



*Changes for the Better*

**MITSUBISHI CNC**

**700/70 Series**

**Programming Manual (M2/M0 Format)**





MELDAS is a registered trademark of Mitsubishi Electric Corporation.  
Other company and product names that appear in this manual are trademarks or registered trademarks of the respective companies.



## Introduction

This manual is a guide for using the MITSUBISHI CNC 700/70 Series.

Programming for M2/M0 format is described in this manual, so read this manual thoroughly before starting programming. Thoroughly study the "Precautions for Safety" on the following page to ensure safe use of this NC unit.

### Details described in this manual

#### CAUTION

-  For items described in "Restrictions" or "Usable State", the instruction manual issued by the machine tool builder takes precedence over this manual.
-  An effort has been made to note as many special handling methods in this user's manual. Items not described in this manual must be interpreted as "not possible".
-  This manual has been written on the assumption that all option functions are added. Refer to the specifications issued by the machine tool builder before starting use.
-  Refer to the Instruction Manual issued by each machine tool builder for details on each machine tool.
-  Some screens and functions may differ depending on the NC system or its version, and some functions may not be possible. Please confirm the specifications before use.

### General precautions

- (1) Refer to the following documents for details on handling  
MITSUBISHI CNC 700/70 Series Instruction Manual ..... IB-1500042



## Precautions for Safety

Always read the specifications issued by the machine maker, this manual, related manuals and attached documents before installation, operation, programming, maintenance or inspection to ensure correct use.

Understand this numerical controller, safety items and cautions before using the unit.

This manual ranks the safety precautions into "DANGER", "WARNING" and "CAUTION".



### DANGER

When the user may be subject to imminent fatalities or major injuries if handling is mistaken.



### WARNING

When the user may be subject to fatalities or major injuries if handling is mistaken.



### CAUTION

When the user may be subject to injuries or when physical damage may occur if handling is mistaken.

Note that even items ranked as "  CAUTION", may lead to major results depending on the situation. In any case, important information that must always be observed is described.



### DANGER

Not applicable in this manual.



### WARNING

Not applicable in this manual.



### CAUTION

#### 1. Items related to product and manual

-  For items described as "Restrictions" or "Usable State" in this manual, the instruction manual issued by the machine tool builder takes precedence over this manual.
-  An effort has been made to describe special handling of this machine, but items that are not described must be interpreted as "not possible".
-  This manual is written on the assumption that all option functions are added. Refer to the specifications issued by the machine tool builder before starting use.
-  Refer to the Instruction Manual issued by each machine tool builder for details on each machine tool.
-  Some screens and functions may differ depending on the NC system or its version, and some functions may not be possible. Please confirm the specifications before use.

(Continued on next page)

## CAUTION

### 2. Items related to operation

-  Before starting actual machining, always carry out dry operation to confirm the machining program, tool compensation amount and workpiece offset amount, etc.
-  If the workpiece coordinate system offset amount is changed during single block stop, the new setting will be valid from the next block.
-  Turn the mirror image ON and OFF at the mirror image center.
-  Refer to the Instruction Manual issued by each machine tool builder for details on each machine tool.
-  If the tool compensation amount is changed during automatic operation (including during single block stop), it will be validated from the next block or blocks onwards.

### 3. Items related to programming

-  The commands with "no value after G" will be handled as "G00".
-  "EOB", "%", and "EOR" are symbols used for explanation. The actual codes for ISO are "CR, LF" ("LF") and "%".  
The programs created on the Edit screen are stored in the NC memory in a "CR, LF" format, however, the programs created with external devices such as the FLD or RS-232C may be stored in an "LF" format.  
The actual codes for EIA are "EOB (End of Block)" and "EOR (End of Record)".
-  When creating the machining program, select the appropriate machining conditions, and make sure that the performance, capacity and limits of the machine and NC are not exceeded. The examples do not consider the machining conditions.
-  Do not change fixed cycle programs without the prior approval of the machine tool builder.
-  When programming a program of the multi-part system, carefully observe the movements caused by other part systems' programs.

# Contents

<b>1. Control Axes</b> .....	<b>1</b>
1.1 Coordinate Words and Control Axis.....	1
1.2 Coordinate Systems and Coordinate Zero Point Symbols.....	2
<b>2. Least Command Increments</b> .....	<b>3</b>
2.1 Input Setting Units.....	3
2.2 Input Command Increment Tenfold.....	5
2.3 Indexing Increment.....	6
<b>3. Data Formats</b> .....	<b>7</b>
3.1 Tape Codes.....	7
3.2 Program Formats.....	10
3.3 Tape Memory Format.....	13
3.4 Optional Block Skip.....	13
3.4.1 Optional Block Skip ; /.....	13
3.4.2 Optional Block Skip Addition ; /n.....	14
3.5 Program/Sequence/Block Numbers ; L(O), N.....	16
3.6 Parity H/V.....	17
3.7 G Code Lists.....	18
3.8 Precautions Before Starting Machining.....	21
<b>4. Buffer Register</b> .....	<b>22</b>
4.1 Input Buffer.....	22
4.2 Pre-read Buffers.....	23
<b>5. Position Commands</b> .....	<b>24</b>
5.1 Position Command Methods ; G90, G91.....	24
5.2 Inch/Metric Command Change; G20, G21.....	26
5.3 Decimal Point Input.....	28
<b>6. Interpolation Functions</b> .....	<b>33</b>
6.1 Positioning (Rapid Traverse); G00.....	33
6.2 Linear Interpolation; G01.....	40
6.3 Plane Selection; G17, G18, G19.....	42
6.4 Circular Interpolation; G02, G03.....	44
6.5 R-specified Circular Interpolation; G02, G03.....	49
6.6 Helical Interpolation ; G17 to G19, G02, G03.....	52
6.7 Thread Cutting.....	56
6.7.1 Constant Lead Thread Cutting; G33.....	56
6.7.2 Inch Thread Cutting; G33.....	60
6.8 Unidirectional Positioning; G60.....	61
6.9 Cylindrical Interpolation; G07.1.....	63
6.10 Polar Coordinate Interpolation; G12.1, G13.1.....	71
6.11 Exponential Function Interpolation; G02.3, G03.3.....	78
6.12 Polar Coordinate Command; G16/G15.....	84
6.13 Spiral/Conical Interpolation; G02.0/G03.1(Type1), G02/G03(Type2).....	90
6.14 3-dimensional Circular Interpolation; G02.4, G03.4.....	95
6.15 NURBS Interpolation.....	100
6.16 Hypothetical Axis Interpolation; G07.....	105
<b>7. Feed Functions</b> .....	<b>107</b>
7.1 Rapid Traverse Rate.....	107
7.2 Cutting Feedrate.....	107
7.3 F1-digit Feed.....	108
7.4 Per-minute/Per-revolution Feed (Asynchronous/Synchronous Feed); G94, G95.....	110

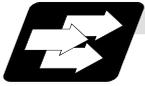
7.5	Inverse Time Feed; G93 .....	112
7.6	Feedrate Designation and Effects on Control Axes .....	116
7.7	Rapid Traverse Constant Inclination Acceleration/Deceleration .....	120
7.8	Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration .....	122
7.9	Exact Stop Check; G09.....	131
7.10	Exact Stop Check Mode; G61 .....	134
7.11	Deceleration Check.....	134
7.11.1	G1 -> G0 Deceleration Check.....	136
7.11.2	G1 -> G1 Deceleration Check.....	137
7.12	Automatic Corner Override .....	138
7.13	Tapping Mode; G63 .....	143
7.14	Cutting Mode; G64.....	143
<b>8.</b>	<b>Dwell.....</b>	<b>144</b>
8.1	Per-second Dwell ; G04 .....	144
<b>9.</b>	<b>Miscellaneous Functions .....</b>	<b>146</b>
9.1	Miscellaneous Functions (M8-digits BCD) .....	146
9.2	Secondary Miscellaneous Functions (B8-digits, A8 or C8-digits) .....	148
9.3	Index Table Indexing.....	149
<b>10.</b>	<b>Spindle Functions.....</b>	<b>151</b>
10.1	Spindle Functions (S6-digits Analog) .....	151
10.2	Spindle Functions (S8-digits) .....	151
10.3	Constant Surface Speed Control; G96, G97 .....	152
10.3.1	Constant Surface Speed Control .....	152
10.4	Spindle Clamp Speed Setting; G92 .....	153
10.5	Spindle/C Axis Control .....	154
10.6	Multiple Spindle Control .....	157
10.6.1	Multiple Spindle Control II .....	158
<b>11.</b>	<b>Tool Functions (T command).....</b>	<b>160</b>
11.1	Tool Functions (T8-digit BCD).....	160
<b>12.</b>	<b>Tool Compensation Functions .....</b>	<b>161</b>
12.1	Tool Compensation .....	161
12.2	Tool Length Compensation/Cancel; G43/G44 .....	165
12.3	Tool Length Compensation in the Tool Axis Direction ; G43.1/G44.....	168
12.4	Tool Radius Compensation; G38, G39/G40/G41,G42 .....	175
12.4.1	Tool Radius Compensation Operation.....	176
12.4.2	Other Commands and Operations During Tool Radius Compensation .....	185
12.4.3	G41/G42 Commands and I, J, K Designation.....	194
12.4.4	Interrupts During Tool Radius Compensation.....	200
12.4.5	General Precautions for Tool Radius Compensation.....	202
12.4.6	Changing of Compensation No. During Compensation Mode .....	203
12.4.7	Start of Tool Radius Compensation and Z Axis Cut in Operation.....	205
12.4.8	Interference Check .....	207
12.4.9	Diameter Designation of Compensation Amount.....	214
12.4.10	Workpiece Coordinate Changing During Radius Compensation .....	216
12.5	Three-dimensional Tool Radius Compensation ; G40/G41,G42.....	218
12.6	Tool Position Offset; G45 to G48 .....	229
12.7	Programmed Compensation Input; G10, G11.1.....	236
12.8	Compensation Data Input to Variable by Program; G11.....	241
12.9	Inputting the Tool Life Management Data; G10, G11 .....	242
12.9.1	Inputting the Tool Life Management Data by G10 L3 Command.....	242
12.9.2	Inputting the Tool Life Management Data by G10 L30 Command.....	244
12.9.3	Precautions for Inputting the Tool Life Management Data.....	247

<b>13. Program Support Functions .....</b>	<b>248</b>
13.1 Fixed Cycles.....	248
13.1.1 Standard Fixed Cycles; G80 to G89, G73, G74, G75, G76 .....	248
13.1.2 Drilling Cycle with High-Speed Retract.....	276
13.1.3 Initial Point and R Point Level Return; G98, G99.....	279
13.1.4 Setting of Workpiece Coordinates in Fixed Cycle Mode.....	281
13.2 Special Fixed Cycle; G34, G35, G36, G37 .....	282
13.3 Subprogram Control; G22, G22 .....	287
13.3.1 Calling Subprogram with G22 and G22 Commands.....	287
13.3.2 Figure Rotation; G22 I_ J_ K_ .....	291
13.4 Variable Commands.....	294
13.5 User Macro Specifications .....	299
13.5.1 User Macro Commands; G65, G66, G66.1, G67, G68(G23).....	299
13.5.2 Macro Call Command .....	300
13.5.3 ASCII Code Macro .....	309
13.5.4 Variables.....	313
13.5.5 Types of Variables .....	315
13.5.6 Arithmetic Commands.....	353
13.5.7 Control Commands.....	358
13.5.8 External Output Commands.....	361
13.5.9 Precautions .....	363
13.5.10 Actual Examples of Using User Macros.....	365
13.6 G Command Mirror Image; G50.1, G51.1 / G62 .....	369
13.7 Corner Chamfering/Corner Rounding I .....	372
13.7.1 Corner Chamfering " ,C_ " .....	372
13.7.2 Corner Rounding " ,R_ " .....	374
13.8 Linear Angle Command .....	375
13.9 Geometric Command .....	376
13.10 Circle Cutting; G12, G13 .....	380
13.11 Parameter Input by Program; G10, G11.1 .....	382
13.12 Macro Interrupt; ION, IOF .....	383
13.13 Tool Change Position Return; G30.1 to G30.6 .....	391
13.14 Normal Line Control ; G40.1/G41.1/G42.1 .....	394
13.15 High-accuracy Control ; G61.1, G08 .....	405
13.16 High-speed Machining Mode; G05, G05.1 .....	419
13.16.1 High-speed Machining Mode I,II; G05 P1, G05 P2.....	419
13.17 High-speed High-accuracy Control; G05, G05.1.....	422
13.17.1 High-speed High-accuracy Control I, II.....	422
13.17.2 SSS Control .....	428
13.18 Spline; G05.1 .....	433
13.19 High-accuracy Spline Interpolation ; G61.2.....	440
13.20 Scaling; G50/G51 .....	442
13.21 Coordinate Rotation by Program; G68.1/G69.1 .....	447
13.22 Coordinate Rotation Input by Parameter; G10.....	455
13.23 3-dimensional Coordinate Conversion; G68.1/69.1 .....	458
13.24 Tool Center Point Control; G43.4/G43.5 .....	475
13.25 Timing-synchronization between Part Systems .....	497
13.26 End Point Error Check Cancellation; G69.....	500
13.27 Coordinate Read Function; G14 .....	502
<b>14. Coordinates System Setting Functions .....</b>	<b>505</b>
14.1 Coordinate Words and Control Axes.....	505
14.2 Basic Machine, Workpiece and Local Coordinate Systems.....	506
14.3 Machine Zero Point and 2nd, 3rd, 4th Reference Positions.....	507
14.4 Basic Machine Coordinate System Selection; G53.....	508

14.5	Coordinate System Setting; G92.....	509
14.6	Automatic Coordinate System Setting .....	510
14.7	Reference (Zero) Position Return; G28, G29.....	511
14.8	2nd, 3rd and 4th Reference (Zero) Position Return; G30 .....	515
14.9	Reference Position Check; G27 .....	518
14.10	Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1) .....	519
14.11	Local Coordinate System Setting; G52 .....	531
14.12	Workpiece Coordinate System Preset; G92.1 .....	535
14.13	Coordinate System for Rotary Axis .....	540
<b>15.</b>	<b>Measurement Support Functions .....</b>	<b>543</b>
15.1	Automatic Tool Length Measurement; G37.1 .....	543
15.2	Skip Function; G31.....	547
15.3	Multi-step Skip Function; G31.n, G04 .....	552
15.4	Multi-step Skip Function 2; G31 .....	554
15.5	Speed Change Skip; G31 .....	556
15.6	Programmable Current Limitation .....	559
15.7	Stroke Check Before Travel; G22.1/G23.1 .....	560
<b>Appendix 1.</b>	<b>Program Error .....</b>	<b>562</b>
<b>Appendix 2.</b>	<b>Order of G Function Command Priority.....</b>	<b>582</b>
<b>INDEX</b> .....		<b>X-1</b>

# 1. Control Axes

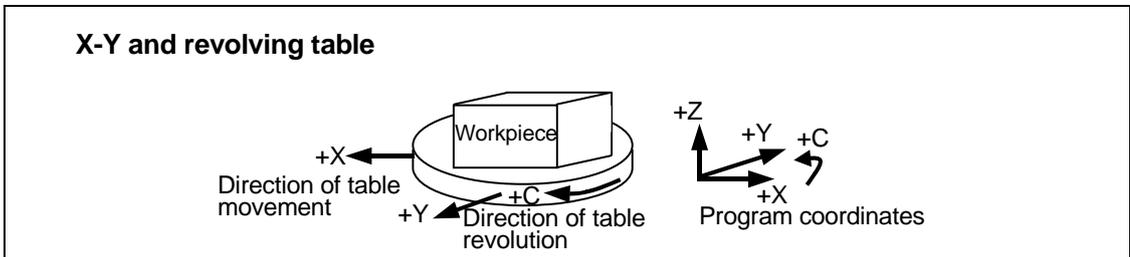
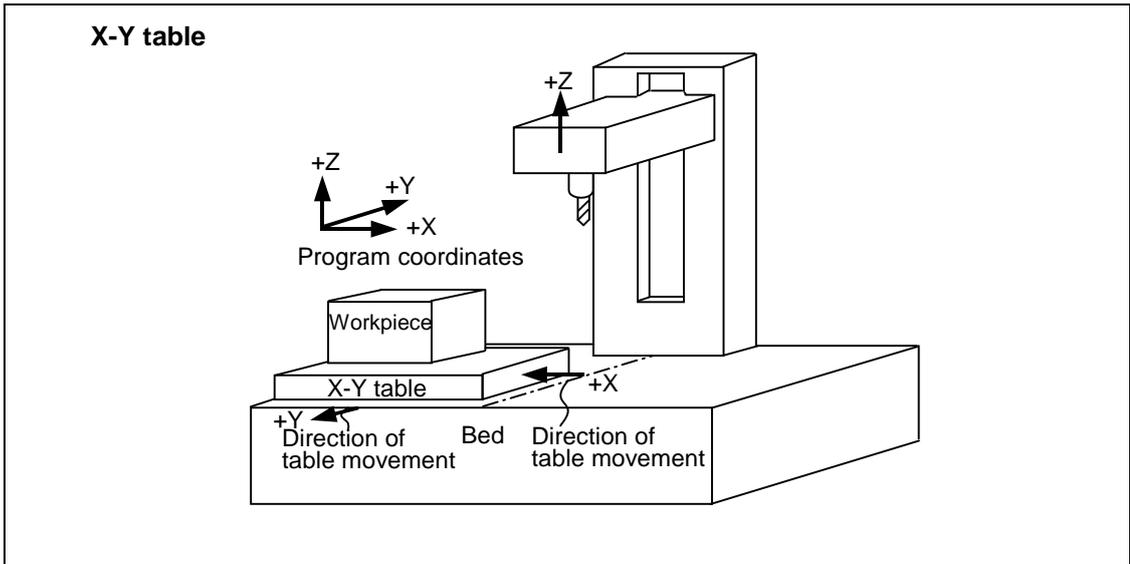
## 1.1 Coordinate Words and Control Axis



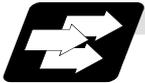
### Function and purpose

In the standard specifications, there are 3 control axes, but, by adding an additional axis, up to 4 axes can be controlled.

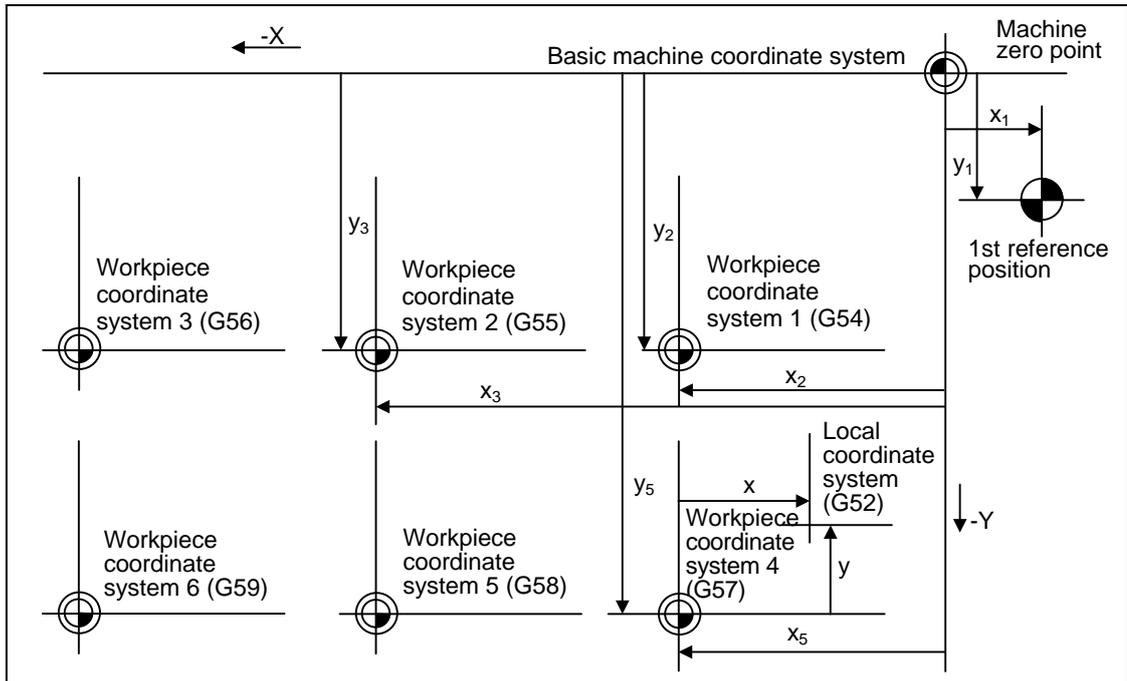
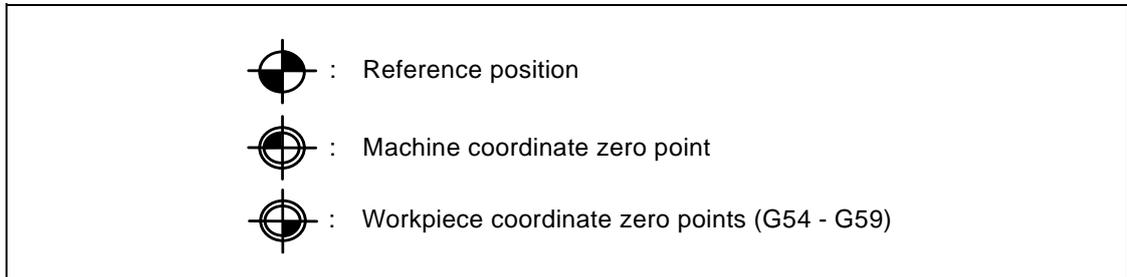
The designation of the processing direction responds to those axes and uses a coordinate word made up of alphabet characters that have been decided beforehand.



1.2 Coordinate Systems and Coordinate Zero Point Symbols



Function and purpose



## 2. Least Command Increments

### 2.1 Input Setting Units



#### Function and purpose

The input setting units are, as with the compensation amounts, the units of setting data used in common for all axes.

The command units are the movement amounts in the program which are commanded with MDI inputs or command tape. These are expressed with mm, inch or degree (°) units.

With the parameters, the command units are decided for each axis, and the input setting units are decided commonly for all axes.

	Parameters	Linear axis		Rotation axis (°)
		Millimeter	Inch	
Input setting unit	#1003 iunit = B	0.001	0.0001	0.001
	= C	0.0001	0.00001	0.0001
	= D	0.00001	0.000001	0.00001
	= E	0.000001	0.0000001	0.000001
Command unit	#1015 cunit = 0	Follow #1003 iunit		
	= 1	0.0001	0.00001	0.0001
	= 10	0.001	0.0001	0.001
	= 100	0.01	0.001	0.01
	= 1000	0.1	0.01	0.1
	= 10000	1.0	0.1	1.0

**(Note 1)** Inch/metric changeover is performed in either of 2 ways: conversion from the parameter screen (#1041 I\_inch: valid only when the power is turned ON) and conversion using the G command (G20 or G21).

However, when a G command is used for the conversion, the conversion applies only to the input command increments and not to the input setting units.

Consequently, the tool offset amounts and other compensation amounts as well as the variable data should be preset to correspond to inches or millimeters.

**(Note 2)** The millimeter and inch systems cannot be used together.

**(Note 3)** During circular interpolation on an axis where the input command increments are different, the center command (I, J, K) and the radius command (R) can be designated by the input setting units. (Use a decimal point to avoid confusion.)



## Detailed description

**(1) Units of various data**

These input setting units determine the parameter setting unit, program command unit and the external interface unit for the PLC axis and handle pulse, etc. The following rules show how the unit of each data changes when the input setting unit is changed. This table applies to the NC axis and PLC axis.

Data	Unit system	Setting value	Input setting unit			
			1 $\mu$ m (B)	0.1 $\mu$ m (C)	10nm (D)	1nm (E)
Speed data Example: rapid	Milli- metre	20000 (mm/min)	20000	20000	20000	20000
		Setting range	1 to 999999	1 to 999999	1 to 999999	1 to 999999
	Inch	2000 (inch/min)	2000	2000	2000	2000
		Setting range	1 to 999999	1 to 999999	1 to 999999	1 to 999999
Position data Example: SoftLimit+	Milli- metre	123.123 (mm)	123.123	123.1230	123.12300	123.123000
		Setting range	$\pm$ 99999.999	$\pm$ 99999.9999	$\pm$ 99999.99999	$\pm$ 99999.999999
	Inch	12.1234 (inch)	12.1234	12.12340	12.123400	12.1234000
		Setting range	$\pm$ 9999.9999	$\pm$ 9999.99999	$\pm$ 9999.999999	$\pm$ 9999.9999999
Interpolation unit data	Milli- metre	1 ( $\mu$ m)	2	20	200	2000
		Setting range	$\pm$ 9999	$\pm$ 9999	$\pm$ 9999	$\pm$ 9999
	Inch	0.001 (inch)	2	20	200	2000
		Setting range	$\pm$ 9999	$\pm$ 9999	$\pm$ 9999	$\pm$ 9999

**(2) Program command**

The program command unit follows the above table.

If the data has a decimal point, the number of digits in the integer section will remain and the number of digits in the decimal point section will increase as the input setting unit becomes smaller.

When setting data with no decimal point, and which is a position command, the data will be affected by the input setting increment and input command increment.

For the feed rate, as the input setting unit becomes smaller, the number of digits in the integer section will remain the same, but the number of digits in the decimal point section will increase.

2.2 Input Command Increment Tenfold



Function and purpose

The program's command increment can be multiplied by an arbitrary scale with the parameter designation.

This function is valid when a decimal point is not used for the command increment.

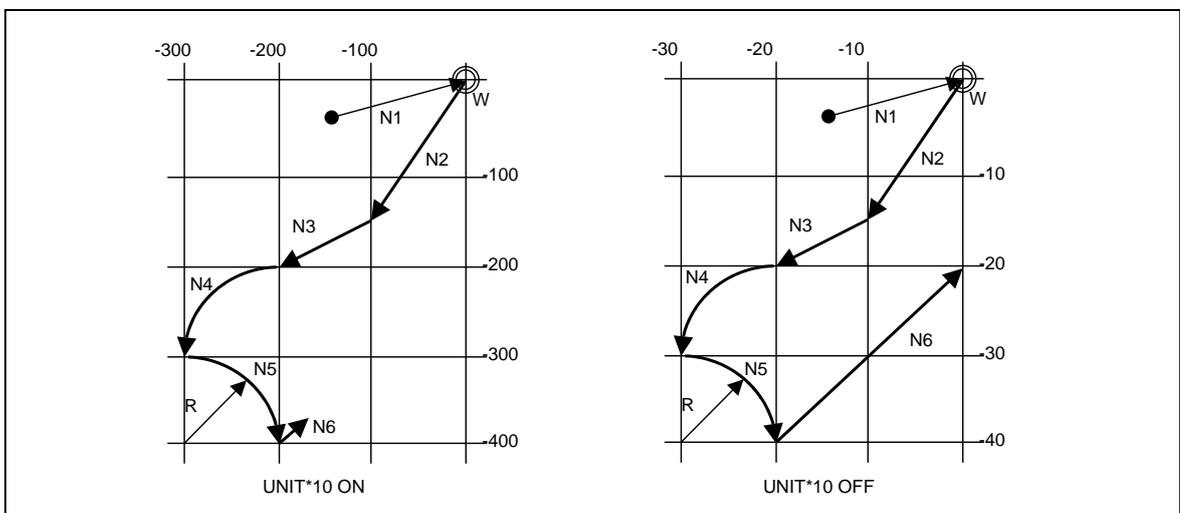
The scale is set with the parameters.



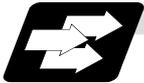
Detailed description

- (1) When running a machining program already created with a 10µm input command increment with a CNC unit for which the command increment is set to 1µm and this function's parameter value is set to "10", machining similar to before this function is possible.
- (2) When running a machining program already created with a 1µm input command increment with a CNC unit for which the command increment is set to 0.1µm and this function's parameter value is set to "10", machining similar to before this function is possible.
- (3) This function cannot be used for the dwell function G04\_X\_(P\_);.
- (4) This function cannot be used for the compensation amount of the tool compensation input.
- (5) This function can be used when decimal point type I is valid, but cannot be used when decimal point type II is valid.

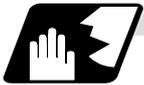
Program example (Machining program: programmed with 1=10µm) (CNC unit is 1=1µm system)	"UNIT*10" parameter			
	10		1	
	X	Y	X	Y
N1 G90 G00 X0 Y0;	0	0	0	0
N2 G91 X-10000 Y-15000;	-100.000	-150.000	-10.000	-15.000
N3 G01 X-10000 Y-5000 F500;	-200.000	-200.000	-20.000	-20.000
N4 G03 X-10000 Y-10000 J-10000;	-300.000	-300.000	-30.000	-30.000
N5 X10000 Y-10000 R5000;	-200.000	-400.000	-20.000	-40.000
N6 G01 X20.000 Y.20.000	-180.000	-380.000	0.000	-20.000



## 2.3 Indexing Increment

**Function and purpose**

This function limits the command value for the rotary axis.  
This can be used for indexing the rotary table, etc. It is possible to cause a program error with a program command other than an indexing increment (parameter setting value).

**Detailed description**

When the indexing increment (parameter) for limiting the command value is set, the rotary axis can be positioned with that indexing increment. If a program other than the indexing increment setting value is commanded, a program error (P20) will occur.

The indexing position will not be checked when the parameter is set to 0.

**(Example)** When the indexing increment setting value is 2 degrees, only command with the 2-degree increment are possible.

G90 G01 C102.000 ; ... Moves to the 102 degree angle.

G90 G01 C101.000 : ... Program error

G90 G01 C102 ; ... Moves to the 102 degree angle. (Decimal point type II)

The following axis specification parameter is used.

#	Item	Contents	Setting range (unit)
2106	Index unit : Indexing increment	Set the indexing increment to which the rotary axis can be positioned.	0 to 360 (°)

**Precautions**

- When the indexing increment is set, degree increment positioning takes place.
- The indexing position is checked with the rotary axis, and is not checked with other axes.
- When the indexing increment is set to 2 degrees, the rotary axis is set to the B axis, and the B axis is moved with JOG to the 1.234 position, an indexing error will occur if "G90B5." or "G91B5." is commanded.

## 3. Data Formats

### 3.1 Tape Codes



#### Function and purpose

The tape command codes used for this controller are combinations of alphabet letters (A, B, C, ... Z), numbers (0, 1, 2 ... 9) and signs (+, -, / ...). These alphabet letters, numbers and signs are referred to as characters. Each character is represented by a combination of 8 holes which may, or may not, be present.

These combinations make up what is called codes.

This controller uses, the ISO code (R-840).

**(Note 1)** If a code not given in the tape code table in Fig. 1 is assigned during operation, program error (P32) will result.

**(Note 2)** For the sake of convenience, a semicolon " ; " has been used in the CNC display to indicate the end of a block (EOB/IF) which separates one block from another. Do not use the semicolon key, however, in actual programming but use the keys in the following table instead.

#### CAUTION

 "EOB", "%", and "EOR" are symbols used for explanation. The actual codes for ISO are "CR, LF" ("LF") and "%".

The programs created on the Edit screen are stored in the NC memory in a "CR, LF" format, however, the programs created with external devices such as the FLD or RS-232C may be stored in an "LF" format.

The actual codes for EIA are "EOB (End of Block)" and "EOR (End of Record)".



#### Detailed description

##### EOB/EOR keys and displays

Key used	Code used	ISO	Screen display
End of block		LF or NL	;
End of record		%	%

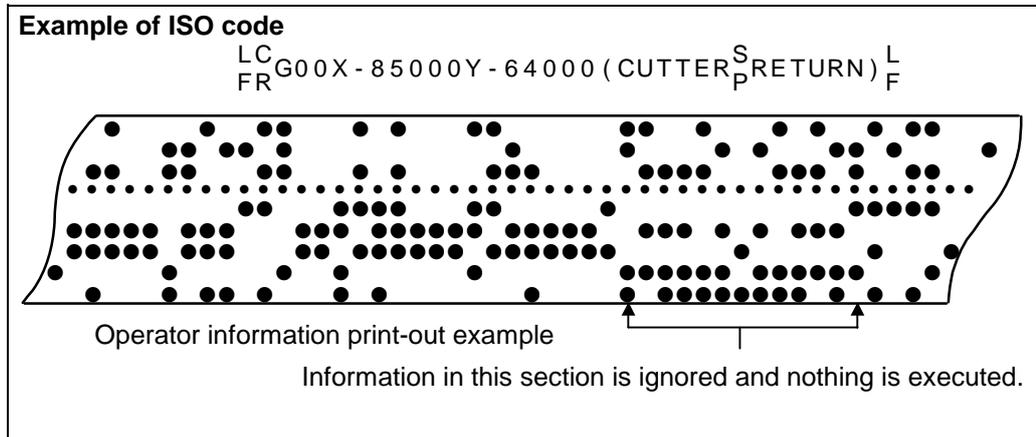
##### (1) Significant data section (label skip function)

All data up to the first EOB ( ; ), after the power has been turned on or after operation has been reset, are ignored during automatic operation based on tape, memory loading operation or during a search operation. In other words, the significant data section of a tape extends from the character or number code after the initial EOB ( ; ) code after resetting to the point where the reset command is issued.

**(2) Control out, control in**

All data between control out "(" and control in ")" or ";" , from "0" to ";" (when label L) are ignored, although these data appear on the setting and display unit. Consequently, the command tape name, No. and other such data not directly related to control can be inserted in this section.

This information (except (B) in the tape codes) will also be loaded, however, during tape loading. The system is set to the "control in" mode when the power is switched on.



The difference of the label and variable code is the following tables.

Code	Control in/out range	Label (L)	Label (O)
ISO/EIA	Form "(" to ")"	Ignore the data.	Ignore the data.
	From "O" to ";" (EOB)"	Ignore the data.	The value after "O" will be handled as label No.

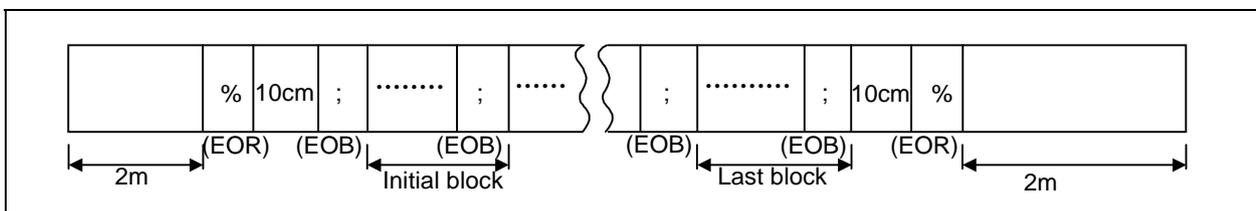
**(Note)** Always set "O" at the head of the block. A program error (P32) will occur if there is "O" besides the head of the block.

**(3) EOR (%) code**

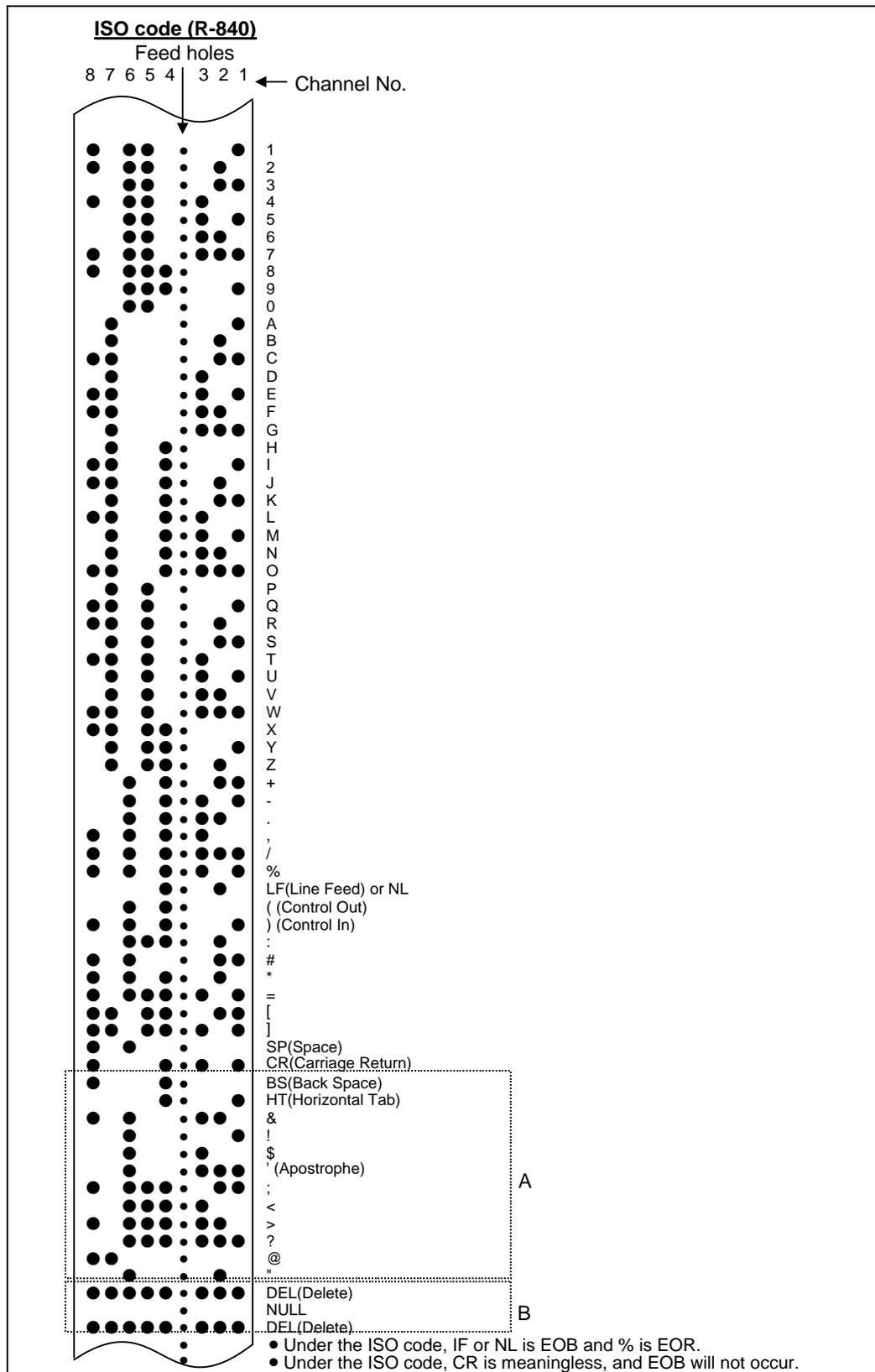
Generally, the end-of-record code is punched at both ends of the tape. It has the following functions:

- (a) Rewind stop when rewinding tape (with tape handler)
- (b) Rewind start during tape search (with tape handler)
- (c) Completion of loading during tape loading into memory

**(4) Tape preparation for tape operation (with tape handler)**



If a tape handler is not used, there is no need for the 2-meter dummy at both ends of the tape and for the head EOR (%) code.

