The spindle is stopped, and a dwell is performed once tapping has been completed. Then the spindle is rotated in reverse direction. The tool is retracted to point R, and the spindle is stopped. Subsequently, rapid traverse to the initial level is carried out. The feedrate override and the spindle override are supposed to be 100% while tapping is being carried out. Yet the rotation speed during retraction can be controlled through GUD `_ZSFI[2]`. Example: `_ZSFI[2]=120`, the retraction is performed at 120% of the tapping speed.

#### 4.1 Program support functions (1)

### Thread lead

The thread lead is obtained from the expression 'feedrate spindle speed' in the feed-per-minute mode. The thread lead equals the feedrate speed within the speed-per-revolution mode.

### Tool length compensation

The offset is applied at the time of positioning to point R if a tool length compensation (G43, G44, or G49) is determined in the canned cycle.

### Axis switching

The canned cycle must be canceled before the drilling axis can be changed. An alarm is issued if the drilling axis is changed in rigid mode.

### S command

An alarm is issued if a speed higher than the maximum speed for the gear being used is specified.

### F command

An alarm is issued if a value exceeding the upper limit of cutting feedrate is specified.

### Unit of F command

	Metric input	Inch input	Remarks
G94	1 mm/min	0.01 inch/min	Decimal point programming allowed
G95	0.01 mm/rev	0.0001 inch/rev	Decimal point programming allowed

### Cancel

G codes of group 01 (G00 to G03) and G84 must not be specified in a single block. Otherwise, G84 is canceled.

## Tool offset

Tool offsets are ignored in the canned cycle mode.

### Example

Z axis feedrate 1000 mm/min

Spindle speed 1000 rpm

Thread lead 1.0 mm

<Programming of feed per minute>

S1000 M3;

G94; Feed-per-minute

G00 X100.0 Y100.0; Positioning

G84 Z-50.0 R-10.0 **F1000**; Rigid tapping

<Programming of feed per revolution>

G95; Feed-per-revolution

G00 X100.0 Y100.0; Positioning

G84 Z-50.0 R-10.0 **F1.0**; Rigid tapping

## 4.1 Program support functions (1)

## 4.1.12 Left-handed rigid tapping cycle (G74)

When the spindle motor is controlled in rigid mode as if it were a servo motor, tapping cycles can be sped up.

## Format

G74 X... Y... Z... R... P... F... K... ;

**X,Y:** Hole position

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

**R:** The distance from the initial level to point R

**P:** Dwell time at the bottom of the hole and at point R when return is made.

**F:** Cutting feedrate

**K:** Number of repeats (if required)

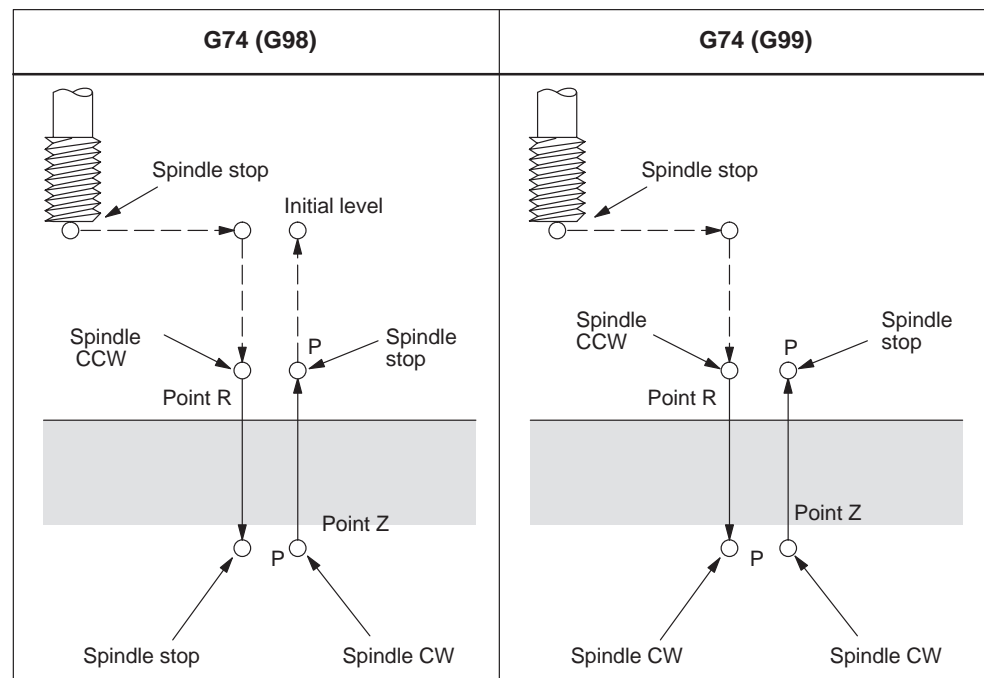


Fig. 4-18 Left-handed rigid tapping cycle (G74)

## Explanations

Rapid traverse is performed to point R after positioning along the X and Y axes. Tapping is carried out from point R to point Z. The spindle is stopped and a dwell is performed once tapping has been completed. Subsequently, the spindle is rotated in the normal direction. The tool is retracted to point R, and the spindle is stopped. Rapid traverse to the initial level is then carried out. The feedrate override and the spindle override are supposed to be 100% while tapping is being carried out. The speed of rotation, however, can be controlled through GUD \_ZSFI[2] during retraction. Example: \_ZSFI[2]=120, the retraction takes place with 120% of the tapping speed.

## Thread lead

The thread lead is obtained from the expression 'feedrate spindle speed' in feed-per-minute mode. The thread lead equals the feedrate speed when in feed-per-revolution mode.

## Tool length compensation

The offset is applied at the time of positioning to point R whenever a tool length compensation (G43, G44, or G49) is specified in the canned cycle.

## Axis switching

The canned cycle must always be canceled before the drilling axis can be changed. An alarm is issued if the drilling axis is changed into rigid mode.

## S command

An alarm is issued if a speed that is higher than the maximum speed for the gear currently in use is specified.

## F command

An alarm is issued if a value overshooting the upper limit of cutting feedrate is specified.

## Unit of F command

	Metric input	Inch input	Remarks
G94	1 mm/min	0.01 inch/min	Decimal point programming allowed
G95	0.01 mm/rev	0.0001 inch/rev	Decimal point programming allowed

---

#### 4.1 Program support functions (1)

### Cancel

A G code of the 01 group (G00 to G03) and G74 should not be specified in a single block or else G74 will be canceled.

### Tool offset

Tool offsets are ignored In the canned cycle mode.

### Example

Z axis feedrate 1000 mm/min  
Spindle speed 1000 rpm  
Thread lead 1.0 mm

<Programming of feed per minute>

S1000 M4

G94; Feed-per-minute

G00 X100.0 Y100.0; Positioning

G74 Z-50.0 R-10.0 **F1000**; Rigid tapping

<Programming of feed per revolution>

G95; Feed-per-revolution

G00 X100.0 Y100.0; Positioning

G74 Z-50.0 R-10.0 **F1.0**; Rigid tapping

### 4.1.13 Peck tapping cycle (G84 or G74)

Due to chips stuck to the tool or increased resistance to cutting, the tapping a deep hole in rigid tapping mode could be difficult. The peck rigid tapping cycle is useful when this is the case.

Cutting is carried out several times in this cycle until the bottom of the hole is reached. For this, two peck tapping cycles are available: High-speed peck tapping cycle (deep hole tapping with chip-breaking) as well as the standard peck tapping cycle (deep hole tapping with swarf removal).

By using GUD7 and setting data `_ZSFI[1]`, these cycles are selected as follows:

`_ZSFI[1] = 2`: High-speed peck tapping cycle

`_ZSFI[1] = 3`: Standard peck tapping cycle

#### Format

G84 (or G74) X... Y... Z... R... P... Q... F... K... ;

**X,Y**: Hole position

**Z**: Distance from point R to the bottom of the hole

**R**: The distance from the initial level to point R level

**P**: Dwell time at the bottom of the hole and at point R when a return is made

**Q**: Depth of cut for each cutting feed

**F**: The cutting feedrate

**K**: Number of repeats

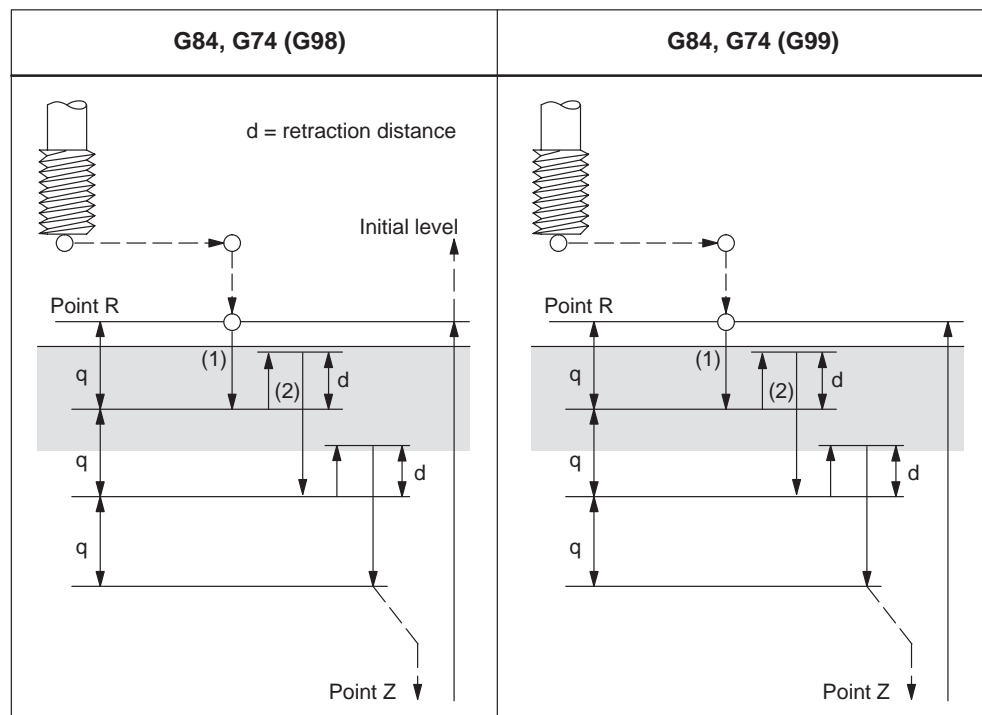


Fig. 4-19 High-speed peck tapping cycle (GUD7 `_ZSFI[1] = 2`)

## 4.1 Program support functions (1)

1. The tool can operate at a normal cutting feedrate. Here the normal time constant is applied.
2. The retraction can be overridden. The retraction speed set to GUD7\_ZSFI[2] is applied in this case.

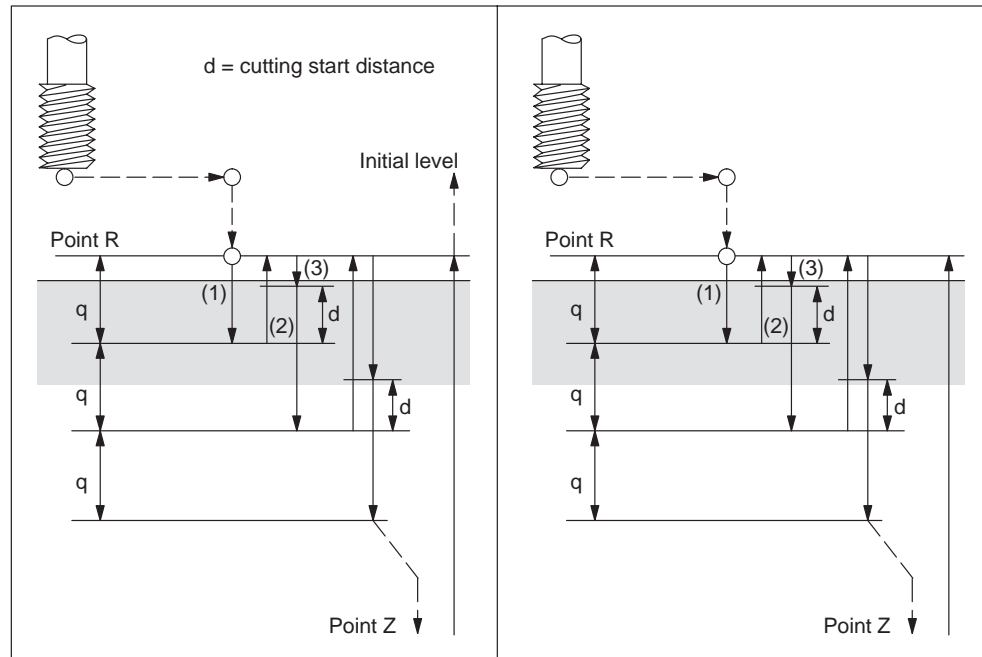


Fig. 4-20 Peck tapping cycle (GUD7\_ZSFI[1] = 3)

1. The tool can operate at a normal cutting feedrate. Here the normal time constant is applied.
2. The retraction can be overridden. The retraction speed set to GUD7\_ZSFI[2] is applied in this case.
3. The retraction can be overridden. Here the normal time constant is applied.

An in-position check is carried out at the end of each operation of 1. and 2. in the peck tapping cycle during a rigid tapping cycle.

## Explanation

### High-speed peck tapping cycle

Rapid traverse is carried out to point R after positioning along the X and Y axes. Cutting is carried out from point R, with depth Q (depth of cut for each cutting feed). Subsequently, the tool is retracted by the distance d. Whether retraction is overridden or not is specified by a value other than 100% set to GUD7\_ZSFI[2]. The spindle is stopped once point Z is reached and then rotated in the reverse direction for retraction. The retraction distance d is to be set in GUD7\_ZSFR[1].



### Peck tapping cycle

Rapid traverse is performed to R level after positioning along the X and Y axes. , Cutting is performed from point R with depth Q (depth of cut for each cutting feed). Subsequently a return is carried out to point R. Whether retraction is overridden or not is specified by a value other than 100% set to GUD7 \_ZSFI[2]. Moving the cutting feedrate F is carried out from point R to a position distance d from the end point of the last cutting. This is where the cutting is restarted.

The spindle is stopped once point Z is reached, and, subsequently, rotated in the reverse direction for retraction.

Set d (distance to the point where the cutting is started) in GUD7 \_ZSFR[1].

### Axis switching

The canned cycle must be canceled before the drilling axis can be changed. An alarm is issued whenever the drilling axis is changed in rigid mode.

### S command

An alarm is issued whenever a speed higher than the maximum speed for the gear in use is specified.

### F command

An alarm is issued, if a value overshooting the upper limit of the cutting feedrate is specified.

### Unit of F command

	Metric input	Inch input	Remarks
G94	1 mm/min	0.01 inch/min	Decimal point programming allowed
G95	0.01 mm/rev	0.0001 inch/rev	Decimal point programming allowed

### Cancel

A G code of the 01 group (G00 to G03) and G74/G84 should not be specified in a single block or else G74/G84 will be canceled.

### Tool offset

Tool offsets are ignored in the canned cycle mode.

---

#### 4.1 Program support functions (1)

##### 4.1.14 Canned cycle cancel (G80)

G80 cancels canned cycles.

##### Format

G80;

##### Explanations

The values of point R and point Z are cleared, all canned cycles are canceled and normal operation is performed. In addition, the values of all addresses programmed with drilling cycles are cleared.