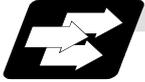


Code A are stored on tape but an error results (except when they are used in the comment section) during operation.

The B codes are non-working codes and are always ignored. Parity V check is not executed.

**Table of tape codes**

### 3.2 Program Formats



#### Function and purpose

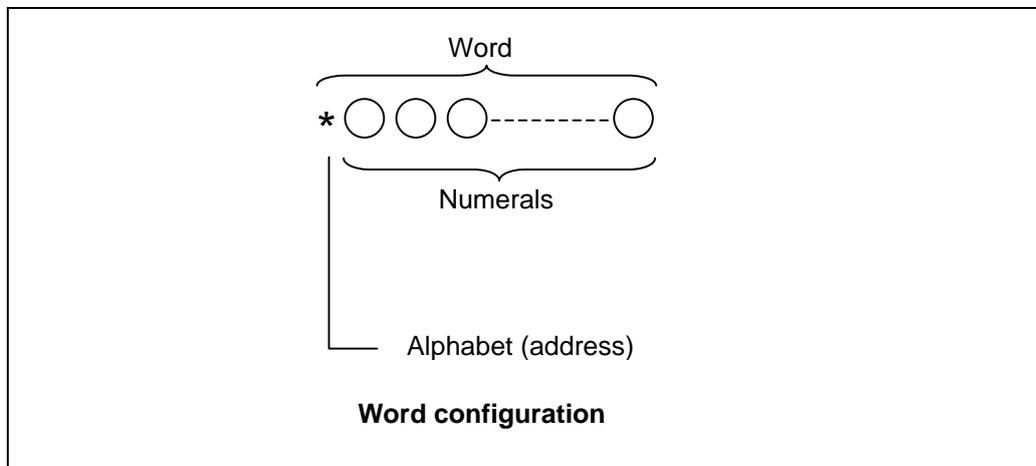
The prescribed arrangement used when assigning control information to the controller is known as the program format, and the format used with this controller is called the "word address format".



#### Detailed description

##### (1) Word and address

A word is a collection of characters arranged in a specific sequence. This entity is used as the unit for processing data and for causing the machine to execute specific operations. Each word used for this controller consists of an alphabet letter and a number of several digits (sometimes with a "-" sign placed at the head of the number.).



The alphabet letter at the head of the word is the address. It defines the meaning of the numerical information which follows it.

For details of the types of words and the number of significant digits of words used for this controller, refer to the "format details".

##### (2) Blocks

A block is a collection of words. It includes the information which is required for the machine to execute specific operations. One block unit constitutes a complete command. The end of each block is marked with an EOB (end-of-block) code.

###### (Example 1)

```
G0X - 1000 ;
G1X - 2000F500 ;    } 2 blocks
```

###### (Example 2)

```
(G0X - 1000 ;)
G1X - 2000F500 ;    } Since the semicolon in the parentheses will not result
                    } in an EOB, it is 1 block.
```

##### (3) Programs

A program is a collection of several blocks.

<Brief summary of format details>

	Metric command	Inch command	Rotary axis (Metric command)	Rotary axis (Inch command)	
Program No.	L(O)8	←	←	←	
Sequence No.	N6	←	←	←	
Preparatory function	G3/G21	←	←	←	
Movement axis	0.001(°) mm/ 0.001 inch	X+53 Y+53 Z+53 α+53	X+44 Y+44 Z+44 α+44	X+53 Y+53 Z+53 α+53	
	0.0001(°) mm/ 0.0001 inch	X+54 Y+54 Z+54 α+54	X+45 Y+45 Z+45 α+45	X+54 Y+54 Z+54 α+54	
	0.00001(°) mm/ 0.00001 inch	X+55 Y+55 Z+55 α+55	X+46 Y+46 Z+46 α+46	X+55 Y+55 Z+55 α+55	
	0.000001(°) mm/ 0.000001 inch	X+56 Y+56 Z+56 α+56	X+47 Y+47 Z+47 α+47	X+56 Y+56 Z+56 α+56	
Arc and cutter radius	0.001(°) mm/ 0.001 inch	I+53 J+53 K+53	I+44 J+44 K+44	I+53 J+53 K+53 (Note 5)	
	0.0001(°) mm/ 0.0001 inch	I+54 J+54 K+54	I+45 J+45 K+45	I+54 J+54 K+54 (Note 5)	
	0.00001(°) mm/ 0.00001 inch	I+55 J+55 K+55	I+46 J+46 K+46	I+55 J+55 K+55 (Note 5)	
	0.000001(°) mm/ 0.000001 inch	I+56 J+56 K+56	I+47 J+47 K+47	I+56 J+56 K+56 (Note 5)	
Dwell	0.001 (rev)/(s)	X53/P8	←	←	
Feed function (Feed per minute)	0.001(°) mm/ 0.001 inch	F63	F54	F63	F54 (Note 6)
	0.0001(°) mm/ 0.0001 inch	F64	F55	F64	F55 (Note 6)
	0.00001(°) mm/ 0.00001 inch	F65	F56	F65	F56 (Note 6)
	0.000001(°) mm/ 0.000001 inch	F66	F57	F66	F57 (Note 6)
Feed function (Feed per revolution)	0.0001(°) mm/ 0.0001 inch	F33	F34	F33	F34 (Note 6)
	0.00001(°) mm/ 0.00001 inch	F34	F35	F34	F35 (Note 6)
	0.000001(°) mm/ 0.000001 inch	F35	F36	F35	F36 (Note 6)
	0.0000001(°) mm/ 0.0000001 inch	F36	F37	F36	F37 (Note 6)
Tool compensation	H3 D3	←	←	←	
Miscellaneous function (M)	M8	←	←	←	
Spindle function (S)	S8	←	←	←	
Tool function (T)	T8	←	←	←	
2nd miscellaneous function	A8/B8/C8	←	←	←	
Subprogram	P8 H5 L4	←	←	←	
Fixed cycle	0.001(°) mm/ 0.001 inch	R+53 Q53 P8 L4	R+44 Q44 P8 L4	R+53 Q53 P8 L4	R+53 Q53 P8 L4
	0.0001(°) mm/ 0.0001 inch	R+54 Q54 P8 L4	R+45 Q45 P8 L4	R+54 Q54 P8 L4	R+54 Q54 P8 L4
	0.00001(°) mm/ 0.00001 inch	R+55 Q55 P8 L4	R+46 Q46 P8 L4	R+55 Q55 P8 L4	R+55 Q55 P8 L4
	0.000001(°) mm/ 0.000001 inch	R+56 Q56 P8 L4	R+47 Q47 P8 L4	R+56 Q56 P8 L4	R+56 Q56 P8 L4

(Note 1) α indicates the additional axis address, such as A, B or C.

(Note 2) The number of digits check for a word is carried out with the maximum number of digits of that address.

(Note 3) Numerals can be used without the leading zeros.

**(Note 4)** The description of the brief summary is explained below:

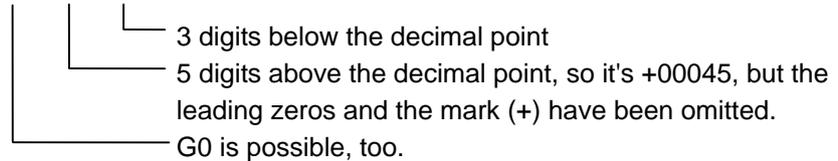
**Example 1 :** L(O)8 :8-digit program No.

**Example 2 :** G21 :Dimension G is 2 digits to the left of the decimal point, and 1 digit to the right.

**Example 3 :** X+53 :Dimension X uses + or - sign and represents 5 digits to the left of the decimal point and 3 digits to the right.

For example, the case for when the X axis is positioned (G00) to the 45.123 mm position in the absolute value (G90) mode is as follows:

G00 X45.123 ;



**(Note 5)** If an arc is commanded using a rotary axis and linear axis while inch commands are being used, the degrees will be converted into 0.1 inches for interpolation.

**(Note 6)** While inch commands are being used, the rotary axis speed will be in increments of 10 degrees.

**Example:** With the F1. (per-minute-feed) command, this will become the 10 degrees/minute command.

**(Note 7)** The decimal places below the decimal point are ignored when a command, such as an S command, with an invalid decimal point has been assigned with a decimal point.

**(Note 8)** This format is the same for the value input from the memory, MDI or setting and display unit.

**(Note 9)** Command the program No. in an independent block. Command the program No. in the head block of the program.

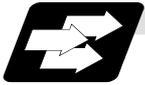
**(Note 10)** The addresses of the program No. and subprogram call No. differ according to the parameter.

The system must be formatted when this parameter is changed.

Setting of "#11009 M2 labelO"	Address of the program No.	Address to call the subprogram
0	L	L
1	O	A

This manual describes on the assumption that the parameter is set to "0".

### 3.3 Tape Memory Format



#### Function and purpose

##### (1) Storage tape and significant sections

The others are about from the current tape position to the EOB. Accordingly, under normal conditions, operate the tape memory after resetting.

The significant codes listed in "Table of tape codes" in "3.1 Tape Codes" in the above significant section are actually stored into the memory. All other codes are ignored and are not stored.

The data between control out "(" and control in ")" are stored into the memory.

### 3.4 Optional Block Skip

#### 3.4.1 Optional Block Skip ; /



#### Function and purpose

This function selectively ignores specific blocks in a machining program which starts with the "/" (slash) code.



#### Detailed description

(1) Provided that the optional block skip switch is ON, blocks starting with the "/" code are ignored. They are executed if the switch is OFF.

Parity check is valid regardless of whether the optional block skip switch is ON or OFF.

When, for instance, all blocks are to be executed for one workpiece but specific block are not to be executed for another workpiece, the same command tape can be used to machine different parts by inserting the "/" code at the head of those specific blocks.



#### Precautions for using optional block skip

(1) Put the "/" code for optional block skip at the beginning of a block. If it is placed inside the block, it is assumed as a user macro, a division instruction.

**Example :** N20 G1 X25./Y25. ;...NG (User macro, a division instruction; a program error results.)  
/N20 G1 X25. Y25. ;.....OK

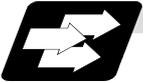
(2) Parity checks (H and V) are conducted regardless of the optional block skip switch position.

(3) The optional block skip is processed immediately before the pre-read buffer. Consequently, it is not possible to skip up to the block which has been read into the pre-read buffer.

(4) This function is valid even during a sequence number search.

(5) All blocks with the "/" code are also input and output during tape storing and tape output, regardless of the position of the optional block skip switch.

## 3.4.2 Optional Block Skip Addition ; /n



## Function and purpose

Whether the block with "/n (n:1 to 9)" (slash) is executed during automatic operation and searching is selected.

By using the machining program with "/n" code, different parts can be machined by the same program.



## Detailed description

The block with "/n" (slash) code is skipped when the "/n" is programmed to the head of the block and the optional block skip signal is turned ON.

For the block with the "/n" code inside the block (not the head of block), the program is operated according to the value of the parameter "#1226 aux10/bit1" setting.

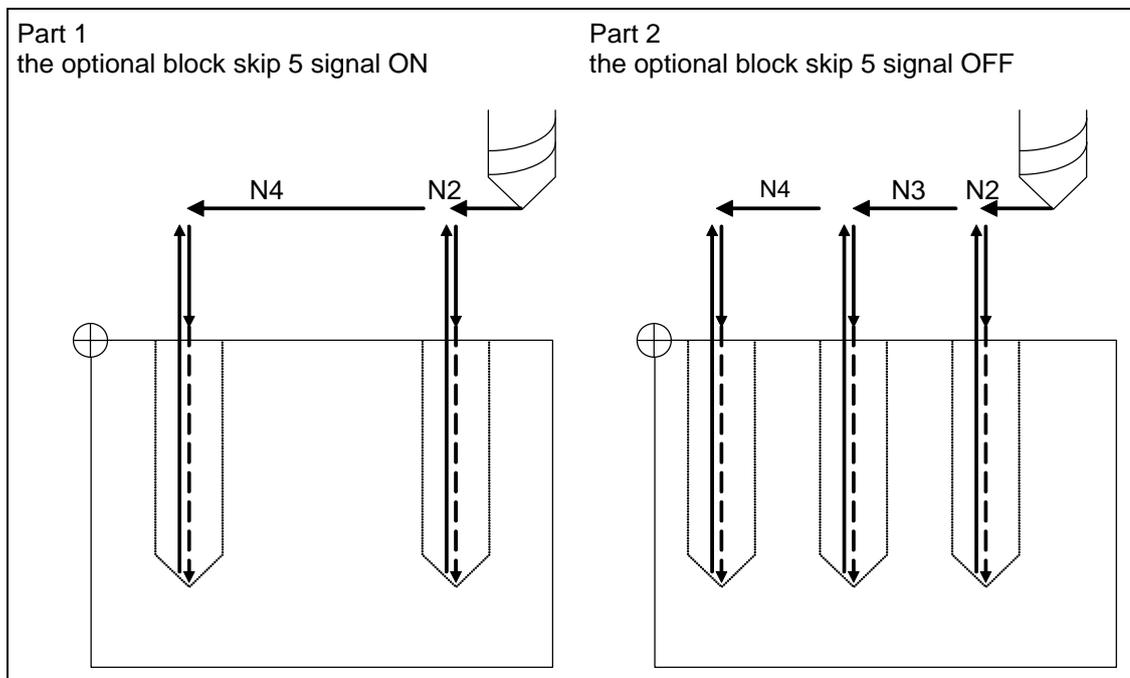
When the optional block skip signal is OFF, the block with "/n" is executed.



## Example of program

- (1) When the 2 parts like the figure below are machined, the following program is used. When the optional block skip 5 signal is ON, the part 1 is created. When the optional block skip 5 signal is OFF, the part 2 is created.

```
<Program>
N1 G54;
N2 G90G81X50. Z-20. R3. F100;
/5 N3 X30.;
N4 X10.;
N5 G80;
M02;
```



- (2) When two or more "/n" codes are commanded to the head of the same block, the block is ignored if either of the optional block skip signal corresponding to the command is ON.

<Program>

N01 G90 Z3. M03 S1000;	(a) Optional block skip 1 signal ON
/1/2 N02 G00 X50.;	(Optional block skip 2, 3 signals OFF)
/1/2 N03 G01 Z-20. F100;	N01 -> N08 -> N09 -> N10 -> N11 -> N12
/1/2 N04 G00 Z3.;	
/1 /3 N05 G00 X30.;	(b) Optional block skip 2 signal ON
/1 /3 N06 G01 Z-20. F100;	(Optional block skip 1, 3 signals OFF)
/1 /3 N07 G00 Z3.;	N01 -> N05 -> N06 -> N07 -> N11 -> N12
/2/3 N08 G00 X10.;	
/2/3 N09 G01 Z-20. F100;	(c) Optional block skip 3 signal ON
/2/3 N10 G00 Z3.;	(Optional block skip 1, 2 signals OFF)
N11 G28 X0 M05;	N01 -> N02 -> N03 -> N04 -> N11 -> N12
N12 M02;	

- (3) When the parameter "#1226 aux10/bit1" is "1", when two or more "/n" are commanded inside the same block, the commands following "/n" in the block are ignored if either of the optional block skip signal corresponding to the command is ON.

N01 G91 G28 X0.Y0.Z0.;	(a) When the optional block skip 1 signal is ON and the optional block skip 2 signal is OFF, "Y1. Z1." is ignored
N02 G01 F1000;	
N03 X1. /1 Y1. /2 Z1.;	
N04 M30;	(b) When the optional block skip 1 signal is OFF and the optional block skip 2 signal is ON, "Z1." is ignored.

## 3.5 Program/Sequence/Block Numbers ; L(O), N



## Function and purpose

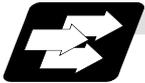
These numbers are used for monitoring the execution of the machining programs and for calling both machining programs and specific stages in machining programs.

- (1) Program numbers are classified by workpiece correspondence or by subprogram units, and they are designated by the address "L" (or "O") followed by a number with up to 8 digits.
- (2) Sequence numbers are attached where appropriate to command blocks which configure machining programs, and they are designated by the address "N" followed by a number with up to 6 digits.
- (3) Block numbers are automatically provided internally. They are preset to zero every time a program number or sequence number is read, and they are counted up one at a time unless program numbers or sequence numbers are commanded in blocks which are subsequently read.

Consequently, all the blocks of the machining programs given in the table below can be determined without further consideration by combinations of program numbers, sequence numbers and block numbers.

Machining program	Monitor display		
	Program No.	Sequence No.	Block No.
L12345678 (DEMO, PROG) ;	12345678	0	0
G92 X0 Y0 ;	12345678	0	1
G90 G51 X-150. P0.75 ;	12345678	0	2
N100 G00 X-50. Y-25. ;	12345678	100	0
N110 G01 X250. F300 ;	12345678	110	0
Y-225. ;	12345678	110	1
X-50. ;	12345678	110	2
Y-25. ;	12345678	110	3
N120 G51 Y-125. P0.5 ;	12345678	120	0
N130 G00 X-100. Y-75. ;	12345678	130	0
N140 G01 X-200. ;	12345678	140	0
Y-175. ;	12345678	140	1
X-100. ;	12345678	140	2
Y-75. ;	12345678	140	3
N150 G00 G50 X0 Y0 ;	12345678	150	0
N160 M02 ;	12345678	160	0
%			

## 3.6 Parity H/V



## Function and purpose

Parity check provides a mean of checking whether the tape has been correctly perforated or not. This involves checking for perforated code errors or, in other words, for perforation errors. There are two types of parity check: Parity H and Parity V.

**(1) Parity H**

Parity H checks the number of holes configuring a character and it is done during tape operation, tape input and sequence number search.

A parity H error is caused in the following cases.

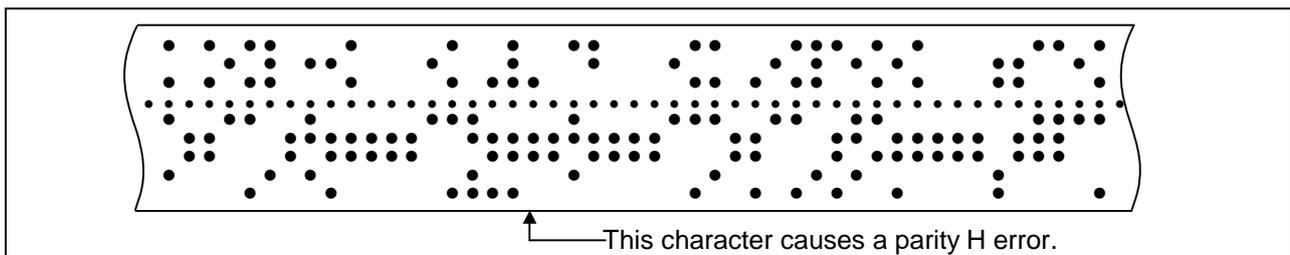
**(a) ISO code**

When a code with an odd number of holes in a significant data section has been detected.

**(b) EIA code**

When a code with an even number of holes in a significant data section has been detected.

## Parity H error example



When a parity H error occurs, the tape stops following the alarm code.

**(2) Parity V**

A parity V check is done during tape operation, tape input and sequence number search when the I/O PARA #9n15 (n is the unit No.1 to 5) parity V check function is set to "1". It is not done during memory operation.

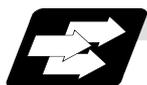
A parity V error occurs in the following case: when the number of codes from the first significant code to the EOB (;) in the significant data section in the vertical direction of the tape is an odd number, that is, when the number of characters in one block is odd.

When a parity V error is detected, the tape stops at the code following the EOB (;).

**(Note 1)** Among the tape codes, there are codes which are counted as characters for parity and codes which are not counted as such. For details, refer to the "Table of tape codes" in "3.1 Tape Codes".

**(Note 2)** Any space codes which may appear within the section from the initial EOB code to the address code or "/" code are counted for parity V check.

## 3.7 G Code Lists



## Function and purpose

G code	Group	Function	Section
Δ 00	01	Positioning	6.1
Δ 01	01	Linear interpolation	6.2
02	01	Circular interpolation CW (clockwise)	6.4
		R-specified circular interpolation CW	6.5
		Helical interpolation CW	6.6
		Spiral/Conical interpolation CW (type2)	6.13
03	01	Circular interpolation CCW (counterclockwise)	6.4
		R-specified circular interpolation CCW	6.5
		Helical interpolation CCW	6.6
		Spiral/Conical interpolation CCW (type2)	6.13
02.1	01	Spiral/Conical interpolation CW (type1)	6.13
03.1	01	Spiral/Conical interpolation CCW (type1)	6.13
02.3	01	Exponential function interpolation positive rotation	6.11
03.3	01	Exponential function interpolation negative rotation	6.11
02.4	01	3-dimensional circular interpolation	6.14
03.4	01	3-dimensional circular interpolation	6.14
04	00	Dwell	8.1
05	00	High-speed machining mode	13.16
		High-speed high-accuracy control II	13.17
05.1	00	High-speed high-accuracy control I	13.17
		Spline	13.18
06.2	01	NURBS interpolation	6.15
07	00	Hypothetical axis interpolation	6.16
07.1 107	21	Cylindrical interpolation	6.9
08	00	High-accuracy control	13.15
09	00	Exact stop check	7.9
10	00	Program data input (parameter /compensation data/parameter coordinate rotation data)	12.7 13.11 13.22
		Program data input cancel	12.7 13.11
		Program compensation data input to variable	12.8
12	00	Circular cut CW (clockwise)	13.10
13	00	Circular cut CCW (counterclockwise)	13.10
12.1	21	Polar coordinate interpolation ON	6.10
* 13.1	21	Polar coordinate interpolation cancel	6.10
14	00	Coordinate data read	13.27
* 15	18	Polar coordinate command OFF	6.12
16	18	Polar coordinate command ON	6.12
Δ 17	02	Plane selection X-Y	6.3
Δ 18	02	Plane selection Z-X	6.3
Δ 19	02	Plane selection Y-Z	6.3
Δ 20	06	Inch command	5.2
Δ 21	06	Metric command	5.2

G code	Group	Function	Section
22	00	Subprogram call / figure rotation ON	13.3
23	00	Subprogram return / figure rotation cancel	13.3
22.1	04	Stroke check before travel ON	15.7
23.1	04	Stroke check before travel cancel	15.7
24			
25			
26			
27	00	Reference position check	14.9
28	00	Reference position return	14.7
29	00	Start position return	14.7
30	00	2nd to 4th reference position return	14.8
30.1	00	Tool change position return 1	13.13
30.2	00	Tool change position return 2	13.13
30.3	00	Tool change position return 3	13.13
30.4	00	Tool change position return 4	13.13
30.5	00	Tool change position return 5	13.13
30.6	00	Tool change position return 6	13.13
31	00	Skip	15.2
		Multi-step skip function 2	15.4
31.1	00	Multi-step skip function 1-1	15.3
31.2	00	Multi-step skip function 1-2	15.3
31.3	00	Multi-step skip function 1-3	15.3
32			
33	01	Thread cutting	6.7
34	00	Special fixed cycle (bolt hole circle)	13.2
35	00	Special fixed cycle (line at angle)	13.2
36	00	Special fixed cycle (arc)	13.2
37	00	Special fixed cycle (grid)	13.2
37.1	00	Automatic tool length measurement	15.1
38	00	Tool radius compensation vector designation	12.4
39	00	Tool radius compensation corner arc	12.4
* 40	07	Tool radius compensation cancel	12.4
		3-dimensional tool radius compensation cancel	12.5
41	07	Tool radius compensation left	12.4
		3-dimensional tool radius compensation left	12.5
42	07	Tool radius compensation right	12.4
		3-dimensional tool radius compensation right	12.5
* 40.1	15	Normal line control cancel	13.14
41.1	15	Normal line control left ON	13.14
42.1	15	Normal line control right ON	13.14
43	08	Tool length compensation	12.2
* 44	08	Tool length compensation cancel	12.2
		Tool center point control cancel	13.24
43.1	08	Tool length compensation along the tool axis	12.3
43.4	08	Tool center point control type 1	13.24
43.5	08	Tool center point control type 2	13.24
45	00	Tool position offset (extension)	12.6
46	00	Tool position offset (reduction)	12.6
47	00	Tool position offset (doubled)	12.6
48	00	Tool position offset (halved)	12.6
49			

G code	Group	Function	Section
* 50	11	Scaling cancel	13.20
51	11	Scaling ON	13.20
* 50.1	19	G command mirror image cancel	13.6
51.1	19	G command mirror image ON	13.6
52	00	Local coordinate system setting	14.11
53	00	Basic machine coordinate system selection	14.4
* 54	12	Workpiece coordinate system 1 selection	14.10
55	12	Workpiece coordinate system 2 selection	14.10
56	12	Workpiece coordinate system 3 selection	14.10
57	12	Workpiece coordinate system 4 selection	14.10
58	12	Workpiece coordinate system 5 selection	14.10
59	12	Workpiece coordinate system 6 selection	14.10
54.1	12	Workpiece coordinate system selection 48 / 96 sets extended	14.10
60	00	Unidirectional positioning	6.8
61	13	Exact stop check mode	7.10
61.1	13	High-accuracy control 1 ON	13.15
61.2	13	High-accuracy spline interpolation	13.19
62	19	G command mirror image	13.6
63	13	Tapping mode	7.13
63.1	13	Synchronous tapping mode (normal tapping)	
63.2	13	Synchronous tapping mode (reverse tapping)	
* 64	13	Cutting mode	7.14
65	00	User macro call	13.5.1
66	14	User macro modal call A	13.5.1
66.1	14	User macro modal call B	13.5.1
* 67	14	User macro modal call cancel	13.5.1
68	00	Subprogram return	13.5.1
69	00	End point error check cancellation	13.26
68.1	16	Programmable coordinate rotation mode ON/3-dimensional coordinate conversion mode ON	13.21 13.23
* 69.1	16	Programmable coordinate rotation mode OFF/3-dimensional coordinate conversion mode OFF	13.21 13.23
70	09	User fixed cycle	
71	09	User fixed cycle	
72	09	User fixed cycle	
73	09	Fixed cycle (step)	13.1.1
74	09	Fixed cycle (reverse tap)	13.1.1
75	09	Fixed cycle (circle cutting cycle)	13.1.1
76	09	Fixed cycle (fine boring)	13.1.1
77	09	User fixed cycle	
78	09	User fixed cycle	
79	09	User fixed cycle	
* 80	09	Fixed cycle cancel	13.1.1
81	09	Fixed cycle (drill/spot drill)	13.1.1
82	09	Fixed cycle (drill/counter boring)	13.1.1
83	09	Fixed cycle (deep drilling)	13.1.1
84	09	Fixed cycle (tapping)	13.1.1
85	09	Fixed cycle (boring)	13.1.1
86	09	Fixed cycle (boring)	13.1.1
87	09	Fixed cycle (back boring)	13.1.1
88	09	Fixed cycle (boring)	13.1.1
89	09	Fixed cycle (boring)	13.1.1

G code	Group	Function	Section
Δ 90	03	Absolute value command	5.1
Δ 91	03	Incremental command value	5.1
92	00	Coordinate system setting / Spindle clamp speed setting	14.5
92.1	00	Workpiece coordinate system pre-setting	14.12
93	05	Inverse time feed	7.5
Δ 94	05	Per-minute feed (asynchronous feed)	7.4
Δ 95	05	Per-revolution feed (synchronous feed)	7.4
Δ 96	17	Constant surface speed control ON	10.3
Δ 97	17	Constant surface speed control OFF	10.3
* 98	10	Fixed cycle Initial level return	13.1.2
99	10	Fixed cycle R point level return	13.1.2
100 to 255	00	User macro (G code call) Max. 10	13.5.2

- (Note 1)** Codes marked with \* are codes that must be or are selected in the initial state.  
The codes marked with Δ are codes that should be or are selected in the initial state by the parameters.
- (Note 2)** If two or more G codes from the same code are commanded, the latter G code will be valid.
- (Note 3)** This G code list is a list of conventional G codes. Depending on the machine, movements that differ from the conventional G commands may be included when called by the G code macro. Refer to the Instruction Manual issued by the tool builder.
- (Note 4)** Whether the modal is initialized or not depends on each reset input.
- (1) "Reset 1"  
The modal is initialized when the reset initial parameter "#1151 rstinit" turns ON.
  - (2) "Reset 2" and "Reset & rewind"  
The modal is initialized when the signal is input.
  - (3) Resetting when emergency stop is canceled  
Follows "Reset 1".
  - (4) When modal is automatically reset at the start of individual functions such as reference position return.  
Follows "Reset & rewind".

### CAUTION

 The commands with "no value after G" will be handled as "G00".

## 3.8 Precautions Before Starting Machining



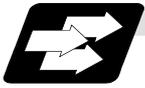
### Precautions before starting machining

### CAUTION

-  When creating the machining program, select the appropriate machining conditions so that the machine, NC performance, capacity and limits are not exceeded. The examples do not allow for the machining conditions.
-  Before starting actual machining, always carry out dry operation to confirm the machining program, tool compensation amount and workpiece offset amount, etc.

## 4. Buffer Register

### 4.1 Input Buffer

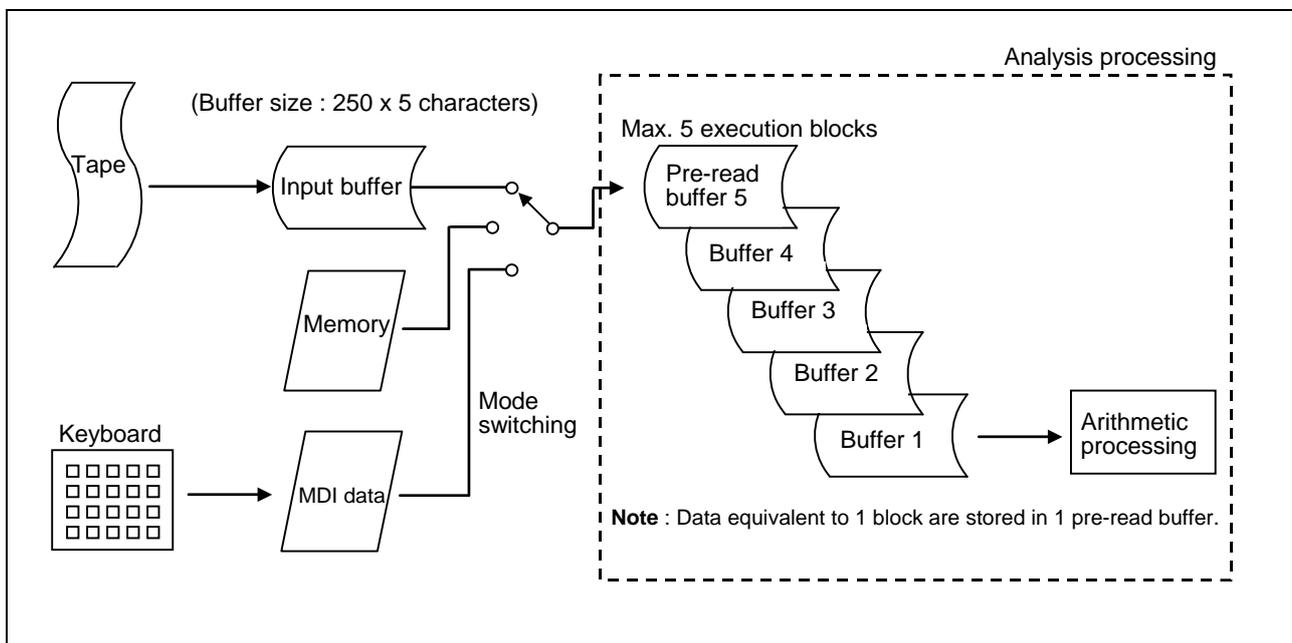


#### Function and purpose

When the pre-read buffer is empty during a tape operation or RS232C operation, the contents of the input buffer are immediately transferred to the pre-read buffers and, provided that the data stored in the input buffer do not exceed 250 x 4 characters, the following data (Max. 250 characters) are read and loaded into the input buffer.

This buffer is designed to eliminate the operational delay originating in the readout time of the tape reader and to smooth out the block joints.

The pre-reading effects are lost, however, when the block execution time is shorter than the tape readout time of the following block.

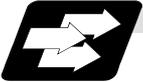


The input buffer has a memory capacity of 250 x 5 characters (including the EOB code).

- (1) The contents of the input buffer register are updated in 250-character units.
- (2) Only the significant codes in the significant data section are read into the input buffer.
- (3) When codes (including "(" and ")") are sandwiched in the control in or control out mode and the optional block skip function is ON, the data extending from the "/" (slash) code up to the EOB code are read into the input buffer.
- (4) The input buffer contents are cleared with resetting.

**(Note 1)** The input buffer size (250 characters) differs according to the model.

## 4.2 Pre-read Buffers



### Function and purpose

During automatic processing, the contents of 1 block are normally pre-read so that program analysis processing is conducted smoothly. However, during tool radius compensation, a maximum of 5 blocks are pre-read for the intersection point calculation including interference check.

The specifications of the data in 1 block are as follows:

- (1) The data of 1 block are stored in this buffer.
- (2) Only the significant codes in the significant data section are read into the pre-read buffer.
- (3) When codes are sandwiched in the control in and control out, and the optional block skip function is ON, the data extending from the "/" (slash) code up to the EOB code are not read into the pre-read buffer.
- (4) The pre-read buffer contents are cleared with resetting.
- (5) When the single block function is ON during continuous operation, the pre-read buffer stores the following block data and then stops operation.



### Other precautions

- (1) Depending on whether the program is executed continuously or by single blocks, the timing of the valid/invalid for the external control signals for the block skip and others will differ.
- (2) If the external control signal such as optional block skip is turned ON/OFF with the M command, the external control operation will not be effective on the program pre-read with the buffer register.
- (3) According to the M command that operates the external controls, it prohibits pre-reading, and the recalculation is as follows:  
The M command that commands the external controls is distinguished at the PLC, and the "recalculation request" for PLC -> NC interface table is turned ON.  
(When the "recalculation request" is ON, the program that has been pre-read is reprocessed.)

## 5. Position Commands

### 5.1 Position Command Methods ; G90, G91



#### Function and purpose

By using the G90 and G91 commands, it is possible to execute the next coordinate commands using absolute values or incremental values.

The R-designated circle radius and the center of the circle determined by I, J, K are always incremental value commands.



#### Command format

**G90** X\_\_ Y\_\_ Z\_\_ α\_\_;

G90 :Absolute command

G91 :Incremental command

α :Additional axis



#### Detailed description

- (1) Regardless of the current position, in the absolute value mode, it is possible to move to the position of the workpiece coordinate system that was designated in the program.

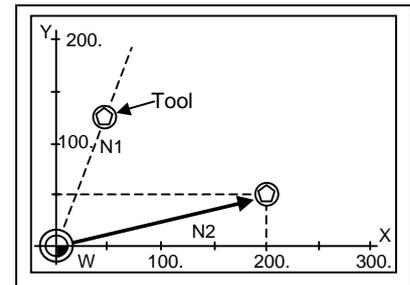
```
N 1 G90 G00 X0 Y0 ;
```

In the incremental value mode, the current position is the start point (0), and the movement is made only the value determined by the program, and is expressed as an incremental value.

```
N 2 G90 G01 X200. Y50. F100;
```

```
N 2 G91 G01 X200. Y50. F100;
```

Using the command from the 0 point in the workpiece coordinate system, it becomes the same coordinate command value in either the absolute value mode or the incremental value mode.



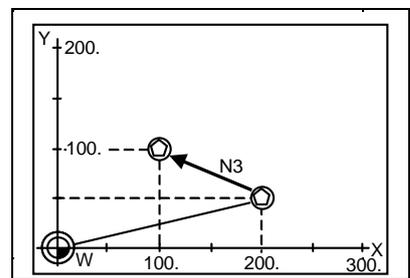
- (2) For the next block, the last G90/G91 command that was given becomes the modal.

```
(G90) N 3 X100. Y100.;
```

The axis moves to the workpiece coordinate system X = 100mm and Y = 100mm position.

```
(G91) N 3 X-100. Y50.;
```

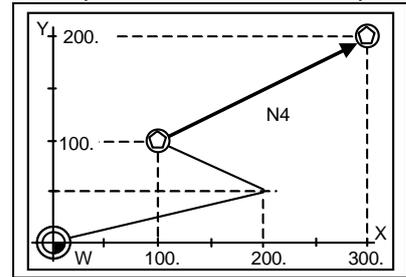
The X axis moves to -100.mm and the Y axis to +50.0mm as an incremental value, and as a result X moves to 100.mm and Y to 100.mm.