4.1.15 Program example using tool length offset and canned cycles

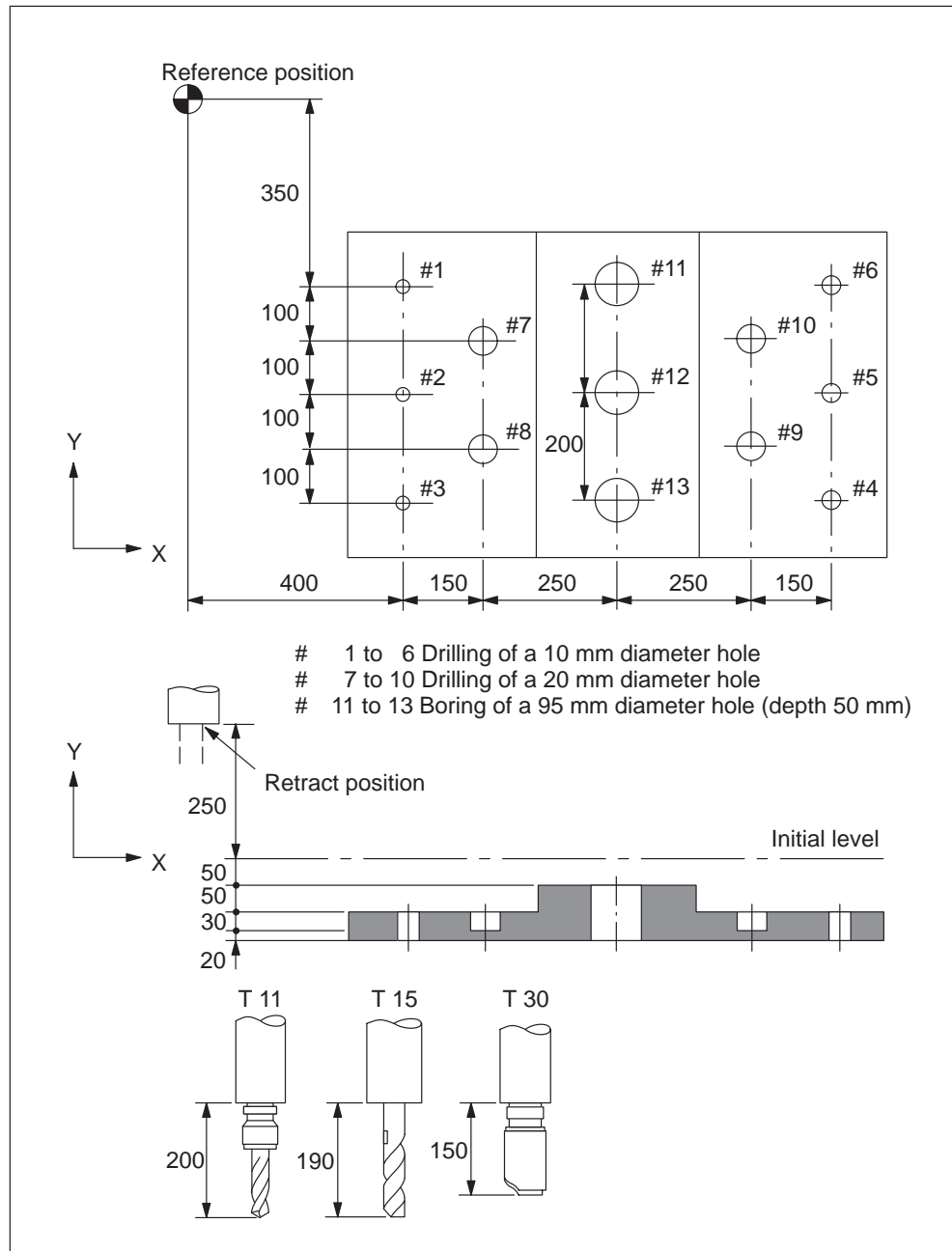


Fig. 4-21 Program example (drilling cycle)

## 4.1 Program support functions (1)

Offset value +200.0 is set in offset No.11, +190.0 is set in offset No.15, and +150.0 is set in offset No.30

**Program example**

```

;
N001 G49;           Tool length offset cancel
N002 G10 L10 P11 R200.; Set tool offset 11 to +200.
N003 G10 L10 P15 R190.; Set tool offset 15 to +190.
N004 G10 L10 P30 R150.; Set tool offset 30 to +150.
N005 G92 X0 Y0 Z0;   Coordinate setting at reference position
N006 G90 G00 Z250.0 T11 M6; Tool change
N007 G43 Z0 H11;     Initial level, tool length offset
N008 S30 M3 ;       Spindle start
N009 g99 G81 X400.0 Y-350.0 Z-153.0 R-97.0 F1200;
                    Positioning, then #1 drilling
N010 Y-550.0;       Positioning, then #2 drilling and point R level return
N011 G98 Y-750.0;   Positioning, then #3 drilling and initial level return
N012 G99 X1200.0;   Positioning, then #4 drilling and point R level return
N013 Y-550.0;       Positioning, then #5 drilling and point R level return
N014 G98 Y-350.0;   Positioning, then #6 drilling and initial level return
N015 G00 X0 Y0 M5;   Reference position return, spindle stop
N016 G49 Z250.0 T15 M6; Tool length offset cancel, tool change
N017 G43 Z0 H15;     Initial level, tool length offset
N018 S20 M3;        Spindle start
N019 G99 G82 X550.0 Y-450.0 Z-130.0 R-97.0 P300 F700;
                    Positioning, then #7 drilling, point R level return
N020 G98 Y-650.0;   Positioning, then #8 drilling, initial level return
N021 G99 X1050.0;   Positioning, then #9 drilling, point R level return
N022 G98 Y-450.0;   Positioning, then #10 drilling, initial level return
N023 G00 X0 Y0 M5;   Reference position return, spindle stop
N024 G49 Z250.0 T30 M6; Tool length offset cancel, tool change
N025 G43 Z0 H30;     Initial level, tool length offset
N026 S10 M3;        Spindle start
N027 G85 G99 X800.0 Y-350.0 Z-153.0 R47.0 F500;
                    Positioning, then #11 drilling, point R level return
N028 G91 Y-200.0 K2; Positioning, then #12, 13 drilling. point R level return
N029 G28 X0 Y0 M5;   Reference position return, spindle stop
N030 G49 Z0;        Tool length offset cancel
N031 M30 ;         Program End

```

## 4.2 Programmable data input (G10)

### 4.2.1 Changing of tool offset value

Existing tool offsets can be overwritten by using the G10. New tool offsets, however, cannot be created.

#### Format

G10 L10 P... R... ; Tool length compensation, geometry  
 G10 L11 P... R... ; Tool length compensation, wear  
 G10 L12 P... R... ; Tool radius compensation, geometry  
 G10 L13 P... R... ; Tool radius compensation, wear

P: Number of the compensation memory

R: Specifies the value

L1 can be programmed instead of L11.

#### Relevant machine data

Machine data 20382 \$MC\_TOOL\_CORR\_MOVE\_MODE defines whether the compensation is applied in the block containing the selection or the next time the axis is programmed.

Machine data 20270 \$MC\_CUTTING\_EDGE\_DEFAULT = 0 defines that no tool length compensation is active initially on a tool change.

Setting data \$SC\_TOOL\_LENGTH\_CONST must contain the value 17 for the assignment of tool length offsets to geometry axes to be independent of the plane selection. Length 1 is then always assigned to the Z axis.

### 4.2.2 Setting the workpiece coordinate system shift data

With the commands of "G10 P00 X (U) ... Y (V) ... Z (W) ... ;", 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

## 4.3 Subprogram call up function (M98, M99)

## 4.3 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 4-5 are used.

Table 4-5 Subprogram call M code

M code	Function
M98	Subprogram call up
M99	End of subprogram

### Subprogram call (M98)

- M98 P nnn mmmm  
m: Program number (max. 4 digits)  
n: Number of repetitions (max. 3 digits)
- For example, if M98 P21 is programmed, the part program memory is searched for program name 21.mpf and the subprogram is executed once. To execute the subprogram 3 times, M98 P30021 needs to be programmed. If the specified program number is not found, an alarm occurs.
- Nesting of subprograms is possible - the allowable nesting level is four. If the nesting level exceeds this limit, an alarm occurs.

### End of subprogram code (M99)

If M99 Pxxxx is programmed, execution resumes at block number xxxx on the return jump to the main program. The system initially searches forward 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 searched backwards (towards the head of the program).

If M99 is specified in a main program, the program returns to the beginning of that main program and the program is repeatedly executed.

## 4.4 Polar coordinate command (G15, G16)

Using polar coordinate command it is possible to program the end point coordinate value in radius and angle. Any dimension word between G16 and G15 command is interpreted as the polar coordinate values for radius and angle in the current plane. The first axis of the plane represents the polar radius, while the second axis represents the polar angle.

### Format

G17 (G18, G19) G90 (G91) G16;	Polar coordinate command ON
G90 (G91) X... Y... Z...	Polar coordinate command
... ;	
... ;	
G15;	Polar coordinate command CANCEL

G16:	Polar coordinate command
G15:	Polar coordinate command CANCEL
G17, G18, G19:	Plane selection
G90:	The pole is at the workpiece zero
G91:	The pole is at the current position
X, Y, Z:	First axis: radius of polar coordinate Second axis: angle of polar coordinate

### Note

If the pole is moved from the current position to the workpiece zero, the radius is calculated as the distance from the current position to the workpiece zero.

### Example

N5 G17 G90 X0 Y0;	
N10 G16 X100. Y45.;	Polar coordinates ON, pole is workpiece zero, position X 70,711 Y 70,711 in Cartesian coordinate system
N15 G91 X100 G90 Y0;	Pole is current position, i.e. position X 170,711 Y 70,711
N20 Y90.;	No X in block, pole is at workpiece zero, Radius = $\text{SORT}(X^2 + Y^2) = 184,776$
G15;	

The polar radius is always traversed as an absolute value while the polar angle can be interpreted as an absolute or incremental value.

## 4.5 Polar coordinate interpolation (G12.1, G13.1)

## 4.5 Polar coordinate interpolation (G12.1, G13.1)

An interpolation between a rotary axis and a linear axis in the machining plane is switched on and off through G12.1 and G13.1. A further linear axis is perpendicular to this plane.

Linear or circular interpolation using coordinates in a Cartesian coordinate system is applied in order to program a linear axis together with a rotary axis (virtual axis).

This function corresponds to the TRANSMIT function within Siemens mode. For a detailed description of the TRANSMIT function see "SINUMERIK 840D/810D(CCU2)/FM" NC Functional Description, Extended Functions" chapter "Kinematic Transformation (M1)" and "SINUMERIK 840D/810D/FM-NC Programming Guide Production Scheduling (PGA)" chapter "Transformation".

### Format

G12.1;      Polar coordinate interpolation mode ON  
 ...  
 ...  
 G13.1;      Polar coordinate interpolation mode cancel



### Caution

When specifying G12.1, the plane (G17, G18, G19) which has been used previously is canceled.

NC Reset will cancel the polar coordinate interpolation mode and re-establish the previously selected plane.

### Possible G codes in the polar coordinate interpolation mode

G01	Linear interpolation
G02, G03	Circular interpolation
G04	Dwell, Exact stop
G40, G41, G42	Cutter compensation
G65, G66, G67	Custom macro command
G90, G91	Absolute command, incremental command
G94, G95	Feed per minute, feed per revolution

### Using G02, G03 in the polar coordinate plane

The addresses used for the specification of the radius of an arc with respect to circular interpolation (G02 or G03) applied to a polar coordinate interpolation plane are dependent on the first axis in the plane (linear axis).

- I and J in the Xp–Yp plane whenever the linear axis is the X axis or, alternatively, an axis parallel to the X axis.
- J and K in the Yp–Zp plane whenever the linear axis is the Y axis



## 4.5 Polar coordinate interpolation (G12.1, G13.1)

- or, alternatively, an axis parallel to the Y axis.
  - K and I in the Zp–Xp plane whenever the linear axis is the Z axis
  - or, alternatively, an axis parallel to the Z axis.
- Address R can also be used to specify the radius of an arc.

**Example**

X axis (linear axis), C axis (rotary axis)

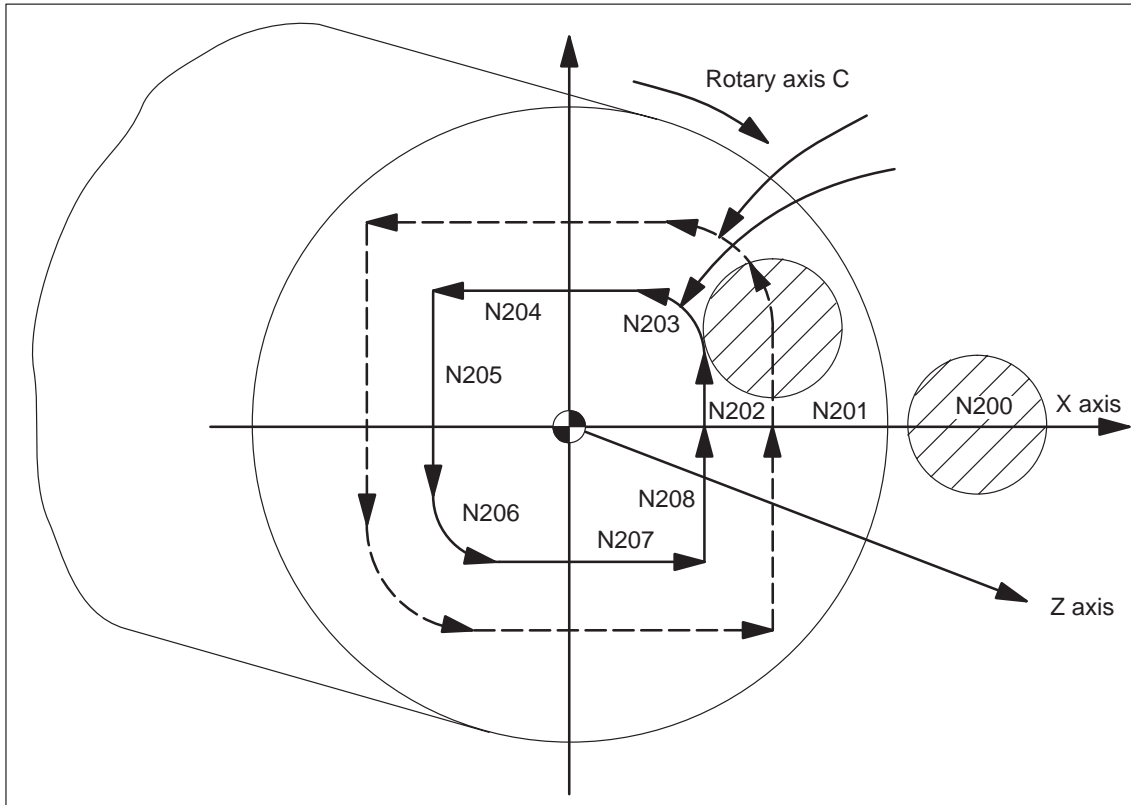


Fig. 4-22

```

N010 T0101;
N0100 G90 G00 X60.0 C0 Z.;
N0200 G12.1;           Polar coordinate interpolation ON
N0201 G42 G01 X20.0 F1000;
N0202 C10.0;
N0203 G03 X10.0 C20.0 R10.0
N0204 G01 X-20.0;
N0205 C-10.0;
N0206 G03 X-10.0 C-20.0 I10.0 J0;
N0207 G01 X20.0;
N0208 C0;
N0209 G40 X60.0;
N0210 G13.1;         Polar coordinate interpolation OFF
N0300 Z.;
N0400 X.. C.;
N0900 M30;

```

## 4.6 Cylindrical interpolation (G07.1)

## 4.6 Cylindrical interpolation (G07.1)

This interpolation feature allows the machining to be accomplished by the combination of tool movements and rotation of a workpiece in the virtual orthogonal coordinate system. Machining is possible on the circumference of cylindrical workpiece by using the commands in an orthogonal coordinate system. To use this function, an additional axis of rotation is necessary in addition to the normal servo axes (X, Y, and Z axes).

### Programming format

The cylindrical interpolation mode is turned ON and OFF by the G codes indicated below.

Table 4-6 G codes used for cylindrical interpolation

G code	Function	Group
G07.1	Cylindrical interpolation mode	16

### Format

G07.1 A (B, C) r ;

Starts the cylindrical interpolation mode (enables cylindrical interpolation).

G07.1 A (B, C) 0 ;

The cylindrical interpolation mode is cancelled.

A, B, C: An address for the rotation axis

r: The radius of the cylinder

Specify the G07.1 command in a block without other commands.

The G07.1 command is modal and once the G07.1 command is specified, the cylindrical interpolation mode remains ON until the G07.1 A (B, C) command is specified. The NC is in the cylindrical interpolation OFF mode when the power is turned ON or when the NC is reset.

### 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.

### Example of programming

The following program is created on the cylindrical plane (the plane obtained by developing the circumference of the cylindrical workpiece) where the Z-axis is taken as the linear axis and the A-axis is taken as the rotary axis.

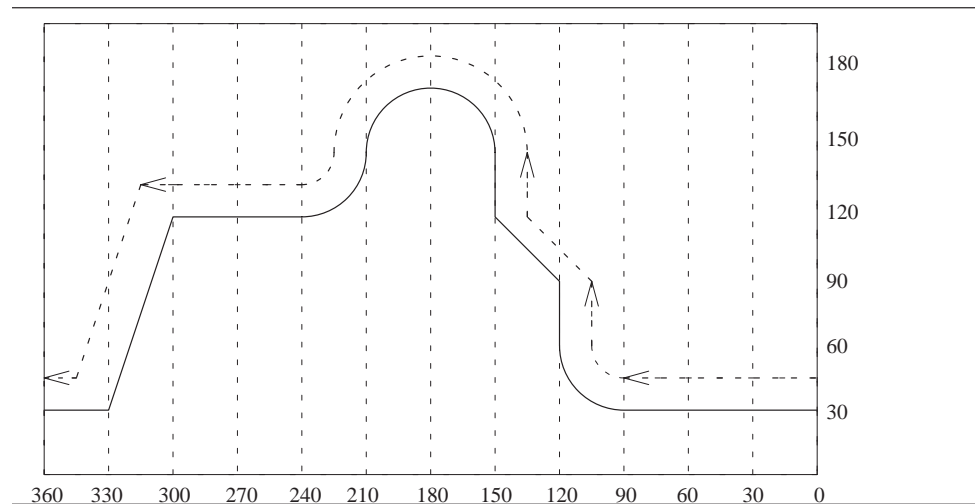


Fig. 4-23 G07.1 – Example of programming

## 4.6 Cylindrical interpolation (G07.1)

**Program**

```
M19
G40;
G00 Z30. A-10.;
G07.1 A57.296;           Cylindrical interpolation mode ON
                          (workpiece radius = 57.926)

G90:
G42 G01 A0 F200;
G00 X50.;
G01 A90. F100;
G02 A120. Z60. R30;
G01 Z90.;
Z120. A150.;
Z150.;
G03 Z150. A210. R30;
G01 Z90.;
Z120. A150.;
Z150.;
G03 Z150. A210. R30.;
G02 Z120. A240. R30;
G01 A300.;
Z30. A330.;
A360.;
G00 X100.;
G40 G01 A370.;
G07.1 A0;               Cylindrical interpolation mode OFF
G00 A0;
```

## Programming in the cylindrical interpolation mode

In the cylindrical interpolation mode, only the following G codes can be used: G00, G01, G02, G03, G04, G40, G41, G42, G65, G66, G67, G90, G91, and G7.1. Concerning the G00 command, only the axes not included in the cylindrical plane can be designated in the G00 mode.

1. G00 (positioning command)
 

The G00 command can be specified only for the axes which are not included in the cylindrical plane. Positioning is not possible on the cylindrical plane. If positioning is required for the axis which is included in the cylindrical plane, the cylindrical interpolation mode must be canceled once.
2. G01 (linear interpolation command)
 

This command can be specified for all axes. However, it is not allowed to specify the axis included in the cylindrical plane and the one not included in the cylindrical plane in the same block.

The designation of the end point for the linear interpolation should be made in either "mm" or "inch" for both the linear and rotary axes.

Feedrates of the axes are controlled so that the vector sum (tangential velocity in the direction of tool movement) of the linear axis feedrate and the rotary axis feedrate will be the feedrate specified in the program.
3. G02/G03 (circular interpolation commands)
 

The circular interpolation commands can be specified only for the axes included in the cylindrical plane.

The designation of the end point for the circular interpolation should be made in either "mm" or "inch" for both the linear and rotary axes.

The radius for the circular interpolation should be specified by an R command or by specifying the center of the arc. When an R command is used, designation of the radius should be made in either "mm" or "inch". If the center of the arc should be designated instead of the R command, specify the distance from the start point to the center of the arc by signed incremental value using addresses I, J, and K.

  - If the linear axis is X-axis, use I and J assuming the XY plane.
  - If the linear axis is Y-axis, use J and K assuming the YZ plane.
  - If the linear axis is Z-axis, use K and I assuming the ZX plane.
4. G40/G41/G42
 

The tool radius offset C function can be used only in the cylindrical plane. The D command specifying the offset memory number may be specified in any block. To execute tool radius offset in the cylindrical plane, turn ON the cylindrical interpolation mode and the tool radius offset mode.

The tool path in the cylindrical plane is offset by the tool radius set in the tool offset data memory. The direction of offset is specified by G41 and G42.

It is necessary to cancel the offset by specifying the G40 command before turning the cylindrical interpolation mode OFF.
5. G90/G91 (absolute/incremental commands)
 

It is allowed to change the dimension data designation mode between absolute and incremental while in the cylindrical interpolation mode. Designation can be made in the same manner as in the normal mode.

---

#### 4.6 Cylindrical interpolation (G07.1)

### Relationship between the Cylindrical Interpolation and Operations

- The following functions cannot be specified in the cylindrical interpolation mode. Similarly, it is not allowed to specify the G07.1 command while any of the functions indicated below is called.
  - Mirror image
  - Scaling (G50, G51)
  - Coordinate rotation (G68)
  - Base coordinate system setting
- Overrides (rapid traverse, jog, spindle speed) are valid.
- When the cylindrical interpolation mode is canceled, the interpolation plane selected before the call of the cylindrical interpolation mode is recovered.
- In the cylindrical interpolation mode, the stored stroke limit function is valid.
- To execute tool length offset, specify the tool length offset command before specifying the G07.1 command.
- The workpiece coordinate (G54 - G59) must be specified before specifying the G07.1 command.

## 4.7 Program support functions (2)

### 4.7.1 Working area limitation (G22, G23) (Under development)

The working area limitation function checks whether the present position of axes operated manually or automatically enters the stored stroke limit (entry prohibited area) which is set by G22. If an axis has entered the stroke end limit, operation is stopped and an alarm occurs.

A protection zone predetermined by machine data setting must exist if G commands G22 and G23 are used. Further, the following machine data need to be set:  
 $\$MN\_NUM\_PROTECT\_AREA\_NCK = 2$  (minimum)  
 $\$MC\_NUM\_PROTECT\_AREA\_ACTIVE = 2$  (minimum)

When programming G22, the area inside the boundary becomes the forbidden area.

An upper (G23) and lower (G22) working area limit is defined for each axis. These values apply immediately and are not lost on Reset and when the control is switched on again. The tool (milling tool) radius can be changed in the channel-specific machine data  $\$MC\_WORKAREA\_WITH\_TOOL\_RADIUS$ .

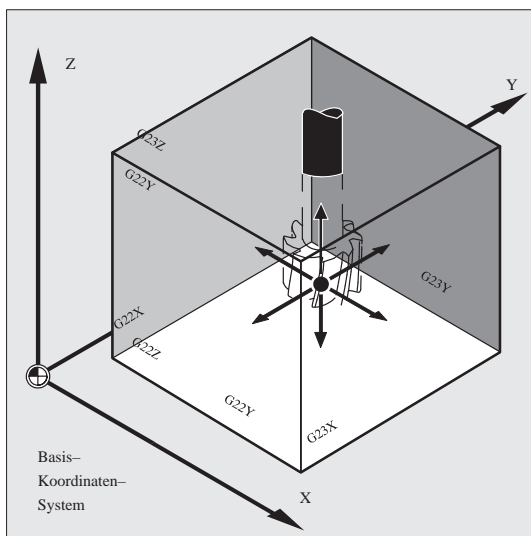


Fig. 4-24

---

#### 4.7 Program support functions (2)

##### **Status at power-on**

Whether working area limitation is enabled or disabled at power-on is decided by the following machine data:

`$MC_EXTERN_GCODE_RESET_VALUES[3]`

This MD is set to value 2 (G23) as default.



### 4.7.2 Chamfering and corner rounding commands

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...; Chamfering  
, 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.

## 4.7 Program support functions (2)

**Example**

```

N10 G1 X10. Y100. F1000 G18
N20 A140 C7.5
N30 X80. Y70. A95.824, R10

```

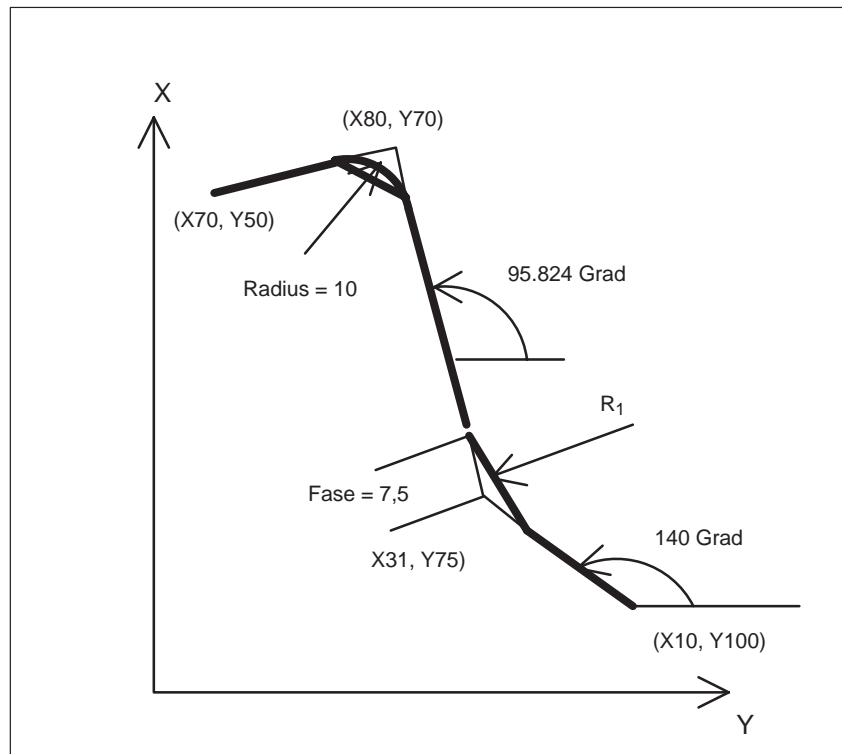


Fig. 4-25 Chamfer and corner R

**Restrictions****ISO dialect mode**

Address C is used in ISO Dialect0 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.

## Siemens mode

The identifiers for radius and chamfer are defined by machine data in Siemens mode. This prevents the occurrence of name conflicts. A comma must not be programmed before the identifier for radius or chamfer. The relevant MD are as follows:

MD for radius: \$MN\_RADIUS\_NAME

MD for chamfer: \$MN\_CHAMFER\_NAME

## Plane selection

It is only possible to carry out chamfering and corner rounding in the plane specified via plane selection (G17, G18, or G19). Parallel axes cannot be treated with these functions.

## Switching planes

A chamfering or corner rounding block can be inserted only for move commands which are performed in the same plane. In a block that comes immediately after plane switching (G17, G18, or G19 is specified), neither chamfering nor corner rounding can be specified.

## Going to the next block

A block that specifies a move command using linear interpolation (G01) or circular interpolation (G02 or G03) must follow a block specifying chamfering or corner rounding. An alarm is issued whenever the next block does not contain these specifications.

## Coordinate system

Neither chamfering nor corner rounding can be applied to a block that immediately succeeds a change of the coordinate system (G92, or G52 to G59) or a specification of a return to the reference position (G28 to G30).

## Travel distance 0

Assuming the angle between the two straight lines is within +1, the chamfering or corner rounding block is regarded as having a travel distance of zero when two linear interpolation operations are performed. Assuming the angle between the straight line and the tangent to the arc at the intersection is within +1, the corner rounding block is regarded as having a travel distance of zero when linear interpolation and circular interpolation operations are performed. Assuming the angle between the tangents to the arcs at the intersection is within +1, the corner rounding block is regarded as having a travel distance of zero when two circular interpolation operations are carried out.

**Cutting a thread**

Within a threading block, corner rounding cannot be specified.

## 4.8 Automating support functions

### 4.8.1 Skip function (G31)

By specifying "G31 X... Y... Z... 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.

Delay from the input of the skip signal to the start of processing corresponding to the input signal is shorter than 0.5 msec; this is processed at extremely high speed.

#### Format

G31 X... Y... Z... F... ;

G31: One-shot G code (It is effective only in the block in which it is specified)

#### Explanations

The coordinate values when the skip signal is turned on can be used in a macro because they are stored as follows:

\$AA\_MW[X]: Position value in work coordinate system

\$AA\_MM[X]: Position value in machine coordinate system

In ISO Dialect mode, the PLC signals are evaluated in every block, irrespective of G31. G31 activates probe1. The deleted distance to go can be calculated via the PLC Var selector.

---

#### Note

An alarm is transmitted whenever the G31 command is issued while cutter compensation is being applied. Before the G31 command is specified, cancel cutter compensation through the G40 command.