- (3) Since multiple commands can be issued in the same block, it is possible to command specific addresses as either absolute values or incremental values.

```
N 4 G90 X300. G91 Y100.;
```

The X axis is treated in the absolute value mode, and with G90 is moved to the workpiece coordinate system 300.mm position. The Y axis is moved +100.mm with G91. As a result, Y moves to the 200.mm position. In terms of the next block, G91 remains as the modal and becomes the incremental value mode.



- (4) When the power is turned ON, it is possible to select whether you want absolute value commands or incremental value commands with the #1073 I\_Absm parameter.
- (5) Even when commanding with the manual data input (MDI), it will be treated as a modal from that block.

## 5.2 Inch/Metric Command Change; G20, G21

**Function and purpose**

These G commands are used to change between the inch and millimeter (metric) systems.

**Command format**

<b>G20/G21;</b>	
G20	: Inch command
G21	: Metric command

**Detailed description**

The G20 and G21 commands merely select the command units. They do not select the Input units. G20 and G21 selection is meaningful only for linear axes and it is meaningless for rotation axes.

**Output unit, command unit and setting unit**

The counter or parameter setting and display unit is determined by parameter "#1041 I\_inch". For the movement/speed command, the followings will be resulted:  
The movement/speed command will be displayed as metric units when "#1041 I\_inch" is ON during the G21 command mode.  
The internal unit metric data of the movement/speed command will be converted into an inch unit and displayed when "#1041 I\_inch" is OFF during the G20 command mode.  
The command unit for when the power is turned ON and reset is decided by combining the parameters "#1041 I\_inch", "#1151 rstint" and "#1210 RstGmd/bit5".

**NC axis**

Item	Initial inch OFF (metric internal unit) #1041 I_inch=0		Initial inch ON (inch internal unit) #1041 I_inch=1	
	G21	G20	G21	G20
Movement/ speed command	Metric	Inch	Metric	Inch
Counter display	Metric	Metric	Inch	Inch
Speed display	Metric	Metric	Inch	Inch
User parameter setting/display	Metric	Metric	Inch	Inch
Workpiece/ tool offset setting/display	Metric	Metric	Inch	Inch
Handle feed command	Metric	Metric	Inch	Inch

**PLC axis**

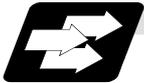
Item	#1042 pcinch=0 (metric)	#1042 pcinch=1 (inch)
Movement/ speed command	Metric	Inch
Counter display	Metric	Inch
User parameter setting/display	Metric	Inch



#### Precautions

- (1) The parameter and tool data will be input/output with the "#1041 I\_inch" setting unit. If "#1041 I\_inch" is not found in the parameter input data, the unit will follow the unit currently set to NC.
- (2) The unit of read/write used in PLC window is fixed to metric unit regardless of a parameter and G20/G21 command modal.
- (3) A program error (P33) will occur if G20/G21 command is issued in the same block as following G code. Command in a separate block.
  - G05 (High-speed machining mode)
  - G7.1 (Cylindrical Interpolation)
  - G12.1 (Polar coordinate interpolation)

## 5.3 Decimal Point Input



## Function and purpose

This function enables the decimal point command to be input. It assigns the decimal point in millimeter or inch units for the machining program input information that defines the tool paths, distances and speeds. The parameter "#1078 Decpt2" selects whether type I (minimum input command unit) or type II (zero point) is to apply for the least significant digit of data without a decimal point.



## Detailed description

- The decimal point command is valid for the distances, angles, times, speeds and scaling rate, in machining programs. (Note, only after G51)
- In decimal point input type 1 and type 2, the values of the data commands without the decimal points are shown in the table below.

Command	Command unit	Type 1	Type 2
X1 ;	cunit = 10000	1000 ( $\mu\text{m}$ , $10^{-4}$ inch, $10^{-3}$ °)	1 (mm, inch, °)
	cunit = 1000	100	1
	cunit = 100	10	1
	cunit = 10	1	1

- The valid addresses for the decimal points are X, Y, Z, U, V, W, A, B, C, I, J, K, E, F, P, Q, and R. However, P is valid only during scaling. For details, refer to the list.
- In decimal point command, the valid range of command value is as shown below. (Input command unit cunit = 10)

	Movement command (linear)	Movement command (rotary)	Feedrate	Dwell
Input unit [mm]	-99999.999 to 99999.999	-99999.999 to 99999.999	0.001 to 1000000.000	0 to 99999.999
Input unit [inch]	-9999.9999 to 9999.9999		0.0001 to 100000.0000	

- The decimal point command is valid even for commands defining the variable data used in subprograms.
- While the smallest decimal point command is validated, the smallest unit for a command without a decimal point designation is the smallest command input unit set in the specifications ( $1\mu\text{m}$ ,  $10\mu\text{m}$ , etc.) or mm can be selected. This selection can be made with parameter "#1078 Decpt2".
- Decimal point commands for decimal point invalid addresses are processed as integer data only and everything below the decimal point is ignored. Addresses which are invalid for the decimal point are D, H, L, M, N, O, S and T. All variable commands, however, are treated as data with decimal points.
- "Input command increment tenfold" is applied in the decimal point type I mode, but not in the decimal point type II mode.



## Example of program

- (1) Example of program for decimal point valid address

Specification division Program example	Decimal point command 1		Decimal point command 2 1 = 1mm
	When 1 = 1 $\mu$ m	When 1 = 10 $\mu$ m	
G0X123.45 (decimal points are all mm points)	X123.450mm	X123.450mm	X123.450mm
G0X12345	X12.345mm (last digit is 1 $\mu$ m unit)	X123.450mm	X12345.000mm
#111 = 123, #112 = 5.55 X#111 Y#112	X123.000mm, Y5.550mm	X123.000mm, Y5.550mm	X123.000mm, Y5.550mm
#113 = #111+#112 (addition)	#113 = 128.550	#113 = 128.550	#113 = 128.550
#114 = #111-#112 (subtraction)	#114 = 117.450	#114 = 117.450	#114 = 117.450
#115 = #111*#112 (multiplication)	#115 = 682.650	#115 = 682.650	#115 = 682.650
#116 = #111/#112 #117 = #112/#111 (division)	#116 = 22.162 #117 = 0.045	#116 = 22.162 #117 = 0.045	#116 = 22.162 #117 = 0.045



## Decimal point input I/II and decimal point command valid/invalid

If a command does not use a decimal point at an address where a decimal point command is valid in the table on the following page, it is handled differently between decimal point input I and II modes as explained below.

A command using a decimal point is handled the same way in either the decimal point input I or II mode.

- (1) Decimal point input I

The least significant digit place of command data corresponds to the command unit.

**(Example)** Command "X1" in the 1 $\mu$ m system is equivalent to command "X0.001".

- (2) Decimal point input II

The least significant digit place of command data corresponds to the decimal point.

**(Example)** Command "X1" in the 1 $\mu$ m system is equivalent to command "X1.".

**(Note)** When a four rules operator is contained, the data will be handled as that with a decimal point.

**(Example)** When the min. input command unit is 1 $\mu$ m :

G0 x 123 + 0 ; ... X axis 123mm command. It will not be 123 $\mu$ m.

Addresses used and validity/invalidity of decimal point commands are shown below.

Address	Decimal point command	Application	Remarks
A	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function code	
	Valid	Angle data	
	Invalid	Data settings, axis numbers (G10)	
	Invalid	Subprogram call : program No. (label O)	
B	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function code	
C	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function code	
	Valid	Corner chamfering amount	,C
D	Invalid	Compensation numbers (tool position, tool radius)	
	Valid	Automatic tool length measurement: deceleration distance d	
	Invalid	Data setting: byte type data	
	Invalid	Subprogram storing device number	,D
E	Valid	Inch thread: number of ridges, precision thread: lead	
	Valid	Dwell time at hole bottom of fixed cycle (label L, O)	
F	Valid	Feedrate, automatic tool length measurement speed	
	Valid	Thread lead	
	Valid	Number of Z axis pitch in synchronous tap	
G	Valid	Preparatory function code	
H	Invalid	Tool length compensation number	
	Invalid	Sequence number in subprogram (label L, O)	
	Invalid	Subprogram return destination sequence No. (label L, O)	
	Invalid	Program parameter input: bit type data	
	Invalid	Basic spindle selection	
I	Valid	Arc center coordinates	
	Invalid	Center of figure rotation (incremental)	
	Valid	Tool radius compensation vector components	
	Valid	Hole pitch in the special fixed cycle	
	Valid	Circle radius of cut circle (increase amount)	
	Valid	G0/G1 imposition width, drilling cycle G0 imposition width	,I
	Valid	Stroke check before travel: lower limit coordinates	

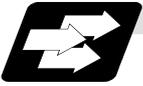
Address	Decimal point command	Application	Remarks
J	Valid	Arc center coordinates	
	Invalid	Center of figure rotation (incremental)	
	Valid	Tool radius compensation vector components	
	Valid	Special fixed cycle's hole pitch or angle	
	Valid	G0/G1 imposition width, drilling cycle G1 imposition width	
	Valid	Stroke check before travel: lower limit coordinates	
K	Valid	Arc center coordinates	
	Invalid	Center of figure rotation (incremental)	
	Valid	Tool radius compensation vector components	
	Invalid	Number of holes of the special fixed cycle	
	Invalid	Number of drilling cycle repetitions	
	Valid	Stroke check before travel: lower limit coordinates	
L	Invalid	Subprogram call program No. (label L)	
	Invalid	Program tool compensation input/workpiece offset input: type selection	L2, L20, L12, L10, L13, L11
	Invalid	Program parameter input: data setting selection	L70
	Invalid	Program parameter input: 2-word type data	4 bytes
	Invalid	Tool life data	
M	Invalid	Miscellaneous function codes	
N	Invalid	Sequence numbers	
	Invalid	Program parameter input: data numbers	
O	Invalid	Program numbers (label O)	
P	Invalid/Valid	Dwell time	Parameter
	Invalid	Subprogram call : number of repetition (label L, O)	
	Invalid/Valid	Dwell at tap cycle hole base	Parameter
	Invalid	Number of holes of the special fixed cycle (label L)	
	Invalid	Amount of helical pitch	
	Invalid	Offset number (G10)	
	Invalid	Constant surface speed control axis number	
	Invalid	Program parameter input: broad classification number	
	Invalid	Multi-step skip function 2 signal command	
	Invalid	2nd, 3rd, 4th reference position return number	
	Valid	Scaling magnification	
	Invalid	High-speed mode type	
	Invalid	Type of the coordinate to be read	
	Invalid	Transmission source compensation No. of program compensation data input to variable	
	Invalid	Extended workpiece coordinate system No.	
Invalid	Tool life data group No.		

Address	Decimal point command	Application	Remarks
Q	Valid	Cut amount of deep hole drill cycle	
	Valid	Shift amount of back boring	
	Valid	Shift amount of fine boring	
	Invalid	Minimum spindle clamp speed	
	Valid	Starting shift angle for screw cutting	
	Invalid	Transmission destination variable No. of program compensation data input to variable	
	Invalid	Tool life data management method	
R	Valid	R-point in the fixed cycle	
	Valid	R-specified arc radius	
	Valid	Corner R arc radius	,R
	Valid	Offset amount (G10)	
	Invalid	Synchronous tap/asynchronous tap changeover	
	Valid	Automatic tool length measurement: deceleration distance r	
	Valid	Rotation angle	
S	Invalid	Spindle function codes	
	Invalid	Maximum spindle clamp speed	
	Invalid	Constant surface speed control: surface speed	
	Invalid	Program parameter input: word type data	2 bytes
T	Invalid	Tool function codes	
U	Valid	Coordinate position data	
V	Valid	Coordinate position data	
W	Valid	Coordinate position data	
X	Valid	Coordinate position data	
	Valid	Dwell time	
Y	Valid	Coordinate position data	
Z	Valid	Coordinate position data	

**(Note 1)** All decimal points are valid for the user macro arguments.

## 6. Interpolation Functions

### 6.1 Positioning (Rapid Traverse); G00



#### Function and purpose

This command is accompanied by coordinate words. It positions the tool along a linear or non-linear path from the present point as the start point to the end point which is specified by the coordinate words.



#### Command format

**G00 X\_\_ Y\_\_ Z\_\_ α\_\_ ; (α represents additional axis)**

X, Y, Z, α : Represent coordinates, and could be either absolute values or incremental values, depending on the setting of G90/G91.



#### Detailed description

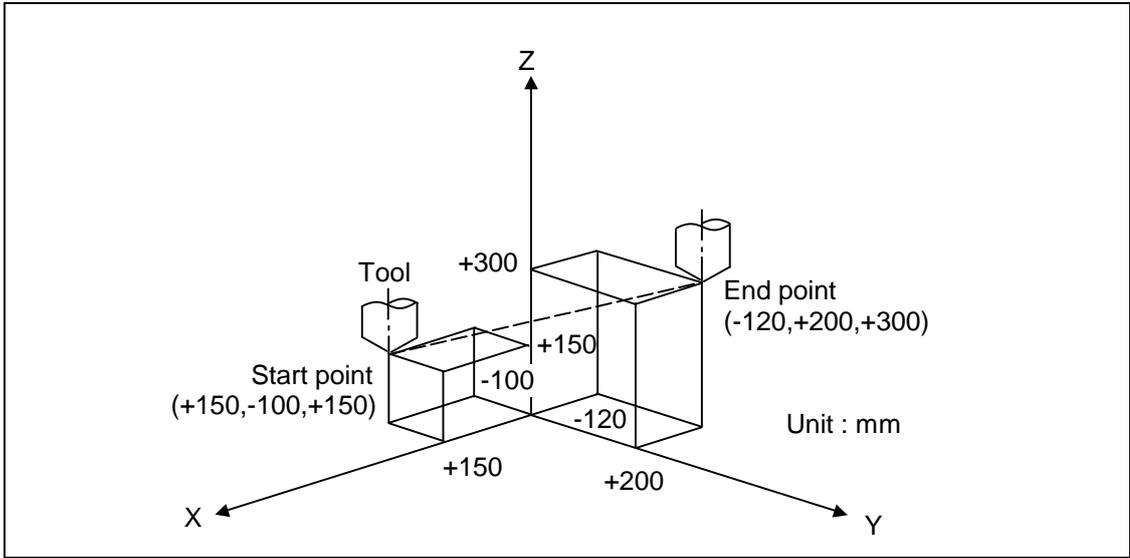
- (1) Once this command has been issued, the G00 mode is retained until it is changed by another G function or until the G01, G02, G03 or G33 command in the 01 group is issued. If the next command is G00, all that is required is simply that the coordinate words be specified.
- (2) In the G00 mode, the tool is always accelerated at the start point of the block and decelerated at the end point. Having no more command pulse in the current block and the following error status of the acceleration/deceleration paths are confirmed before advancing to the next block. The in-position width is set with the parameters.
- (3) Any G command (G72 to G89) in the 09 group is cancelled (G80) by the G00 command.
- (4) The tool path can be selected from linear or non-linear.  
The positioning time is the same for the linear and non-linear paths.
  - (a) Linear path..... : This is the same as linear interpolation (G01), and the speed is limited by the rapid traverse rate of each axis.
  - (b) Non-linear path .. : The tool is positioned at the rapid traverse rate independently for each axis.

#### CAUTION

 The commands with "no value after G" will be handled as "G00".



Example of program

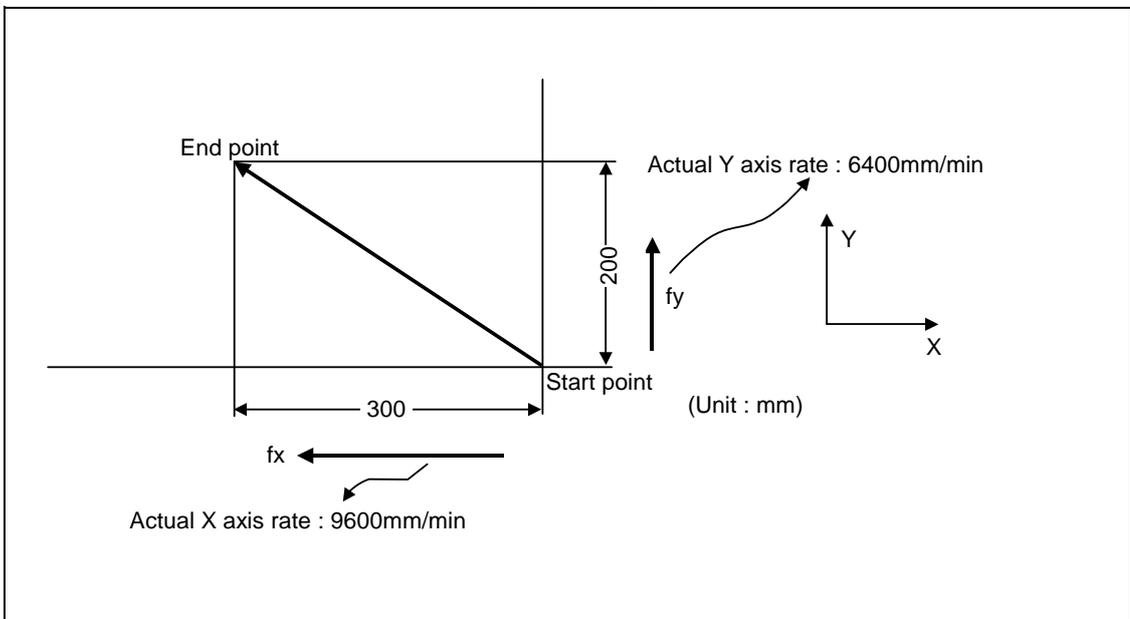


```
G91 G00 X-270000 Y300000 Z150000 ; (For input setting unit: 0.001mm)
```

**(Note 1)** When parameter "#1086 G0Intp" is set to "0", the path along which the tool is positioned is the shortest path connecting the start and end points. The positioning speed is automatically calculated so that the shortest distribution time is obtained in order that the commanded speeds for each axis do not exceed the rapid traverse rate.

When for instance, the Y axis and Z axis rapid traverse rates are both 9600mm/min, the tool will follow the path in the figure below if the following is programmed:

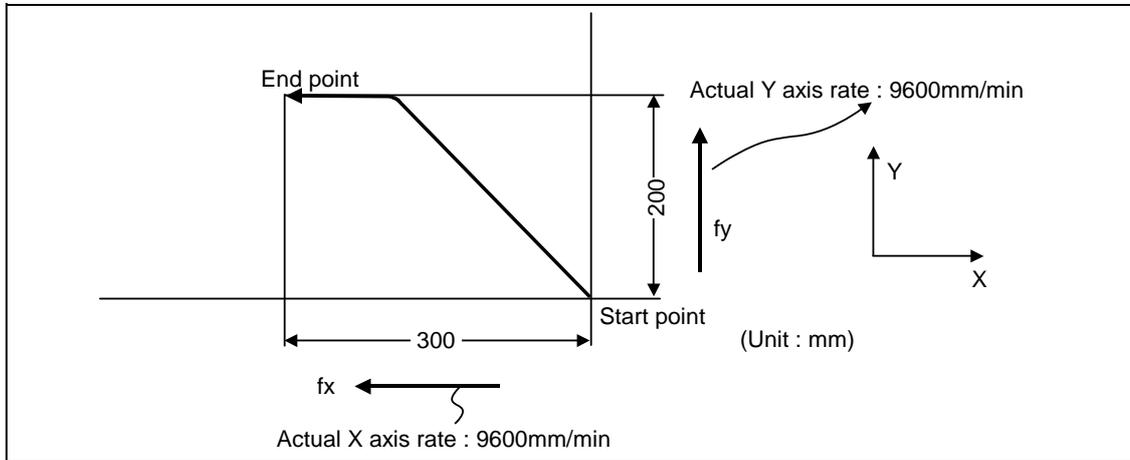
```
G91 G00 X-300000 Y200000 ; (With an input setting unit of 0.001mm)
```



**(Note 2)** When parameter "#1086 G0Intp" is set to 1, the tool will move along the path from the start point to the end point at the rapid traverse rate of each axis.

When, for instance, the Y axis and Z axis rapid traverse rates are both 9600mm/min, the tool will follow the path in the figure below if the following is programmed:

G91 G00 X-300000 Y200000 ; (With an input setting unit of 0.001mm)



**(Note 3)** The rapid traverse rate for each axis with the G00 command differs according to the individual machine and so reference should be made to the machine specifications.

**(Note 4)** Rapid traverse (G00) deceleration check

There are two methods for the deceleration check at rapid traverse; commanded deceleration method and in-position check method. Select a method with the parameter "#1193 inpos".

■ When "inpos" = "1"

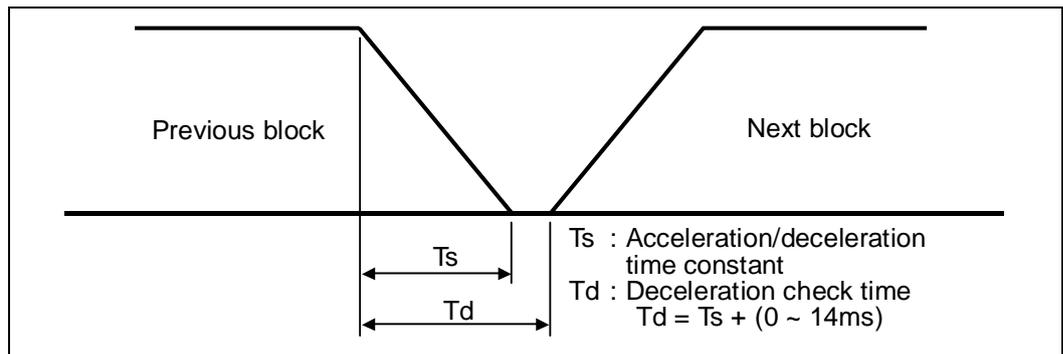
Upon completion of the rapid traverse (G00), the next block will be executed after confirming that the remaining distances for each axis are below the fixed amounts. (Refer to following drawing.) The confirmation of the remaining distance should be done with the imposition width,  $L_R$ .  $L_R$  is the setting value for the servo parameter "#2224 SV 024".

The purpose of checking the rapid feedrate is to minimize the time it takes for positioning. The bigger the setting value for the servo parameter "#2224 SV024", the longer the reduced time is, but the remaining distance of the previous block at the starting time of the next block also becomes larger, and this could become an obstacle in the actual processing work. The check for the remaining distance is done at set intervals. Accordingly, it may not be possible to get the actual amount of time reduction for positioning with the setting value SV 024.

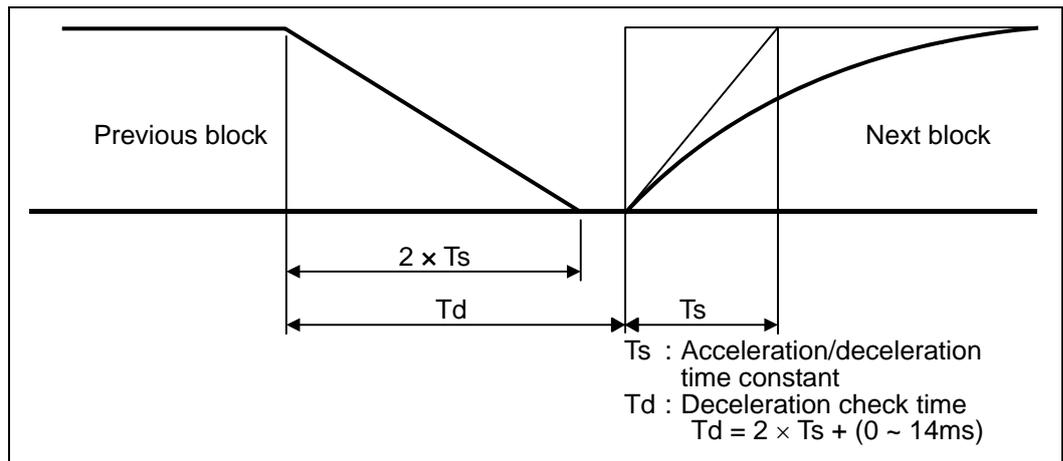
■ When "inpos" = "0"

Upon completion of the rapid traverse (G00), the next block will be executed after the deceleration check time ( $T_d$ ) has elapsed. The deceleration check time ( $T_d$ ) is as follows, depending on the acceleration/deceleration type.

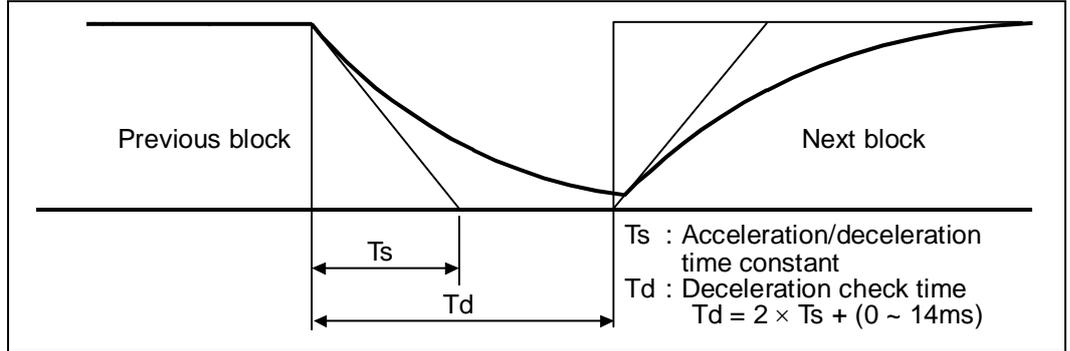
(1) Linear acceleration/linear deceleration.....  $T_d = T_s + \alpha$



(2) Exponential acceleration/linear deceleration.....  $T_d = 2 \times T_s + \alpha$



(3) Exponential acceleration/exponential deceleration .....  $T_d = 2 \times T_s + \alpha$



Where  $T_s$  is the acceleration time constant,  $\alpha = 0$  to 14ms

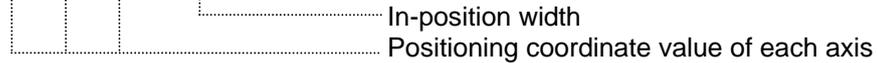
The time required for the deceleration check during rapid traverse is the longest among the rapid traverse deceleration check times of each axis determined by the rapid traverse acceleration/deceleration time constants and by the rapid traverse acceleration/deceleration mode of the axes commanded simultaneously.



**Programmable in-position width command for positioning**

This command commands the in-position width for the positioning command from the machining program.

```
G00 X__ Y__ Z__, I__;
```



**Operation during in-position check**

Execution of the next block starts after confirming that the position error amount of the positioning (rapid traverse: G00) command block and the block that carries out deceleration check with the linear interpolation (G01) command is less than the in-position width issued in this command.

The in-position width in this command is valid only in the command block, so the deceleration check method set in base specification parameter "#1193 inpos" is used for blocks that do not have the in-position width command.

When there are several movement axes, the system confirms that the position error amount of each movement axis in each part system is less than the in-position width issued in this command before executing the next block.

The differences of when the in-position check is validated with the parameter (base specification parameter "#1193 inpos" set to 1; refer to next page for in-position width) and when validated with this command are shown in the following drawing.

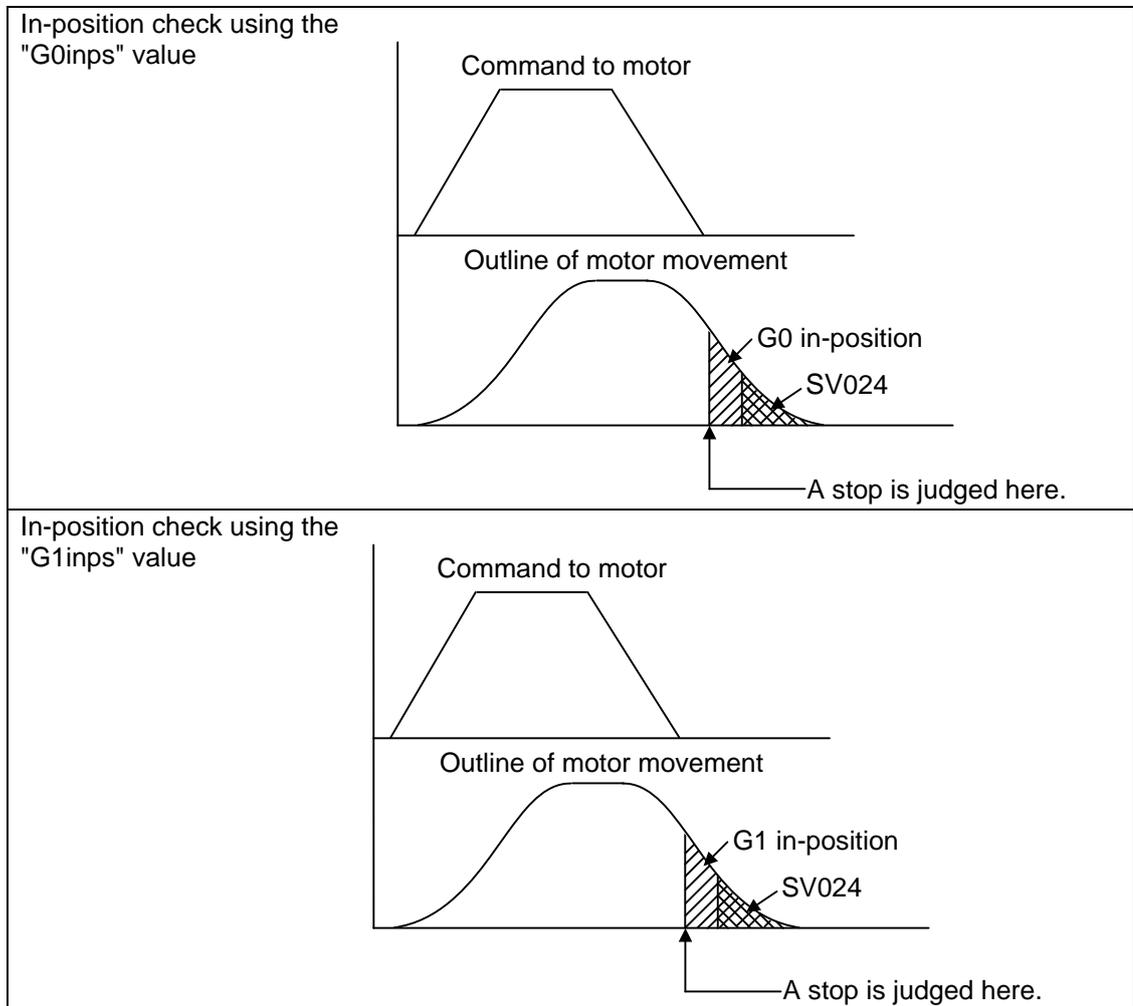
Differences between in-position check with this command and in-position check with parameter

In-position check with "I" address command	In-position check with parameter
After starting deceleration of the command system, the position error amount and commanded in-position width are compared.	After starting deceleration of the command system, the servo system's position error amount and the parameter setting value (in-position width) are compared.
<p>Ts : Acceleration/deceleration time constant                      Td : Deceleration check time                      Td = Ts + (0 to 14ms)</p>	



## In-position width setting

When the servo parameter "#2224 SV024" setting value is smaller than the setting value of the G0 in-position width "#2077 G0inps" and the G1 in-position width "#2078 G1inps", the in-position check is carried out with the G0 in-position width and the G1 in-position width.

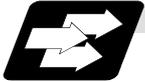


When the SV024 value is larger, the in-position check is completed when the error amount is smaller than the SV024 setting value.

The in-position check method depends on the method set in the deceleration check parameter.

- (Note 1)** When the in-position width (programmable in-position check width) is set in the machining program, either the in-position width set with the parameter (SV024, G0inps, G1inps) or that set in the program, whichever larger, is applied when performing an in-position check.
- (Note 2)** When the SV024 setting value is larger than the G0 in-position width/G1 in-position width, the in-position check is carried out with the SV024 value.
- (Note 3)** When the error detect is ON, the in-position check is forcibly carried out.

## 6.2 Linear Interpolation; G01



## Function and purpose

This command is accompanied by coordinate words and a feedrate command. It makes the tool move (interpolate) linearly from its present position to the end point specified by the coordinate words at the speed specified by address F. In this case, the feedrate specified by address F always acts as a linear speed in the tool nose center advance direction.



## Command format

**G01 X\_\_ Y\_\_ Z\_\_  $\alpha$ \_\_ F\_\_ ,I\_\_ ; ( $\alpha$  represents additional axis)**

X, Y, Z, $\alpha$	:Represents the coordinate value. An absolute position or incremental position is indicated based on the state of G90/G91 at that time.
F	:Feedrate (mm/min or °/min)
I	:In-position width. A program error occurs if a decimal point command is issued. This is valid only in the commanded block. A block that does not contain this address will follow the parameter "#1193 inpos" settings.



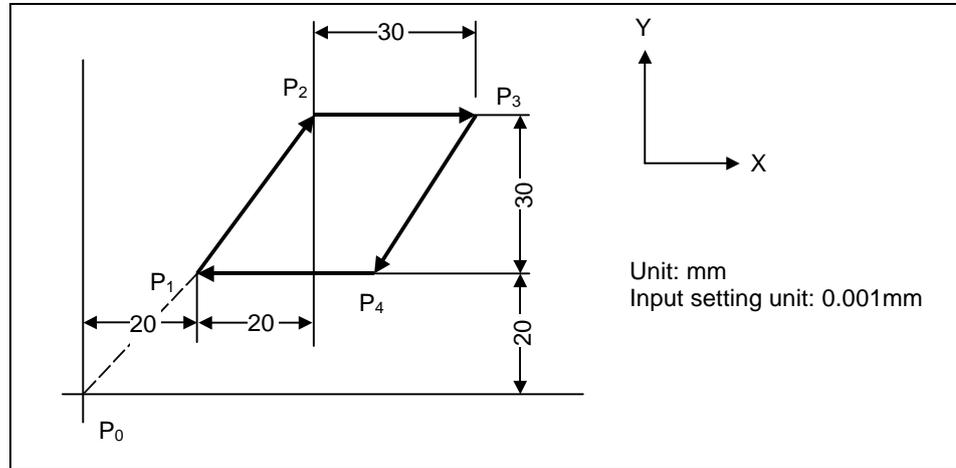
## Detailed description

- (1) Once this command is issued, the mode is maintained until another G function (G00, G02, G03, G33) in the 01 group which changes the G01 mode is issued. Therefore, if the next command is also G01 and if the feedrate is the same, all that is required to be done is to specify the coordinate words. If no F command is given in the first G01 command block, program error (P62) results.
- (2) The feedrate for a rotary axis is commanded by °/min (decimal point position unit). (F300 = 300°/min)
- (3) The G functions (G70 - G89) in the 09 group are cancelled (G80) by the G01 command.



Example of program

(Example 1) Cutting in the sequence of  $P_1 \rightarrow P_2 \rightarrow P_3 \rightarrow P_4 \rightarrow P_1$  at 300 mm/min feedrate  
 $P_0 \rightarrow P_1$  is for tool positioning



G90	G00	X20000	Y20000 ;	$P_0 \rightarrow P_1$
	G01	X20000	Y30000 F300	$P_1 \rightarrow P_2$
		X30000	;	$P_2 \rightarrow P_3$
		X-20000	Y-30000 ;	$P_3 \rightarrow P_4$
		X-30000 ;		$P_4 \rightarrow P_1$

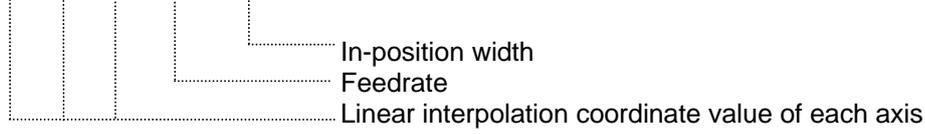


Programmable in-position width command for linear interpolation

This command commands the in-position width for the linear interpolation command from the machining program.  
 The commanded in-position width is valid in the linear interpolation command only when carrying out deceleration check.

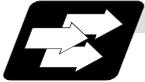
- When the error detect switch is ON.
- When G09 (exact stop check) is commanded in the same block.
- When G61 (exact stop check mode) is selected.

```
G01 X__Y__Z__F__, I__;
```



(Note 1) Refer to section "6.1 Positioning (rapid traverse); G00" for details on the in-position check operation.

## 6.3 Plane Selection; G17, G18, G19



## Function and purpose

The plane to which the movement of the tool during the circle interpolation (including helical cutting) and tool radius compensation command belongs is selected.

By registering the basic three axes and the corresponding parallel axis as parameters, a plane can be selected by two axes that are not the parallel axis. If the rotary axis is registered as a parallel axis, a plane that contains the rotary axis can be selected.

The plane selection is as follows:

- Plane that executes circular interpolation (including helical cutting)
- Plane that executes tool radius compensation
- Plane that executes fixed cycle positioning.



## Command format

<b>G17 ;</b>	<b>(ZX plane selection)</b>
<b>G18 ;</b>	<b>(YZ plane selection)</b>
<b>G19 ;</b>	<b>(XY plane selection)</b>

X, Y and Z indicate each coordinate axis or the parallel axis.



## Parameter entry

Table 1 Example of plane selection parameter entry

	<b>#1026 to 1028 base_I,J,K</b>	<b>#1029 to 1039 aux_I,J,K</b>
I	X	U
J	Y	
K	Z	V

As shown in the above example, the basic axis and its parallel axis can be registered.

The basic axis can be an axis other than X, Y and Z.

Axes that are not registered are irrelevant to the plane selection.



## Plane selection system

In Table 1,

I is the horizontal axis for the G17 plane or the vertical axis for the G18 plane

J is the vertical axis for the G17 plane or the horizontal axis for the G19 plane

K is the horizontal axis for the G18 plane or the vertical axis for the G19 plane

In other words,

G17 ..... IJ plane

G18 ..... KI plane

G19 ..... JK plane

- (1) The axis address commanded in the same block as the plane selection (G17, G18, G19) determines which basic axis or parallel axis is used for the plane selection.

For the parameter registration example in Table 1.

G17X\_\_Y\_\_ ; XY plane

G18X\_\_V\_\_ ; VX plane

G18U\_\_V\_\_ ; VU plane

G19Y\_\_Z\_\_ ; YZ plane

G19Y\_\_V\_\_ ; YV plane

- (2) The plane will not changeover at a block where a plane selection G code (G17, G18, G19) is not commanded.

G17X\_\_Y\_\_ ; XY plane

Y\_\_Z\_\_ ; XY plane (plane does not change)

- (3) If the axis address is omitted in the block where the plane selection G code (G17, G18, G19) is commanded, it will be viewed as though the basic three axes address has been omitted.

For the parameter registration example in Table 1.

G17 ; XY plane

G17U\_\_ ; UY plane

G18U\_\_ ; ZU plane

G18V\_\_ ; VX plane

G19Y\_\_ ; YZ plane

G19V\_\_ ; YV plane

- (4) The axis command that does not exist in the plane determined by the plane selection G code (G17, G18, G19) is irrelevant to the plane selection.

For the parameter registration example in Table 1.

G17U\_\_Z\_\_ ;

- (5) If the above is commanded, the UY plane will be selected, and Z will move regardless of the plane.

If the basic axis and parallel axis are commanded in duplicate in the same block as the plane selection G code (G17, G18, G19), the plane will be determined in the priority order of basic axis and parallel axis.

For the parameter registration example in Table 1.

G17U\_\_Y\_\_W\_\_-;

If the above is commanded, the UY plane will be selected, and W will move regardless of the plane.

**(Note 1)** The plane set with parameter "#1025 I\_plane" will be selected when the power is turned ON or reset.

## 6.4 Circular Interpolation; G02, G03



## Function and purpose

These commands serve to move the tool along an arc.



## Command format

**G02 (G03) X\_\_ Y\_\_ I\_\_ J\_\_ K\_\_ F\_\_;**

G02	: Clockwise (CW)
G03	: Counterclockwise (CCW)
X, Y	: End point
I, J	: Arc center
F	: Feedrate

For the arc command, the arc end point coordinates are assigned with addresses X, Y (or Z, or parallel axis X, Y, Z), and the arc center coordinate value is assigned with addresses I, J (or K).

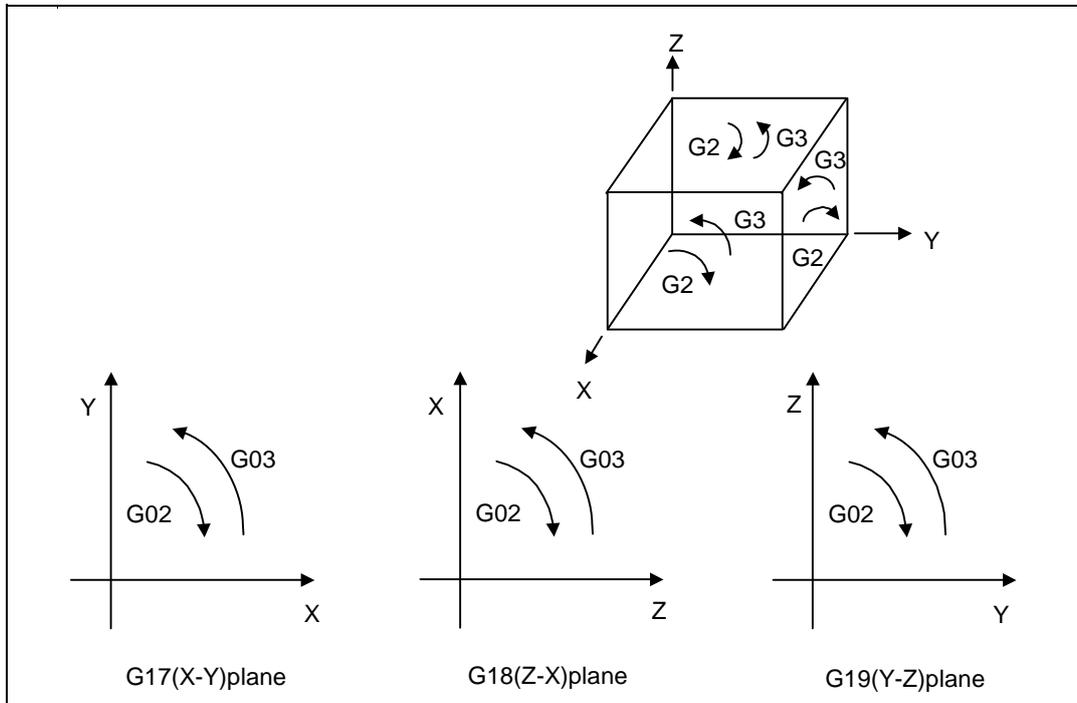
Either an absolute value or incremental value can be used for the arc end point coordinate value command, but the arc center coordinate value must always be commanded with an incremental value from the start point.

The arc center coordinate value is commanded with an input setting unit. Caution is required for the arc command of an axis for which the input command value differs. Command with a decimal point to avoid confusion.



## Detailed description

- (1) G02 (or G03) is retained until another G command (G00, G01 or G33) in the 01 group that changes its mode is issued.  
 The arc rotation direction is distinguished by G02 and G03.  
 G02 Clockwise (CW)  
 G03 Counterclockwise (CCW)

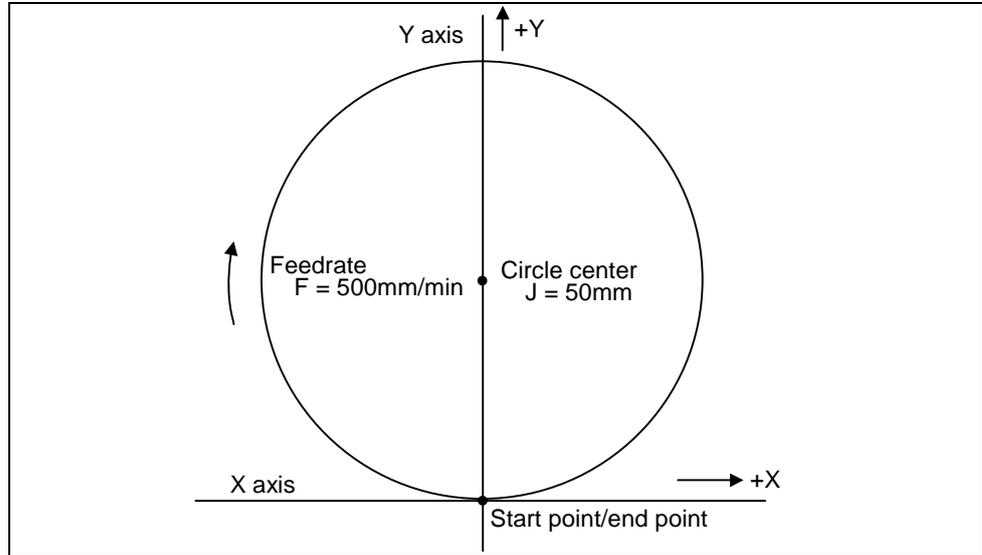


- (2) An arc which extends for more than one quadrant can be executed with a single block command.
- (3) The following information is needed for circular interpolation.
- Plane selection ..... : Is there an arc parallel to one of the XY, ZX or YZ planes?
  - Rotation direction ..... : Clockwise (G02) or counterclockwise (G03)?
  - Arc end point coordinates... : Given by addresses X, Y, Z
  - Arc center coordinates ..... : Given by addresses I, J, K (incremental commands)
  - Feed rate ..... : Given by address F



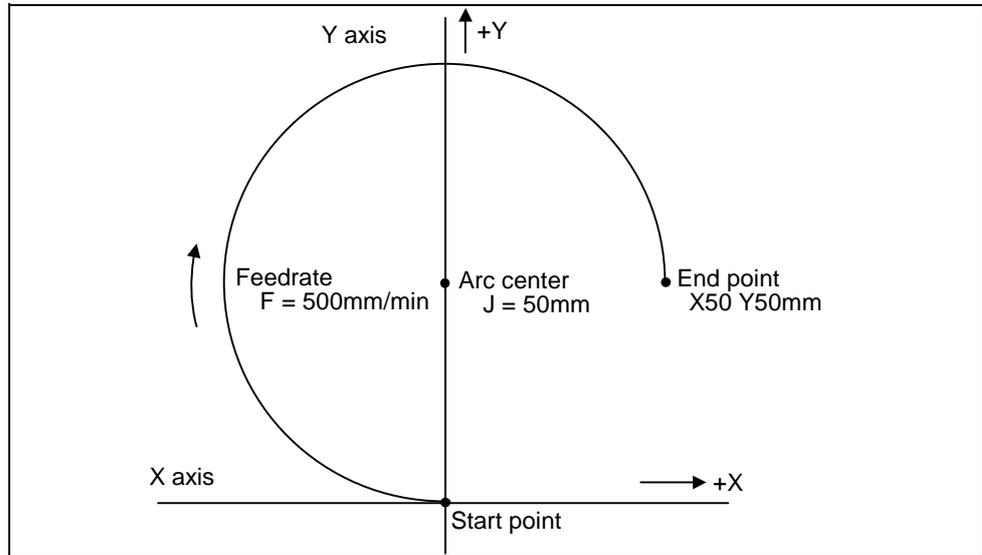
Example of program

(Example 1)



G02 J50000 F500 ;	Circle command
-------------------	----------------

(Example 2)



G91 G02 X50000 Y50000 J50000 F500 ;	3/4 command
-------------------------------------	-------------



### Plane selection

The planes in which the arc exists are the following three planes (refer to the detailed drawings), and are selected with the following method.

XY plane

G17; Command with a (plane selection G code)

ZX plane

G18; Command with a (plane selection G code)

YZ plane

G19; Command with a (plane selection G code)

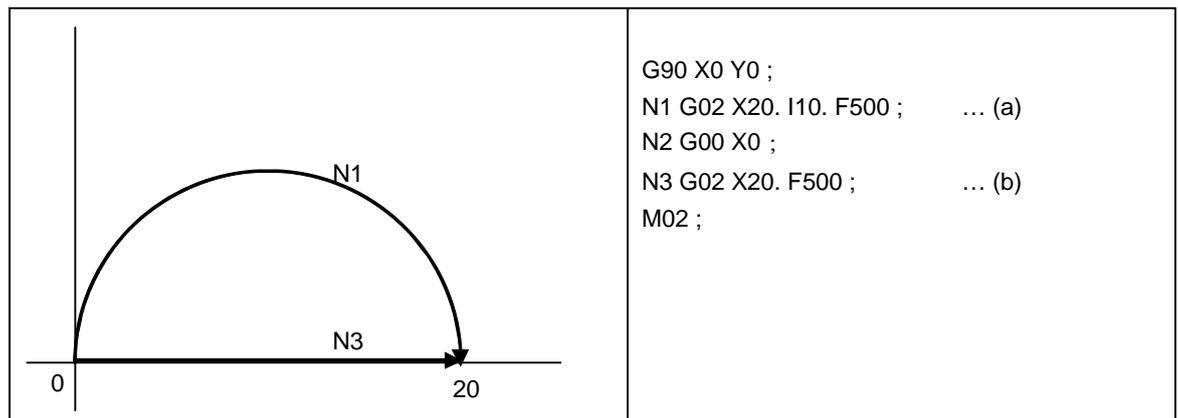


### Change into linear interpolation command

Program error (P33) will occur when the center and radius are not designated at circular command. When the parameter "#11029 Arc to G1 no Cent (Change command from arc to linear when no arc center designation)" is set, the linear interpolation can be applied to terminal coordinates value for only the block. However, a modal is the circular modal.

This function is not applied to a circular command by a geometric function.

(Example) The parameter "#11029 Arc to G1 no Cent (Change command from arc to linear when no arc center designation)" = "1"



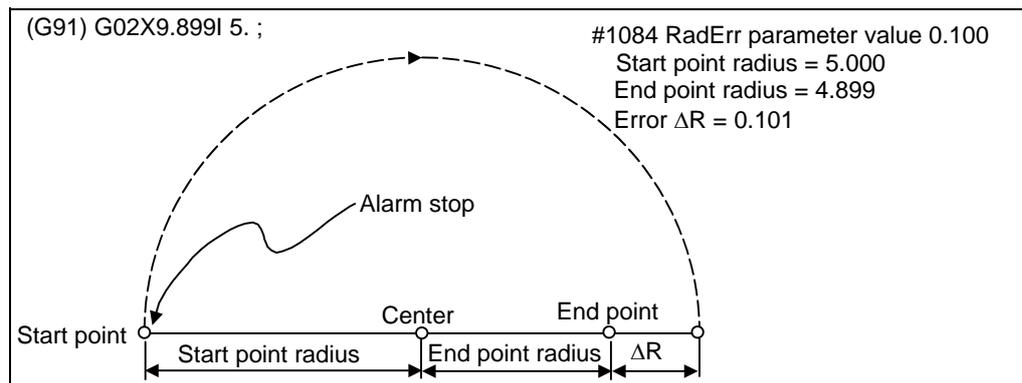
(a) The circular interpolation (G02) is executed because there is a center command.

(b) The linear interpolation (G01) is executed because there is no center and radius command.

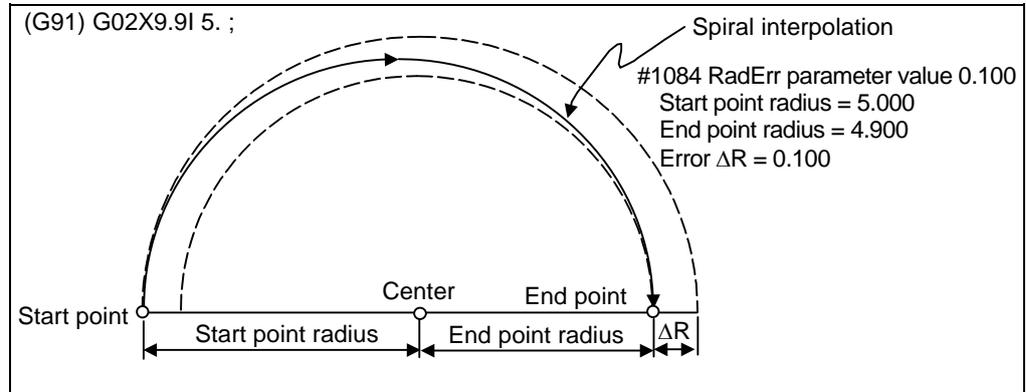


## Precautions for circular interpolation

- (1) The terms "clockwise" (G02) and "counterclockwise" (G03) used for arc operations are defined as a case where in a right-hand coordinate system, the negative direction is viewed from the position direction of the coordinate axis which is at right angles to the plane in question.
- (2) When all the end point coordinates are omitted or when the end point is the same position as the start point, a 360° arc (full circle) is commanded when the center is commanded using I, J and K.
- (3) The following occurs when the start and end point radius do not match in an arc command :
  - (a) Program error (P70) results at the arc start point when error  $\Delta R$  is greater than parameter "#1084 RadErr".

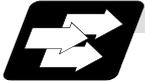


- (b) Spiral interpolation in the direction of the commanded end point results when error  $\Delta R$  is less than the parameter value.



The parameter setting range is from 0.001mm to 1.000mm.

## 6.5 R-specified Circular Interpolation; G02, G03



## Function and purpose

Along with the conventional circular interpolation commands based on the arc center coordinate (I, J, K) designation, these commands can also be issued by directly designating the arc radius R.



## Command format

**G02 (G03) X\_\_ Y\_\_ R\_\_ F\_\_ ;**

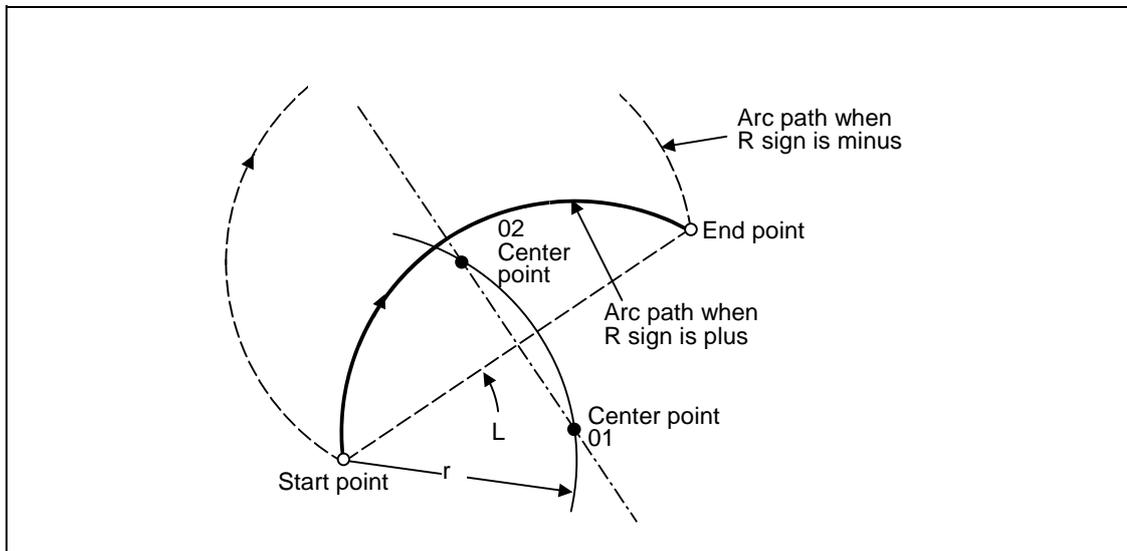
X : X axis end point coordinate  
 Y : Y axis end point coordinate  
 R : Arc radius  
 F : Feedrate

The arc radius is commanded with an input setting unit. Caution is required for the arc command of an axis for which the input command value differs. Command with a decimal point to avoid confusion.



## Detailed description

The arc center is on the bisector line which is perpendicular to the line connecting the start and end points of the arc. The point, where the arc with the specified radius whose start point is the center intersects the perpendicular bisector line, serves as the center coordinates of the arc command. If the R sign of the commanded program is plus, the arc is smaller than a semisphere; if it is minus, the arc is larger than a semisphere.



The following condition must be met with an R-specified arc interpolation command:

$L/(2r) \leq 1$  An error will occur when  $L/2 - r > (\text{parameter : \#1084 RadErr})$

Where L is the line from the start point to end point.

When the R specification and I, J, K specification are contained in the same block, the R specification has priority in processing.

When the R specification and I, J, K specification are contained in the same block, the R specification has priority in processing.

The plane selection is the same as for the I, J, K-specified arc command.



## Example of program

## (Example 1)

G02 Xx <sub>1</sub> Yy <sub>1</sub> Rr <sub>1</sub> Ff <sub>1</sub> ;	XY plane R-specified arc
---	--------------------------

## (Example 2)

G03 Zz <sub>1</sub> Xx <sub>1</sub> Rr <sub>1</sub> Ff <sub>1</sub> ;	ZX plane R-specified arc
---	--------------------------

## (Example 3)

G02 Xx <sub>1</sub> Yy <sub>1</sub> Ii <sub>1</sub> Jj <sub>1</sub> Rr <sub>1</sub> Ff <sub>1</sub> ;	XY plane R-specified arc (When the R specification and I, J, (K) specification are contained in the same block, the R specification has priority in processing.)
---	---

## (Example 4)

G17 G02 Ii <sub>1</sub> Jj <sub>1</sub> Rr <sub>1</sub> Ff <sub>1</sub> ;	XY plane This is an R-specified arc, but as this is a circle command, it is already completed.
---	--

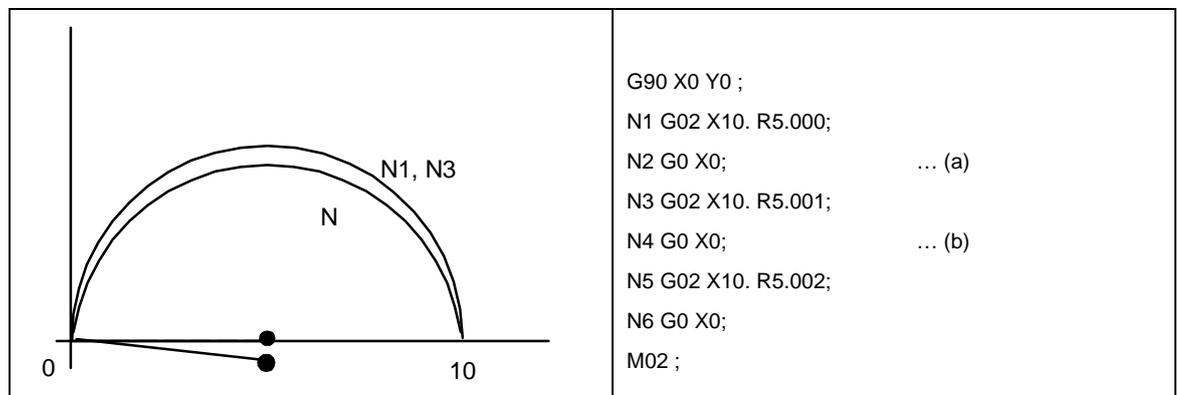


### Circular center coordinate compensation

When "the error margin between the segment connecting the start and end points" and "the commanded radius  $\times 2$ " is less than the setting value because the required semicircle is not obtained by calculation error in R specification circular interpolation, "the midpoint of segment connecting the start and end points" is compensated as the circular center. Set the setting value to the parameter "#11028 Tolerance Arc Cent (Tolerable correction value of arc center error)".

(Ex.) "#11028 Tolerance Arc Cent" = "0.000 (mm)"

Setting value	Tolerance value
Setting value < 0	0 (Center error will not be interpolated)
Setting value = 0	2 $\times$ minimum setting increment
Setting value > 0	Setting value



- (a) Compensate the center coordinate: Same as N1 path  
 (b) Do not compensate the center coordinate: Inside path a little than N1

Calculation error margin compensation allowance value: 0.002 mm

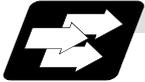
Segment connecting the start and end points: 10.000

N3: Radius  $\times 2 = 10.002$  "Error 0.002 -> Compensate"

N5: Radius  $\times 2 = 10.004$  "Error 0.004 -> Do not compensate"

Therefore, this example is shown in the above figure.

## 6.6 Helical Interpolation ; G17 to G19, G02, G03



## Function and purpose

While circular interpolating with G02/G03 within the plane selected with the plane selection G code (G17, G18, G19), the 3rd axis can be linearly interpolated.

Normally, the helical interpolation speed is designated with the tangent speed  $F'$  including the 3rd axis interpolation element as shown in the lower drawing of Fig. 1. However, when designating the arc plane element speed, the tangent speed  $F$  on the arc plane is commanded as shown in the upper drawing of Fig. 1.

The NC automatically calculates the helical interpolation tangent speed  $F'$  so that the tangent speed on the arc plane is  $F$ .

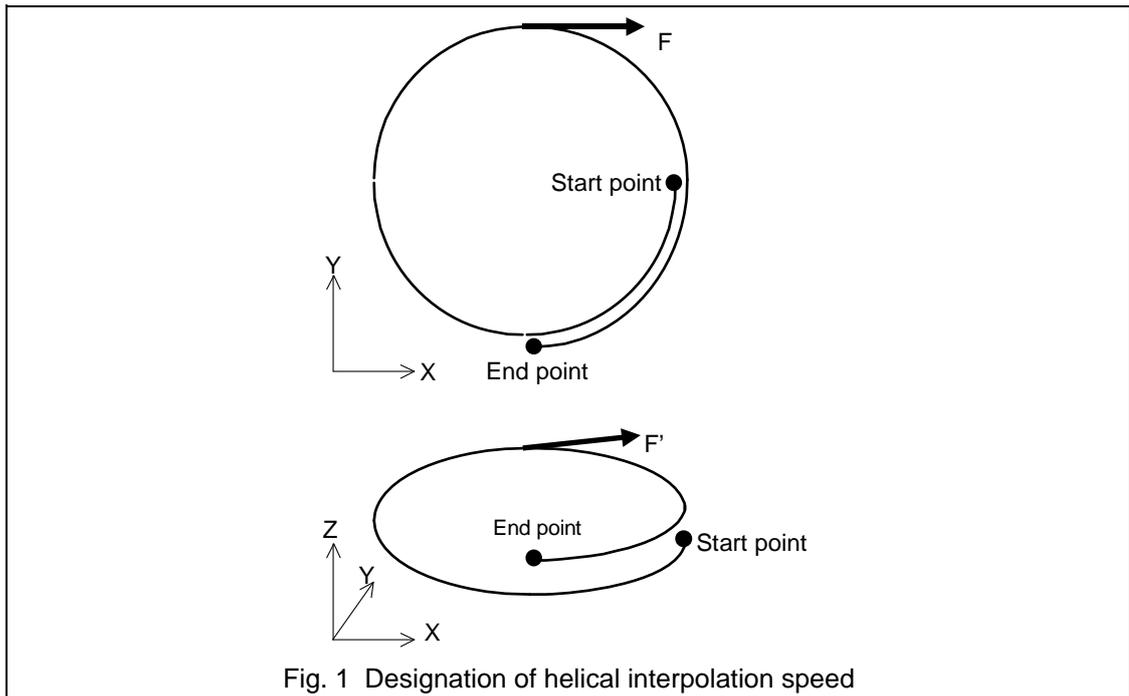


Fig. 1 Designation of helical interpolation speed



## Command format

**G17 G02 (G03) Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> Ii<sub>1</sub> Jj<sub>1</sub> Pp<sub>1</sub> Ff<sub>1</sub> ; Helical interpolation command (Specify arc center)**  
**G17 G02 (G03) Xx<sub>2</sub> Yy<sub>2</sub> Zz<sub>2</sub> Rr<sub>2</sub> Ff<sub>2</sub> ; Helical interpolation command (Specify radius (R))**

G17(G18, G19) : Plane selection (G17: XY plane, G18: ZX plane, G19: YZ plane)  
 G02(G03) : Arc rotation direction  
 Xx<sub>1</sub> Yy<sub>1</sub> Xx<sub>2</sub> Yy<sub>2</sub> : Arc end point coordinate  
 Zz<sub>1</sub> Zz<sub>2</sub> : Linear axis end point coordinate  
 Ii<sub>1</sub> Jj<sub>1</sub> : Arc center coordinate  
 Pp<sub>1</sub> : Pitch No.  
 Ff<sub>1</sub> Ff<sub>2</sub> : Feedrate  
 Rr<sub>2</sub> : Arc radius

The arc center coordinate and arc radius are commanded with an input setting input. Caution is required for the helical interpolation command of an axis for which the input command value differs. Command with a decimal point to avoid confusion.

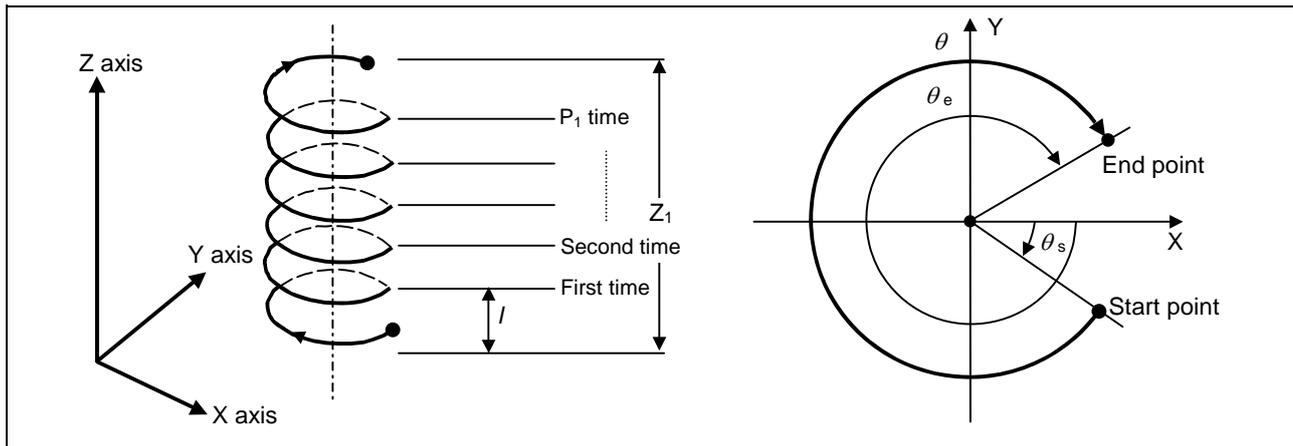
Absolute or incremental values can be assigned for the arc end point coordinates and the end point coordinates of the linear axis, but incremental values must be assigned for the arc center coordinates.

The arc plane element speed designation and normal speed designation can be selected with the parameter.

#1235 set07/bit0	Meaning
1	Arc plane element speed designation is selected.
0	Normal speed designation is selected.



### Normal speed designation



- (1) This command should be issued with a linear axis (multiple axes can be commanded) that does not contain a circular axis in the circular interpolation command combined.
- (2) For feedrate F, command the X, Y Z axis composite element directions speed.
- (3) Pitch l is obtained with the following expression.

$$l = \frac{Z1}{(2\pi \cdot P1 + \theta) / 2\pi}$$

$$\theta = \theta_e - \theta_s = \tan^{-1} \frac{y_e}{x_e} - \tan^{-1} \frac{y_s}{x_s} \quad (0 \leq \theta < 2\pi)$$

Where  $x_s, y_s$  are the start point coordinates from the arc center  
 $x_e, y_e$  are the end point coordinates from the arc center

- (4) If pitch No. is 0, address P can be omitted.

**(Note)** The pitch No. P command range is 0 to 9999.

The pitch No. designation (P command) cannot be made with the R-specified arc.

- (5) Plane selection

The helical interpolation arc plane selection is determined with the plane selection mode and axis address as for the circular interpolation. For the helical interpolation command, the plane where circular interpolation is executed is commanded with the plane selection G code (G17, G18, G19), and the 2 circular interpolation axes and linear interpolation axis (axis that intersects with circular plane) 3 axis addresses are commanded.

XY plane circular, Z axis linear

Command the X, Y and Z axis addresses in the G02 (G03) and G17 (plane selection G code) mode.

ZX plane circular, Y axis linear

Command the X, Y and Z axis addresses in the G02 (G03) and G18 (plane selection G code) mode.

YZ plane circular, X axis linear

Command the X, Y and Z axis addresses in the G02 (G03) and G19 (plane selection G code) mode.

The plane for an additional axis can be selected as with circular interpolation.

UY plane circular, Z axis linear

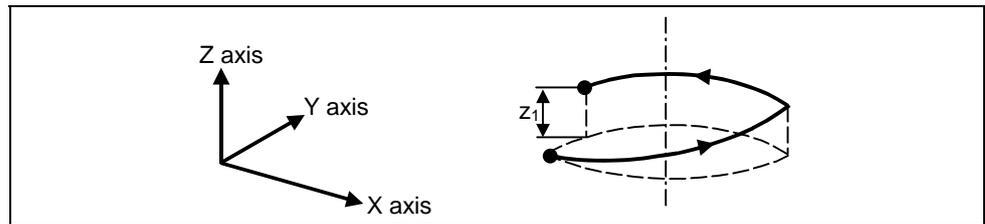
Command the U, Y and Z axis addresses in the G02 (G03) and G19 (plane selection G code) mode.

In addition to the basic command methods above, the command methods following the program example can be used. Refer to the section "6.3 plane selection" for the arc planes selected with these command methods.



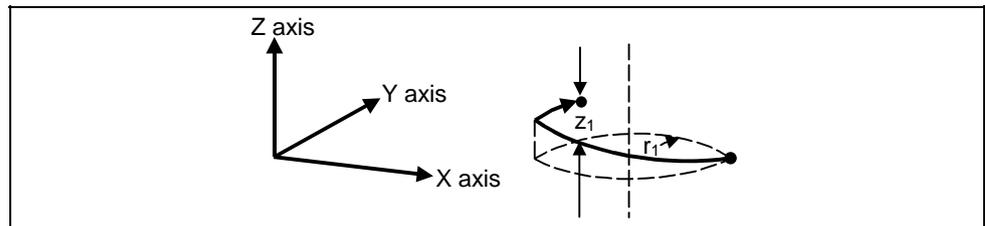
Example of program

(Example 1)



G17 ;	XY plane
G03 Xx <sub>1</sub> Yy <sub>1</sub> Zz <sub>1</sub> Ii <sub>1</sub> Jj <sub>1</sub> P0 Ff <sub>1</sub> ;	XY plane arc, Z axis linear

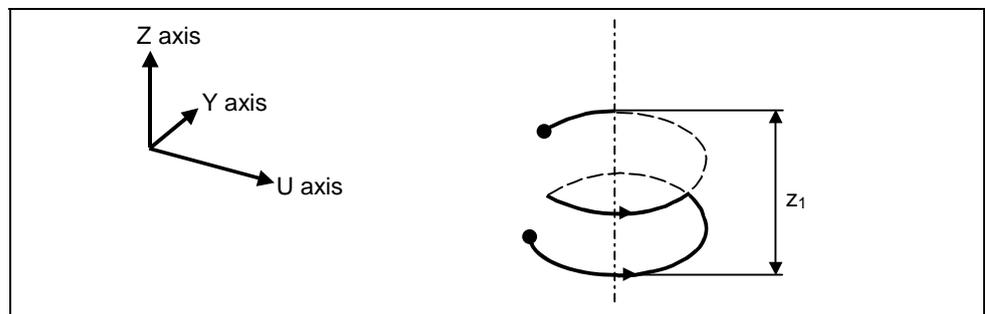
(Note) If pitch No. is 0, address P can be omitted.



(Example 2)

G17 ;	XY plane
G02 Xx <sub>1</sub> Yy <sub>1</sub> Zz <sub>1</sub> Rr <sub>1</sub> Ff <sub>1</sub> ;	XY plane arc, Z axis linear

(Example 3)



G17 G03 Uu <sub>1</sub> Yy <sub>1</sub> Zz <sub>1</sub> Ii <sub>1</sub> Jj <sub>1</sub> P2 Ff <sub>1</sub> ;	UY plane arc, Z axis linear
--	-----------------------------

**(Example 4)**

G18 G03 Xx <sub>1</sub> Uu <sub>1</sub> Zz <sub>1</sub> Ii <sub>1</sub> Kk <sub>1</sub> Ff <sub>1</sub> ;	ZX plane arc, U axis linear
---	-----------------------------

**(Note)** If the same system is used, the standard axis will perform circular interpolation and the additional axis will perform linear interpolation.

**(Example 5)**

G18 G02 Xx <sub>1</sub> Uu <sub>1</sub> Yy <sub>1</sub> Zz <sub>1</sub> Ii <sub>1</sub> Jj <sub>1</sub> Kk <sub>1</sub> Ff <sub>1</sub> ;	ZX plane arc, U axis, Y axis linear (The J command is ignored)
---	---

**(Note)** Two or more axes can be designated for the linear interpolation axis.

**Arc plane element speed designation**

If arc plane element speed designation is selected, the F command will be handled as modal data in the same manner as the normal F command. This will also apply to the following G01, G02 and G03 commands.

For example, the program will be as follows.

**(Example)**

G17 G91 G02 X10. Y10. Z-4. I10, F100 ;	Helical interpolation at speed at which arc plane element is F100
G01 X20. ;	Linear interpolation at F100
G02 X10. Y-10. Z4. J10. ;	Helical interpolation at speed at which arc plane element is F100
G01 Y-40. F120 ;	Linear interpolation at F120
G02 X-10. Y-10. Z-4. I10. ;	Helical interpolation at speed at which arc plane element is F120
G01 X-20. ;	Linear interpolation at F120

When the arc plane element speed designation is selected, only the helical interpolation speed command is converted to the speed commanded with the arc plane element. The other linear and arc commands operate as normal speed commands.

- (1) The actual feedrate display (Fc) indicates the tangent element of the helical interpolation.
- (2) The modal value speed display (FA) indicates the command speed.
- (3) The speed data acquired with API functions follows the Fc and FA display.
- (4) This function is valid only when feed per minute (asynchronous feed: G94) is selected. If feed per revolution (synchronous feed: G95) is selected, the arc plane element speed will not be designated.
- (5) The helical interpolation option is required to use this function.

## 6.7 Thread Cutting

## 6.7.1 Constant Lead Thread Cutting; G33



## Function and purpose

The G33 command exercises feed control over the tool which is synchronized with the spindle rotation and so this makes it possible to conduct constant-lead straight thread-cutting and tapered thread-cutting. Multiple thread screws, etc., can also be machined by designating the thread cutting angle.