---

4.8 Automating support functions

**Example**

The next block to G31 represents an incremental command

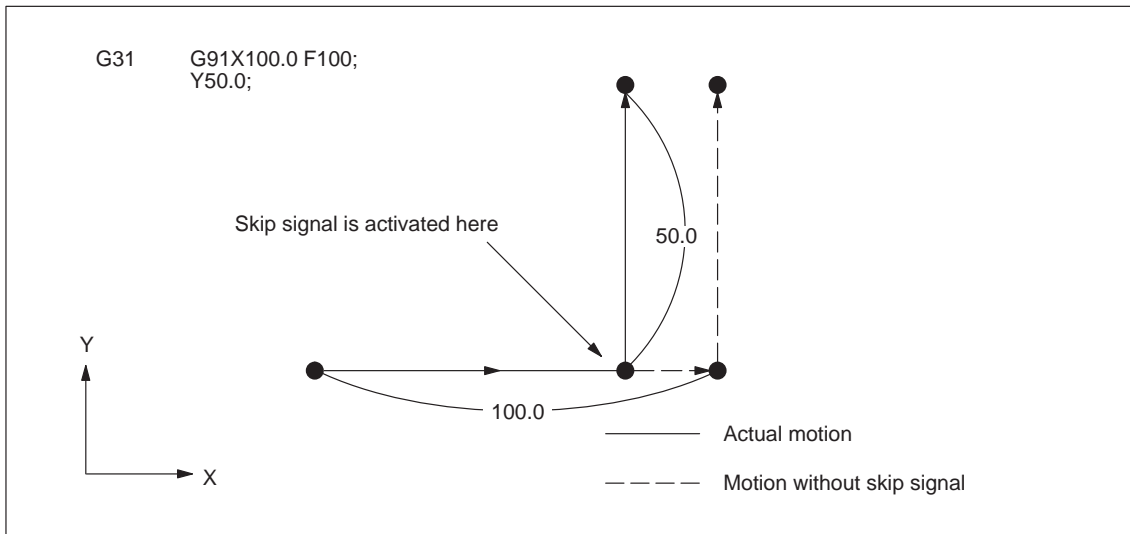


Fig. 4-26 The next block represents an incremental command

The next block to G31 represents an absolute command for 1 axis

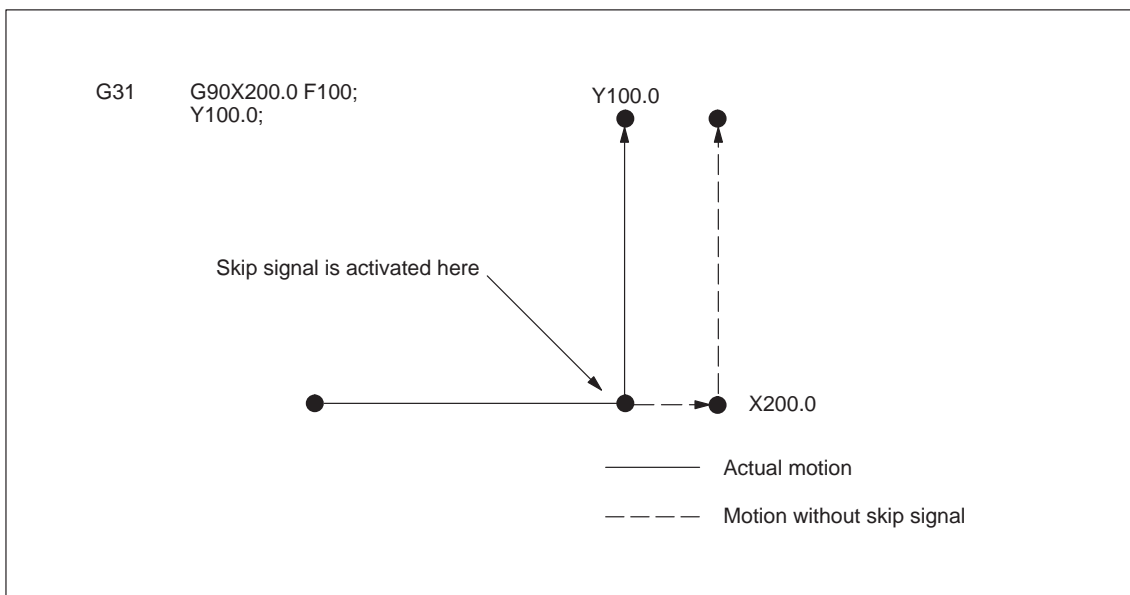


Fig. 4-27 The next block represents an absolute command for 1 axis

The next block to G31 represents an absolute command for 2 axes

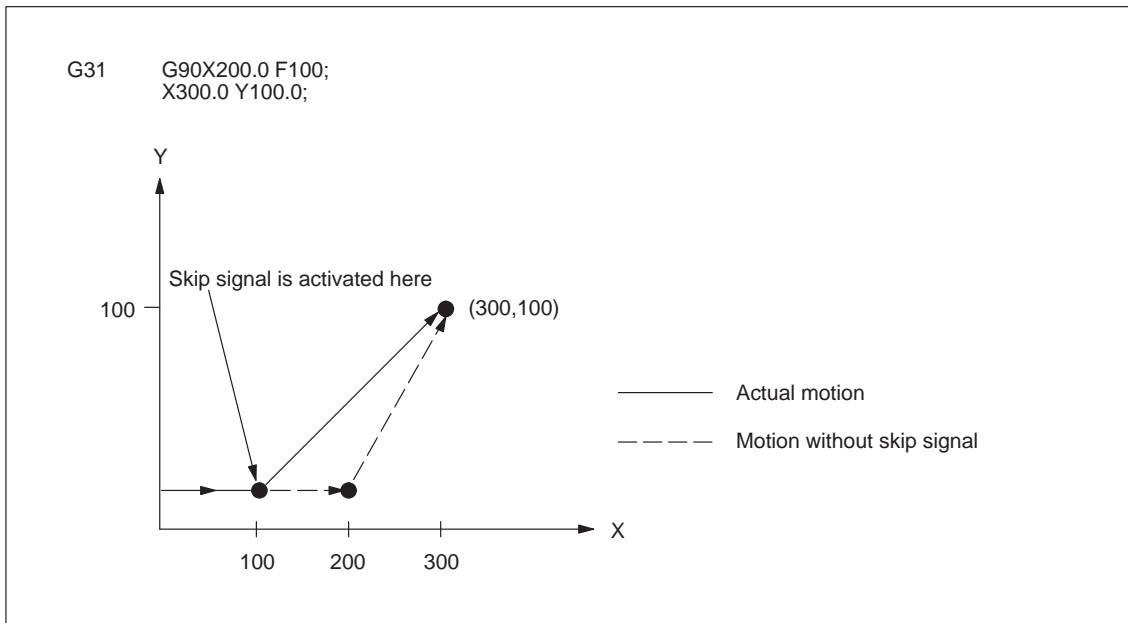


Fig. 4-28 The next block represents an absolute command for 2 axes

---

## 4.8 Automating support functions

### 4.8.2 Multistage skip (G31, P1 – P4)

The multistage skip function stores coordinates in a macro variable within a block specifying P1 to P4 after G31 whenever a skip signal (4–point) is turned on. In order to match multiple Pn (n=1,2,3,4) 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... Y... Z... F... P... ;

X, Y, Z : End point  
F... : Feedrate  
P... : P1–P4

#### Explanation

Multistage skip is caused by specifying P1, P2, P3, or P4 in a G31 block. The digital inputs are assigned to addresses P1 – P4 through machine data as follows:

P1: \$MN\_EXTERN\_MEAS\_G31\_P\_SIGNAL[0]  
P2: \$MN\_EXTERN\_MEAS\_G31\_P\_SIGNAL[1]  
P3: \$MN\_EXTERN\_MEAS\_G31\_P\_SIGNAL[2]  
P4: \$MN\_EXTERN\_MEAS\_G31\_P\_SIGNAL[3]

For an explanation of selecting (P1, P2, P3, or P4), refer to the manual supplied by the machine tool builder.



### 4.8.3 Program interrupt function (M96, M97)

By activating an external interrupt signal from the machine, another program can be called while a program is being executed. This function is referred to as program interrupt function. It is emulated using the Siemens syntax SETINT(1) <program name> [PRIO=1].

Program an interrupt command in the following format:

#### Format

M96 Pxxxx;	Enables program interrupt
M97;	Disables program interrupt

M97 and M96 P\_ should be specified in a block without other commands. If other commands such as axis move commands are specified with M97 or M96 P\_ in the same block, an alarm occurs.

#### Programming format

#### Start of interruption (M96)

By specifying "M96P . . . ;", if the program interrupt signal goes ON during the execution of the program before the execution of M97, the program presently executed is interrupted (axis move is decelerated and stopped), and the program jumps to the one specified by P.

## 4.8 Automating support functions

## Example

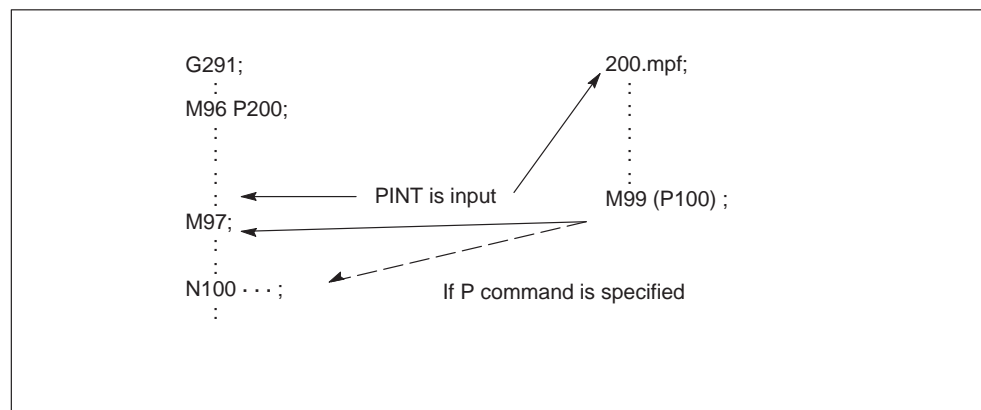


Fig. 4-29

- While the interrupt program, where jump has been made in response to the input of the interrupt signal during the execution of a program in the M96 mode, is executed, another interrupt signal is invalid.
- It is possible to specify the sequence number of the block where the interrupt program should start by using a Q command in the M96 P\_ block.

**End of interruption (M97)**

The program interrupt function is canceled by specifying "M97;".

**Supplements to the program interrupt function**

- The behaviour of the program interrupt function can be determined by setting the relevant bits of the following machine data:  
\$MN\_EXTERN\_INTERRUPT\_BITS\_M96:  
Bit 0 = 0: No interrupt function possible.  
M96/M97 are treated as standard M functions  
Bit 0 = 1: Activation of program interrupt function possible  
  
Bit 1 = 0: Part program execution is continued with the end position of the NC block subsequent to the interruption block.  
Bit 1 = 1: Part program execution is continued from the interruption position  
  
Bit 2 = 0: NC block execution is interrupted immediately and the subprogram is called.  
Bit 2 = 1: The subprogram is called after completion of the currently executed NC block.  
  
Bit 3 = 0: Machining cycle is interrupted if an interrupt signal occurs.  
Bit 3 = 1: Machining cycle is completed prior to subprogram call.

(The data bits are evaluated by the shell cycles)

- The M function to enable/disable the program interrupt function can be determined by machine data. However, M96, M97 is set as default.  
\$MN\_EXTERN\_M\_NO\_SET\_INT: enable  
\$MN\_EXTERN\_M\_NO\_DISABLE\_INT: disable
- In the program that is called up after interrupting the execution of another program, it is not allowed to specify M97 or M96. If specified, an alarm occurs.
- The M96 command can be specified in a subprogram. Jump to an interrupt program is not counted as a nesting level. Therefore, the level saved to the macro local variable does not change.
- By the execution of M99 specified in the interrupt program, the program returns to the block next to the one where the interrupt program has been called up. It is also possible to specify the return block by specifying a P command with M99. When returning to the previous program by the execution of M99, the modal information which was valid before the interruption, is recovered. However, if M99P\_ is used to return to the previous program, the modal information changed during the execution of the interrupt program is used for the execution of the previous program.
- If the interrupt signal is input during the block stop state, the program jumps to the interrupt program when the operation is started by depressing of the cycle start switch.
- The program interrupt signal is invalid if input during the execution of high-speed cutting.
- If the program interrupt signal is input during the execution of G31 (skip), the skip mode is canceled and the program interrupt function is executed.
- If the program interrupt signal is input during the execution of a block including M, S, T, or B command, the program jumps to the interrupt program. Before jumping to the interrupt program, axis move is stopped after deceleration if the interrupt signal is input during axis move. If the M or T function is being executed when the interrupt signal is input, the program does not jump until the M or T function completion signal is input.
- If the program interrupt signal is input during the execution of tapping in the solid tap mode, execution of the interrupt program starts only after the completion of the solid tap block.

#### 4.8.4 Tool life control function

Tool management, tool life and workpiece count monitoring can be reproduced with the Siemens tool management system.

## 4.9 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.

### 4.9.1 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.

### 4.9.2 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 4-7.

Table 4-7 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 4-8 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 4-9 Modal call up conditions

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

## 4.9 Macroprograms

**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.

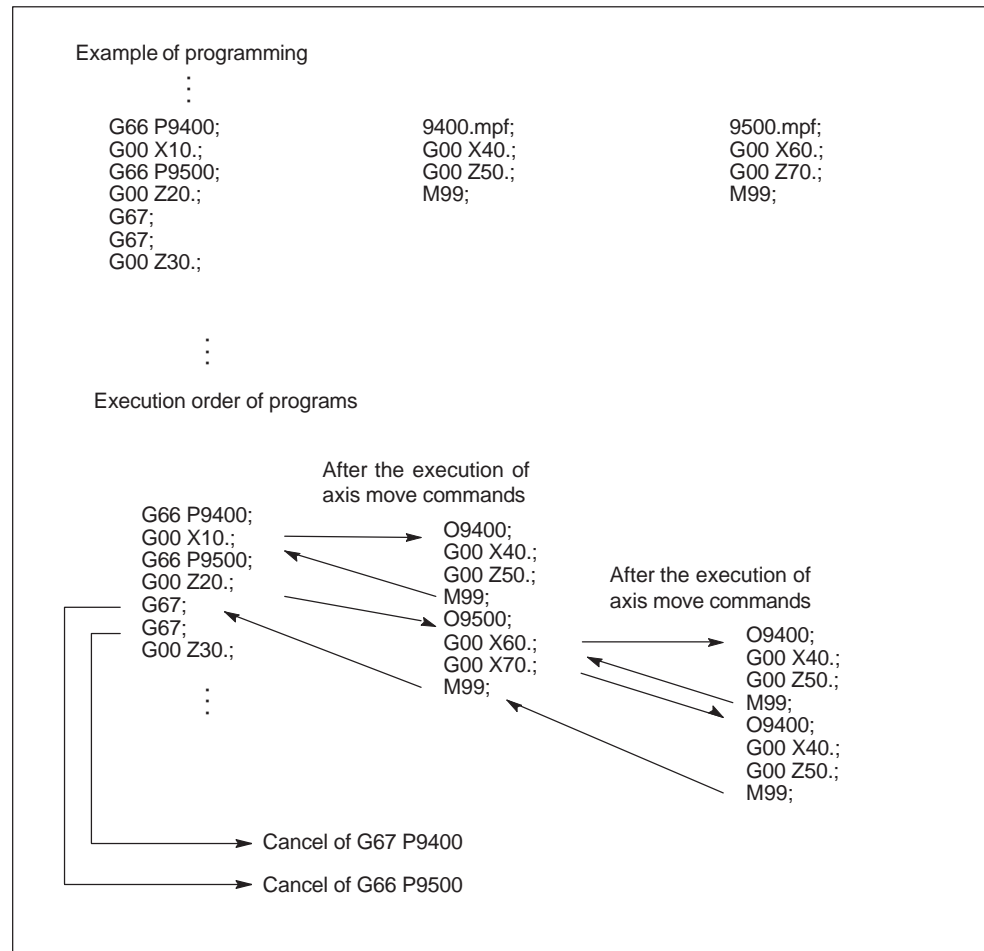


Fig. 4-30 Nesting of macroprogram call

**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 4-10 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]

## 4.9 Macroprograms

Table 4-11 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.