## Command format

**G33 Z\_(X\_ Y\_  $\alpha$ \_ ) F\_ Q\_ ; (Normal lead thread cutting commands)**

Z (X Y  $\alpha$ ) : Thread end point  
 F : Lead of long axis (axis which moves most) direction  
 Q : Thread cutting start shift angle, (0.000 to 360.000°)

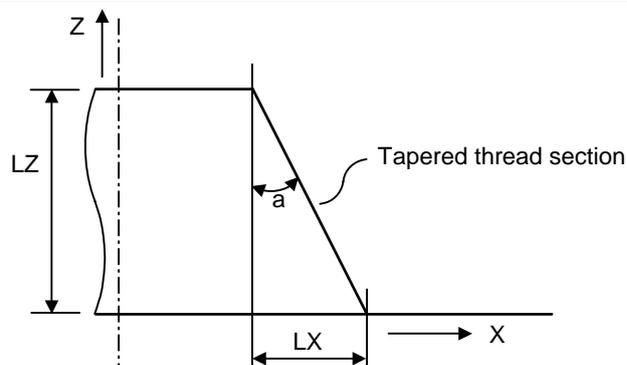
**G33 Z\_(X\_ Y\_  $\alpha$ \_ ) E\_ Q\_ ; (Precision lead thread cutting commands)**

Z (X Y  $\alpha$ ) : Thread end point  
 E : Lead of long axis (axis which moves most) direction  
 Q : Thread cutting start shift angle, (0.000 to 360.000°)



## Detailed description

- (1) The E command is also used for the number of ridges in inch thread cutting, and whether the ridge number or precision lead is to be designated can be selected by parameter setting. (Precision lead is designated by setting the parameter "#1229 set 01/bit 1" to 1.)
- (2) The lead in the long axis direction is commanded for the taper thread lead.



When  $a < 45^\circ$ , the lead is LZ.

When  $a > 45^\circ$ , the lead is LX.

When  $a = 45^\circ$ , the lead can be in either LX or LZ.

## Thread cutting metric input

Input unit system	B (0.001mm)			C (0.0001mm)		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)	F (mm/rev)	E (mm/rev)	E (ridges/inch)
Minimum command unit	1(=1.000) (1.=1.000)	1(=1.00000) (1.=1.00000)	1(=1.00) (1.=1.00)	1(=1.0000) (1.=1.0000)	1(=1.000000) (1.=1.000000)	1(=1.000) (1.=1.000)
Command range	0.001~ 999.999	0.00001~ 999.99999	0.03~999.99	0.0001~ 999.9999	0.000001~ 999.999999	0.026~ 999.999

Input unit system	D (0.00001mm)			E (0.000001mm)		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)	F (mm/rev)	E (mm/rev)	E (ridges/inch)
Minimum command unit	1(=1.00000) (1.=1.00000)	1(=1.0000000) (1.=1.0000000)	1(=1.0000) (1.=1.0000)	1(=1.000000) (1.=1.000000)	1(=1.00000000) (1.=1.00000000)	1(=1.00000) (1.=1.00000)
Command range	0.00001~ 999.99999	0.0000001~ 999.999999	0.0255~ 999.9999	0.000001~ 999.99999	0.00000001~ 999.9999999	0.02541~ 999.99999

## Thread cutting inch input

Input unit system	B (0.0001inch)			C (0.00001inch)		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)	F (inch/rev)	E (inch/rev)	E (ridges/inch)
Minimum command unit	1(=1.0000) (1.=1.0000)	1(=1.000000) (1.=1.000000)	1(=1.0000) (1.=1.0000)	1(=1.00000) (1.=1.00000)	1(=1.0000000) (1.=1.0000000)	1(=1.00000) (1.=1.00000)
Command range	0.0001~99.9999	0.000001~ 39.370078	0.0101~ 9999.9999	0.00001~ 99.99999	0.0000001~ 39.3700787	0.01001~ 9999.99999

Input unit system	D (0.000001inch)			E (0.0000001inch)		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)	F (inch/rev)	E (inch/rev)	E (ridges/inch)
Minimum command unit	1(=1.000000) (1.=1.000000)	1(=1.00000000) (1.=1.00000000)	1(=1.000000) (1.=1.000000)	1(=1.0000000) (1.=1.0000000)	1(=1.000000000) (1.=1.000000000)	1(=1.0000000) (1.=1.0000000)
Command range	0.000001~ 99.999999	0.00000001~ 39.37007874	0.010001~ 9999.99999	0.0000001~ 99.999999	0.000000001~ 39.370078740	0.0100001~ 9999.999999

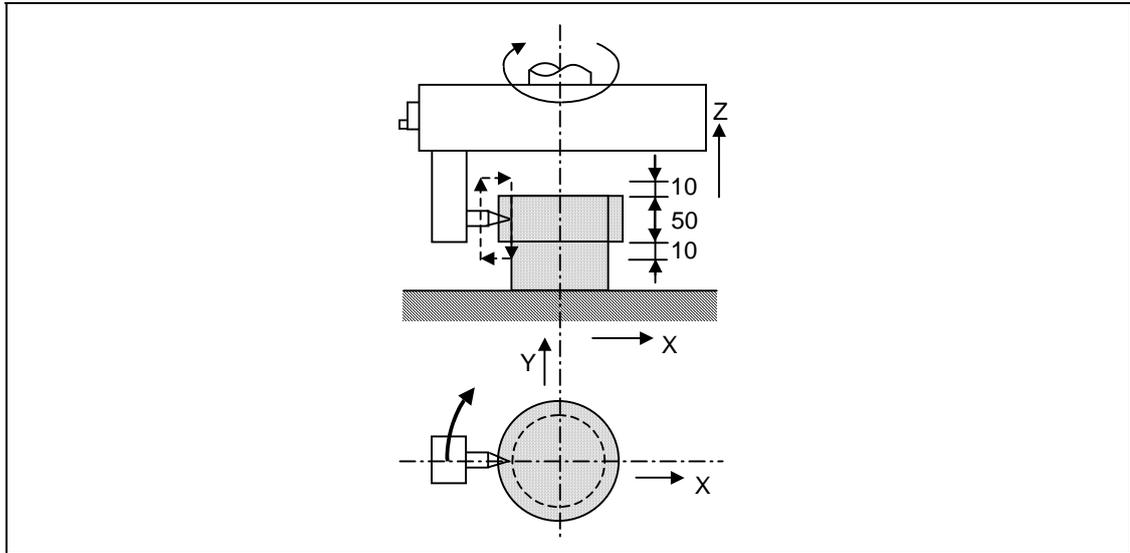
**(Note 1)** It is not possible to assign a lead where the feed rate as converted into per-minute feed exceeds the maximum cutting feed rate.

- (3) The constant surface speed control function should not be used for taper thread cutting commands or scrolled thread cutting commands.
- (4) The thread cutting command waits for the "single rotation sync" signal of the rotary encoder and starts movement. Make sure to carry out timing-synchronization between part systems before issuing a thread cutting command with multiple part systems.  
For example, with the 1-spindle specifications with two part systems, if one part system issues a thread cutting command during ongoing thread cutting by another part system, the movement will start without waiting for the rotary encoder "single rotation sync" signal causing an illegal operation.
- (5) The spindle speed should be kept constant throughout from the rough cutting until the finishing.

- (6) If the feed hold function is employed during thread cutting to stop the feed, the thread ridges will lose their shape. For this reason, feed hold does not function during thread cutting. Note that this is valid from the time the thread cutting command is executed to the time the axis moves.  
If the feed hold switch is pressed during thread cutting, block stop will result at the end point of the block following the block in which thread cutting is completed (no longer G33 mode).
- (7) The converted cutting feedrate is compared with the cutting feed clamp rate when thread cutting starts, and if it is found to exceed the clamp rate, an operation error will result.
- (8) In order to protect the lead during thread cutting, a cutting feed rate which has been converted may sometimes exceed the cutting feed clamp rate.
- (9) An illegal lead is normally produced at the start of the thread and at the end of the cutting because of servo system delay and other such factors.  
Therefore, it is necessary to command a thread length which is determined by adding the illegal lead lengths to the required thread length.
- (10) The spindle speed is subject to the following restriction :
- $$1 \leq R \leq \frac{\text{Maximum feedrate}}{\text{Thread lead}}$$
- Where R ≤ Permissible speed of encoder (r/min)
- R : Spindle speed (r/min)
- Thread lead : mm or inches
- Maximum feedrate : mm/min or inch/mm (This is subject to the restrictions imposed by the machine specifications).
- (11) When the thread lead is extremely large to the maximum cutting feedrate enough to satisfy "R<1" in the formula of (10) above, the program error (P93) may occur.
- (12) Though dry run is valid for thread cutting, the feed rate based on dry run is not synchronized with the spindle rotation.  
The dry run signal is checked at the start of thread cutting and any switching during thread cutting is ignored.
- (13) Synchronous feed applies for the thread cutting commands even with an asynchronous feed command (G94).
- (14) Spindle override and cutting feed override are invalid and the speeds are fixed to 100% during thread cutting.
- (15) When a thread cutting is commanded during tool radius compensation, the compensation is temporarily canceled and the thread cutting is executed.
- (16) When the mode is switched to another automatic mode while G33 is executed, the following block which does not contain a thread cutting command is first executed and then the automatic operation stops.
- (17) When the mode is switched to the manual mode while G33 is executed, the following block which does not contain a thread cutting command is first executed and then the automatic operation stops.  
In the case of a single block, the following block which does not contain a thread cutting command (when G33 mode is cancelled) is first executed and then the automatic operation stops.  
Note that automatic operation will stop immediately if the mode is switched before the G33-commanded axis starts moving.
- (18) The handle interruption for automatic operation is valid while thread cutting.
- (19) The thread cutting start shift angle is not a modal. If there is no Q command with G33, this will be handled as "Q0".
- (20) If a value more than 360.000 is commanded with G33 Q, the program error (P35) will occur.
- (21) G33 cuts one row with one cycle. To cut two rows, change the Q value, and issue the same command.

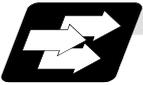


## Example of program



N110 G90 G0 X-200. Y-200. S50 M3 ; N111 Z110. ;	The spindle center is positioned to the workpiece center, and the spindle rotates in the forward direction.
N112 G33 Z40. F6.0 ;	The first thread cutting is executed. Thread lead = 6.0mm
N113 M19 ;	Spindle orientation is executed with the M19 command.
N114 G0X-210. ; N115 Z110. M0 ;	The tool is evaded in the X axis direction. The tool rises to the top of the workpiece, and the program stops with M00. Adjust the tool if required.
N116 X-200. ; M3 ;	Preparation for second thread cutting is done.
N117 G04 X5.0 ;	Command dwell to stabilize the spindle rotation if necessary.
N11 G33 Z40. ;	The second thread cutting is executed.

## 6.7.2 Inch Thread Cutting; G33



## Function and purpose

If the number of ridges per inch in the long axis direction is assigned in the G33 command, the feed of the tool synchronized with the spindle rotation will be controlled, which means that constant-lead straight thread-cutting and tapered thread-cutting can be performed.



## Command format

**G33 Z\_\_ E\_\_ Q\_\_ ;**

Z	:	Thread cutting direction axis address (X, Y, Z, $\alpha$ ) and thread length
E	:	Number of ridges per inch in direction of long axis (axis which moves most) (decimal point command can also be assigned)
Q	:	Thread cutting start shift angle, 0 to 360°.



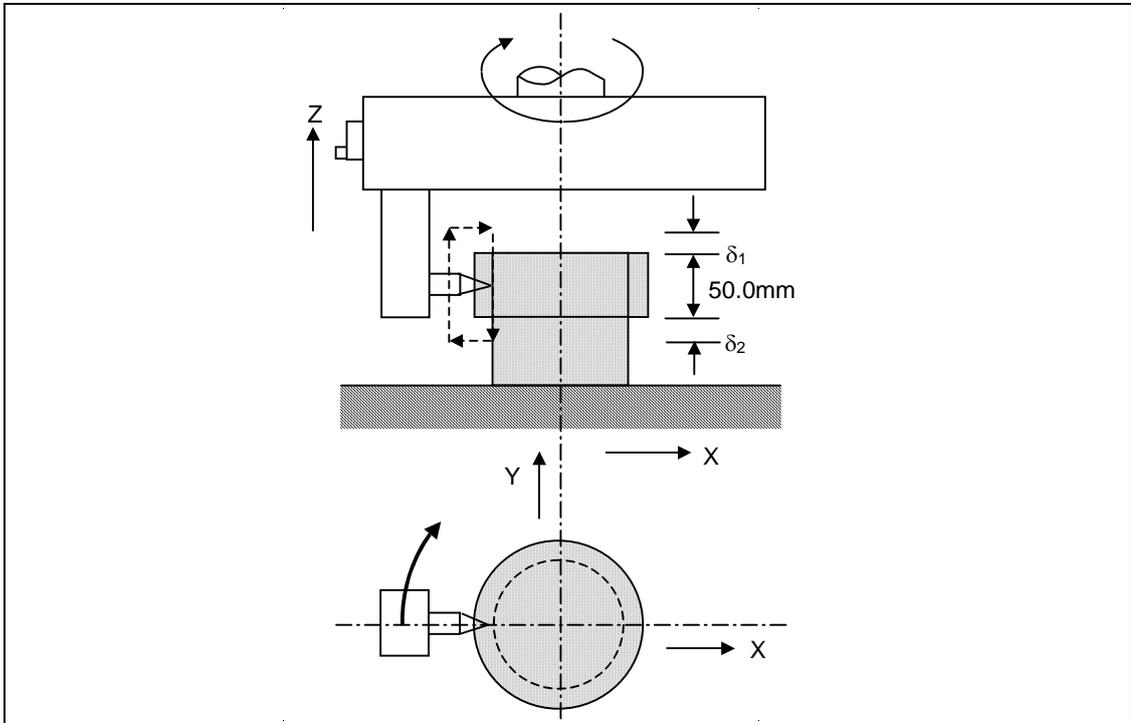
## Detailed description

- (1) The number of ridges in the long axis direction is assigned as the number of ridges per inch.
- (2) The E code is also used to assign the precision lead length, and whether the ridge number of precision lead length is to be designated can be selected by parameter setting. (The number of ridges is designated by setting the parameter "#1229 set01/bit1" to 0.)
- (3) The E command value should be set within the lead value range when the lead is converted.
- (4) The other matters are the same as uniform lead thread cutting.



## Example of program

Thread lead ..... 3 threads/inch (= 8.46666 ...)  
 When programmed with  $\delta_1 = 10\text{mm}$ ,  
 $\delta_2 = 10\text{mm}$  using metric input



N210 G90 G0X-200. Y-200. S50M3;	
N211 Z110.;	
N212 G91 G33 Z-70.E3.0;	(First thread cutting)
N213 M19;	
N214 G90 G0X-210.;	
N215 Z110.M0;	
N216 X-200.;	
M3;	
N217 G04 X2.0;	
N218 G91 G33 Z-70.;	(Second thread cutting)

## 6.8 Unidirectional Positioning; G60



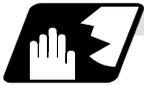
## Function and purpose

The G60 command can position the tool at a high degree of precision without backlash error by locating the final tool position from a single determined direction.



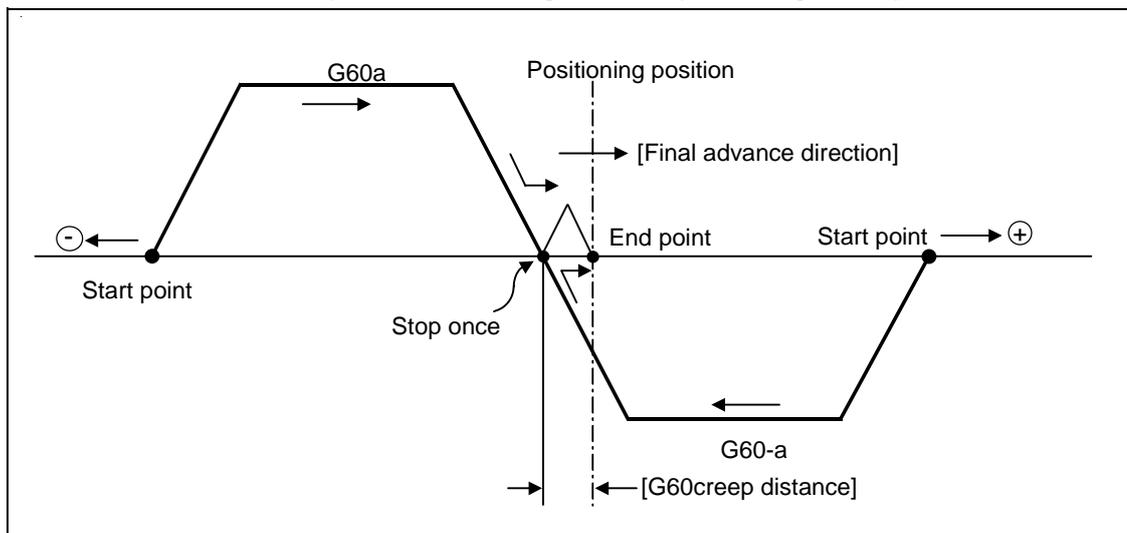
## Command format

**G60 X\_\_ Y\_\_ Z\_\_ α\_\_ ;**  
 α : Optional axis



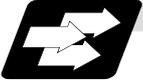
## Detailed description

- (1) The creep distance for the final positioning as well as the final positioning direction is set by parameter.
- (2) After the tool has moved at the rapid traverse rate to the position separated from the final position by an amount equivalent to the creep distance, it moves to the final position in accordance with the rapid traverse setting where its positioning is completed.



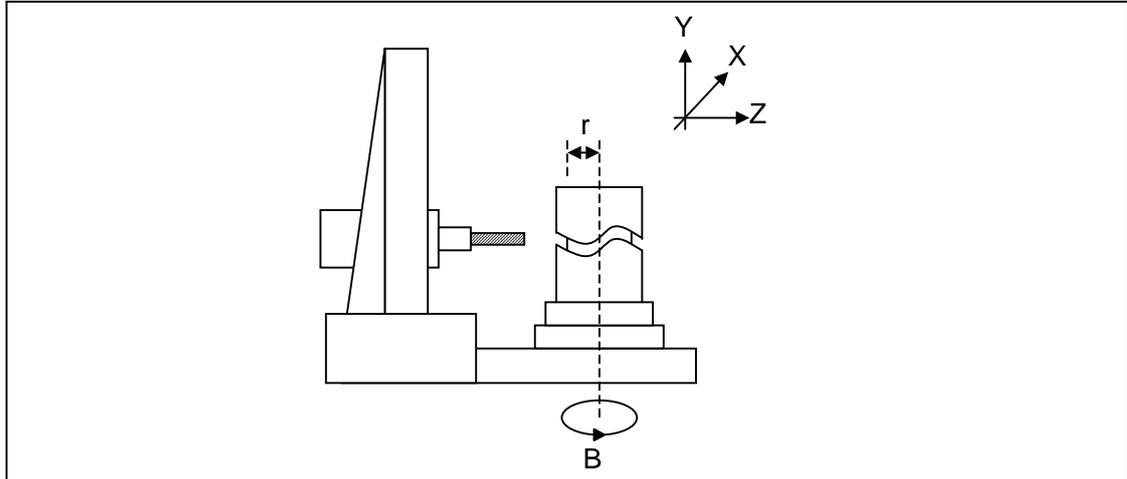
- (3) The above positioning operation is performed even when Z axis commands have been assigned for Z axis cancel and machine lock. (Display only)
- (4) When the mirror image function is ON, the tool will move in the opposite direction as far as the intermediate position due to the mirror image function but the operation within the creep distance during its final advance will not be affected by mirror image.
- (5) The tool moves to the end point at the dry run speed during dry run when the G0 dry run function is valid.
- (6) Feed during creep distance movement with final positioning can be stopped by resetting, emergency stop, interlock, feed hold and rapid traverse override zero.  
The tool moves over the creep distance at the rapid traverse setting. Rapid traverse override is valid.
- (7) Uni-directional positioning is not performed for the drilling axis during drilling fixed cycles.
- (8) Uni-directional positioning is not performed for shift amount movements during the fine boring or back boring fixed cycle.
- (9) Normal positioning is performed for axes whose creep distance has not been set by parameter.
- (10) Uni-directional positioning is always a non-interpolation type of positioning.
- (11) When the same position (movement amount of zero) has been commanded, the tool moves back and forth over the creep distance and is positioned at its original position from the final advance direction.
- (12) Program error (P61) results when the G60 command is assigned with an NC system which has not been provided with this particular specification.

## 6.9 Cylindrical Interpolation; G07.1

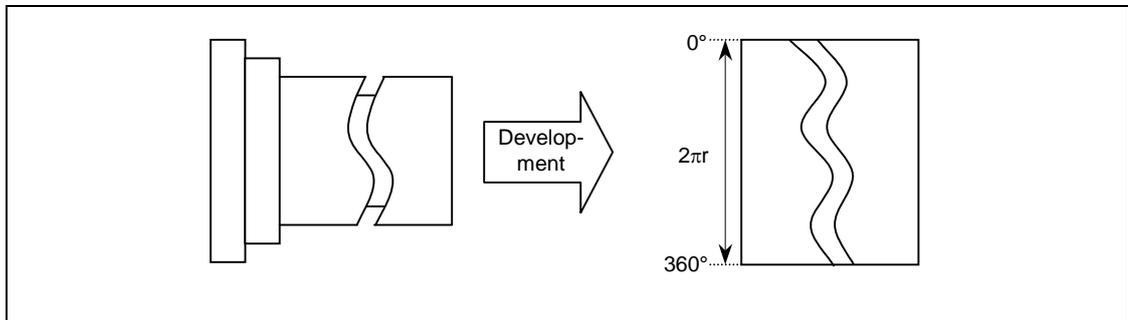


## Function and purpose

This function develops a shape with a cylindrical side (shape in cylindrical coordinate system) into a plane. When the developed shape is programmed as the plane coordinates, that is converted into the linear axis and rotation axis movement in the cylindrical coordinates and the contour is controlled during machining.



As programming can be carried out with a shape with which the side on the cylinder is developed, this is effective for machining cylindrical cams, etc. When programmed with the rotation axis and its orthogonal axis, slits, etc., can be machined on the cylinder side.







## Detailed description

- (1) Command G07.1 in an independent block. A program error (P33) will occur if this command is issued in the same block as another G code.
- (2) Program the rotation axis with an angle degree.
- (3) Linear interpolation or circular interpolation can be commanded during the cylindrical interpolation mode. Note that the plane selection command must be issued just before the G07.1 block.
- (4) The coordinates can be commanded with either an absolute command or incremental command.
- (5) Tool radius compensation can be applied on the program command. Cylindrical interpolation will be executed on the path after tool radius compensation.
- (6) Command the segment feed in the cylinder development with F. The F unit is mm/min or inch/min.
- (7) Cylindrical interpolation accuracy

In the cylindrical interpolation mode, the movement amount of the rotation axis commanded with an angle is converted on the circle periphery, and after operating the linear and circular interpolation between the other axes, the amount is converted into an angle again. Thus, the actual movement amount may differ from the commanded value such as when the cylinder radius is small.

Note that the error generated at this time is not cumulated.

- (8) F command during cylindrical interpolation
 

As for the F command in the cylindrical interpolation mode, whether the previous F command is used or not depends on that the mode just before G07.1 is the feed per minute command (G94/G98) or feed per rotation command (G95/G99).

  - (a) When G94 is commanded just before G07.1
 

If there is no F command in the cylindrical interpolation, the previous F command feedrate will be used.

The feedrate after the cylindrical interpolation mode is canceled will remain the F command feedrate issued when the cylindrical interpolation mode was started or the final F command feedrate set during cylindrical interpolation.
  - (b) When G95 is commanded just before G07.1
 

The previous F command feedrate cannot be used during cylindrical interpolation, thus a new F command must be issued.

The feedrate after the cylindrical interpolation mode is canceled will return to that applied before the cylindrical interpolation mode was started.

**When there is no F command in G07.1**

Previous mode	No F command	After G07.1 is canceled
G94 (G98)	Previous F is used	←
G95 (G99)	Program error (P62)	F just before G07.1 is used

**When F is commanded in G07.1**

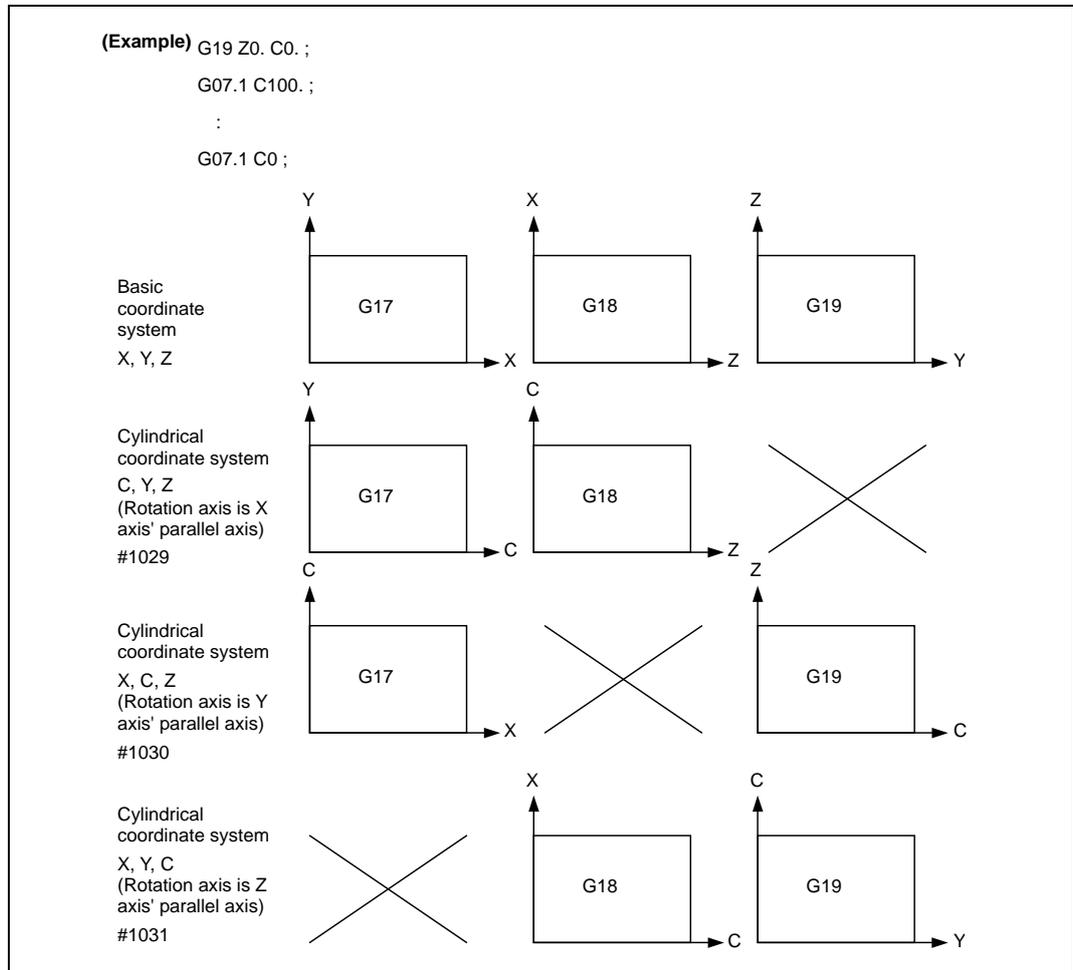
Previous mode	No F command	After G07.1 is canceled
G94 (G98)	Commanded F is used	←
G95 (G99)	Commanded F is used *1	F just before G07.1 is used

\*1) Moves with the feed per minute command during G07.1.

(9) Plane selection

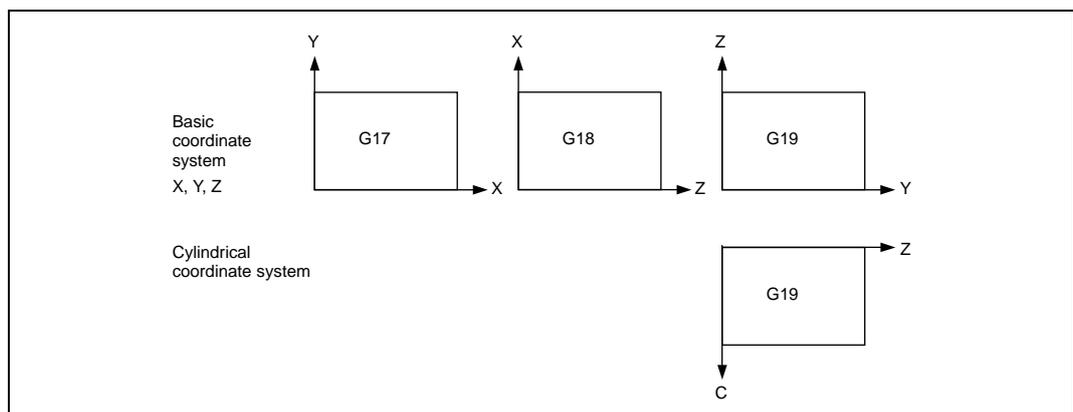
The axis used for cylindrical interpolation must be set with the plane selection command.<sup>(Note)</sup>  
 The correspondence of the rotation axis to an axis' parallel axis is set with the parameters (#1029, #1030, #1031).

The circular interpolation and tool radius compensation, etc., can be designated on that plane.  
 The plane selection command is set immediately before or after the G07.1 command. If not set and a movement command is issued, a program error (P485) will occur.



**(Note)** Depending on the model or version, the Z-C plane (Y-Z cylinder plane) will be automatically selected with G07.1 and G19.

The circular interpolation and tool radius compensation, etc., can be designated on that plane.



## (10) Related parameters

#	Item	Details	Setting range
1516	mill_ax Milling axis name	Set the name of the rotation axis for milling interpolation (pole coordinate interpolation, cylindrical interpolation). Only one of the rotation axes can be set.	A to Z
8111	Milling Radius	Select the diameter and radius of the linear axis for milling interpolation (pole coordinate interpolation, cylindrical interpolation). 0: Radius command for all axes 1: Each axis setting (follows #1019 dia diameter designation axis)	0 / 1
1267 (PR)	ext03 (bit0) G code type	The type of G code is changed. 0: Conventional format 1: Mitsubishi special format	0 / 1
1270 (PR)	ext06 (bit7) Handle C axis coordinate during cylindrical interpolation	Specify whether the rotary axis coordinate before the cylindrical interpolation start command is issued is kept during the cylindrical interpolation or not. 0: Do not keep 1: keep	0 / 1



## Relation with other functions

## (1) The following G code commands can be used during the cylindrical interpolation mode.

G code	Details
G00	Positioning
G01	Linear interpolation
G02	Circular interpolation (CW)
G03	Circular interpolation (CWW)
G04	Dwell
G09	Exact stop check
G40-42	Tool nose R compensation
G61	Exact stop check mode
G64	Cutting mode
G65	Macro call (simple call)
G66	Macro modal call A (modal call)
G66.1	Macro modal call B (block call per macro)
G67	Macro modal call cancel (modal call cancel)
G80-89	Hole drilling fixed cycle
G90/91	Absolute/incremental value command
G94	Asynchronous feed
G98	Hole drilling cycle initial return
G99	Hole drilling cycle R point return

A program error (P481) may occur if a G code other than those listed above is commanded during cylindrical interpolation.

## (2) Circular interpolation

- (a) Circular interpolation between the rotation axis and linear axis is possible during the cylindrical interpolation mode.
- (b) An R specification command can be issued with circular interpolation. (I, J and K cannot be designated.)

**(3) Tool radius compensation**

The tool radius can be compensated during the cylindrical interpolation mode.

- (a) Command the plane selection in the same manner as circular interpolation.  
When using tool radius compensation, start up and cancel the compensation within the cylindrical interpolation mode.
- (b) A program error (P485) will occur if G07.1 is commanded during tool radius compensation.
- (c) If the G07.1 command is issued with no movement command given after the tool radius compensation is canceled, the position of the axis in the G07.1 command block is interpreted as the position applied after the tool radius compensation is canceled and the following operations are performed.

**(4) Tool length compensation**

- (a) A program error (P481) will occur if the tool length compensation is carried out in the cylindrical interpolation mode.

```

      :
      :
G43H12 ;      ... Tool length compensation before cylindrical interpolation → Valid
G0 X100. Z0 ;
G19 Z C ;
G07.1 C100. ;
      :
G43H11 ; ... Tool length compensation in cylindrical interpolation mode → Program error
      :
G07.1 C0 ;

```

- (b) Complete the tool compensation movement (movement of tool length and wear compensation amount) before executing the cylindrical interpolation. If the tool compensation movement is not completed when the cylindrical interpolation start command has been issued, the followings will be resulted:
  - Machine coordinate is not changed even if G07.1 is executed.
  - The workpiece coordinate is changed to that of the post tool length compensation when G07.1 is executed. (Even if canceling the cylindrical interpolation, this workpiece coordinate will not be canceled.)

**(5) Cutting asynchronous feed**

- (a) The asynchronous mode is forcibly set when the cylindrical interpolation mode is started.
- (b) When the cylindrical interpolation mode is canceled, the synchronization mode will return to the state before the cylindrical interpolation mode was started.
- (c) A program error (P485) will occur if G07.1 is commanded in the constant surface speed control mode (G96).

**(6) Miscellaneous functions**

- (a) The miscellaneous function (M) and 2nd miscellaneous function can be issued even in the cylindrical interpolation mode.
- (b) The S command in the cylindrical interpolation mode issues the rotary tool's rotation speed instead of the spindle rotation speed.



#### Restrictions and precautions

- (1) The cylindrical interpolation mode is canceled when the power is turned ON or reset.
- (2) A program error (P484) will occur if any axis commanded for cylindrical interpolation has not completed reference position return.
- (3) Tool radius compensation must be canceled before the cylindrical interpolation mode can be canceled.
- (4) When the cylindrical interpolation mode is canceled, the mode will change to the cutting mode, and the plane will return to that selected before cylindrical interpolation.
- (5) The program of the block during the cylindrical interpolation cannot be restarted (program restart).
- (6) A program error (P486) will occur if the cylindrical interpolation command is issued during the mirror image.
- (7) When the cylindrical interpolation mode is started or canceled, the deceleration check is performed.
- (8) A program error (P481) will occur if the cylindrical interpolation or the pole coordinate interpolation is commanded during the cylindrical interpolation mode.



Example of program

<Program>

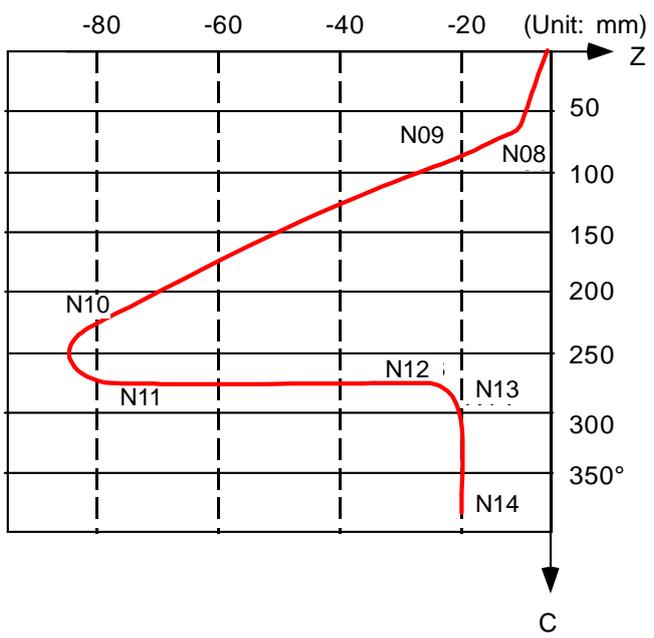
```

N01 G28XZC;
N02 G97S100M23;
N03 G00X50.Z0.;
N04 G94G01X40.F100.;
N05 G19C0Z0;
N06 G07.1C20.;
N07 G41;
N08 G01Z-10.C80.F150;
N09 Z-25.C90.;
N10 Z-80.C225.;
N11 G03Z-75.C270.R55.;
N12 G01Z-25;
N13 G02Z-20.C280.R80.;
N14 G01C360.
N15 G40;
N16 G07.1C0;
N17 G01X50.;
N18 G0X100.Z100.;
N19 M25;
N20 M30;
    
```

<Parameter>	
#1029 aux_I	
#1030 aux_J	C
#1031 aux_K	

← Command of plane selection for cylindrical interpolation and command of two interpolation axes  
 ← Cylindrical interpolation start

← Cylindrical interpolation cancel



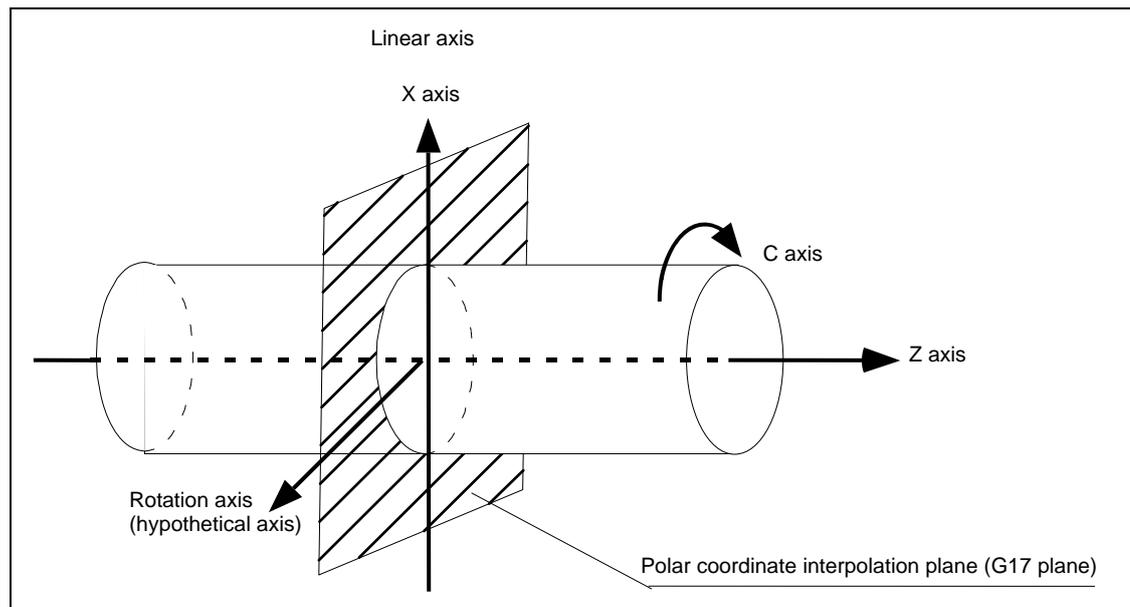
## 6.10 Polar Coordinate Interpolation; G12.1, G13.1



## Function and purpose

This function converts the commands programmed with the orthogonal coordinate axis into linear axis movement (tool movement) and rotation axis movement (workpiece rotation), and controls the contour.

The plane that uses the linear axis as the plane's 1st orthogonal axis, and the intersecting hypothetical axis as the plane's 2nd axis (hereafter "pole coordinate interpolation plane") is selected. Pole coordinate interpolation is carried out on this plane. The workpiece coordinate system zero point is used as the coordinate system zero point during pole coordinate interpolation.



This is effective for cutting a notch section on a linear section of the workpiece diameter, and for cutting cam shafts, etc.



## Command format

**G12.1 ; Pole coordinate interpolation mode start**

**G13.1 ; Pole coordinate interpolation mode cancel**

- (1) The coordinates commanded in the interval from the start to cancellation of the pole coordinate interpolation mode will be the pole coordinate interpolation.

G12.1 ;	Pole coordinate interpolation mode start
:	(Pole coordinate interpolation will start)
:	(The coordinate commands in this interval will be the pole coordinate interpolation)
:	
G13.1 ;	Pole coordinate interpolation mode cancel
	(Pole coordinate interpolation is canceled)



## Detailed description

- (1) Command G12.1 and G13.1 in an independent block. A program error (P33) will occur if this command is issued in the same block as another G code.
- (2) Linear interpolation or circular interpolation can be commanded during the pole coordinate interpolation mode.
- (3) The coordinates can be commanded with either an absolute command or incremental command.
- (4) Tool radius compensation can be applied on the program command. Pole coordinate interpolation will be executed on the path after tool radius compensation.
- (5) Command the segment feed in the pole coordinate interpolation plane (orthogonal coordinate system) with F. The F unit is mm/min or inch/min.
- (6) When the G12.1 command is issued, the deceleration check is executed.
- (7) Plane selection

The linear axis and rotation axis used for pole coordinate interpolation must be set beforehand with the parameters.

- (a) Determine the deemed plane for carrying out pole coordinate interpolation with the parameter (#1533) for the linear axis used for pole coordinate interpolation.

#1533 setting value	Deemed plane
X	G17 (XY plane)
Y	G19 (YZ plane)
Z	G18 (ZX plane)
Blank (no setting)	G17 (XY plane)

- (b) A program error (P485) will occur if the plane selection command (G16 to G19) is issued during the pole coordinate interpolation mode.

**(Note)** Depending on the model or version, parameter (#1533) may not be provided. In this case, the operation will be the same as if the parameter (#1533) is blank (no setting).

- (8) F command during pole coordinate interpolation  
 As for the F command in the pole coordinate interpolation mode, whether the previous F command is used or not depends on that the mode just before G12.1 is the feed per minute command (G94/G98) or feed per rotation command (G95/G99).
- (a) When G94(G98) is commanded just before G12.1  
 If there is no F command in the pole coordinate interpolation, the previous F command feedrate will be used.  
 The feedrate after the pole coordinate interpolation mode is canceled will remain the F command feedrate issued when the pole coordinate interpolation mode was started or the final F command feedrate set during pole coordinate interpolation.  
 The previous F command feedrate cannot be used during pole coordinate interpolation.
- (b) When G95(G99) is commanded just before G12.1  
 The previous F command feedrate cannot be used during pole coordinate interpolation. A new F command must be issued.  
 The feedrate after the pole coordinate interpolation mode is canceled will return to that applied before the pole coordinate interpolation mode was started.

**When there is no F command in G12.1**

Previous mode	No F command	After G13.1 is canceled
G94(G98)	Previous F is used	←
G95(G99)	Program error (P62)	F just before G12.1 is used

**When F is commanded in G12.1**

Previous mode	No F command	After G07.1 is canceled
G94(G98)	Commanded F is used	←
G95(G99)	Commanded F is used *1	F just before G12.1 is used

\*1) Moves with the feed per minute command during G12.1.

(9) Related parameters

#	Item	Details	Setting range
1516	mill_ax Milling axis name	Set the name of the rotation axis for milling interpolation (pole coordinate interpolation, cylindrical interpolation). Only one of the rotation axes can be set.	A to Z
1517	mill_c Milling interpolation hypothetical axis name	Select the hypothetical axis command name for milling interpolation (pole coordinate interpolation, cylindrical interpolation). 0: Y axis command 1: Command rotation axis name	0 / 1
8111	Milling Radius	Select the diameter and radius of the linear axis for milling interpolation. 0: Radius command for all axes 1: Each axis setting (follows #1019 dia diameter designation axis)	0 / 1
1533	mill_Pax Polar coordinate linear axis name	Set the linear axis for polar coordinate interpolation.	Axis name



## Relation with other functions

- (1) The following G code commands can be used during the pole coordinate interpolation mode.

G code	Details
G00	Positioning
G01	Linear interpolation
G02	Circular interpolation (CW)
G03	Circular interpolation (CCW)
G04	Dwell
G09	Exact stop check
G40-42	Tool radius compensation
G61	Exact stop check mode
G64	Cutting mode
G65	Macro call (simple call)
G66	Macro modal call A (modal call)
G66.1	Macro modal call B (block call per macro)
G67	Macro modal call cancel (modal call cancel)
G80-89	Hole drilling fixed cycle
G90/91	Absolute/incremental value command
G94	Asynchronous feed
G98	Hole drilling cycle initial return
G99	Hole drilling cycle R point return

A program error (P481) may occur if a G code other than those listed above is commanded during pole coordinate interpolation.

- (2) Program commands during pole coordinate interpolation

- (a) The program commands in the pole coordinate interpolation mode are commanded with the orthogonal coordinate value of the linear axis and rotation axis (hypothetical axis) on the pole coordinate interpolation plane.  
The axis address of the rotation axis (C) is commanded as the axis address for the plane's 2nd axis (hypothetical axis) command.  
The command unit is not degree, and instead is the same unit (mm or inch) as the command issued with the axis address for the plane's 1st axis (linear axis).
- (b) The hypothetical axis coordinate value will be set to "0" when G12.1 is commanded. In other words, the position where G12.1 is commanded will be interpreted as angle = 0, and the pole coordinate interpolation will start.

- (3) Circular interpolation on pole coordinate plane

The arc radius address for carrying out circular interpolation during the pole coordinate interpolation mode is determined with the linear axis parameter (#1533).

#1533 setting value	Center designation command
X	I, J (pole coordinate plane is interpreted as XY plane)
Y	J, K (pole coordinate plane is interpreted as YZ plane)
Z	K, I (pole coordinate plane is interpreted as ZX plane)
Blank (no setting)	I, J (pole coordinate plane is interpreted as XY plane)

The arc radius can also be designated with the R command.

- (Note)** Depending on the model or version, parameter (#1533) may not be provided. In this case, the operation will be the same as if the parameter (#1533) is blank (no setting).

## (4) Tool radius compensation

The tool radius can be compensated during the pole coordinate interpolation mode.

- (a) Command the plane selection in the same manner as pole coordinate interpolation. When using tool radius compensation, it must be started up and canceled within the pole coordinate interpolation mode.
- (b) A program error (P485) will occur if polar coordinate interpolation is executed during tool radius compensation.
- (c) If the G12.1 and G13.1 commands are issued with no movement command given after the tool radius compensation is canceled, the position of the axis in the G12.1 and G13.1 commands block is interpreted as the position applied after the tool radius compensation is canceled and the following operations are performed.

## (5) Tool length compensation

- (a) A program error (P481) will occur if the tool length compensation is carried out in the polar coordinate interpolation mode.

⋮

```
G43 H12 ;      ...Tool length compensation before polar coordinate interpolation → Valid
G0 X100. Z0 ;
G12.1 ;
```

⋮

```
G43 H11 ;      ...Tool length compensation in polar coordinate interpolation mode →
Program error
```

⋮

```
G13.1 ;
```

- (b) Complete the tool compensation operation (movement of tool length and wear compensation amount) before executing the polar coordinate interpolation. If the tool compensation operation is not completed when the polar coordinate interpolation start command has been issued, the followings will be resulted:
  - Machine coordinate is not changed even if G12.1 is executed..
  - The workpiece coordinate is changed to that of the post tool length compensation when G12.1 is executed. (Even if canceling the polar coordinate interpolation, this workpiece coordinate will not be canceled.)

## (6) Cutting asynchronous feed

- (a) The asynchronous mode is forcibly set when the pole coordinate interpolation mode is started.
- (b) When the pole coordinate interpolation mode is canceled, the synchronization mode will return to the state before the pole coordinate interpolation mode was started.
- (c) A program error (P485) will occur if G12.1 is commanded in the constant surface speed control mode (G96).

## (7) Miscellaneous functions

- (a) The miscellaneous function (M) and 2nd miscellaneous function can be issued even in the pole coordinate interpolation mode.
- (b) The S command in the pole coordinate interpolation mode issues the rotary tool's rotation speed instead of the spindle rotation speed.

- (8) Hole drilling axis in the hole drilling fixed cycle command during the pole coordinate interpolation

Hole drilling axis in the hole drilling fixed cycle command during the pole coordinate interpolation is determined with the linear axis parameter (#1533).

#1533 setting value	Hole drilling axis
X	Z (pole coordinate plane is interpreted as XY plane)
Y	X (pole coordinate plane is interpreted as YZ plane)
Z	Y (pole coordinate plane is interpreted as ZX plane)
Blank (no setting)	Z (pole coordinate plane is interpreted as XY plane)

- (9) Shift amount in the G76 (fine boring) or G87 (back boring) command during the pole coordinate interpolation

Shift amount in the G76 (fine boring) or G87 (back boring) command during the pole coordinate interpolation is determined with the linear axis parameter (#1533).

#1533 setting value	Center designation command
X	I, J (pole coordinate plane is interpreted as XY plane)
Y	J, K (pole coordinate plane is interpreted as YZ plane)
Z	K, I (pole coordinate plane is interpreted as ZX plane)
Blank (no setting)	I, J (pole coordinate plane is interpreted as XY plane)



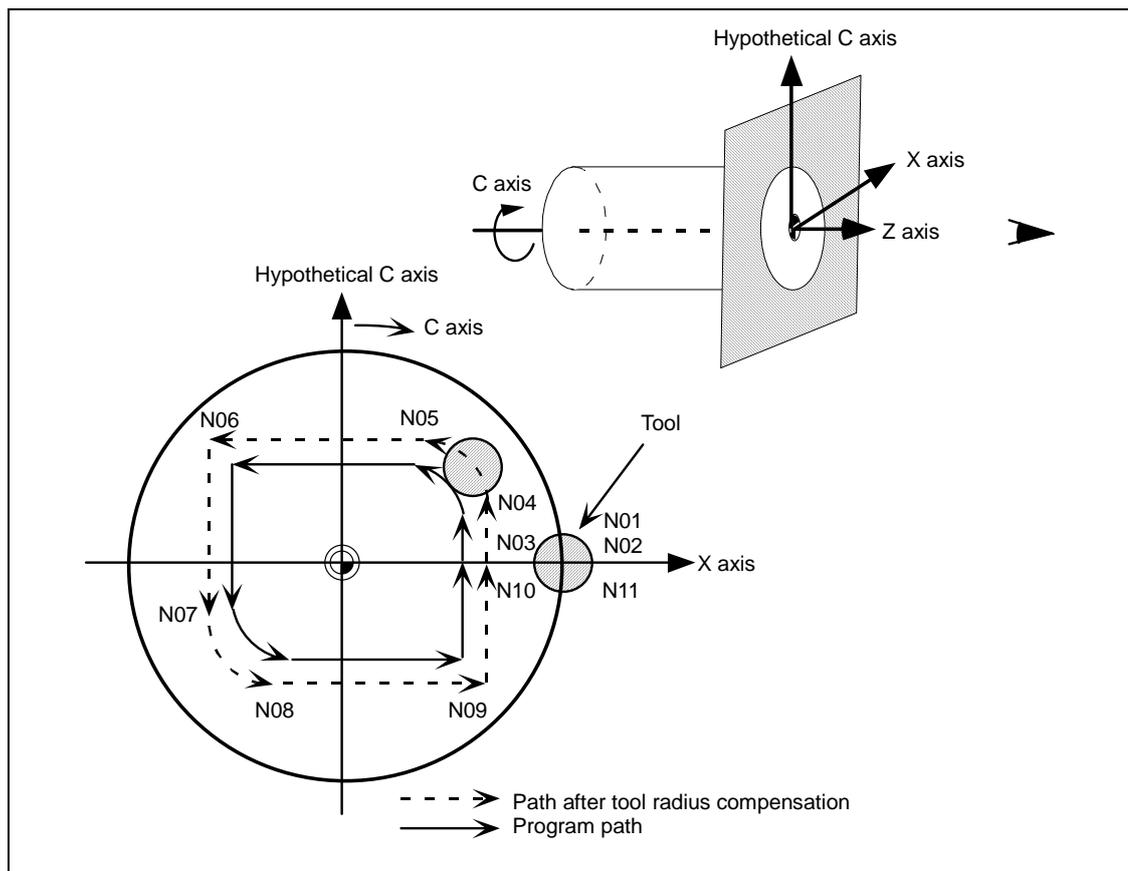
### Restrictions and precautions

- (1) The program cannot be restarted (resumed) for a block in pole coordinate interpolation.
- (2) Before commanding pole coordinate interpolation, set the workpiece coordinate system so that the center of the rotation axis is at the coordinate system zero point. Do not change the coordinate system during the pole coordinate interpolation mode. (G50, G52, G53, relative coordinate reset, G54 to G59, etc.)
- (3) The feedrate during pole coordinate interpolation will be the interpolation speed on the pole coordinate interpolation plane (orthogonal coordinate system).  
(The relative speed with the tool will be converted with pole coordinate conversion.)  
When passing near the center of the rotation axis on the pole coordinate interpolation plane (orthogonal coordinate system), the rotation axis side feedrate after pole coordinate interpolation will be very high.
- (4) The axis movement command outside of the plane during pole coordinate interpolation will move unrelated to the pole coordinate interpolation.
- (5) The current position displays during pole coordinate interpolation will all indicate the actual coordinate value. However, the "remaining movement amount" will be the movement amount on the pole coordinate input plane.
- (6) The pole coordinate interpolation mode will be canceled when the power is turned ON or reset.
- (7) A program error (P484) will occur if any axis commanded for pole coordinate interpolation has not completed zero point return.
- (8) Tool radius compensation must be canceled before the pole coordinate interpolation mode can be canceled.
- (9) When the pole coordinate interpolation mode is canceled, the mode will change to the cutting mode, and the plane will return to that selected before pole coordinate interpolation.
- (10) A program error (P486) will occur if the pole coordinate interpolation command is issued during the mirror image.

- (11) A program error (P486) will occur if the cylindrical interpolation or the pole coordinate interpolation is commanded during the pole coordinate interpolation mode.
- (12) During pole coordinate interpolation, if X axis moveable range is controlled in the plus side, X axis has to be moved to the plus area that includes "0" and above before issuing the polar coordinate interpolation command. If X axis moveable range is controlled in the minus side, X axis has to be moved to the area that does not include "0" before issuing the polar coordinate interpolation command.



## Example of program



## &lt;Program&gt;

```

:
N01 G17 G90 G0 X40.0 C0 Z0;   Setting of start position
N02 G12.1;                  Polar coordinate interpolation mode: Start
N03 G1 G42 X20.0 F2000;     Actual machining start
N04 C10.0;
N05 G3 X10.0 C20.0 R10.0;
N06 G1 X-20.0;              Shape program
N07 C-10.0;
N08 G3 X-10.0 C-20.0 I10.0 J0; (Command the position with the orthogonal
N09 G1 X20.0;               coordinate on X-C hypothetical axis plane.)
N10 C0;
N11 G40 X40.0;
N12 G13.1;                  Polar coordinate interpolation mode: Cancel
:
:
M30 ;

```

6.11 Exponential Function Interpolation; G02.3, G03.3



Function and purpose

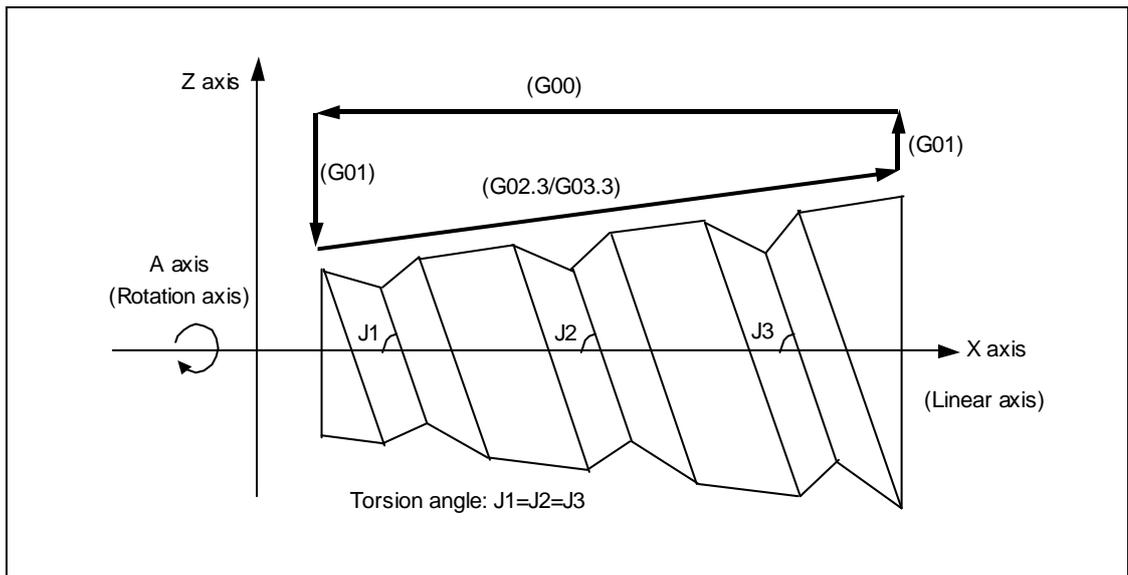
Exponential function interpolation changes the rotation axis into an exponential function shape in respect to the linear axis movement.

At this time, the other axes carry out linear interpolation between the linear axis.

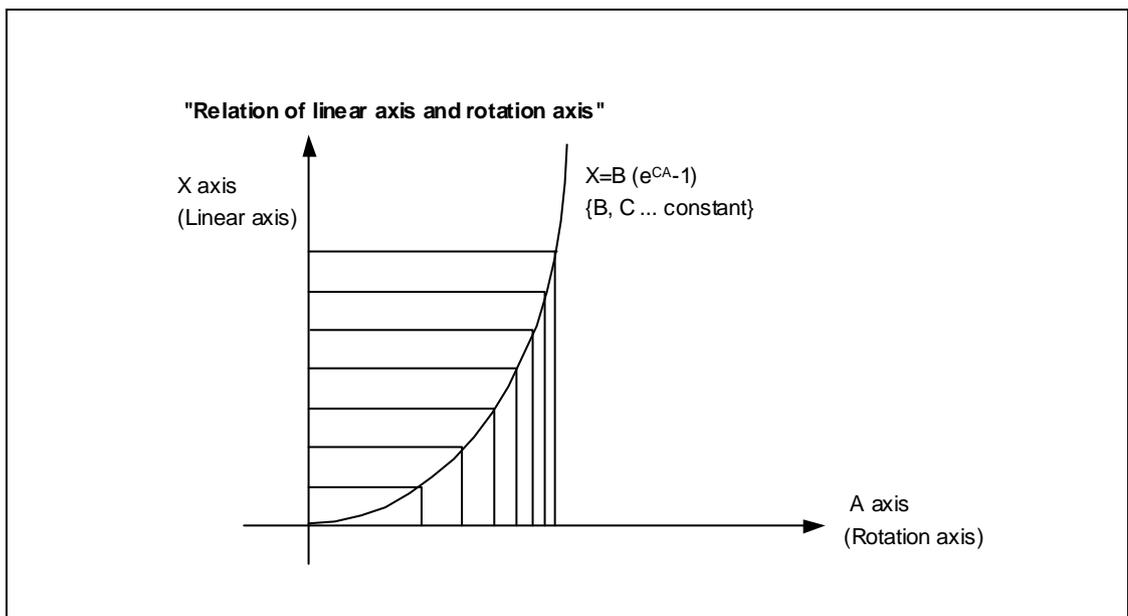
This allows a machining of a taper groove with constant torsion angle (helix angle) (uniform helix machining of taper shape).

This function can be used for slotting or grinding a tool for use in an end mill, etc.

• Uniform helix machining of taper shape



• Relation of linear axis and rotation axis





## Command format

**G02.3/G03.3 Xx1 Yy1 Zz1 Ii1 Jj1 Rr1 Ff1 Qq1 Kk1 ;**  
 G02.3 : Forward rotation interpolation (modal)  
 G03.3 : Negative rotation interpolation (modal)  
 X : X axis end point (**Note 1**)  
 Y : Y axis end point (**Note 1**)  
 Z : Z axis end point (**Note 1**)  
 I : Angle i1 (**Note 2**)  
 J : Angle j1 (**Note 2**)  
 R : Constant value r1 (**Note 3**)  
 F : Initial feedrate (**Note 4**)  
 Q : Feedrate at end point (**Note 5**)  
 K : Command will be ignored.

**(Note 1)** Designate the end point of the linear axis designated with parameter "#1514 expLinax" and the axis that carries out linear interpolation between that axis.

If the end point on of the rotation axis designated with parameter "#1515 expRotax" is designated, linear interpolation without exponential function interpolation will take place.

**(Note 2)** The command unit is as follows.

Setting unit	#1003 = B	#1003 = C	#1003 = D	#1003 = E
(Unit = °)	0.001	0.0001	0.00001	0.000001

The command range is -89 to +89°.

A program error (P33) will occur if there is no address I or J command.

A program error (P35) will occur if the address I or J command value is 0.

**(Note 3)** The command unit is as follows.

Setting unit	#1003 = B	#1003 = C	#1003 = D	#1003 = E	Unit
Metric system	0.001	0.0001	0.00001	0.000001	mm
Inch system	0.0001	0.00001	0.000001	0.0000001	inch

The command range is a positive value that does not include 0.

A program error (P33) will occur if there is no address R command.

A program error (P35) will occur if the address R command value is 0.

**(Note 4)** The command unit and command range is the same as the normal F code. (Command as a per minute feed.)

Command the composite feedrate that includes the rotation axis.

The normal F modal value will not change by the address F command.

A program error (P33) will occur if there is no address F command.

A program error (P35) will occur if the address F command value is 0.

(Note 5) The command unit is as follows.

Setting unit	#1003 = B	#1003 = C	#1003 = D	#1003 = E	Unit
Metric system	0.001	0.0001	0.00001	0.000001	mm
Inch system	0.0001	0.00001	0.000001	0.0000001	inch

The command unit and command range is the same as the normal F code.

Command the composite feedrate that includes the rotation axis.

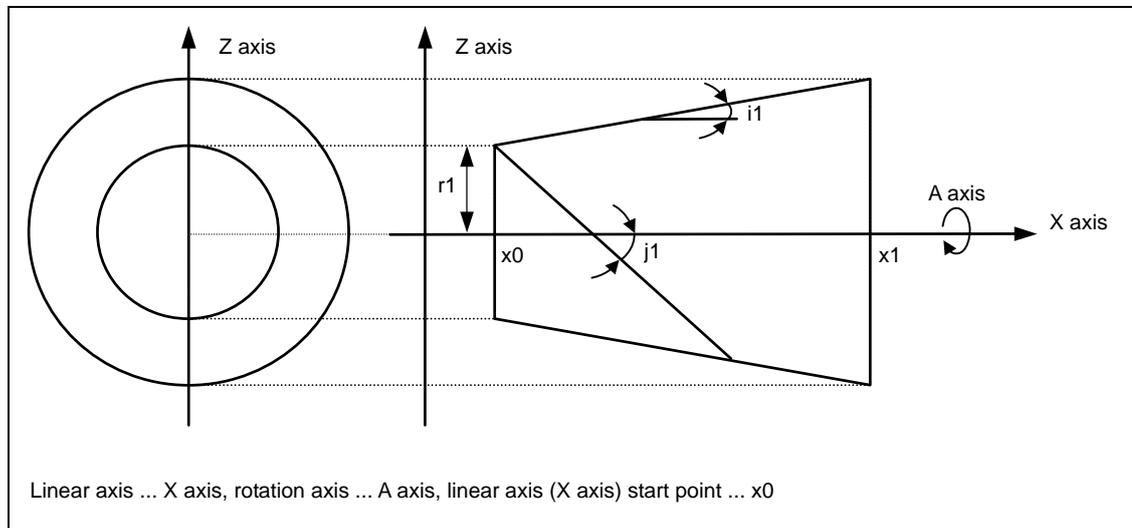
The normal F modal value will not change by the address Q command.

The axis will interpolate between the initial speed (F) and end speed (Q) in the CNC according to the linear axis.

If there is no address Q command, interpolation will take place with the same value as the initial feedrate (address F command). (The start point and end point feedrates will be the same.)

A program error (P35) will occur if the address Q command value is 0.

• Example of uniform helix machining of taper shape



Relational expression of exponential function

The exponential function relational expression of the linear axis (X) and rotation axis (A) in the G02.3/G03.3 command is defined in the following manner.

$$X(\theta) = r1 * (e^{\theta/D} - 1) / \tan(i1) \quad (\text{linear axis (X) movement (1)})$$

$$A(\theta) = (-1)^\omega * 360 * \theta / (2\pi) \quad (\text{rotation axis (A) movement})$$

$$D = \tan(j1) / \tan(i1)$$

$\omega = 0$  during forward rotation (G02.3), and  $\omega = 1$  during reverse rotation (G03.3)

$\theta$  is the rotation angle (radian) from the rotation axis' start point

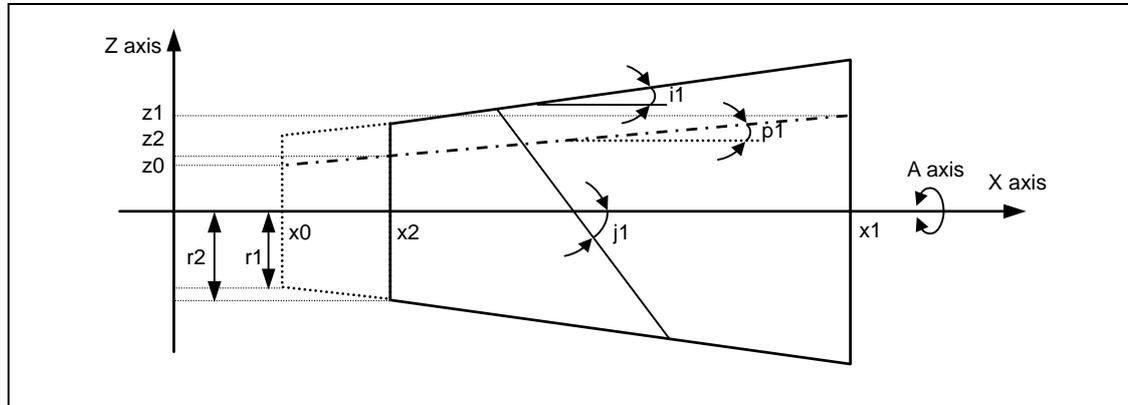
The rotation axis' rotation angle ( $\theta$ ) is as follows according to expression (1).

$$\theta = D * \ln \{ (X * \tan(i1) / r1) + 1 \}$$



## Machining example

## • Example of uniform helix machining of taper shape



## &lt;Relational expression of exponential function in machining example&gt;

$$Z(\theta) = r1 * (e^{\theta/D} - 1) * \tan(p1) / \tan(i1) + z0 \quad \dots (1)$$

$$X(\theta) = r1 * (e^{\theta/D} - 1) / \tan(i1) \quad \dots (2)$$

$$A(\theta) = (-1)^\omega * 360 * \theta / (2\pi)$$

$$D = \tan(j1) / \tan(i1)$$

Z(θ) Absolute value from zero point of Z axis (axis that linearly interpolates between interval with linear axis (X axis))

X(θ) Absolute value from X axis (linear axis) start point

A(θ) Absolute value from A axis (rotation axis) start point

r1 Exponential function interpolation constant value (address R command)

r2 Workpiece left edge radius

x2 X axis (linear axis) position at workpiece left edge

x1 X axis (linear axis) end point (address X command)

x0 X axis (linear axis) start point (Set as "x0 ≤ x1" so that workpiece does not interfere with tool)

z1 End point of Z axis (axis that linearly interpolates between interval with linear axis (X axis)) (address Z command)

z0 Start point of Z axis (axis that linearly interpolates between interval with linear axis (X axis))

i1 Taper gradient angle (address I command)

p1 Slot base gradient angle

j1 Torsion angle (helix angle) (address J command)

ω Torsion direction (0: Forward rotation, 1: reverse direction)

θ Workpiece rotation angle (radian)

f1 Initial feedrate (address F command)

q1 Feedrate at end point (address Q command)

k1 Insignificant data (address K command)

According to expressions (1) and (2):

$$Z(\theta) = X(\theta) * \tan(p1) + z0 \quad \dots (3)$$

According to expression (3), the slot base gradient angle (p1) is determined from the X axis and Z axis end point positions (x1, z1).

The Z axis movement amount is determined by the slot base gradient angle (p1) and X axis position.

In the above diagram, the exponential function interpolation's constant value (r1) is determined with the following expression using the workpiece left edge radius (r2), X axis start point (x0), X axis position at workpiece left edge (x2) and taper gradient angle (i1).

$$r1 = r2 - \{ (x2 - x0) * \tan(i1) \} \quad \dots (4)$$

## 6. Interpolation Functions

### 6.11 Exponential Function Interpolation; G02.3, G03.3

The taper gradient angle ( $i1$ ) and torsion angle ( $j1$ ) are each issued with the command address I and J.

Note that if the shape is a reverse taper shape, the taper gradient angle ( $i1$ ) is issued as a negative value.

The torsion direction ( $\omega$ ) is changed with the G code. (Forward rotation when G02.3 is commanded, negative rotation when G03.3 is commanded)

The above settings allow uniform helix machining of a taper shape (or reverse taper shape).



#### Command and operation

G2.3(Equivalent to G3.3 if $j1 < 0$ )					
		X movement direction > 0		X movement direction < 0	
		$i1 > 0$	$i1 < 0$	$i1 > 0$	$i1 < 0$
Command					
Operation					
Machining program example	N10 G28XYZC; N20 G91G0 X100. Z100.; N30 G2.3 X100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X100. Z200.; N30 G2.3 X100. Z-100. I-50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z100.; N30 G2.3 X-100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z200.; N30 G2.3 X-100. Z-100. I-50. J80. R105. F500.; N40 M30;	
G3.3(Equivalent to G2.3 if $j1 < 0$ )					
		X movement direction > 0		X movement direction < 0	
		$i1 > 0$	$i1 < 0$	$i1 > 0$	$i1 < 0$
Command					
Operation					
Machining program example	N10 G28XYZC; N20 G91G0 X100. Z100.; N30 G3.3 X100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X100. Z200.; N30 G3.3 X100. Z-100. I-50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z100.; N30 G3.3 X-100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z200.; N30 G3.3 X-100. Z-100. I-50. J80. R105. F500.; N40 M30;	

**Precautions for programming**

- (1) When G02.3/G03.3 is commanded, interpolation takes place with the exponential function relational expression using the start position of the linear axis and rotation axis as 0.
- (2) Linear interpolation will take place in the following cases, even if in the G02.3/G03.3 mode. The feedrate for linear interpolation will be the F command in that block. (Note that the normal F modal is not updated.)
  - The linear axis designated with the parameter (#1514 expLinax) is not commanded, or the movement amount for that axis is 0.
  - The rotation axis designated with the parameter (#1515 expRotax) is commanded.
- (3) A program error will occur if the following commands are issued in G02.3 or G03.3 mode. A program error will also occur if G02.3 or G03.3 command is issued in the following modes.:
  - Tool length compensation  
(A program error will occur only when the compensation starts at the same time as the movement by exponential function interpolation. The tool length compensation will operate normally if it has started before the G02.3/G03.3 mode starts.)
  - Tool radius compensation
  - High-speed high-accuracy control
  - High-speed machining
  - Scaling
  - Tool length compensation along tool axis
  - Figure rotation
  - Coordinate rotation by program
  - Coordinate rotation by parameter
  - 3-dimensional coordinate conversion
- (4) A program error (P481) will occur if commands are issued during the pole coordinate interpolation, cylindrical interpolation or milling interpolation modes.
- (5) Program error (P612) will occur if commands are issued during the scaling or mirror image.
- (6) Program error (P34) will occur if commands are issued during the high-speed high-accuracy control II.
- (7) G02.3/G03.3 will function with asynchronous feed even during the synchronous feed mode.
- (8) If the parameter "#1515 expRota" setting is the same axis name as the initial C axis, the axis selected with the C axis selection signal will interpolate as the rotation axis.

## 6.12 Polar Coordinate Command; G16/G15



## Function and purpose

With this function, the end point coordinate value is commanded with the polar coordinate of the radius and angle.



## Command format

**G16 ; Polar coordinate command mode ON**

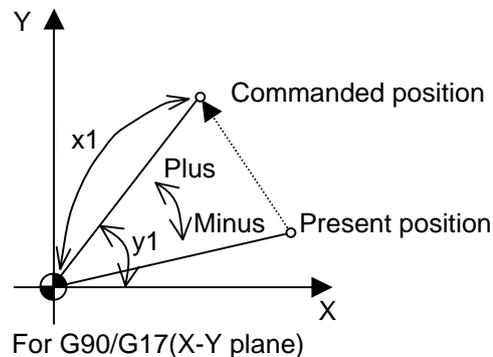
**G15 ; Polar coordinate command mode OFF**



## Detailed description

- (1) The polar coordinate command is applied in the interval from turning ON to OFF of the polar coordinate command mode.

G1x ;	Plane selection for polar coordinate command (G17/G18/G19)
G16 ;	Polar coordinate command mode ON
G9x G01 Xx1 Yy1 F2000 ;	Polar coordinate command
:	G9x : Center selection for polar coordinate command (G90/G91)
	G90 ...The workpiece coordinate system zero point is the polar coordinate center.
	G91 ...The present position is the polar coordinate center.
	x1 : 1st axis for the plane ...The radius commanded
	y1 : 2nd axis for the plane ...The angle commanded



G15 ; Polar coordinate command mode OFF

- (2) The plane selection during the polar coordinate command mode is carried out with G17, G18 and G19.
- (3) The polar coordinate command is a modal. The polar coordinate command mode when the power is turned ON is OFF (G15). Whether to initialize the modal at reset or not can be selected with the parameter (#1210 RstGmd/bit 11) setting.
- (4) During polar coordinate command mode, command the radius with the 1st axis for the selected plane, and the angle with the 2nd axis. For example, when the X-Y plane is selected, command the radius with the address "X", and the angle with the address "Y".
- (5) For the angle, the counterclockwise direction of the selected plane is positive and the clockwise direction is negative.
- (6) The radius and angle can be commanded with both the absolute value and incremental value (G90, G91).

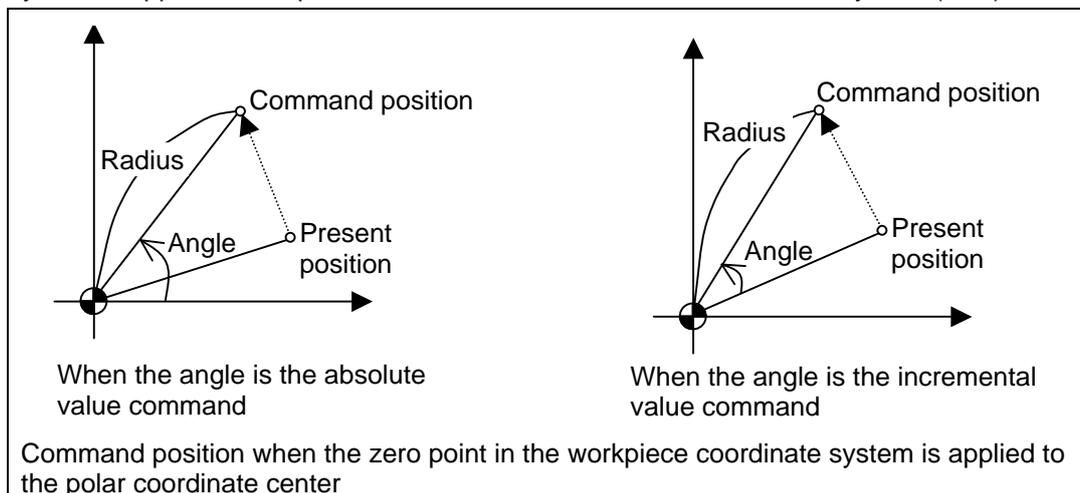
- (7) When the radius is commanded with the absolute value, command the distance from the zero point in the workpiece coordinate system (note that the local coordinate system is applied when the local coordinate system is set).
- (8) When the radius is commanded with the incremental value command, considering the end point of the previous block as the polar coordinate center, command the incremental value from that end point. The angle is commanded with the incremental value of the angle from the previous block.
- (9) When the radius is commanded with the negative value, the same operation as the command that the radius command value is changed to the absolute value and  $180^\circ$  is added to the angle command value.