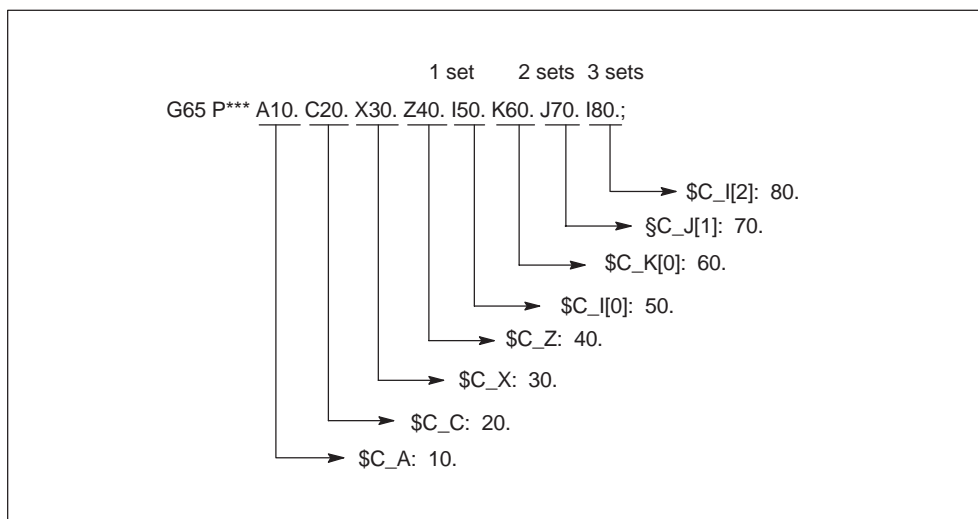


Fig. 4-31 Example of argument specification

### 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.

## 4.9 Macroprograms

**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:
G290 ; Switching into Siemens mode,
      ; if transfer parameters need to be read
N15 X_AXIS = $C_X Y_AXIS = $C_Y S_SPEED = $C_S FEED = $C_F
N20 G01 F=$C_F G95 S=$C_S
N10 G1 X=$C_X Y=$C_Y
G291 ; Switching into ISO mode
N15 M3 G54 T1
N20
...
N80 M99
```

## 4.10 Additional functions

### 4.10.1 Figure copy (G72.1, G72.2)

Using the figure copy function, a once programmed contour can be easily repeated or, respectively, copied. A linear (G72.2) or rotational (G72.1) copy can be carried out by means of this function.

#### Format

G72.1 X... Y... (Z...) P... L... R...

**X, Y, Z:** Reference point for the rotation of coordinates  
**P:** Sub-program number  
**L:** Number of sub-program repeats  
**R:** Rotation angle

Through G72.1, a sub-program containing the contour to be copied can be called repeatedly. Prior to calling each sub-program, the coordinate system is rotated by a certain angle. The coordinate rotation is carried out along the axis perpendicular to the selected plane.

G72.2 I... J... K... P... L...

**I, J, K:** X, Y, Z position prior to sub-program call  
**P:** Sub-program number  
**L:** Number of sub-program repeats

Through G72.2, a sub-program, in which the contour to be repeated is programmed, is repeatedly called. Prior to each sub-program call, the axes programmed through I, J, K are traversed incrementally. The cycle calls the sub-program by the number of times specified by address L. Prior to each sub-program call, a path programmed in I, J, K and calculated from the initial point is traversed incrementally.

## 4.10 Additional functions

## Examples

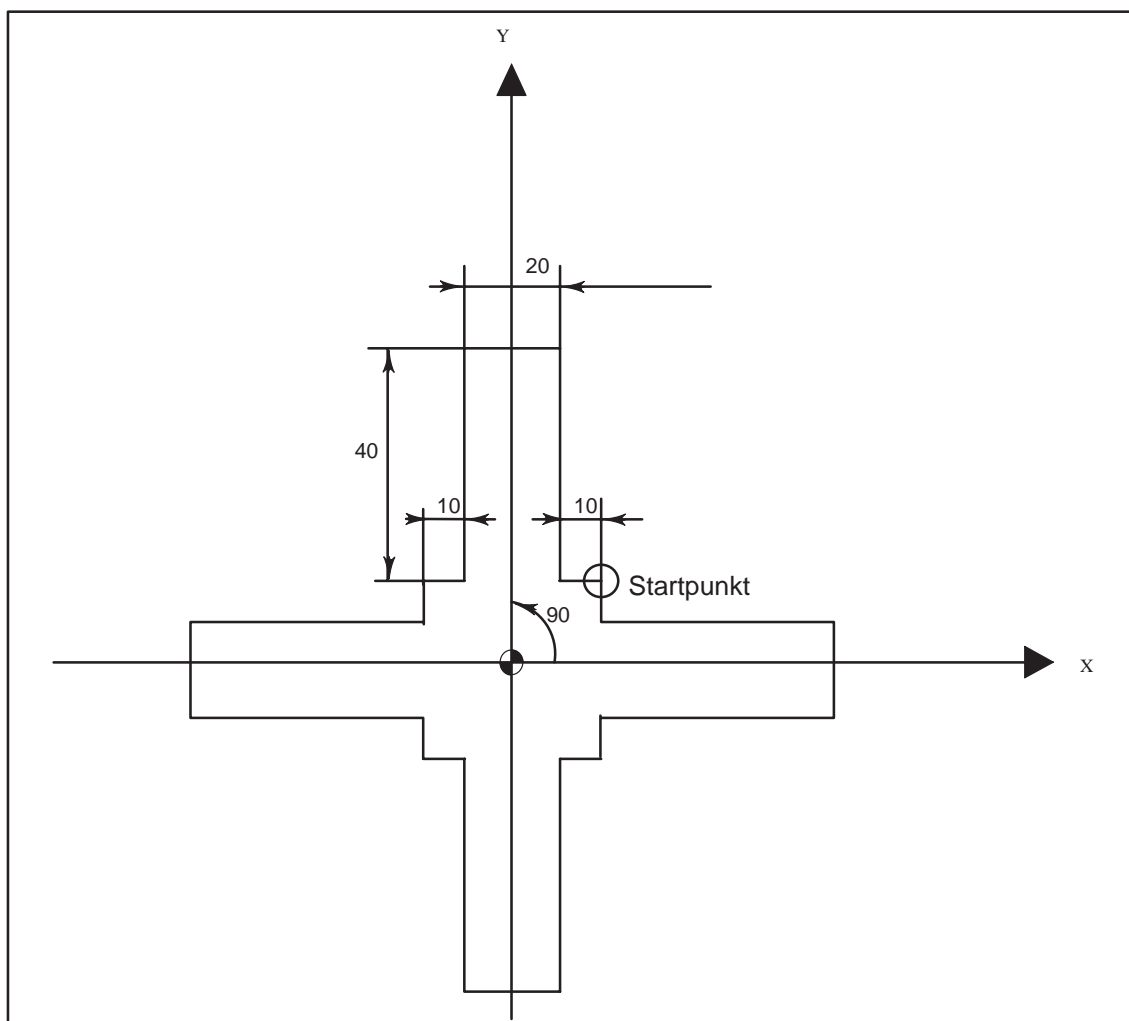


Fig. 4-32

Hauptprogramm  
 N10 G92 X40.0 Y50.0 ;  
 N20 G01 G90 G17 G41 20 Y20 D01 F1000  
 N30 G72.1 P1234 L4 X0 Y0 R90.0  
 N40 G40 G01 X100 Y50 Z0  
 N50 G00 X40.0 Y50.0 ;  
 N60 M30 ;

Unterprogramm 1234.spf  
 N100 G01 X10  
 N200 Y50  
 N300 X-10  
 N400 Y10

N500 X-20  
N600 M99

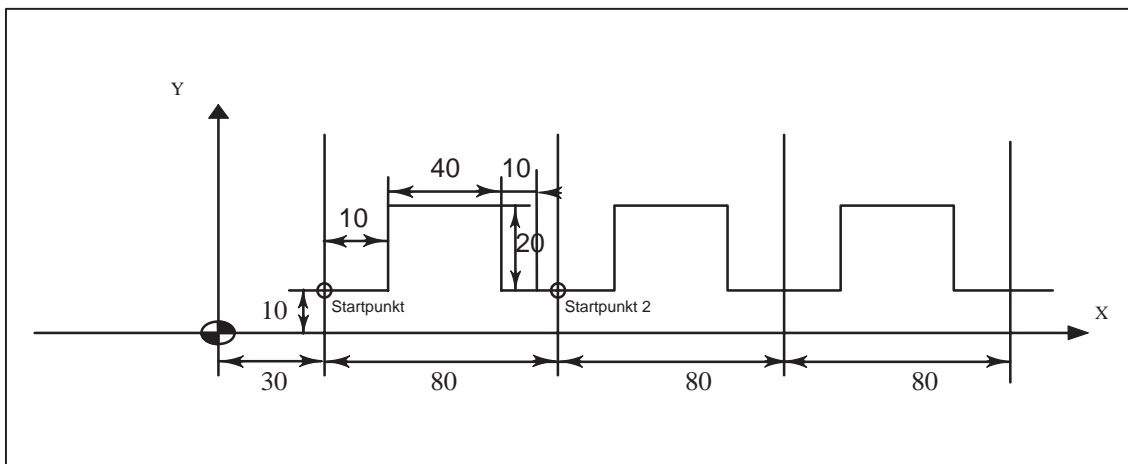


Fig. 4-33

N10 G00 G90 X0 Y0  
N20 G01 G17 G41 X30. Y0 D01 F1000  
N30 Y10.  
N40 X30.  
N50 G72.2 P2000 L3 I80. J0

O2000 G90 G01 X40.  
N100 Y30.  
N200 G01 X80.  
N300 G01 Y10.  
N400 X90.  
N500 M99

# Abbreviations

# A

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

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

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



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

---

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

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

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

---

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

<b>UFR</b>	User Frame: Zero offset
<b>V.24</b>	Serial Interface (definition of interchange lines between DTE and DCE)
<b>WCS</b>	Workpiece Coordinate System
<b>WPD</b>	Work Piece Directory
<b>ZO</b>	Zero Offset
<b>ZOA</b>	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.

# B



<b>Activate/deactivate</b>	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.
<b>Address</b>	Addresses are fixed or variable identifiers for axes (X, Y, ...), spindle speed (S), feedrate (F), circle radius (CR), etc.
<b>Alarms</b>	<p>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.</p> <p>Alarms and messages are displayed separately.</p> <p>3. Alarms and messages in the part program Alarms and messages can be displayed directly from the part program in plaintext.</p> <p>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.</p>
<b>Analog input/output module</b>	<p>Analog input/output modules are signal transducers for analog process signals.</p> <p>Analog input modules convert analog measured values into digital values that can be processed in the CPU.</p> <p>Analog output modules convert digital values into manipulated variables.</p>
<b>Approach fixed machine point</b>	Approach motion towards one of the predefined → fixed machine points.
<b>Archiving</b>	Exporting files and/or directories to an <b>external</b> storage device.

<b>Asynchronous subroutine</b>	<ul style="list-style-type: none"> <li>• 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).</li>   <li>• 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).</li> </ul>
<b>Automatic</b>	Control system operating mode (block-sequential to DIN): Mode in NC systems in which a → part program is selected and continuously executed.
<b>Auxiliary functions</b>	Auxiliary functions can be used to pass → parameters to the → PLC in → part programs, triggering reactions there which are defined by the machine manufacturer.
<b>Axes</b>	<p>CNC axes are classified according to their functional scope as:</p> <ul style="list-style-type: none"> <li>• Axes: Interpolative path axes</li> <li>• 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.</li> </ul>
<b>Axis address</b>	See → axis identifier
<b>Axis identifier</b>	<p>In compliance with DIN 66217, axes are identified as X, Y and Z for a right-handed rectangular → coordinate system.</p> <p>→ 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.</p>
<b>Axis name</b>	See → axis identifier
<b>Axis/spindle replacement</b>	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.

<b>Block search</b>	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.
<b>Booting</b>	Loading the system program after Power ON.
<b>Bus connector</b>	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.
<b>C</b>	
<b>C axis</b>	Axis about which the tool spindle describes a controlled rotational and positioning movement.
<b>C spline</b>	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.
<b>Channel structure</b>	The channel structure makes it possible to process the → programs of individual channels simultaneously and asynchronously.
<b>Circular interpolation</b>	The → tool is required to travel in a circle between defined points on the contour at a specified feed while machining the workpiece.
<b>Clearance control (3D), sensor-driven</b>	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.
<b>CNC</b>	→ NC
<b>CNC high-level language</b>	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.

<b>CNC programming language</b>	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).
<b>COM</b>	Numerical control component for the implementation and coordination of communication.
<b>Command axis</b>	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.
<b>Compensation axis</b>	Axis having a setpoint or actual value modified by the compensation value.
<b>Compensation table</b>	Table of interpolation points. It supplies the compensation values of the compensation axis for selected positions of the base axis.
<b>Compensation value</b>	Difference between the axis position measured by the position sensor and the desired, programmed axis position.
<b>Connecting cables</b>	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.
<b>Continuous-path mode</b>	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.
<b>Contour</b>	Outline of a → workpiece.
<b>Contour monitoring</b>	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.

<b>Coordinate system</b>	See → machine coordinate system, → workpiece coordinate system
<b>CPU</b>	Central Processor Unit → programmable controller
<b>Cycle</b>	Protected subroutine for executing a recurring machining operation on the → workpiece.
<b>Cycles support</b>	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.
<b>D</b>	
<b>Data block</b>	<ol style="list-style-type: none"><li>3. Data unit of the → PLC which can be accessed by → HIGHSTEP programs.</li><li>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.</li></ol>
<b>Data transfer program PCIN</b>	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.
<b>Data word</b>	A data unit, two bytes in size, within a → PLC data block.
<b>Deletion of distance-to-go</b>	Command in part program which stops machining and clears the remaining path distance to go.

<b>Design</b>	<ul style="list-style-type: none"> <li>• 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.</li> <li>• 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.</li> <li>• 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.</li> </ul>
<b>Diagnosis</b>	<ol style="list-style-type: none"> <li>3. Control operating area</li> <li>4. The control incorporates a self–diagnosis program and test routines for servicing: Status, alarm and service displays.</li> </ol>
<b>Digital input/output module</b>	Digital modules are signal transducers for binary process signals.
<b>Dimensions in metric and inch systems</b>	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).
<b>DRF</b>	Differential Resolver Function NC function which generates an incremental zero offset in AUTOMATIC mode in conjunction with an electronic handwheel.
<b>Drift compensation</b>	When the CNC axes are in the constant motion phase, automatic drift compensation is implemented in the analog speed control. (SINUMERIK FM–NC).
<b>Drive</b>	<ul style="list-style-type: none"> <li>• SINUMERIK FM–NC has an analog <math>\pm 10V</math> interface to the SIMODRIVE 611A converter system.</li> <li>• The SINUMERIK 840D control system is linked to the SIMODRIVE 611D converter system via a high–speed digital parallel bus.</li> </ul>

**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.



<b>Feedrate override</b>	<p>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.</p> <p>An offset can also be applied via motion–synchronous actions independently of the running program.</p>
<b>Finished–part contour</b>	Contour of the finished workpiece. See also → blank.
<b>Fixed machine point</b>	A point defined uniquely by the machine tool, such as the reference point.
<b>Fixed–point approach</b>	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.
<b>Frame</b>	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.
<b>G</b>	
<b>General reset</b>	<p>The following memories of the → CPU are erased by a general reset operation:</p> <ul style="list-style-type: none"><li>• → Working memory</li><li>• Read/write area of the → load memory</li><li>• → System memory</li><li>• → Backup memory</li></ul>
<b>Geometry</b>	Description of a → workpiece in the → workpiece coordinate system.
<b>Geometry axis</b>	Geometry axes are used to describe a 2 or 3–dimensional area in the workpiece coordinate system.

<b>Global main run/subroutine</b>	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.
<b>Ground</b>	"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.
<b>H</b>	
<b>Helical interpolation</b>	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.
<b>High-speed digital inputs/outputs</b>	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).
<b>HIGHSTEP</b>	Combination of the programming features for the → PLC in the S7-300/400 range.
<b>I</b>	
<b>Identifier</b>	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</b>	System of measurement that defines distances in "inches" and fractions thereof.
<b>Inclined axis</b>	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.

<b>Increment</b>	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.
<b>Increment</b>	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.
<b>Initialization block</b>	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.
<b>Initialization file</b>	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.
<b>Intermediate blocks</b>	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.
<b>Interpolation cycle</b>	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.
<b>Interpolative compensation</b>	Interpolative compensation provides a means of compensating for leadscrew errors (LEC) and measuring-system errors (MSEC) resulting from the production process.
<b>Interpolator</b>	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.

<b>Interrupt routine</b>	<p>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.</p> <p>See → ASUB</p>
<b>Inverse–time feedrate</b>	<p>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).</p>
<b>I/O module</b>	<p>I/O modules create the link between the CPU and the process. I/O modules are:</p> <ul style="list-style-type: none"> <li>• →Digital input/output modules</li> <li>• →Analog input/output modules</li> <li>• →Simulator modules</li> </ul>
<b>J</b>	
<b>Jog</b>	<p>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).</p>
<b>K</b>	
<b>Keyswitch</b>	<p><b>3. S7–300:</b> The keyswitch is the mode selector switch on the → CPU. The keyswitch is operated by means of a removable key.</p> <p><b>4. 840D/FM–NC:</b> 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.</p>
<b>K<sub>ü</sub></b>	Transmission Ratio
<b>K<sub>v</sub></b>	Servo gain factor, control variable of a control loop

**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.

<b>Machine coordinate system</b>	System of coordinates based on the axes of the machine tool.
<b>Machine zero</b>	A fixed point on the machine tool which can be referenced by all (derived) measurement systems.
<b>Machining channel</b>	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.
<b>Macros</b>	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.
<b>Main block</b>	A block prefixed by ":" containing all the parameters required to start execution of a → part program.
<b>Main program</b>	→ Part program identified by a number or name in which other main programs, subroutines or → cycles may be called.
<b>Main run</b>	Part program blocks which have been decoded and prepared by the preprocessor are executed during the "main run".
<b>MDA</b>	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.

<b>Measuring circuits</b>	<ul style="list-style-type: none"><li>• 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.</li><li>• 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.</li></ul>
<b>Messages</b>	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.
<b>Metric system</b>	Standardized system of units for lengths in millimeters (mm), meters (m), etc.
<b>Mirroring</b>	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.
<b>Mode</b>	An operating concept on a SINUMERIK control. The modes → Jog, → MDA, → Automatic are defined.
<b>Mode group</b>	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.
<b>Motion synchronization</b>	<p>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:</p> <p>Transfer M and H auxiliary functions to the PLC or deletion of distance–to–go for specific axes.</p>

**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

<b>PLC</b>	Programmable Logic Control → Speicherprogrammierbare Steuerung. Component of the → NC: Programmable controller for processing the control logic on the machine tool.
<b>PLC program memory</b>	<ul style="list-style-type: none"><li>• 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.</li><li>• 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.</li><li>• 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.</li></ul>
<b>PLC programming</b>	The PLC is programmed with the <b>STEP 7</b> software. The STEP 7 programming software is based on the standard <b>WINDOWS</b> operating system and incorporates the functionality of STEP 5 programming with innovative expansions and developments.
<b>Polar coordinates</b>	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.
<b>Polynomial interpolation</b>	Polynomial interpolation provides a means of generating a very wide range of curves, including <b>straight–line, parabolic and exponential functions</b> (SINUMERIK 840D/810D).
<b>Positioning axis</b>	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.
<b>Power ON</b>	The action of switching the control off and then on again.

<b>Preprocessing memory, dynamic</b>	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.
<b>Preprocessing stop</b>	<p>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.</p> <p>See also "Preprocessing memory".</p>
<b>Preset</b>	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.
<b>Program</b>	<ol style="list-style-type: none"><li>3. Control operating area</li><li>4. Sequence of instructions to the control system.</li></ol>
<b>Programmable frames</b>	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.
<b>Programmable logic controller</b>	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.
<b>Programmable working area limitation</b>	Limitation of the movement area of the tool to within defined, programmable limits.

<b>Programming key</b>	Characters and character sequences which have a defined meaning in the programming language for → part programs (see Programming Guide).
<b>Protection zone</b>	Three-dimensional area within a → working area which the tool tip is not permitted to enter (programmable via MD).
<b>Q Quadrant error compensation</b>	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.
<b>R R parameter</b>	Calculation parameter. The programmer can assign or request the values of the R parameter in the → part program as required.
<b>Rail</b>	This rail is used to mount the modules of the S7–300 system.
<b>Rapid traverse</b>	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.
<b>Reference point</b>	Point on the machine tool with which the measuring system of the → machine axes is referenced.
<b>Reference point approach</b>	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.

<b>REPOS</b>	<p>3. Reapproach contour, triggered by operator REPOS allows the tool to be returned to the interrupt position by means of the direction keys.</p> <p>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.</p>
<b>Revolutional feedrate</b>	The axis feedrate is adjusted as a function of the speed of the master spindle in the channel (programmed with G95).
<b>Rigid tapping</b>	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).
<b>Rotary axis</b>	Rotary axes cause the tool or workpiece to rotate to a specified angle position.
<b>Rotary axis, continuously turning</b>	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.
<b>Rotation</b>	Component of a → frame which defines a rotation of the coordinate system through a specific angle.
<b>Rounding axis</b>	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.

**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

<b>Setting data</b>	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.
<b>Softkey</b>	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.
<b>Software limit switches</b>	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.
<b>Spindles</b>	The spindle functionality is a two-level construct:  3. Spindles: Speed-controlled or position-controlled spindle drives, analog +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.
<b>Spline interpolation</b>	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.
<b>Standard cycles</b>	Standard cycles are used to program machining operations which repeat frequently: <ul style="list-style-type: none"> <li>• For drilling/milling</li> <li>• For measuring tools and workpieces</li> <li>• For turning (SINUMERIK FM-NC)</li> </ul> <p>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.</p>



<b>Subblock</b>	Block prefixed by "N" containing information for a machining step such as a position parameter.
<b>Subroutine</b>	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.
<b>Synchronization</b>	Instructions in → part programs for coordination of the operations in different → channels at specific machining points.
<b>Synchronized actions</b>	<p>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.</p> <p>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.</p> <p>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.</p>
<b>Synchronized axes</b>	Synchronized axes require the same amount of time to traverse their path as → geometry axes for their path.
<b>Synchronous spindle</b>	<p>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.</p> <p>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.</p> <p>Several pairs of synchronous spindles can be implemented.</p>

<b>System variable</b>	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.
<b>T</b>	
<b>Teach In</b>	<b>Teach In</b> 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.
<b>Text editor</b>	→ Editor
<b>Tool</b>	A tool employed to shape the workpiece, for example, a turning tool, milling cutter, drill, laser beam, grinding wheel, etc.
<b>Tool</b>	A tool for machining workpieces (e.g. drill, cutter, etc.).
<b>Tool nose radius compensation</b>	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.
<b>Tool offset</b>	A tool is selected by programming a <b>T function</b> (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.
<b>Tool radius compensation</b>	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).

<b>Transformation</b>	<p>Programming in a Cartesian coordinate system, execution in a non-Cartesian coordinate system (e.g. with machine axes as rotary axes).</p> <p>Employed in conjunction with Transmit, Inclined Axis, 5-Axis Transformation.</p>
<b>Transmit</b>	<p>This function is used to mill the outside contours on turned parts, e.g. four-sided parts (linear axis with rotary axis).</p> <p>3D interpolation with two linear axes and one rotary axis is also possible.</p> <p>The benefits afforded by Transmit are simplified programming and improved machine efficiency through complete machining: Turning and milling on the same machine without reclamping.</p>
<b>Travel to fixed stop</b>	<p>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.</p>
<b>Traversing range</b>	<p>The maximum permissible travel range for linear axes is <math>\pm 9</math> decades. The absolute value depends on the selected input and position control resolution and the unit of measurement (inch or metric).</p>
<b>U</b>	
<b>User-defined variable</b>	<p>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.</p>
<b>User memory</b>	<p>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.</p>
<b>User program</b>	<p>→ Part program</p>
<b>V</b>	
<b>Variable definition</b>	<p>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.</p>

<b>Velocity control</b>	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.
<b>Vocabulary words</b>	Words with a specific notation which have a defined meaning in the programming language for → part programs.
<b>W</b>	
<b>Working memory</b>	The working storage is a Random Access Memory in the → CPU which the processor accesses as it executes the application program.
<b>Working space</b>	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.
<b>Workpiece</b>	Part to be produced/machined by the machine tool.
<b>Workpiece contour</b>	Setpoint contour of the → workpiece to be produced/machined.
<b>Workpiece coordinate system</b>	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.
<b>Workpiece zero</b>	The workpiece zero is the origin for the → workpiece coordinate system. It is defined by its distance from the machine zero.

**X****Y****B**

**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 C describes the G codes and the functions.

**C**

## C.1 G code table

## C.1 G code table

Table C-1 G code table

G code		Description
<b>Group 1</b>		
G00 <sup>1)</sup>	1	Rapid traverse
G01	2	Linear motion
G02	3	Circle/helix, clockwise
G02.2	6	Involute interpolation, clockwise
G03	4	Circle/helix, counterclockwise
G03.2	7	Involute interpolation, counterclockwise
G33	5	Thread cutting with constant lead
<b>Group 2</b>		
G17 <sup>1)</sup>	1	XY plane
G18	2	ZX plane
G19	3	YZ plane
<b>Group 3</b>		
G90 <sup>1)</sup>	1	Absolute programming
G91	2	Incremental programming
<b>Group 4</b>		
G22	1	Working area limitation, protection zone 3 on
G23 <sup>1)</sup>	2	Working area limitation, protection zone 3 off
<b>Group 5</b>		
G93	3	Inverse time feed (rev/min)
G94 <sup>1)</sup>	1	Feed in [mm/min, inch/min]
G95	2	Feed in [mm/rev, inch/rev]
<b>Group 6</b>		
G20 <sup>1)</sup> (G70)	1	Input system inch
G21 (G71)	2	Input system metric
<b>Group 7</b>		
G40 <sup>1)</sup>	1	Deselect cutter radius compensation
G41	2	Compensation to left of contour
G42	3	Compensation to right of contour
<b>Group 8</b>		
G43	1	Tool length compensation positive on
G44	2	Tool length compensation negative on
G49 <sup>1)</sup>	3	Tool length compensation off

Table C-1 G code table

G code		Description
<b>Group 9</b>		
G73	1	Deep hole drilling cycle with chipbreaking
G74	2	Counterclockwise tapping cycle
G76	3	Fine drilling cycle
G80 <sup>1)</sup>	4	Cycle off
G81	5	Counterbore drilling cycle
G82	6	Countersink drilling cycle
G83	7	Deep hole drilling cycle with swarf removal
G84	8	Clockwise tapping cycle
G85	9	Drilling cycle
G86	10	Drilling cycle, retract using G00
G87	11	Back boring cycle
G89	13	Drilling cycle, retract using G01
<b>Group 10</b>		
G98 <sup>1)</sup>	1	Return to starting point for fixed cycles
G99	2	Return to point R for fixed cycles
<b>Group 11</b>		
G50 <sup>1)</sup>	1	Scaling off
G51	2	Scaling on
<b>Group 12</b>		
G66	1	Modal macro call
G67 <sup>1)</sup>	2	Delete modal macro call
<b>Group 13</b>		
G96	1	Constant cutting rate on
G97 <sup>1)</sup>	2	Constant cutting rate off
<b>Group 14</b>		
G54 <sup>1)</sup>	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
G54 P{1...100}		Extended zero offsets
	1	
G54 P0	1	"external zero offsets EXOFS"



## C.1 G code table

Table C-1 G code table

G code		Description
<b>Group 15</b>		
G61	1	Exact stop modal
G63	2	Tapping mode
G64 <sup>1)</sup>	3	Continuous-path mode
<b>Group 16</b>		
G17 <sup>1)</sup>	1	XY plane
G18	2	ZX plane
G19	3	YZ plane
G68	1	Coordinate rotation ON, 2D/3D
G69	2	Coordinate rotation OFF
<b>Group 17</b>		
G15 <sup>1)</sup>	1	Polar coordinates off
G16	2	Polar coordinates on
<b>Group 18 (non-modal)</b>		
G04	1	Dwell
G05	18	High-speed cycle cutting
G07.1	16	Cylindrical interpolation
G08	12	Look-ahead control
G09	2	Exact stop
G10	3	Write zero offset/tool offset
G11	4	Terminate parameter input
G27	13	Referencing check
G28	5	Approach 1st reference point
G30	6	Approach 2nd/3rd/4th reference point
G30.1	19	Floating reference position
G31	7	Measurement with touch-trigger probe
G52	8	Additive zero offset
G53	9	Approach position in machine coordinate system
G65	10	Call macro
G72.1	14	Contour repeating – rotational copy
G72.2	15	Contour repeating – linear copy
G92	11	Preset actual value memory
<b>Group 22</b>		
G50.1	1	Programmable mirror image cancel
G51.1	2	Programmable mirror image

Table C-1 G code table

G code		Description
<b>Group 25</b>		
G13.1	1	Polar coordinate interpolation mode cancel
G12.1	2	Polar coordinate interpolation mode
<b>Group 31</b>		
G290 <sup>1)</sup>	1	Select Siemens mode
G291	2	select ISO dialect mode

Note: In general, the NC establishes the G code modes identified by 1), when the power is turned ON or when the NC is reset. However, please refer to the machine tool builders documentation for actual setting.



## Machine and Setting Data

### D.1 Machine/Setting data

<b>10604</b>	<b>WALIM_GEOAX_CHANGE_MODE</b>		
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		

<b>10615</b>	<b>NCFRAME_POWERON_MASK</b>		
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.1 Machine/Setting data

<b>10652</b>	<b>CONTOUR_DEF_ANGLE_NAME</b>		
MD number	Definable name for angle in the contour short description		
Default setting: "ANG"	Minimum input limit: –	Maximum input limit: 0	
Changes effective after Power On	Protection level: 2/7	Unit: –	
Datatype: STRING	Applies with effect from SW version: 5		
Meaning:	<p>The setting is effective for Siemens G code programming only, i.e. G290.</p> <p>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 <math>\angle A</math>, it is programmed in the same way with Siemens and ISO Dialect0.</p> <p>The name must be unique, i.e. axes, variables, macros, etc. must not exist with the same name.</p>		

<b>10654</b>	<b>RADIUS_NAME</b>		
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:	<p>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 <math>R</math>, it is programmed in the same way with Siemens and ISO Dialect0.</p> <p>The name must be unique, i.e. axes, variables, macros, etc. must not exist with the same name.</p> <p>The setting is effective for Siemens G code programming, i.e. G290.</p>		

<b>10656</b>	<b>CHAMFER_NAME</b>		
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:	<p>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 Dialect0.</p> <p>The name must be unique, i.e. axes, variables, macros, etc. must not exist with the same name.</p> <p>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.</p>		

<b>10715</b>	<b>M_NO_FCT_CYCLE[0]</b>		
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:	<p>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.</p> <p>A subprogram call may not be superimposed on M functions with fixed meanings. In the event of a conflict, alarm 4150 is output:</p> <ul style="list-style-type: none"> <li>- M0 to M5,</li> <li>- M17, M30,</li> <li>- M40 to M45,</li> <li>- M function for spindle/axis mode switchover according to \$MC_SPIND_RIGID_TAPPING_M_NR (default M70)</li> <li>- M functions for nibbling/punching according to configuration via \$MC_NIBBLE_PUNCH_CODE if activated via \$MC_PUNCHNIB_ACTIVATION.</li> <li>- With applied external language (\$MN_MM_EXTERN_LANGUAGE) M19, M96-M99.</li> </ul> <p>Exception: The M functions defined for the tool change with \$MC_TOOL_CHANGE_M_CODE.</p> <p>\$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.</p> <p>Alarm 14016 is output in the event of a conflict.</p>		

## D.1 Machine/Setting data

<b>10716</b>	<b>M_NO_FCT_CYCLE_NAME[0]</b>		
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:	<p>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 movement.</p> <p>\$MN_M_NO_FCT_CYCLE is effective both in Siemens mode G290 and in external language mode G291.</p> <p>If a T number is programmed in the calling block, the programmed T number can be scanned in the cycle in variable \$P_TOOL.</p> <p>\$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.</p> <p>Alarm 14016 is output in the event of a conflict.</p>		

<b>10717</b>	<b>T_NO_FCT_CYCLE_NAME</b>		
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:	<p>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.</p> <p>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).</p> <p>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.</p> <p>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 substitution takes place, but the T word is output to the PLC.</p> <p>\$MN_T_NO_FCT_CYCLE_NAME and system variables \$C_T / \$C_TS_PROG are effective both in Siemens mode G290 and in external language mode G291.</p> <p>\$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.</p> <p>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.</p> <p>Alarm 14016 is output in the event of a conflict.</p>		

<b>10760</b>	<b>G53_TOOLCORR</b>		
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:	<p>The MD is effective in both Siemens mode and in external language mode.</p> <p>This machine data defines whether tool length compensation and tool radius compensation are suppressed with language commands G53, G153 and SUPA.</p> <p>0 = G53/G153/SUPA is non-modal suppression of zero offsets, tool length compensation and tool radius compensation remain active.</p> <p>1 = G53/G153/SUPA is non-modal suppression of zero offsets, and active tool length and tool radius compensation.</p>		

<b>10800</b>	<b>EXTERN_CHAN_SYNC_M_NO_MIN</b>		
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.		