### Command position

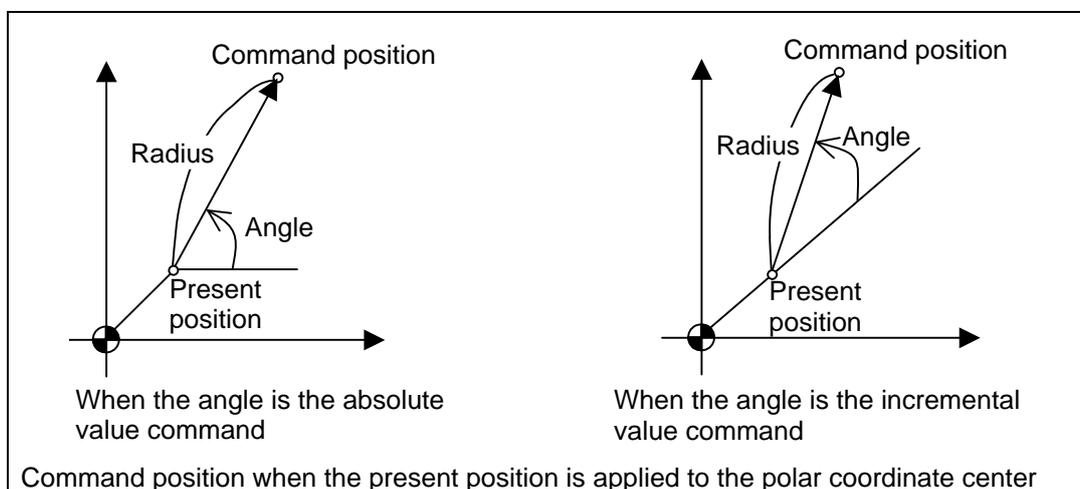
#### (1) When the zero point in the workpiece coordinate system is applied to the polar coordinate center

The zero point in the workpiece coordinate system is applied to the polar coordinate center by commanding the radius with the absolute value. Note that the zero point in the local coordinate system is applied to the polar coordinate center if the local coordinate system (G52) is used.



#### (2) When the present position is applied to the polar coordinate center

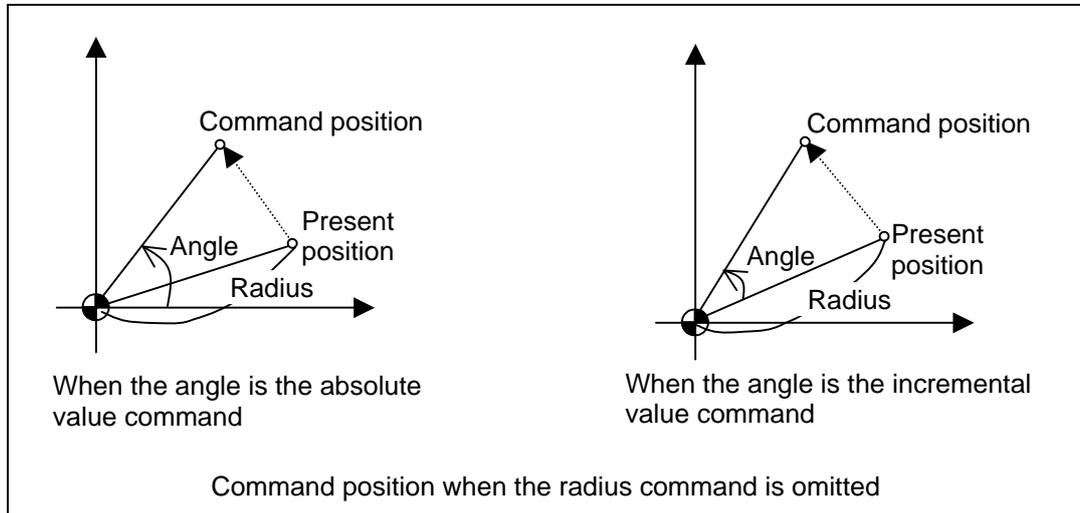
The present position is applied to the polar coordinate center by commanding the radius with the incremental value.



**(3) When the radius command is omitted**

When the radius command is omitted, the zero point in the workpiece coordinate system is applied to the polar coordinate center, and the distance between the polar coordinate center and current position is regarded as the radius.

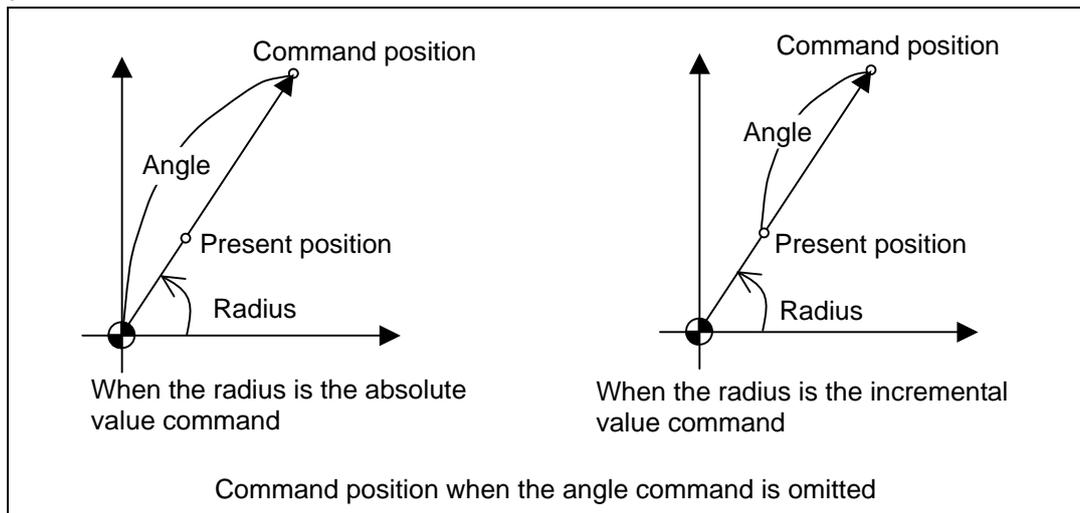
Note that the zero point in the local coordinate system is applied to the polar coordinate center if the local coordinate system (G52) is used.

**(4) When the angle command is omitted**

When the angle command is omitted, the angle of the present position in the workpiece coordinate system is applied to the angle command.

The zero point in the workpiece coordinate system is applied to the polar coordinate center by commanding the radius with the absolute value. Note that the zero point in the local coordinate system is applied to the polar coordinate center if the local coordinate system (G52) is used.

If the radius is commanded with the incremental value, the present position is applied to the polar coordinate center.




**Axis command not interpreted as polar coordinate command**

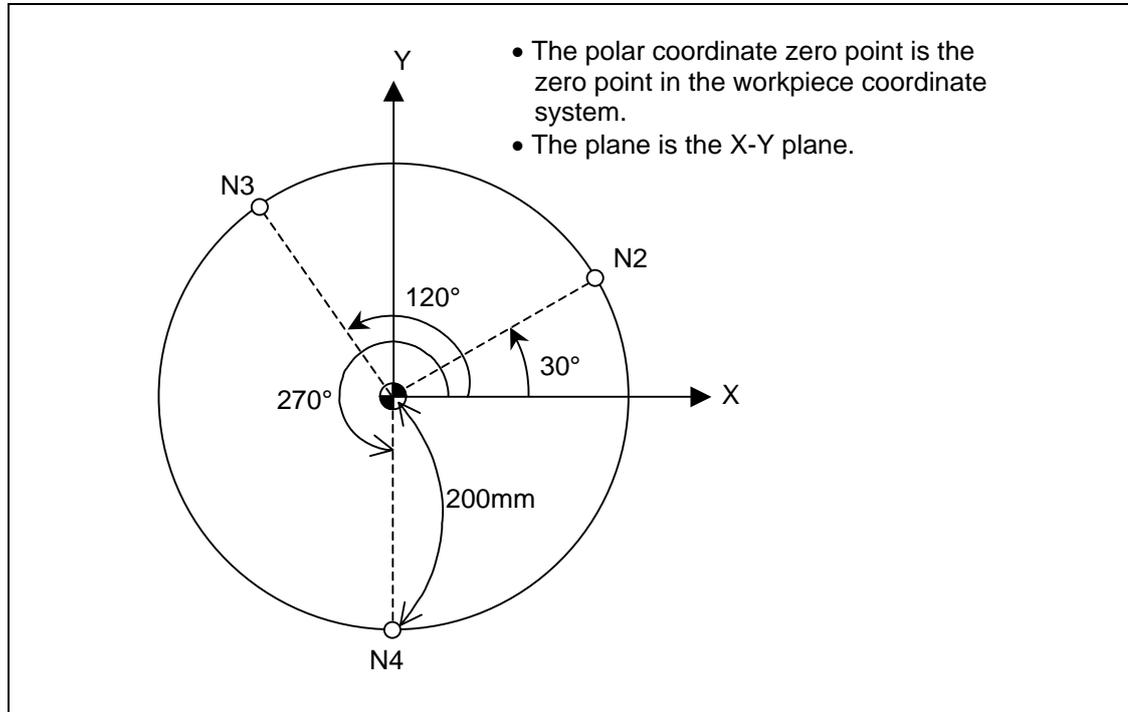
The axis command with the following command is not interpreted as the polar coordinate command during the polar coordinate command mode. The movement command that has no axes commands for the 1st axis and 2nd axis in the selected plane mode is also not interpreted as polar coordinate command during the polar coordinate command mode.

Function	G code
Dwell	G04
Program parameter input/compensation input	G10
Local coordinate system setting	G52
Machine coordinate system setting	G92
Machine coordinate system selection	G53
Coordinate rotation by program	G68.1
Scaling	G51
G command mirror image	G51.1
Reference position check	G27
Reference position return	G28
Start position return	G29
2nd to 4th reference position return	G30
Tool change position return 1	G30.1
Tool change position return 2	G30.2
Tool change position return 3	G30.3
Tool change position return 4	G30.4
Tool change position return 5	G30.5
Tool change position return 6	G30.6
Automatic tool length measurement	G37
Skip	G31
Multi-step skip function 1-1	G31.1
Multi-step skip function 1-2	G31.2
Multi-step skip function 1-3	G31.2
Linear angle command	G01 Aa1



## Example of program

When the zero point in the workpiece coordinate system is the polar coordinate zero point



## (1) When the radius and angle are the absolute value command

N1 G17 G90 G16 ;	Polar coordinate command, X-Y plane selection The polar coordinate zero point is the zero point in the workpiece coordinate system.
N2 G85 X200. Y30. Z-20. F200. ;	Radius 200mm, angle 30°
N3 Y120. ;	Radius 200mm, angle 120°
N4 Y270. ;	Radius 200mm, angle 270°
N5 G15 G80 ;	Polar coordinate command cancel

## (2) When the radius is the absolute value command and the angle is the incremental value command

N1 G17 G90 G16 ;	Polar coordinate command, X-Y plane selection The polar coordinate zero point is the zero point in the workpiece coordinate system.
N2 G85 X200. Y30. Z-20. F200. ;	Radius 200mm, angle 30°
N3 G91 Y90. ;	Radius 200mm, angle +90°
N4 Y150. ;	Radius 200mm, angle +150°
N5 G15 G80 ;	Polar coordinate command cancel

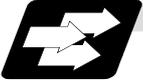
**Precautions**

- (1) If the following commands are carried out during the polar coordinate command mode, or if the polar coordinate command is carried out during the following command mode, a program error (P34) will occur.

Function	G code
High-speed high-accuracy control I	G05.1 Q1
High-speed high-accuracy control II	G05 P10000
Spline	G05.1 Q2

- (2) When the mirror image (G code/parameter/external signal) is canceled anywhere except at the mirror image center during the polar coordinate command mode, the absolute value and machine position will deviate. The mirror center is set with an absolute value and so if another mirror center is assigned in this state, the center may be set at an unforeseen position. Cancel the mirror image above the mirror center or, after cancellation, assign a positioning command using absolute value command that the radius and angle of the polar coordinate command are designated.

## 6.13 Spiral/Conical Interpolation; G02.0/G03.1(Type1), G02/G03(Type2)



## Function and purpose

This function carries out interpolation that smoothly joins the start and end points in a spiral. This interpolation is carried out for arc commands in which the start point and end point are not on the same circumference.

There are two types of command formats, and they can be switched with the parameters.



## Command format

<b>G17 G02.1/G03.1 X__ Y__ I__ J__ P__ F__ ; (Type 1: #1272 ex08/bit2=0)</b>	
<b>G17 G02/G03 X__ Y__ I__ J__ Q_/L_/K__ F__ ; (Type 2: #1272 ex08/bit2=1)</b>	
G17	: Rotation plane
G02.1/G03.1 (Type 1)	: Arc rotation direction (Type 1)
G02/G03 (Type 2)	: Arc rotation direction (Type 2)
X Y	: End point coordinates (Conical Interpolation when the axis other than rotation plane axes is included.)
I J	: Arc center
P	: Number of pitches (number of rotations) (Type 1)
Q/L/K (Type 2)	: Incremental/decremental amount of radius /Number of pitches(Number of spirals)/ Increment/decrement amount of height (Type 2)
F	: Feedrate (tool path direction speed)

Circular interpolation operations are carried out at the f1 speed by the commands above. The path is toward the end point, following a spiral arc path centered at the position designated by distance i (X axis direction) and distance j (Y axis direction) in respect to the start point.

- (1) The arc plane is designated by G17, G18 and G19. (Common for type 1 and 2)
  - G17 . . . . XY plane
  - G18 . . . . ZX plane
  - G19 . . . . YZ plane
- (2) The arc rotation direction is designated by G02.1(G02) or G03.1(G03). (Common for type 1 and 2)
  - G02.1(G02) . . . Clockwise (CW)
  - G03.1(G03) . . . Counterclockwise (CCW)
- (3) The end point coordinates are designated with XYZ. (Common for type 1 and 2) (Decimal point command is possible. Use mm (or inch) as the unit).  
When designation of arc plane axes is omitted, the coordinates of the start point are inherited. If the axis other than arc plane axes is designated, conical interpolation is applied.
- (4) The arc center is designated with IJK. (Common for type 1 and 2) (Decimal point command is possible. Use mm (or inch) as the unit.)
  - I : Incremental designation in the X axis direction from the start point
  - J : Incremental designation in the Y axis direction from the start point
  - K : Incremental designation in the Z axis direction from the start point
 When either 1 axis of arc plane is omitted, the coordinates of the start point are inherited.

- (5) P designates the number of pitches (number of spirals). (Type 1)  
The number of pitches and rotations is as shown below.

Number of pitches (0 to 99)	Number of rotations
P0	Less than 1 rotation (Can be omitted.)
P1	1 or more rotation, less than 2 rotations
Pn	n or more rotation, less than (n+1) rotations

- (6) Q designates the increment/decrement amount of radius per spiral rotation. (Type 2)  
The number of spiral rotations when the radius increment/decrement amount is specified can be calculated with the following expression.  
Number of rotations =  $\left| \frac{\text{arc end point radius} - \text{arc start point radius}}{\text{increment/decrement amount of radius}} \right|$

- (7) L designates the number of pitches (number of spirals). (Type 2) (range: 0 to 99)  
When omitted, L1 is designated.  
The number of pitches and rotations is as shown below.

Number of pitches (0 to 99)	Number of rotations
L1	Less than 1 rotation
L2	1 or more rotation, less than 2 rotations
Ln	(n-1) or more rotations, less than n rotations

Q takes precedence over L if both Q and L have been designated at the same time.

- (8) K designates the increment/decrement amount of height per spiral rotation in conical interpolation. (Type 2)  
The increment/decrement amount of height is designated with I/J/K for the axis other than arc plane.  
The relation between increment/decrement amount of height and the rotation plane is as shown below.

Rotation plane	Increment/decrement amount of height
G18	J command
G19	I command
Other than G18/G19	K command

The number of rotations when the designation of increment/decrement amount of height is specified can be calculated with the following expression.

$$\text{Number of rotations} = \frac{\text{Height}}{\text{Increment/decrement amount of height}}$$

If Q, K and L have been designated at the same time, the order of precedence is Q>K>L.

Decimal point command is possible in the range of the increment/decrement amount of radius and height. Use mm (or inch) as the unit.

(9) In the following cases, a program error will occur.

(a) Items common for type 1 and 2

Settings	Command range (Unit)	Error
End point coordinate	Range of coordinate command (mm/inch) (Decimal point command is possible.)	<ul style="list-style-type: none"> <li>• If a value exceeding the command range is issued, program error (P35) will occur.</li> <li>• If an axis other than one which can be controlled with the command system is commanded, a program error (P33) will occur.</li> </ul>
Arc center	Range of coordinate command (mm/inch) (Decimal point command is possible.)	<ul style="list-style-type: none"> <li>• If a value exceeding the command range is issued, a program error (P35) will occur.</li> <li>• If an axis other than one which can be controlled with the command system is commanded, a program error (P33) will occur.</li> <li>• If rotation plane axis is not designated completely, a program error (P33) will occur.</li> </ul>
Number of pitches	0 to 99	<ul style="list-style-type: none"> <li>• If a value exceeding the command range is issued, a program error (P35) will occur.</li> </ul>
Feedrate	Range of speed command (mm/min,inch/min) (Decimal point command is possible.)	<ul style="list-style-type: none"> <li>• If a value exceeding the command range is issued, a program error (P35) will occur.</li> </ul>

(b) Items for type 2 only

Settings	Command range (Unit)	Error
Increment/decrement amount of radius	Range of coordinate command (mm/inch) (Decimal point command is possible.)	<ul style="list-style-type: none"> <li>• If the sign of designated increment/decrement amount is opposite from that of the difference between the start point radius and the end point radius, a program error (P33) will occur.</li> <li>• If the end point position obtained from the speed and increment/decrement amount is larger than "SpiralEndErr (#8075)", a program error (P70) will occur.</li> </ul>
Increment/decrement amount of height	Range of coordinate command (mm/inch) (Decimal point command is possible.)	<ul style="list-style-type: none"> <li>• If the sign of designated increment/decrement amount is opposite from that of the movement direction of height, a program error (P33) will occur.</li> <li>• If the end point position obtained from the speed and increment/decrement amount is larger than "SpiralEndErr (#8075)", a program error (P70) will occur.</li> </ul>
G02.1/G03.1		<ul style="list-style-type: none"> <li>• Program error (P34) will occur if G02.1/G03.1 are used during type 2.</li> </ul>

**Detailed description**

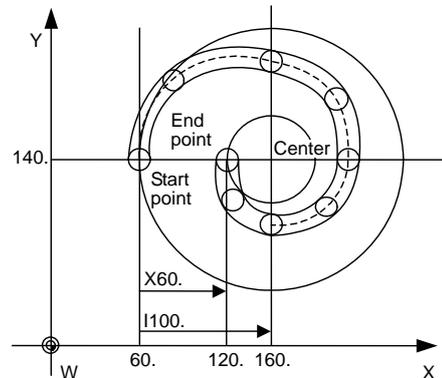
- (1) The arc rotation direction G02.1 is the same as G02, and G03.1 is the same as G03.
- (2) There are no R-designated arcs in spiral interpolation.
- (3) Conical cutting, tapered thread-cutting and other such machining operations can be conducted by changing the start point and end point radius and commanding the linear axis simultaneously.
- (4) Normally the spiral interpolation is automatically enabled with the arc commands (G02, G03) when the difference between the start point radius and the end point radius is less than the parameter setting value.
- (5) The axis combination that can be simultaneously commanded depends on the specifications. The combination within that range is random.
- (6) The feedrate is the constant tangential speed.
- (7) Simultaneous control by combining with tool radius compensation (G41, G42) is not possible.
- (8) The arc plane always follows G17, G18 and G19. The plane arc control is carried out by G17, G18 and G19, even if designated by two addresses that do not match the plane.
- (9) Conical interpolation  
When an axis designation other than the spiral interpolation plane is simultaneously designated, other axes are also interpolated in synchronization with the spiral interpolation.
- (10) With M2/M0 format, the command such as "G69 G02" or "G69 G03" can be also used for the spiral interpolation.  
For details, refer to the section "13.25 End point error check cancel; G69".



## Example of program

## (Example 1)

```
G91 G17 G01 X60. F500 ;
Y140. ;
G02.1 X60. Y0 I100. P1 F300 ;
G01 X-120 ;
G90
G17 G01 X60. F500 ;
Y140. ;
G02.1 X120. Y140. I100. P1 F300 ;
G01 X0 ;
```

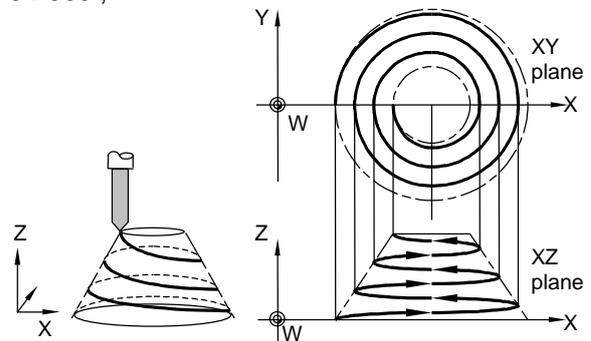


## (Example 2)

```
G91 G17 G01 X60. F500 ;
Y140. ;
G02.1 X60.0 Z100.0 I100. P1 F300 ; → Because this is the G17 plane, arc control
G01 X-120 ;                               is not carried out by X-Z.
                                             Arc control is carried out by X-Y.
```

(Example 3) In this example, the interpolation is truncated cone interpolation.

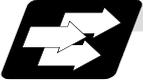
```
G17 G91 G02.1 X100. Z150. I150. P3 F500 ;
```



## Relation with other functions

- (1) Items common for type 1 and 2
  - The start point and end point are not on the same arc, so normal line control is not applied correctly.
  - If there is no center command when geometric is valid, a program error (P33) will occur.
- (2) Items for type 2 only
  - If the spiral interpolation command is issued during the mirror image, a program error (P34) will occur.
  - If the spiral interpolation command is issued during the scaling, a program error (P34) will occur.
  - If the spiral interpolation command is issued during the corner chamfering/corner rounding command, a program error (P33) will occur.

## 6.14 3-dimensional Circular Interpolation; G02.4, G03.4

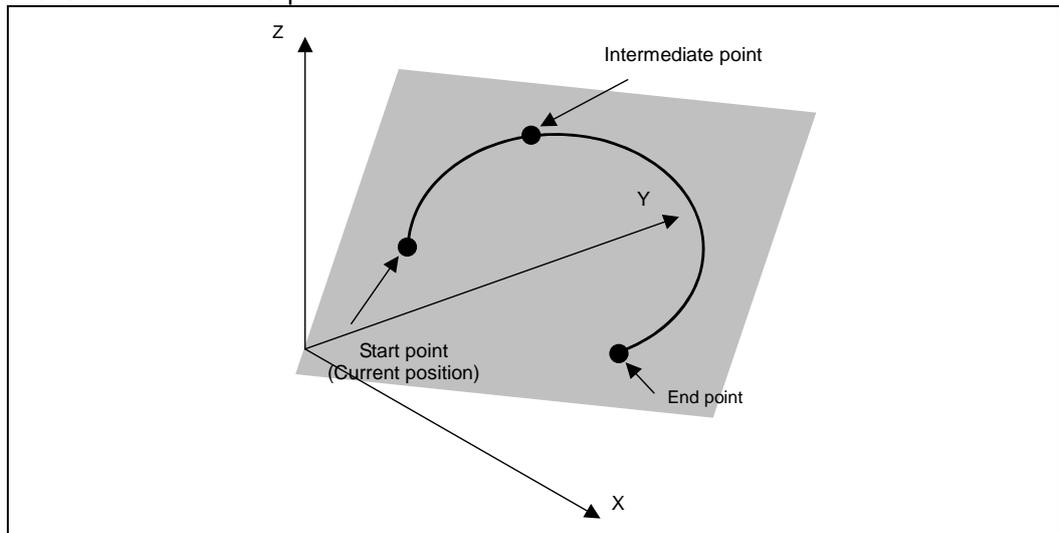


## Function and purpose

To issue a circular command over a 3-dimensional space, an arbitrary point (intermediate point) must be designated on the arc in addition to the start point (current position) and end point. By using the 3-dimensional circular interpolation command, an arc shape determined by the three points (start point, intermediate point, end point) designated on the 3-dimensional space can be machined.

An option is required to validate this function. If the option is not provided and the 3-dimensional circular interpolation command is issued, a program error (P39) will occur.

- 3-dimensional circular interpolation command





## Command format

<b>G02.4(G03.4) Xx1 Yy1 Zz1 <math>\alpha\alpha1</math></b>	<b>; Intermediate point designation (1st block)</b>
<b>Xx2 Yy2 Zz2 <math>\alpha\alpha2</math></b>	<b>; End point designation (2nd block)</b>
G02.4(G03.4)	: 3-dimensional circular interpolation command (Cannot designate the rotation direction)
x1, y1, z1	: Intermediate point coordinates
x2, y2, z2	: End point coordinates
$\alpha$	: Arbitrary axis other than axis used as the reference in 3-dimensional circular interpolation (May be omitted)

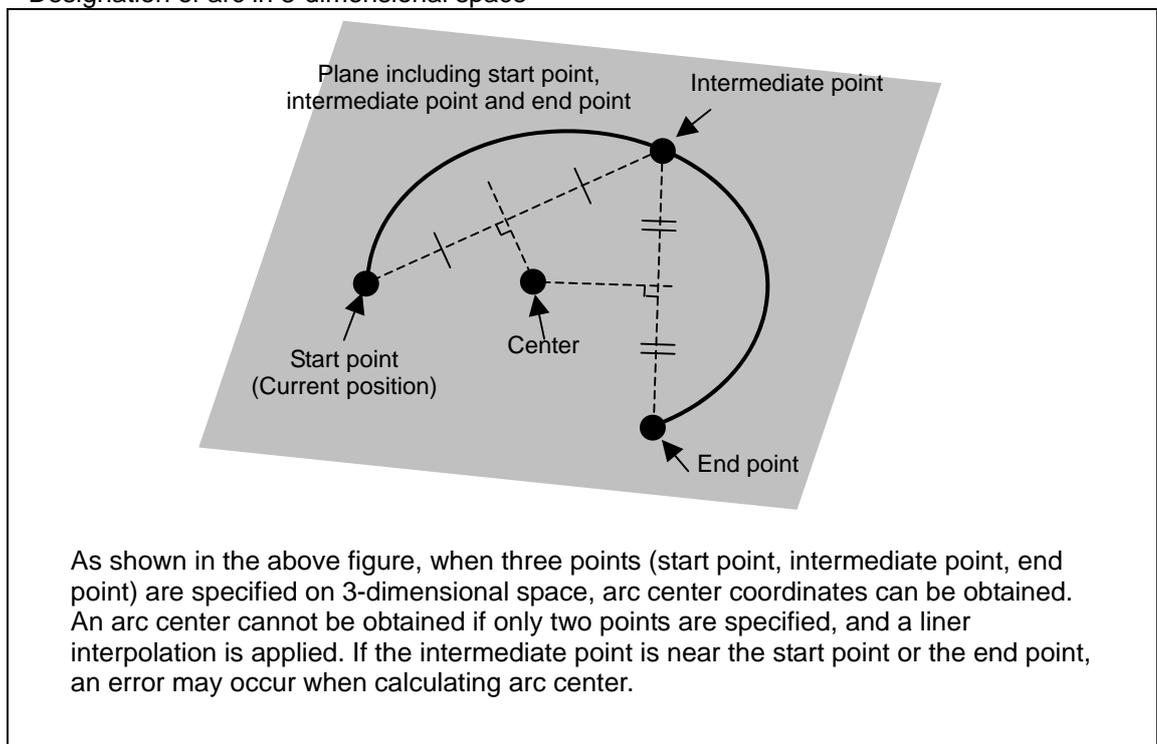
- The G02.4 and G03.4. operations are the same. (The rotation direction cannot be designated.)
- The axes used as the reference in 3-dimensional circular interpolation are the three basic axes set with the parameters.
- The X, Y, Z address in the block may be omitted. The intermediate point coordinates omitted in the 1st block become the start point coordinates, and the end point coordinates omitted in the 2nd block become the intermediate point coordinates.
- When using the 3-dimensional circular interpolation command, an arbitrary axis can be commanded in addition to the orthogonal coordinate system (X, Y, Z) used as the reference. The arbitrary axis designated in the intermediate point designating block (1st block) will interpolate to the command point when moving from the start point to intermediate point. The arbitrary axis designated in the end point command block (2nd block) will interpolate to the command point when moving from the intermediate point to the end point. The number of arbitrary axes that can be commanded differs according to the number of simultaneous contour control axes. The total of the basic three axes used as the reference of the 3-dimensional circular interpolation and the arbitrary axes commanded simultaneously must be less than the number of simultaneous contour control axes.



### Designating intermediate point and end point

When using the 3-dimensional circular interpolation command, an arc that exists over the 3-dimensional space can be determined by designating the intermediate point and end point in addition to the start point (current position). (Refer to following figure) So according to the command format, it is necessary to designate an intermediate point in the 1st block and an end point in the 2nd block. If only one block is designated, a program error (P74) will occur. Linear interpolation is applied when the end point match the start point in the 3-dimensional circular interpolation command. (Refer to "When linear interpolation is applied") Thus, a true circle (360-degree rotation) cannot be designated in the 3-dimensional circular interpolation. In addition, designate that an intermediate point is located in the middle of a start point and an end point, arc accuracy may fall.

#### • Designation of arc in 3-dimensional space



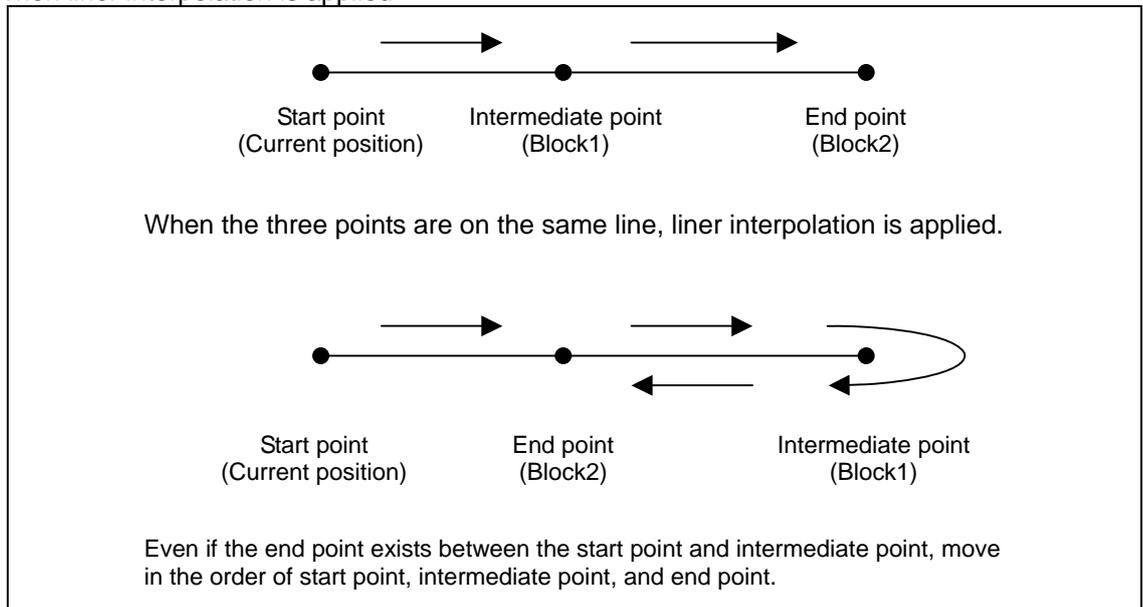


### When liner interpolation is applied

In the following case, liner interpolation but 3-dimensional circular interpolation is applied.

- (1) When the start point, intermediate point, and end point are on the same line (refer to the following figure)  
(If the end point exists between the start point and intermediate point, axes move in the order of start point, intermediate point, and end point.)
- (2) When two points match in start point, a intermediate point, end point  
(Liner interpolation is applied even if the end point matches the start point to command true circle. When the start point matches the end point, axes move in order of the start point, an intermediate point, and an end point.)

#### • When liner interpolation is applied



### Modal command

The 3-dimensional circular interpolation command G02.4 (G03.4) is a modal command belonging to 01 group. The command will remain valid until the other G command in the 01 group is issued. When the 3-dimensional circular interpolation command is carried out continuously, the end point of present command is the start point of next command.



### Precautions

- (1) If single block is valid and this command is operated, a block stop is carried out at an intermediate point and the end point.
- (2) The speed command during 3-dimensional circular interpolation is the tangential speed on arc.
- (3) When 3-dimensional circular interpolation is commanded while incremental command is valid, the relative position of the intermediate point in respect to the start point is designated in the intermediate point designation block, and the relative position of the end point in respect to the intermediate point is designated in the intermediate point designation block.
- (4) The path of 3-dimensional circular interpolation during graphic check is drawn as linear at each range from start point to intermediate point and from intermediate point to end point.



## Relation with other functions

- (1) Commands that cannot be used  
 (a) G code command which leads to a program error during 3-dimensional circular interpolation modal

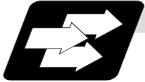
G code	Function name	Program error
G05 Pn	High-speed machining mode	P34
G05 P10000	High-speed high-accuracy control II	P34
G05.1 Q0/G05.1 Q1	High-speed high-accuracy control I	P34
G07.1	Cylindrical interpolation	P485
G12/G13	Circular cutting CW/CCW	P75
G12.1	Polar coordinate interpolation	P485
G16	Polar coordinate command	P75
G41/G42	Tool radius compensation	P75
G41/G42	3-dimensional tool radius compensation	P75
G41.1/G42.1	Normal line control	P75
G43	Tool length compensation	P75
G51	Scaling	P75
G51.1	Mirror image	P75
G66/G66.1	User macro	P75
G67	User macro	P276
G68.1	Programmable coordinate rotation	P75
G68.1	3-dimensional coordinate conversion	P921
G73/G74/G76/G81/G82/G83/ G84/G85/G86/G87/G88/G89	Fixed cycle	P75

- (b) G code command which leads to a program error when 3-dimensional circular interpolation is commanded

G code modal	Function name	Program error
G05 Pn	High-speed machining mode	P34
G05 P10000	High-speed high-accuracy control II	P34
G05.1 Q1	High-speed high-accuracy control I	P34
G07.1	Cylindrical interpolation	P481
G12.1	Polar coordinate interpolation	P481
G16	Polar coordinate command	P75
G41/G42	Tool radius compensation	P75
G41/G42	3-dimensional tool radius compensation	P75
G41.1/G42.1	Normal line control	P75
G43	Tool length compensation	P75
G51	Scaling	P75
G51.1	Mirror image	P75
G66/G66.1	User macro	P75
G68.1	Programmable coordinate rotation	P75
G68.1	3-dimensional coordinate conversion	P922

- (2) Functions that cannot be used  
 If following functions are used in 3-dimensional circular interpolation, alarm will occur.
- Chopping
  - Macro interruption
  - Corner chamfering / corner R
  - Mirror image by parameter setting
  - Mirror image by external input
- Restrictions may be added for other functions. Refer to explanation of each function.

## 6.15 NURBS Interpolation



## Function and purpose

This function realizes NURBS (Non-Uniform Rational B-Spline) curve machining by simply commanding NURBS curve parameters (stage, weight, knot, control point), which is used for the curved surface/line machining, without replacing the path with minute line segments.

This function operates only in the high-speed high-accuracy control II mode, so the high-speed high-accuracy control II option is required.

During NURBS interpolation, interpolation takes place at the commanded speed. However, if the curvature is large, the speed is clamped so that the machine's tolerable acceleration rate is not exceeded.