<b>10802</b>	<b>EXTERN_CHAN_SYNC_M_NO_MAX</b>		
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:	<p>Highest number M code out of an M code number area which is reserved for channel synchronization.</p> <p>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).</p> <p>Alarm 4170 is issued if an excessive M code area is specified.</p>		

<b>10804</b>	<b>EXTERN_M_NO_SET_INT</b>		
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).		



## D.1 Machine/Setting data

<b>10806</b>	<b>EXTERN_M_NO_DISABLE_INT</b>		
MD number	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).		

<b>10808</b>	<b>EXTERN_INTERRUPT_BITS_M96</b>		
MD number	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:	<p>Using the data bits described below, the behaviour of the interruption type subprogram activated by M96 P .. can be specified.</p> <p>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</p> <p>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</p> <p>Bit 2: =0, The current NC block is interrupted immediately and the subprogram is called if the interrupt signal is detected.          =1, The subprogram is called after completion of the current NC block</p> <p>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</p>		

<b>10810</b>	<b>EXTERN_MEAS_G31_P_SIGNAL</b>		
MD number	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: –	
Datatype: BYTE	Applies with effect from SW version: 6.2		
Meaning:	<p>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.</p> <p>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.</p> <p>Bit 0: =0: Deactivate measuring input 1 for G31 P1 (–P4)          =1 Activate measuring input 1 for G31 P1 (–P4)</p> <p>Bit 1: =0 Deactivate measuring input 2 for G31 P1 (–P4)          =1 Activate measuring input 2 for G31 P1 (–P4)</p>		

<b>10880</b>	<b>EXTERN_CNC_SYSTEM</b>		
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:	<p>Selection of the external language</p> <p>1 = ISO–2: System Fanuc0 Milling (from 5.1) 2 = ISO–3: System Fanuc0 Turning (from 5.2)</p> <p>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.</p>		

<b>10882</b>	<b>NC_USER_EXTERN_GCODES_TAB [n]:0..59</b>		
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:	<p>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" —&gt; G20 is reassigned to G70; If G70 already exists, an error message appears on NCK reset.</p>		

<b>10884</b>	<b>EXTERN_FLOATINGPOINT_PROG</b>		
MD number	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:	<p>This machine data is effective for external programming languages, i.e. if MD 18800: MM_EXTERN_LANGUAGE = 1.</p> <p>The machine data defines how programmed values without decimal points are evaluated.</p> <p>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</p> <p>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</p>		

## D.1 Machine/Setting data

<b>10886</b>	<b>EXTERN_INCREMENT_SYSTEM</b>		
MD number	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:	<p>This machine data is effective for external programming languages, i.e. if MD 18800: MM_EXTERN_LANGUAGE = 1.</p> <p>This machine data defines which increment system is active</p> <p>0: Increment system IS-B = 0.001 mm/degree = 0.0001 inch</p> <p>1: Increment system IS-C = 0.0001 mm/degree = 0.00001 inch</p>		

<b>10888</b>	<b>EXTERN_DIGITS_TOOL_NO</b>		
MD number	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:	<p>The machine data is only effective with \$MN_EXTERN_CNC_SYSTEM = 2. Number of digits for tool number in programmed T value.</p> <p>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.</p>		

<b>18800</b>	<b>MM_EXTERN_LANGUAGE</b>		
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:	<p>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.</p> <p>Bit 0 (LSB): Execution of part programs in ISO_2 or ISO_3 mode. For coding see \$MN_MM_EXTERN_CNC_SYSTEM (10880)</p>		

## D.2 Channel-specific machine data

<b>20094</b>	<b>SPIND_RIGID_TAPPING_M_NR</b>		
MD number	M number for switchover to controlled spindle mode (Siemens mode)		
Default setting: 70	Minimum input limit: 0	Maximum input limit: 0xFF	
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 effective in Siemens mode and in external language mode. This machine data defines the M function number used to switch the spindle to controlled spindle mode (axis mode). This number is substituted by M70 in Siemens mode and by M29 in external language mode. Only M numbers which have not already been defined as defaults are allowed. M codes M1, M2, M3, M4, M5, M30, etc. are not allowed, for example.		

<b>20095</b>	<b>EXTERN_RIGID_TAPPING_M_NR</b>		
MD number	M number for switchover to controlled spindle mode (external language mode)		
Default setting: 29	Minimum input limit: 6	Maximum input limit: 0xFF	
Changes effective after POWER ON	Protection level: 2/7	Unit: –	
Data type: BYTE	Applies with effect from SW version:		
Meaning:	This machine data defines the M function number used to switch the spindle to controlled spindle mode (axis mode) in external language mode. This number can be used in external language mode to substitute M29 with another M function. Only M numbers which have not already been defined as defaults are allowed. M codes M0, M1, M3, M4, M5, M30, M99 etc. are not allowed, for example.		

## D.2 Channel-specific machine data

<b>20154</b>	<b>EXTERN_GCODE_RESET_VALUES[n]: 0, ..., 30</b>																								
MD number	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:	<p>The following external programming languages are possible:</p> <ul style="list-style-type: none"> <li>– ISO dialect milling</li> <li>– ISO dialect turning</li> </ul> <p>The G group classification to be used is specified in the current SINUMERIK documentation.</p> <p>The following groups can be defined within MD EXTERN_GCODE_RESET_VALUES:</p> <p><b>ISO dialect M:</b></p> <table style="margin-left: 40px;"> <tr><td>G code group 2:</td><td>G17/G18/G19</td></tr> <tr><td>G code group 3:</td><td>G90/G91</td></tr> <tr><td>G code group 5:</td><td>G94/G95</td></tr> <tr><td>G code group 6:</td><td>G20/G21</td></tr> <tr><td>G code group 13:</td><td>G96/G97</td></tr> <tr><td>G code group 14:</td><td>G54–G59</td></tr> </table> <p><b>ISO dialect T:</b></p> <table style="margin-left: 40px;"> <tr><td>G code group 2:</td><td>G96/G97</td></tr> <tr><td>G code group 3:</td><td>G90/G91</td></tr> <tr><td>G code group 5:</td><td>G94/G95</td></tr> <tr><td>G code group 6:</td><td>G20/G21</td></tr> <tr><td>G code group 16:</td><td>G17/G18/G19</td></tr> </table>			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	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
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																								
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.2 Channel-specific machine data

<b>20380</b>	<b>TOOL_CORR_MODE_G43/G44</b>		
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:	<p>The machine data is only effective if MD EXTERN_CNC_LANGUAGE = 1.</p> <p>When G43/G44 is active, it determines how length offsets programmed with H are processed.</p> <p>0: mode A The tool length H always acts on the Z axis, independent of the current plane.</p> <p>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)</p> <p>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.</p> <p>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 described under mode B.</p>		

<b>20382</b>	<b>TOOL_CORR_MOVE_MODE</b>		
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:	<p>The machine data determines how the tool length offsets are applied.</p> <p>FALSE: A tool length offset is only applied if the associated axis was programmed. (Same behaviour as in previous SW versions)</p> <p>TRUE: Tool length offsets are always applied, regardless of whether the associated axes were programmed.</p>		

<b>20732</b>	<b>EXTERN_G0_LINEAR_MODE</b>		
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:	<p>This MD determines G00 interpolation behaviour.</p> <p>0: axes move like positioning axes 1: linear interpolation</p>		

## D.2 Channel-specific machine data

<b>20734</b>	<b>EXTERN_FUNCTION_MASK</b>		
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:	<p>This MD affects functions included in the ISO mode.</p> <p>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.</p> <p>Bit 1 =0: ISO mode M G10 P&lt;100 tool geometry  &gt;100 tool wear  =1: G10 P&lt;10 000 tool geometry  &gt;10 000 tool wear</p> <p>Bit 2 =0: G04 dwell always [s] or [ms]  =1: if G95 active, dwell in rpm</p> <p>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.</p> <p>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.</p>		

<b>22420</b>	<b>FGROUP_DEFAULT_AXIS[n]: 0, ..., 7</b>		
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:	<p>You can specify up to 8 channel axes whose resulting velocity corresponds to the programmed 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.</p> <p>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</p>		

## D.2 Channel-specific machine data

<b>22512</b>	<b>EXTERN_GCODE_GROUPS_TO_PLC[n]: 0, ..., 7</b>		
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:	<p>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.</p> <p>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</p>		

<b>22900</b>	<b>STROKE_CHECK_INSIDE</b>		
MD number	Determine internal/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:	<p>This machine data applies in combination with external programming languages. It is effective with \$MN_MM_EXTERN_LANGUAGE = 1.</p> <p>It defines whether protection zone 3 is an internal or external protection zone.</p> <p>Meaning: 0: Protection zone 3 is an internal protection zone 1: Protection zone 3 is an external protection zone</p>		

<b>22910</b>	<b>WEIGHTING_FACTOR_FOR_SCALE</b>		
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:	<p>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</p> <p>Meaning: 0: Scale factor in 0.001 1: Scale factor in 0.00001</p>		



## D.2 Channel-specific machine data

<b>22914</b>	<b>AXES_SCALE_ENABLE</b>		
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: –
Datatype: 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)		

<b>22920</b>	<b>EXTERN_FEEDRATE_F1_F9_ACTIV</b>		
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:
Datatype: 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.		

<b>22930</b>	<b>EXTERN_PARALLEL_GEOAX</b>		
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.		

<b>24004</b>	<b>CHBFRAME_POWERON_MASK</b>		
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

<b>43120</b>	<b>DEFAULT_SCALE_FACTOR_AXIS</b>		
MD number	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:	<p>This machine data applies in combination with external programming languages. It is effective with \$MN_MM_EXTERN_LANGUAGE = 1.</p> <p>If no axial scale factor I, J or K is programmed in the G51 block, DEFAULT_SCALEFACTOR_AXIS is effective.</p> <p>This MD is valid only if MD AXES_SCALE_ENABLE is set.</p>		

<b>43240</b>	<b>M19_SPOS</b>		
MD number	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

<b>42110</b>	<b>DEFAULT_FEED</b>		
SD number	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:	<p>If no path feed is programmed in the part program, the value stored in \$SC_DEFAULT_FEED is used.</p> <p>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).</p>		

<b>42140</b>	<b>DEFAULT_SCALE_FACTOR_P</b>		
SD number	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.4 Channel-specific setting data

<b>42140</b>	<b>DEFAULT_SCALE_FACTOR_P</b>		
SD number	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.		

<b>42150</b>	<b>DEFAULT_ROT_FACTOR_R</b>		
SD number	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.		

<b>42160</b>	<b>EXTERN_FIXED_FEEDRATE_F1_F9</b>		
SD number	F1 digit feed F1 – 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 F1 – F9 when G01 is active.		

# Data Fields, Lists

# E

## E.1 Machine data

Number	Identifier	Name	Reference
<b>General (\$MN_ ... )</b>			
10604	WALIM_GEOAX_CHANGE_MODE	Work area limitation when switching geometrical axes	
10615	NCFRAME_POWERON_MASK	Delete global base frames on Power On	K2
10652	CONTOUR_DEF_ANGLE_NAME	Definable name for angle in the contour short description	
10654	RADIUS_NAME	Definable name for radius non-modally in the contour short description	
10656	CHAMFER_NAME	Definable name for chamfer in the contour short description	
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	

E

## E.1 Machine data

<b>General (\$MN_ ...)</b>			
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	
<b>Channel-specific (\$MC_ ...)</b>			
20050	AXCONF_GEOAX_ASSIGN_TAB[ ]	Assignment geometrical axis / channel axis	K2
20060	AXCONF_GEOAX_NAME_TAB[ ]	Geometrical axis in channel	K2
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	Reference
<b>Axis-specific</b>			
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	
<b>Channel-specific</b>			
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	



# 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 defined	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 language MD18800: \$MN_MM_EX-TERN_LANGUAGE or option bit 19800 \$ON_EXTERN_LANGUAGE is not set



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 programmed	CYCLE384M	The programmed spindle direction conflicts with the spindle direction used for the cycle
61808	Final drilling depth or single drilling 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 code not possible	CYCLE383T, CYCLE384T, CYCLE385T	
61811	ISO axis designation not admissible	CYCLE328, CYCLE330, CYCLE371T, CYCLE374T, CYCLE376T, CYCLE383T, 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 admissible numeral value in cycle setting data
61814	Polar coordinate not possible	CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	
61815	G40 not active	CYCLE374T, CYCLE376T	G40 was not active prior to the cycle call

## References

### General Documentation

**/BU/** SINUMERIK 840D/840Di/810D/802S, C, D  
Ordering Information  
Catalog NC 60.1  
Order number: E86060–K4460–A101–A8–7600

**/ST7/** **SIMATIC**  
SIMATIC S7 Programmable Logic Controller  
Catalog ST 70  
Order number: E86060–K4670–A111–A3

**/ZI/** SINUMERIK, SIROTEC, SIMODRIVE  
Accessories and Equipment for Special Purpose Machines  
Catalog NC Z  
Order number: E86060–K4490–A001–A7–7600

### Electronic Documentation

**/CD6/** The SINUMERIK system (10.00 Edition)  
DOC ON CD  
(includes all SINUMERIK 840D/840Di/810D/FM–NC and  
SIMODRIVE publications)  
Order number: 6FC5 298–6CA00–0BG0

## User Documentation

<b>/AUE/</b>	SINUMERIK 840D/810D/FM-NC <b>AutoTurn Graphic Programming System</b> (07.99 Edition) Part 2: Setup Order No: 6FC5 298-4AA50-0BP2
<b>/AUK/</b>	SINUMERIK 840D/810D/FM-NC <b>Short Guide AutoTurn Operation</b> (07.99 Edition) Order number: 6FC5 298-4AA30-0BP2
<b>/AUP/</b>	SINUMERIK 840D/810D/FM-NC <b>AutoTurn Graphic Programming System</b> (07.99 Edition) Operator's Guide Part 1: Programming Order number: 6FC5 298-4AA40-0BP2
<b>/BA/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>Operator's Guide</b> (10.00 Edition) Order number: 6FC5 298-6AA00-0BP0  <ul style="list-style-type: none"> <li>• Operator's Guide</li> <li>• Operator's Guide HMI Advanced</li> </ul>
<b>/BAE/</b>	SINUMERIK 840D/810D/FM-NC <b>Operator's Guide Unit Operator Panel</b> (04.96 Edition) Order number: 6FC5 298-3AA60-0BP1
<b>/BAH/</b>	SINUMERIK 840D/840Di/810D <b>Operator's Guide HT 6 (HPU New)</b> (06.00 Edition) Order number: 6FC5 298-0AD60-0BP0

<b>/BAK/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>Short Operation Guide</b> Order number: 6FC5 298-6AA10-0BP0	(10.00 Edition)
<b>/BAM/</b>	SINUMERIK 810D/840D <b>Operator's Guide ManualTurn</b> Order number: 6FC5 298-5AD00-0BP0	(02.00 Edition)
<b>/KAM/</b>	SINUMERIK 840D/810D <b>Short Guide ManualTurn</b> Order number: 6FC5 298-2AD40-0BP0	(11.98 Edition)
<b>/BAS/</b>	SINUMERIK 840D/810D <b>Operator's Guide ShopMill</b> Order number: 6FC5 298-5AD10-0BP1	(08.00 Edition)
<b>/KAS/</b>	SINUMERIK 840D/810D <b>Short Guide ShopMill</b> Order number: 6FC5 298-2AD30-0BP0	(01.98 Edition)
<b>/BAP/</b>	SINUMERIK 840D/840Di/810D <b>Short Guide Handheld Programming Unit</b> Order number: 6FC5 298-5AD20-0BP1	(04.00 Edition)
<b>/BNM/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>User's Guide Measuring Cycles</b> Order number: 6FC5 298-5AA70-0BP2	(06.00 Edition)
<b>/DA/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>Diagnostics Guide</b> Order number: 6FC5 298-6AA20-0BP0	(10.00 Edition)

<b>/PG/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>Programming Guide Fundamentals</b> (10.00 Edition) Order number: 6FC5 298-6AB00-0BP0
<b>/PGA/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>Programming Guide Advanced</b> (10.00 Edition) Order number: 6FC5 298-6AB10-0BP0
<b>/PGK/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>Short Guide Programming</b> (10.00 Edition) Order number: 6FC5 298-6AB30-0BP0
<b>/PGZ/</b>	SINUMERIK 840D/840Di/810D/FM-NC <b>Programming Guide Cycles</b> (10.00 Edition) Order number: 6FC5 298-6AB40-0BP0
<b>/PI/</b>	<b>PCIN 4.4</b> Software for Data Transfer to/from MMC Module Order number: 6FX2 060-4AA00-4XB0 (German, English, French) Order from: WK Fürth
<b>/SYI/</b>	SINUMERIK 840Di <b>System Overview</b> (01.00 Edition) Order number: 6FC5 298-5AE40-0BP0

## Manufacturer/Service Documentation

### a) Lists

<b>/LIS/</b>	SINUMERIK 840D/840Di/810D/FM-NC SIMODRIVE 611D	
	<b>Lists</b>	(10.00 Edition)
	Order number: 6FC5 297-6AB70-0BP0	

### b) Hardware

<b>/BH/</b>	SINUMERIK 840D/840Di/810D/FM-NC	
	<b>Operator Components Manual (HW)</b>	(10.00 Edition)
	Order number: 6FC5 297-6AA50-0BP0	

<b>/BHA/</b>	SIMODRIVE <b>Sensor</b> <b>Absolute Encoder with Profibus DP</b>	
	User's Guide (HW)	(02.99 Edition)
	Order number: 6SN1 197-0AB10-0YP1	

<b>/EMV/</b>	SINUMERIK, SIROTEC, SIMODRIVE <b>EMC Installation Guideline</b>	(06.99 Edition)
	Planning Guide (HW)	
	Order number: 6FC5 297-0AD30-0BP1	

<b>/PHC/</b>	SINUMERIK 810D	
	<b>Configuring Manual (HW)</b>	(10.00 Edition)
	Order number: 6FC5 297-4AD10-0BP0	

<b>/PHD/</b>	SINUMERIK 840D	
	<b>NCU 561.2-573.2 Configuring Manual (HW)</b>	(10.00 Edition)
	Order number: 6FC5 297-6AC10-0BP0	

---

<b>/PHF/</b>	SINUMERIK FM-NC <b>NCU 570 Configuring Manual (HW)</b> Order number: 6FC5 297-3AC00-0BP0	(04.96 Edition)
<b>/PMH/</b>	SIMODRIVE <b>Sensor</b> <b>Measuring System for Main Spindle Drives</b> SIMAG-H Configuring/Installation Guide (HW) Order number: 6SN1197-0AB30-0BP0	(05.99 Edition)

**c) Software**  
**/FB1/**

SINUMERIK 840D/840Di/810D/FM-NC

**Description of Functions, Basic Machine** (Part 1) (10.00 Edition)

(the various manuals are listed below)

Order number: 6FC5 297-6AC20-0BP0

- A2 Various Interface Signals
- A3 Axis Monitoring, Protection Zones
- B1 Continuous Path Mode, Exact Stop and Look Ahead
- B2 Acceleration
- D1 Diagnostic Tools
- D2 Interactive Programming
- F1 Travel to Fixed Stop
- G2 Velocities, Setpoint/Actual Value Systems, Closed-Loop Control
- H2 Output of Auxiliary Functions to PLC
- K1 Mode Group, Channels, Program Operation
- K2 Coordinate Systems, Axis Types, Axis Configurations, Actual-Value System for Workpiece, External Zero Offset
- K4 Communication
- N2 EMERGENCY STOP
- P1 Transverse Axes
- P3 Basic PLC Program
- R1 Reference Point Approach
- S1 Spindles
- V1 Feeds
- W1 Tool Compensation



**/FB2/**

SINUMERIK 840D/840Di/810D(CCU2)/FM-NC

**Description of Functions, Extended Functions** (Part 2) (10.00 Edition)

including FM-NC: Turning, Stepping Motor

(the various manuals are listed below)

Order number: 6FC5 297-6AC30-0BP0

- A4 Digital and Analog NCK I/Os
- B3 Several Operator Panels and NCUs
- B4 Operation via PG/PC
- F3 Remote Diagnostics
- H1 Jog with/without Handwheel
- K3 Compensations
- K5 Mode Groups, Channels, Axis Replacement
- L1 FM-NC Local Bus
- M1 Kinematic Transformation
- M5 Measurements
- N3 Software Cams, Position Switching Signals
- N4 Punching and Nibbling
- P2 Positioning Axes
- P5 Oscillation
- R2 Rotary Axes
- S3 Synchronous Spindles
- S5 Synchronized Actions (SW 3 and earlier)
- S6 Stepper Motor Control
- S7 Memory Configuration
- T1 Indexing Axes
- W3 Tool Change
- W4 Grinding

**/FB3/**

SINUMERIK 840D/840Di/810D(CCU2)/FM-NC

**Description of Functions, Special Functions** (Part 3) (10.00 Edition)

(the various manuals are listed below)

Order number: 6FC5 297-6AC80-0BP0

- F2 3-Axis to 5-Axis Transformation
- G1 Gantry Axes
- G3 Cycle Times
- K6 Contour Tunnel Monitoring
- M3 Coupled Motion and Master/Slave Couplings
- S8 Constant Workpiece Speed for Centerless Grinding
- T3 Tangential Control
- V2 Preprocessing
- W3 3D Tool Radius Compensation
- TE1 Clearance Control
- TE2 Analog Axis
- TE3 Master-Slave for Drives
- TE4 Transformation Package Handling
- TE5 Setpoint Exchange
- TE6 MCS Coupling

- /FBA/** SIMODRIVE 611D/SINUMERIK 840D/810D  
**Description of Functions, Drive Functions** (10.00 Edition)  
(the various sections are listed below)  
Order number: 6SN1 197-0AA80-0BP6
- DB1 Operational Messages/Alarm Reactions
  - DD1 Diagnostic Functions
  - DD2 Speed Control Loop
  - DE1 Extended Drive Functions
  - DF1 Enable Commands
  - DG1 Encoder Parameterization
  - DM1 Calculation of Motor/Power Section Parameters and Controller Data
  - DS1 Current Control Loop
  - DÜ1 Monitors/Limitations
- /FBAN/** SINUMERIK 840D/SIMODRIVE 611 DIGITAL  
Description of Functions  
**ANA MODULE** (02.00 Edition)  
Order number: 6SN1 197-0AB80-0BP0

<b>/FBD/</b>	SINUMERIK 840D	
	Description of Functions <b>Digitizing</b>	(07.99 Edition)
	Order number: 6FC5 297–4AC50–0BP0	
	DI1	Start–Up
	DI2	Scanning with Tactile Sensors (scancad scan)
	DI3	Scanning with Lasers (scancad laser)
	DI4	Milling Program Generation (scancad mill)
<b>/FBDN/</b>	CAM Integration DNC NT–2000	
	Description of Functions	
	<b>System for NC Data Management and Data Distribution</b>	(05.00 Edition)
	Order number: 6FC5 297–6AE50–0BP0	
<b>/FBFA/</b>	SINUMERIK 840D/840Di/810D	
	Description of Functions	
	<b>ISO Dialects for SINUMERIK</b>	(10.00 Edition)
	Order number: 6FC5 297–6AE10–0BP0	
<b>/FBHLA/</b>	SINUMERIK 840D/SIMODRIVE 611 digital	
	Description of Functions	
	<b>HLA Module</b>	(08.99 Edition)
	Order number: 6SN1 197–0AB60–0BP1	
<b>/FBMA/</b>	SINUMERIK 840D/810D	
	Description of Functions <b>ManualTurn</b>	(02.00 Edition)
	Order number: 6FC5 297–5AD50–0BP0	

- /FBO/** SINUMERIK 840D/810D/FM-NC  
 Description of Functions  
**Configuring of OP 030 Operator Interface** (03.96 Edition)  
 (the various sections are listed below)  
 Order number: 6FC5 297-3AC40-0BP0
- BA Operator's Guide  
 EU Development Environment (Configuring Package)  
 PS Online only: Configuring Syntax (Configuring Package)  
 PSE Introduction to Configuring of Operator Interface  
 IK Screen Kit: Software Update and Configuration
- /FBP/** SINUMERIK 840D  
 Description of Functions **C-PLC Programming** (03.96 Edition)  
 Order number: 6FC5 297-3AB60-0BP0
- /FBR/** SINUMERIK 840D/810D  
 Description of Functions  
**SINCOM Computer Link** (02.00 Edition)  
 Order number: 6FC5 297-5AD60-0BP0
- NFL Interface to Central Production Computer  
 NPL Interface to PLC/NCK
- /FBSI/** SINUMERIK 840D/SIMODRIVE  
 Description of Functions **SINUMERIK Safety Integrated** (05.00 Edition)  
 Order number: 6FC5 297-5AB80-0BP1
- /FBSP/** SINUMERIK 840D/810D  
 Description of Functions **ShopMill** (08.00 Edition)  
 Order number: 6FC5 297-5AD80-0BP1

<b>/FBST/</b>	<p><b>SIMATIC FM STEPDRIVE/SIMOSTEP</b></p> <p>Description of Functions (11.98 Edition) Order number: 6SN1 197-0AA70-0YP3</p>	
<b>/FBSY/</b>	<p>SINUMERIK 840D/840Di/810D(CCU2)</p> <p>Description of Functions <b>Synchronized Actions</b> (10.00 Edition) for Wood, Glass, Ceramics and Presses Order number: 6FC5 297-6AD40-0BP0</p>	
<b>/FBTD/</b>	<p>SINUMERIK 840D/810D</p> <p>Description of Functions <b>SINTDI Tool Data Information System</b> with Online Help (04.99 Edition) Order number: 6FC5 297-5AE00-0BP0</p>	
<b>/FBU/</b>	<p><b>SIMODRIVE 611 universal</b></p> <p>Description of Functions (05.00 Edition) Closed-Loop Control Component for Speed Control and Positioning Order number: 6SN1 197-0AB20-0BP3</p>	
<b>/FBW/</b>	<p>SINUMERIK 840D/840Di/810D</p> <p>Description of Functions <b>Tool Management</b> (07.00 Edition) Order number: 6FC5 297-5AC60-0BP2</p>	
<b>/HBI/</b>	<p>SINUMERIK 840Di</p> <p><b>Manual</b> (06.00 Edition) Order number: 6FC5 297-5EA60-0BP0</p>	
<b>/IK/</b>	<p>SINUMERIK 840D/810D/FM-NC</p> <p><b>Screen Kit for MMC 100/Unit Operator Panel</b> (06.96 Edition) Description of Functions: Software Update and Configuration Order number: 6FC5 297-3EA10-0BP1</p>	

<b>/KBU/</b>	<b>SIMODRIVE 611 universal</b>	
	Short Description Closed-Loop Control Component for Speed Control Order number: 6SN1 197-0AB40-0BP3	(05.00 Edition)
<b>/PJLM/</b>	<b>SIMODRIVE</b> Planning Guide <b>Linear Motors</b>	(05.00 Edition)
	(on request) ALL General Information about Linear Motors 1FN1 1FN1 Three-Phase AC Linear Motor 1FN3 1FN3 Three-Phase AC Linear Motor CON Connections Order number: 6SN1 197-0AB70-0BP1	
<b>/PJM/</b>	<b>SIMODRIVE</b> Planning Guide <b>Motors</b>	
	AC Motors for Feed and Main Spindle Drives Order number: 6SN1 197-0AA20-0BP4	(09.00 Edition)
<b>/PJMS/</b>	<b>SIMODRIVE</b> Planning Guide <b>1FE1 Synchronous Built-In Motors</b>	
	AC Motors for Main Spindle Drives Order number: (on request)	(03.00 Edition)
<b>/PJU/</b>	<b>SIMODRIVE 611-A/611-D</b> Planning Guide <b>Converters</b>	(08.98 Edition)
	Transistor PWM Inverters for AC Feed Drives and AC Main Spindle Drives Order number: 6SN1 197-0AA00-0BP4	

<b>/POS1/</b>	<b>SIMODRIVE POSMO A</b> Operator's Guide Distributed Positioning Motor on PROFIBUS DP, Order number: 6SN2 197-0AA00-0BP1	(02.00 Edition)
<b>/POS2/</b>	<b>SIMODRIVE POSMO A</b> Installation Instructions (supplied with every POSMO A) Order number: 462 008 0815 00	(12.98 Edition)
<b>/POS3/</b>	<b>SIMODRIVE POSMO SI/CD/CA</b> Operator's Guide Distributed Servo Drive Systems Order number: 6SN2 197-0AA20-0BP0	(01.01 Edition)
<b>/S7H/</b>	<b>SIMATIC S7-300</b> Manual: Assembly, CPU Data (HW) Reference Manual: Module Data Order number: 6ES7 398-8AA03-8AA0	(10.98 Edition)
<b>/S7HT/</b>	<b>SIMATIC S7-300</b> STEP 7 Manual, Basic Information, V. 3.1 Order number: 6ES7 810-4CA02-8AA0	(03.97 Edition)
<b>/S7HR/</b>	<b>SIMATIC S7-300</b> Manual STEP 7, Reference Manuals, V. 3.1 Order number: 6ES7 810-4CA02-8AR0	(03.97 Edition)
<b>/S7SI/</b>	<b>SIMATIC S7-300</b> <b>FM 353</b> Stepper Drive Positioning Module Order in conjunction with configuring package	(04.97 Edition)
<b>/S7LI/</b>	<b>SIMATIC S7-300</b> <b>FM 354</b> Servo Drive Positioning Module Order in conjunction with configuring package	(04.97 Edition)



- /S7M/ SIMATIC S7–300**  
**FM 357** Multi–Axis Module for Servo and Stepper Drives (10.99 Edition)  
 Order in conjunction with configuring package
- /SHM/ SIMODRIVE 611**  
 Manual (01.98 Edition)  
 Single–Axis Positioning Control for **MCU 172A**  
 Order number: 6SN 1197–4MA00–0BP0
- /SPI/ SIMODRIVE 611–A/611–D,**  
**SimoPro 3.1**  
 Program for Configuring Machine–Tool Drives  
 Order number: 6SC6 111–6PC00–0AAo  
 Order from: WK Fürth
- d) Installation and start–up**
- /IAA/ SIMODRIVE 611A**  
**Installation and Start–Up Guide** (09.00 Edition)  
 Order number: 6SN 1197–0AA60–0BP6  
 (incl. description of start–up software)
- /IAC/ SINUMERIK 810D**  
**Installation and Start–Up Guide** (10.00 Edition)  
 (including description of SIMODRIVE 611D start–up software)  
 Order number: 6FC5 297–4AD20–0BP0
- /IAD/ SINUMERIK 840D/SIMODRIVE 611D**  
**Installation and Start–Up Guide** (10.00 Edition)  
 (including description of SIMODRIVE 611D start–up software)  
 Order number: 6FC5 297–6AB10–0BP0

**/IAF/** SINUMERIK FM-NC  
**Installation and Start-Up Guide** (04.96 Edition)  
Order number: 6FC5 297-3AB00-0BP0

**/IAM/** SINUMERIK 840D/840Di/810D  
**HMI/MMC Installation and Start-Up Guide** (10.00 Edition)  
Order number: 6FC5 297-6AE20-0BP0

IM1 Start-up functions for the MMC 100.2  
IM3 Start-up functions for the MMC 103  
IM4 Start-up functions for HMI Advanced (PCU 50)  
HE1 Online help  
BE1 Expand the operator interface



# Index

## A

Absolute/incremental designation, 3-57  
 Additional functions, 4-155  
 Alarms, F-225  
 Argument specification, 4-151  
 Automatic coordinate system, 3-52  
 Automatic reference point return for rotary axes, 2-38  
 Automatic return to reference point, 2-37  
 Automating support functions, 4-141

## B

Base coordinate system, 3-45, 3-46

## C

Canned cycles, 4-86  
 Chamfering and corner rounding commands, 4-137  
 Coordinate system, 3-44  
 Cutting feed, 1-19

## D

Designation of multiple M codes in a single block, 3-84  
 Determining the coordinate value input modes, 3-57  
 Dwell, 3-66

## E

Entry prohibited area, 4-135  
 Error messages, F-225

## F

F command, 1-19  
 F1–digit feed, 1-22  
 Feed per minute function, 1-23

## G

G code table, C-200  
 G02.2, ISO–M, C-200  
 General purpose M codes, 3-83

## H

Helical interpolation, 2-35  
 HMI, A-162

## I

Inch/metric input designation, 3-58  
 Interference check, 3-76  
 Internally processed M codes, 3-83  
 Interpolation commands, 2-26

## L

Least input increment, 1-16

## M

M codes relating to stop operation, 3-82  
 M function, 3-82  
 Machine data, D-205  
   channel–specific, D-213  
   general, D-205  
   list, E-221  
 Macroprogram call, 4-148  
 Macroprograms, 4-148  
 Maximum programmable values for axis movement, 1-16, 1-17  
 Miscellaneous function, 3-82  
 MMC, A-164  
 Modal call up, 4-149, 4-150

## N

Numerically controlled axes, 1-16

**O**

Optional block skip, 1-17

**P**

Positioning, 2-26

Positioning in the Error Detect ON Mode, 2-26

Program interrupt function, 4-145

Program support functions, 4-86, 4-135

Programmable data input, 4-125

**R**

Rapid traverse, 1-19

Reference point return, 2-37

Reference point return check, 2-39

**S**

S function, 3-81

Scaling, 3-60

Second miscellaneous function, 3-84

Second to fourth reference point return, 2-41

Setting data

axis-specific, D-219

channel-specific, D-219

list, E-223

Simple call up, 4-149

Skip function, 4-141

Spindle function, 3-81

Stored stroke limit B and C, 4-135

Subprograms, 4-148

**T**

T function, 3-82

Tool function, 3-82

Tool length offset, 3-68

Tool offset data memory, 3-68

Tool offset functions, 3-68

Tool radius offset C function, 3-71

## Commands

### G

G03.2, ISO-M, C-200

G08, ISO-M, C-202

G09, ISO-M, C-202

G11, ISO-M, C-202

G121, ISO-M, C-203

G131, ISO-M, C-203

G15 , ISO-M, C-202

G16, ISO-M, C-202

G17 , ISO-M, C-200

G18, ISO-M, C-200

G19, ISO-M, C-200

G22, ISO-M, C-200

G23 , ISO-M, C-200

G43, ISO-M, C-200

G44, ISO-M, C-200

G49 , ISO-M, C-200

G50 , ISO-M, C-201

G50.1, ISO-M, C-202

G51, ISO-M, C-201

G51.1, ISO-M, C-202

G54 , ISO-M, C-201

G61, ISO-M, C-202

G63, ISO-M, C-202

G64 , ISO-M, C-202

G68, ISO-M, C-202

G69, ISO-M, C-202

G72.1, ISO-M, C-202

G72.2, ISO-M, C-202

G73, ISO-M, C-201

G74, ISO-M, C-201

G76, ISO-M, C-201

G80 , ISO-M, C-201

G81, ISO-M, C-201

G82, ISO-M, C-201

G83, ISO-M, C-201

G84, ISO-M, C-201

G85, ISO-M, C-201

G86, ISO-M, C-201

G87, ISO-M, C-201

G89, ISO-M, C-201

G93, ISO-M, C-200

G96, ISO-M, C-201

G97 , ISO-M, C-201

G98 , ISO-M, C-201

G99, ISO-M, C-201



## 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

© Siemens Japan K.K. All rights reserved.