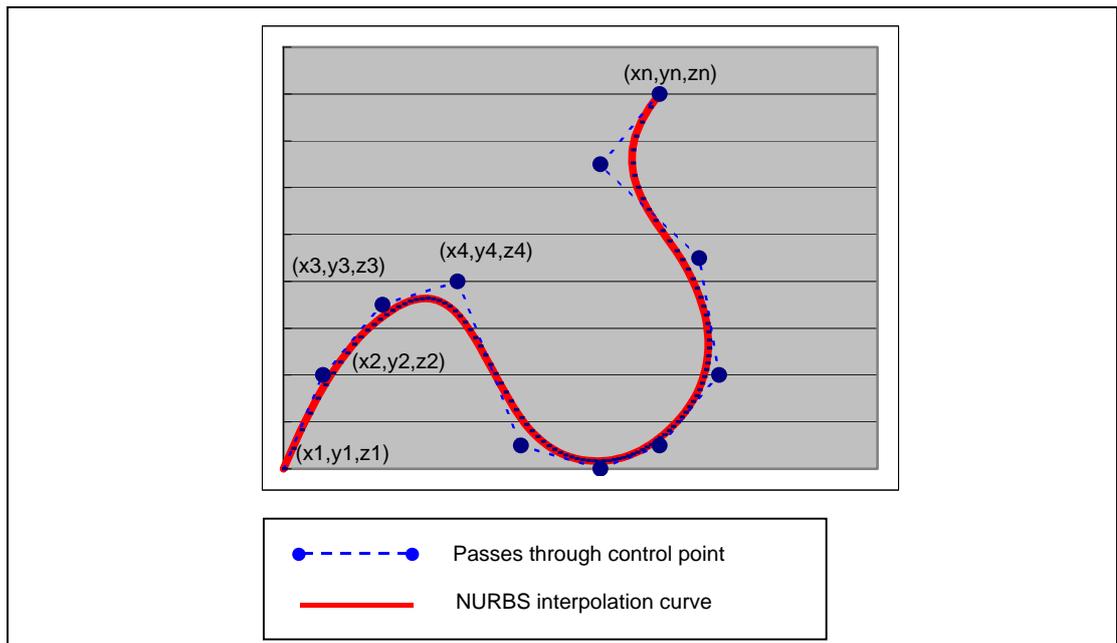
## Command format

<b>G05 P10000 ;</b>	<b>High-speed high-accuracy control II mode ON</b>
<b>G06.2 Pp Kk1 Xx1 Yy1 Zz1 Rr1 Ff;</b>	<b>NURBS interpolation ON</b>
<b>Kk2 Xx2 Yy2 Zz2 Rr2;</b>	
<b>Kk3 Xx3 Yy3 Zz3 Rr3;</b>	
<b>Kk4 Xx4 Yy4 Zz4 Rr4;</b>	
-----	
<b>Kkn Xxn Yyn Zzn Rrn;</b>	
<b>Kkn+1;</b>	
<b>Kkn+2;</b>	
<b>Kkn+3;</b>	
<b>Kkn+4;</b>	<b>NURBS interpolation OFF</b>
<b>G05 P0;</b>	<b>High-speed high-accuracy control II mode OFF</b>
G05 P10000	: High-speed high-accuracy control II mode
G06.2 Qq	: NURBS interpolation
Pp	: Set the stage of the NURBS curve. Designate in the same block as G06.2 command. The NURBS curve of the stage p will be (p-1)th curve. When omitted, Pp means the same as P4. (Example) P2: Primary curve (liner)
Kkn	: Knot Set the knot for each NURBS interpolation block. Set the same value for the knot in the 1st block to the stage p block. NURBS interpolation is terminated if there is a block exclusively with knot.
Xxn Yyn Zzn	: Control point coordinate value Designate the same coordinate value for the 1st block control point as that designated right before NURBS interpolation.
Rrn	: Control point weight Set the weight of each NURBS interpolation control point.
Ff	: Interpolation speed (May be omitted)



## Detailed description

- (1) Designate the stage P for the 1st block of NURBS interpolation.
- (2) Designate the same coordinate value for the 1st block control point of NURBS interpolation as that designated right before NURBS interpolation.
- (3) Designate all axes to be used in the subsequent NURBS interpolation blocks for 1st block of NURBS interpolation
- (4) Set the same value for knot K from the 1st block of NURBS interpolation to setting value block of the stage P.
- (5) Command knot K exclusive block of the same number as the setting value of the stage P for terminating NURBS interpolation. At this time, set the same value for knot K setting.



- (Note)** If an exclusive knot is commanded after NURBS interpolation immediately, NURBS interpolation mode is active again.  
 An exclusive knot that is commanded after NURBS interpolation immediately is the same meaning as following command.  
 G06.2 Pp Km Xxn Yyn Zzn R1.0



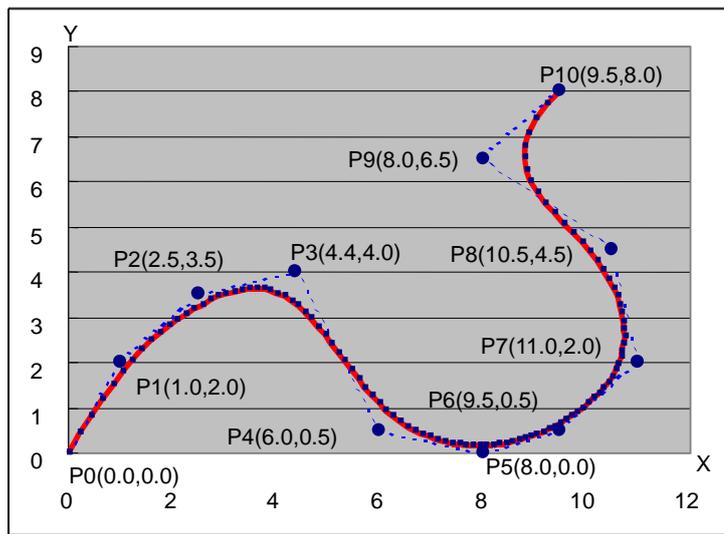
Example of program

The example of program that has 4 stages (cubic curve) and 11 control points is shown below.

Control point	P0	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10				
Knot	0.0	0.0	0.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	8.0	8.0	8.0

```

:
:
G05 P10000;
G90 G01 X0. Y0. Z0. F300;
G06.2 P4 X0. Y0. R1. K0; P0
X1.0 Y2.0 R1. K0; P1
X2.5 Y3.5 R1. K0; P2
X4.4 Y4.0 R1. K0; P3
X6.0 Y0.5 R1. K1; P4
X8.0 Y0.0 R1. K2; P5
X9.5 Y0.5 R1. K3; P6
X11.0 Y2.0 R1. K4; P7
X10.5 Y4.5 R1. K5; P8
X8.0 Y6.5 R1. K6; P9
X9.5 Y8.0 R1. K7; P10
K8;
K8;
K8;
K8;
G05 P0;
:
:
    
```



● ..... ● Passes through control point  
— NURBS interpolation curve



## Relation with other functions

- (1) G code/Feed/Miscellaneous functions  
All the G code, feedrate and MSTB code cannot be set during NURBS interpolation. However, when the fixed cycle G code is commanded in the same block where G06.2 is commanded, the fixed cycle G code is ignored. If a command other than the axis address designated in the 1st block of NURBS interpolation, R and K is commanded, a program error will occur.
- (2) Data format
- Optional block skip "/"  
Cannot be set in the NURBS interpolation 2nd block or after.
  - Control IN ("and Control OUT ")  
Cannot be set in the NURBS interpolation 2nd block or after.
  - Local variables and common variables  
Can be referred but cannot be set in the NURBS interpolation. Setting the variables causes a program error (P29).
  - System variables  
Cannot be referred nor set in the NURBS interpolation; program error (P29) will occur.
- (3) Interruption/restart  
The validity of program interruption/restart is shown below.
- | Type                | During NURBS interpolation |
|---------------------|----------------------------|
| Single block        | Valid (Note 1)             |
| Feed hold           | Valid                      |
| Reset               | Valid (Note 2)             |
| Program stop        | Invalid                    |
| Optional stop       | Invalid                    |
| Manual interruption | Invalid (Note 3)           |
| MDI interruption    | Invalid                    |
| Restart search      | Invalid                    |
| Macro interruption  | Invalid (Note 4)           |
| PLC interruption    | Invalid (Note 5)           |
- (Note 1)** A single block stop is carried out at only the last control points.  
The single block stop is not applied during NURBS interpolation.
- (Note 2)** NURBS interpolation mode is canceled with Reset (Reset1/Reset2/Reset&Rewind).
- (Note 3)** The operation differs according to the manual absolute signal status.
- When the manual absolute signal OFF  
NURBS interpolation is carried out in the state where axis-coordinate system is shifted by the manual absolute movement amount.
  - When the manual absolute signal ON  
At automatic start after manual interruption, a program error (P554) will occur after moving the by the remaining distance.  
Note that the operation can run continually if returning the axis to the original position after manual interruption.
- (Note 4)** "Macro interrupt" signal (UIT) is ignored.
- (Note 5)** "PLC interrupt" signal (PIT) is ignored.
- (4) Graphic check  
NURBS interpolation cannot be applied during graphic check (continuous/step check). Linear interpolation that connects the control points is applied during graphic check.

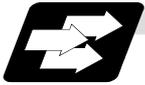


## Precautions

- (1) Target axes for NURBS interpolation are 3 basic axes.
- (2) Command the control point for all the axes for which NURBS interpolation is carried out in the 1st block (G06.2 block). A program error (P32) will occur if an axis which was not commanded in the 1st block is commanded in the 2nd block or after.
- (3) The first control point (G06.2 block coordinate value) should be commanded as the start point of the NURBS curve. Thus, command so that it matches the end point of the previous block. A program error will occur if the points do not match.
- (4) The command range of the weight is 0.0001 to 99.9999. Even if the decimal point is omitted, the value will be handled as the one with a decimal point.  
If "1" is commanded, the result will be the same as "1.0". If more than 5 digits are commanded after the decimal point, a program error (P33) will occur.
- (5) The knot command cannot be omitted, and must be commanded in each block. A program error (P33) will occur if omitted.
- (6) As with knot, in the same manner as weight, up to 4 digits can be commanded after the decimal point. Even if the decimal point is omitted, the value will be handled as the one with a decimal point.  
If "1" is commanded, the result will be the same as "1.0". If more than 5 digits are commanded after the decimal point, a program error (P33) will occur.
- (7) As with knot, command the same or greater value than the previous block. If a smaller value than previous block is set, a program error (P551) will occur.
- (8) NURBS interpolation cannot be applied during graphic check (continuous/step check). Linear interpolation that connects the control points is applied during graphic check.
- (9) NURBS interpolation mode is canceled with Reset(Reset1/Reset2/Reset&Rewind).
- (10) NURBS interpolation can be commanded in only the following modes. If NURBS interpolation is commanded in other than the following modes, the program error (P29) will occur.

Type	Mode in which NURBS interpolation can be commanded
G group 5	Asynchronous feed (G94)
G group 7	Tool radius compensation cancel (G40)
G group 8	Tool length compensation(G43) Tool length compensation cancel (G44)
G group 9	Fixed cycle cancel (G80)
G group 11	Scaling cancel (G50)
G group 13	High-accuracy control 1 ON (G61.1) Cutting mode (G64)
G group 14	User macro modal call cancel (G67)
G group 15	Normal line control cancel (G40.1)
G group 16	Programmable coordinate rotation mode OFF /3-dimensional coordinate conversion mode OFF (G69.1)
G group 17	Constant surface speed control OFF (G97)
G group 18	Polar coordinate command OFF (G15)
G group 19	G command mirror image cancel (G50.1)
G group 21	Polar coordinate interpolation cancel (G13.1)
-	Not during the parameter coordinate rotation
-	Not during the mirror image by parameter setting
-	Not during the mirror image by external input

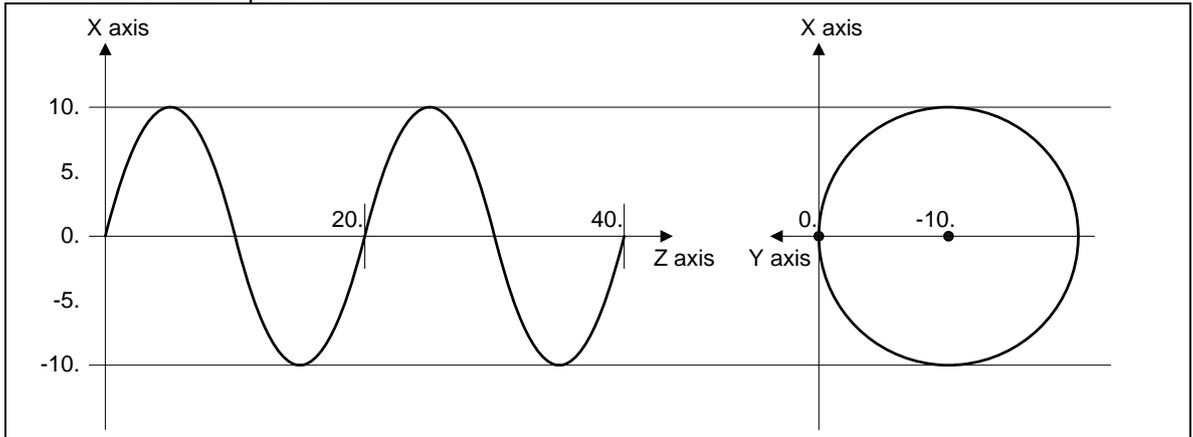
## 6.16 Hypothetical Axis Interpolation; G07



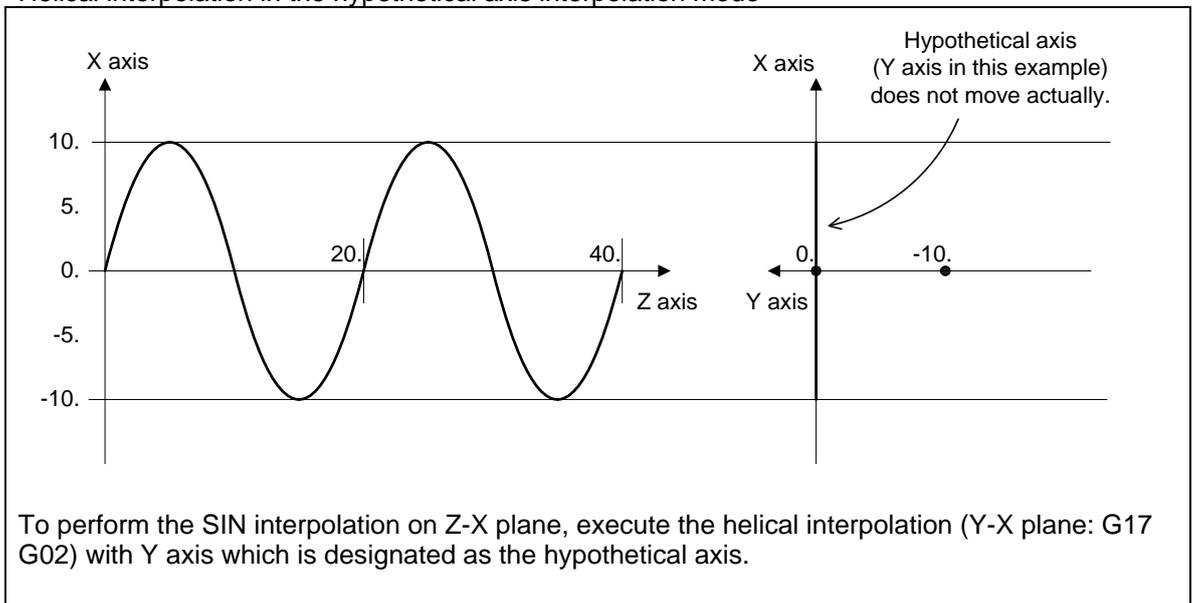
## Function and purpose

Take one of the axes of the helical interpolation or spiral interpolation, including a linear axis, as a hypothetical axis (axis with no actual movement) and perform pulse distribution. With this procedure, an interpolation equivalent to the helical interpolation or spiral interpolation looked from the side (hypothetical axis), or SIN or COS interpolation, will be possible.

## Normal helical interpolation



## Helical interpolation in the hypothetical axis interpolation mode



## Command format

**G07 α0 ; Hypothetical axis interpolation mode ON**

**G07 α1 ; Hypothetical axis interpolation mode cancel**

α : Designate the axis for which hypothetical axis interpolation is performed.

**Detailed description**

- (1) During "G07 $\alpha$ 0" to "G07 $\alpha$ 1",  $\alpha$  axis will be the hypothetical axis.
- (2) Any axis among the NC axes can be designated as the hypothetical axis.
- (3) Multiple axes can be designated as the hypothetical axis.
- (4) The number other than "0" (hypothetical axis interpolation mode ON) or "1" (cancel) is commanded, it will be handled as "1" (cancel). However, when only the axis name is designated with no number, it will be handled as "0" (mode ON).

**Example of program**

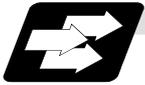
N01 G07 Y0 ;	Y axis is handled as hypothetical axis.
N02 G17 G02 X0. Y0. Z40. I0. J-10. P2 F50;	SIN interpolation is executed on X-Z plane.
N03 G07 Y1 ;	Y axis is returned to the actual axis.

**Precautions**

- (1) Interpolation functions that are used for hypothetical axis interpolation are helical interpolation and spiral interpolation.
- (2) Cancel the hypothetical axis interpolation before the high-speed high-accuracy control 2 (G05P10000) is commanded.
- (3) The hypothetical axis interpolation is valid only in the automatic operation. It is invalid in the manual operation mode. Handle interruption is valid even for the hypothetical axis, that is, axis will move by the interrupted amount.
- (4) Movement command for the hypothetical axis will be ignored. The feedrate will be distributed in the same manner as actual axis.
- (5) The protection functions such as interlock or stored stroke limit are valid for the hypothetical axis.
- (6) Even when the hypothetical axis is applied for the hypothetical axis again, no error will occur and the hypothetical mode will be continued.
- (7) When the hypothetical axis cancel is commanded to the actual axis, no error will occur and the axis is actual as it is.
- (8) The hypothetical axis will be canceled by carrying out the reset 2 or reset & rewind.

## 7. Feed Functions

### 7.1 Rapid Traverse Rate



#### Function and purpose

The rapid traverse rate can be set independently with parameters for each axis. The available speed ranges are from 1 mm/min to 10000000 mm/min. The upper limit is subject to the restrictions imposed by the machine specifications.

Refer to the specifications manual of the machine for the rapid traverse rate settings.

The feedrate is valid for the G00, G27, G28, G29, G30 and G60 commands.

Two paths are available for positioning: the interpolation type where the area from the start point to the end point is linearly interpolated or the non-interpolation type where movement proceeds at the maximum speed of each axis. The type is selected with parameter "#1086 G0Intp". The positioning time is the same for each type.

If the high-accuracy control mode's rapid traverse rate is set, the axis will move at that feed rate during high-accuracy control, high-speed high-accuracy control I/II, high-accuracy spline control or SSS control.

- If the value set for the high-accuracy control mode rapid traverse rate is 0, the axis will move at the rapid traverse rate.
- The high-accuracy control mode rapid traverse rate can be set independently for each axis.
- The high-accuracy control mode rapid traverse rate is effective for the G00, G27, G28, G29, G30 and G60 commands.
- Override can be applied on the high-accuracy control mode rapid traverse rate using the external signal supplied.

### 7.2 Cutting Feedrate



#### Function and purpose

The cutting feedrate is assigned with address F and 8 digits (F8-digit direct designation).

The F8 digits are assigned with a decimal point for a 5-digit integer and a 3-digit fraction. The cutting feedrate is valid for the G01, G02, G03, G02.1 and G03.1 commands.

If the high-accuracy control mode's cutting clamp speed is set, the cutting feed rate will be clamped at that feedrate during high-accuracy control, high-speed high-accuracy control, high-accuracy spline control or SSS control.

- If the value set for the high-accuracy control mode cutting clamp speed is 0, the axis will be clamped at the cutting feed clamp speed.
- The cutting feedrate is clamped with high-accuracy control mode cutting clamp speed in the parameter.

#### (Example)

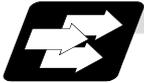
	Feedrate	
G1 X100. Y100. F200 ;	200.0mm/min	F200 or F200.000 gives the same rate.
G1 X100. Y100. F123.4 ;	123.4mm/min	
G1 X100. Y100. F56.789 ;	56.789mm/min	

Speed range that can be commanded (when input setting unit is 1 $\mu$ m)

Command mode	Feed rate command range	Remarks
mm/min	0.001 to 10000000 mm/min	
inch/min	0.0001 to 1000000 inch/min	
°/min	0.001 to 10000000 °/min	

**(Note 1)** A program error (P62) results when there is no F command in the first cutting command (G01, G02, G03) after the power has been switched on.

### 7.3 F1-digit Feed



#### Function and purpose

By setting the F1-digit feed parameter, the feedrate which has been set to correspond to the 1-digit number following the F address serves as the command value.

When F0 is assigned, the rapid traverse rate is established and the speed is the same as for G00. (G modal does not change, but the acceleration/deceleration method is followed by the settings for the rapid traverse.)

When F1 to F5 is assigned, the feedrate set to correspond to the command serves as the command value.

The command greater than F6 is considered to be the normal cutting feedrate.

The F1-digit command is valid in a G01, G02, G03, G02.1 or G03.1 modal.

The F1-digit command can also be used for fixed cycle.



#### Detailed description

The override function of the feedrate which is set in accordance to the F1-digit is performed by using the 1st manual handle. (Feedrate cannot be changed with the 2nd or 3rd handle.)

The amount by which the feedrate can be increased or reduced is determined by the following formula.

$$\pm\Delta F = \frac{FM}{K} \times (\pm \text{number of manual handle pulse generator pulses})$$

Where "+" means increase, and "-" means reduction.

K : Operation constant (This is the number of FM divisions, and is the calculated constant of the increment/decrement speed per scale of the manual handle pulse generator.)  
This is set with the base specification parameter "#1507 F1\_K".

FM : This is the clamp speed of F1 to F5  
This is set with the base specification parameter "#1506 F1\_FM".

Set the corresponding speed of F1 to F5 with the base specification parameters "#1185 spd\_F1" to "#1189 spd\_F5" respectively.

The increase/reduction range is from "0" to the set value of the parameter "#1506 F1\_FM".

Operation alarm (104) will occur when the feedrate is 0.

#### (1) Operation method

- (a) Make the F1-digit command valid. (Set the base specification parameter "#1079 F1digit" to 1.)
- (b) Set FM and K.
 

Setting range	
K : 1 to 32767	(Base specification parameter "#1507 F1_K")
FM : 0 to Fmax (mm/min)	(Base specification parameter "#1506 F1_FM")
- (c) Set F1 to F5. (Base specification parameter "1185 spd\_F1" to "#1189 spd\_F5")

#### (2) Special notes

- (a) Use of both the F1-digit command and normal cutting feedrate command is possible when the F1-digit is valid.

##### (Example 1)

F0	Rapid traverse rate
F1 to F5	F1-digit
F6 or more	Normal cutting feedrate command

- (b) F1 to F5 are invalid in the G00 mode and the rapid traverse rate is established instead.
- (c) If F0 is used in the G02 or G03 mode, a program error (P121) will result.

- (d) When F1. to F5. (with decimal point) are assigned, the 1mm/min to 5mm/min direct commands are established instead of the F1-digit command.
- (e) When the commands are used with the millimeter or degree units, the feedrate set to correspond to F1 to F5 serves as the assigned speed mm (°)/min.
- (f) When the commands are used with inch units, one-tenth of the feedrate set correspond to F1 to F5 serves at the assigned speed inch/min.
- (g) The number of manual handle pulses is 1 pulse per scale unit regardless of the scaling factor.
- (h) During a F1-digit command, the F1-digit number and F1-digit command signal are output as the PLC signals.
- (i) Even if the F1-digit feed commanded during the feed per rotation (G95) is considered as a regular F command (direct value command).

### (3) F1-digit and G commands

- (a) 01 group G command in same block as F1-digit commands

	Executed feedrate	Modal display rate	G modal
G0F0 F0G0	Rapid traverse rate	0	G0
G0F1 F1G0	Rapid traverse rate	1	G0
G1F0 F0G1	Rapid traverse rate	0	G1
G1F1 F1G1	F1 contents	1	G1

- (b) F1-digit and unmodal commands may be assigned in the same block. In this case, the unmodal command is executed and at the same time the F1-digit modal command is updated.

### (4) Example of arithmetic constant K setting

When the handle scale unit is to be made  $\pm 10$ mm/min.

FM is made 15000 mm/min:

$$\Delta F = 10 = \frac{15000}{K}$$

Therefore, K is 1500.

The feed rate is made F (1 to 5)  $\pm 10$  (mm/min) by rotating the handle through one scale unit.

### (5) Valid manual handle conditions

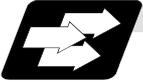
The manual handle is valid during cutting feed (F1 to F5), automatic start, F1-digit valid and manual handle valid switch ON at the machine side as well as in the MDI mode, tape mode or memory mode provided that the machine lock (machine lock rapid traverse) or dry run status has not been established.

The function cannot be used when the handle specifications have not been provided.

## 7. Feed Functions

### 7.4 Per-minute/Per-revolution Feed (Asynchronous/Synchronous Feed); G94, G95

#### 7.4 Per-minute/Per-revolution Feed (Asynchronous/Synchronous Feed); G94, G95



##### Function and purpose

Using the G95 command, it is possible to assign the feed amount per rotation with an F code. When this command is used, the rotary encoder must be attached to the spindle.  
When the G94 command is issued the per-minute feed rate will return to the designated per-minute feed (asynchronous feed) mode.



##### Command format

<b>G94;</b>	
<b>G95;</b>	
G94	: Per-minute feed (mm/min) (asynchronous feed)
G95	: Per-revolution feed (mm/rev) (synchronous feed)

The G95 command is a modal command and so it is valid until the G94 command (per-minute feed) or G93 command (inverse time feed) is next assigned.

- (1) The F code command range is as follows.  
The movement amount per spindle revolution with synchronous feed (per-revolution feed) is assigned by the F code and the command range is as shown in the table below.

##### Metric input

Input command unit system	B (0.001mm)		C (0.0001mm)	
	Command mode	Feed per minute	Feed per rotation	Feed per minute
Command address	F (mm/min)	E (mm/rev)	F (mm/min)	E (mm/rev)
Minimum command unit	1 (= 1.000), (1. = 1.000)	1 (= 0.001), (1. = 1.000)	1 (= 1.0000), (1. = 1.0000)	1 (= 0.0001), (1. = 1.0000)
Command range	0.001 ~1000000.000	0.001 ~999.999	0.0001 ~1000000.0000	0.0001 ~999.9999

Input command unit system	D (0.00001mm)		E (0.000001mm)	
	Command mode	Feed per minute	Feed per rotation	Feed per minute
Command address	F (mm/min)	E (mm/rev)	F (mm/min)	E (mm/rev)
Minimum command unit	1 (= 1.00000), (1. = 1.00000)	1 (= 0.00001), (1. = 1.00000)	1 (= 1.000000), (1. = 1.000000)	1 (= 0.000001), (1. = 1.000000)
Command range	0.00001 ~1000000.00000	0.00001 ~999.99999	0.000001 ~1000000.000000	0.000001 ~999.999999

## 7. Feed Functions

### 7.4 Per-minute/Per-revolution Feed (Asynchronous/Synchronous Feed); G94, G95

#### Inch input

Input command unit system	B (0.0001inch)		C (0.00001inch)	
	Command mode	Feed per minute	Feed per rotation	Feed per minute
Command address	F (inch/min)	E (inch/rev)	F (inch/min)	E (inch/rev)
Minimum command unit	1 (= 1.0000), (1. = 1.0000)	1 (= 0.0001), (1. = 1.0000)	1 (= 1.00000), (1. = 1.00000)	1 (= 0.00001), (1. = 1.00000)
Command range	0.0001 ~ 100000.0000	0.0001 ~ 999.9999	0.00001 ~ 100000.00000	0.00001 ~ 999.99999

Input command unit system	D (0.000001inch)		E (0.0000001inch)	
	Command mode	Feed per minute	Feed per rotation	Feed per minute
Command address	F (inch/min)	E (inch/rev)	F (inch/min)	E (inch/rev)
Minimum command unit	1 (= 1.000000), (1. = 1.000000)	1 (= 0.000001), (1. = 1.000000)	1 (= 1.0000000), (1. = 1.0000000)	1 (= 0.0000001), (1. = 1.0000000)
Command range	0.000001 ~100000.000000	0.000001 ~999.999999	0.0000001 ~100000.0000000	0.0000001 ~999.9999999

- (2) The effective speed (actual movement speed of machine) under per-revolution feed conditions is given in the following formula (Formula 1).

$$FC = F \times N \times OVR \dots \text{(Formula 1)}$$

Where FC = Effective rate (mm/min, inch/min)  
 F = Commanded feedrate (mm/rev, inch/rev)  
 N = Spindle speed (r/min)  
 OVR = Cutting feed override

When a multiple number of axes have been commanded at the same time, the effective rate FC in formula 1 applies in the vector direction of the command.

**(Note 1)** The effective rate (mm/min or inch/min), which is produced by converting the commanded speed, the spindle speed and the cutting feed override into the per-minute speed, appears as the FC on the monitor 1. Screen of the setting and display unit.

**(Note 2)** When the above effective rate exceeds the cutting feed clamp rate, it is clamped at that clamp rate.

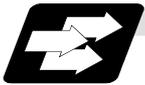
**(Note 3)** If the spindle speed is zero when synchronous feed is executed, operation alarm "105" results.

**(Note 4)** Under dry run conditions, asynchronous speed applies and the axes move at the manual feed rate (mm/min, inch/min, °/min).

**(Note 5)** The fixed cycle G84 (tapping cycle) and G74 (reverse tapping cycle) are executed to the feed mode that is already designated.

**(Note 6)** Whether asynchronous feed (G94) or synchronous feed (G95) is applied when the power is switched on or when M02 or M30 is executed is set with the parameter "#1074 I\_Sync".

## 7.5 Inverse Time Feed; G93

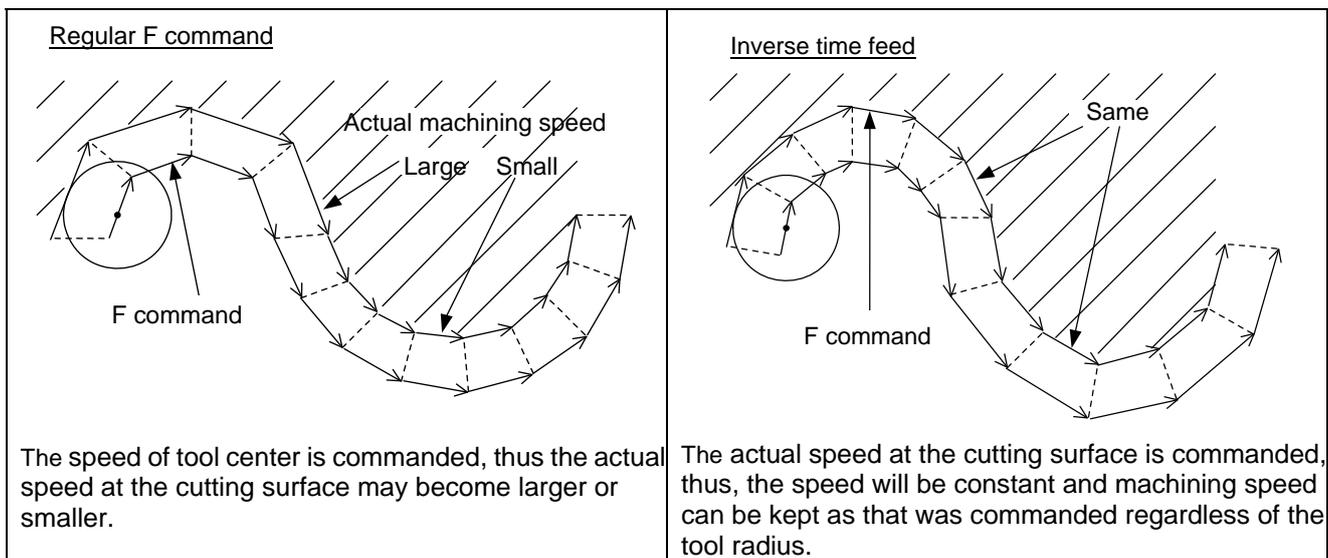


## Function and purpose

During inside cutting when machining curved shapes with radius compensation applied, the machining speed on the cutting surface becomes faster than the tool center feedrate. Therefore, problems such as reduced accuracy may occur.

This reduced accuracy can be prevented with inverse time feed. This function can, in place of normal feed commands, issue one block of machining time (inverse) in F commands. The machining speed on the cutting surface is constantly controlled, even if radius compensation is applied to the machining program that expresses the free curve surface with fine segment lines.

Note that when the calculated machining time exceeds the cutting feed clamp speed, the F command value in the inverse time feed follows the cutting feed clamp speed.



## Command format

**G93 ;                      Inverse time feed**

Inverse time feed (G93) is a modal command. Once commanded, it is valid until feed per minute or feed per revolution is commanded.

G00 Xx1 Yy1 ;	
G93;	→ Inverse time feed mode ON
G01 Xx2 Yy2 Ff2;	→ In inverse time feed mode
G02 Xx3 Yy3 Ii3 Jj3 Ff3;	:
G94(G95);	→ Inverse time feed mode OFF

In movement blocks, since processing time is commanded to a line segment, command the feedrate "F" each time.



## Detailed description

- (1) Inverse time feed (G93) is a modal command. Once commanded, it is valid until feed per minute (G94) or feed per revolution (G95) is commanded, or until a reset (M02, M30, etc.) is executed.
- (2) Command method of F command values in inverse time feed

		Metric command (G21)	Inch command (G20)
During linear mode (G01)		Cutting point feedrate (mm/min)	Cutting point feedrate (inch/min)
		Line segment length (mm)	Line segment length (inch)
During arc mode (G02, G03) (G02.1, G03.1)		Cutting point feedrate (mm/min)	Cutting point feedrate (inch/min)
		Start point arc radius (mm)	Start point arc radius (inch)
Command range	B	0.001 to 999999.999(1/min)	
	C	0.0001 to 999999.9999(1/min)	
	D	0.00001 to 999999.99999(1/min)	
	E	0.000001 to 999999.999999(1/min)	

- (3) The initial modal after a restart is G94 (feed per minute) or G95 (feed per revolution).
- (4) The feedrate of the block inserted in tool radius compensation and corner R/C is the same speed as the feedrate of the block immediately before it.
- (5) The feedrate of the block inserted in C axis normal line control (normal line control type II) is the same speed as the feedrate of the movement block after turning.



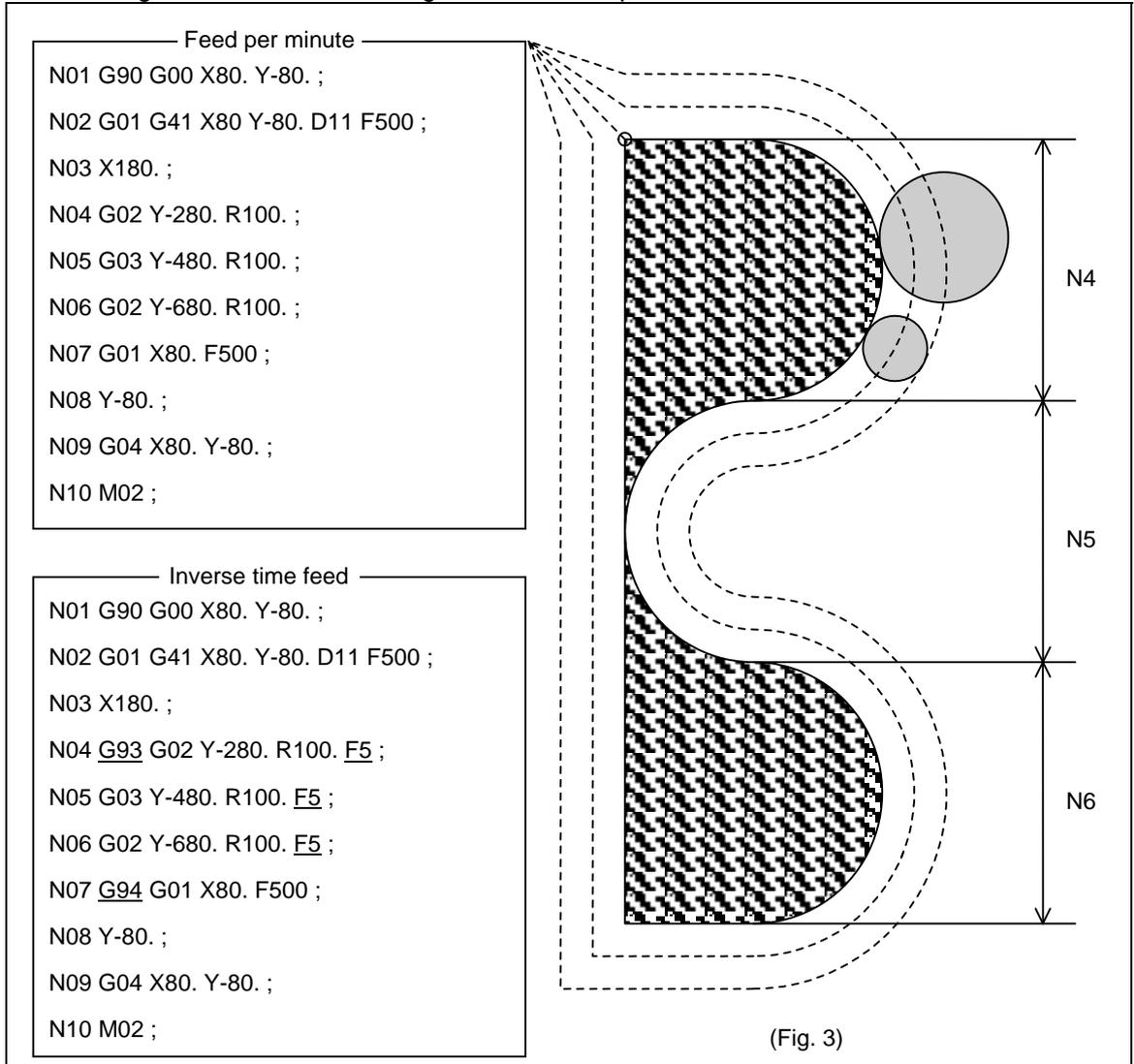
## Precautions

- (1) The initial modal after a restart is G94 (feed per minute) or G95 (feed per revolution).
- (2) The F command in G93 modal is unmodal. Issue an F command for each block. The program error (P62) will occur in blocks with no F command.
- (3) The program error (P62) will occur when F0 is commanded.
- (4) An F command is necessary when changing from G93 to G94/G95. The program error (P62) will occur if there is no F command.
- (5) The feed function is clamped at the maximum cutting speed. Consequently, the feed may be slower than the commanded speed.
- (6) If an extremely slow speed such as F0.001 is designated, an error will occur in the machining time.



Example of program

When using inverse time feed during tool radius compensation



Comparison between feed per minute and inverse time feed

(Assuming that tool radius is 10. [mm]) (Unit: mm/min)

Sequence No.	Location	Feed per minute		Inverse time feed	
		Feedrate of tool center	Feedrate of cutting point	Feedrate of tool center	Feedrate of cutting point
N04		F500	F450	F550	F500
N05		F500	F550	F450	F500
N06		F500	F450	F550	F500

↓

The block seam protrudes due to the cutting speed change at the block seam.

↓

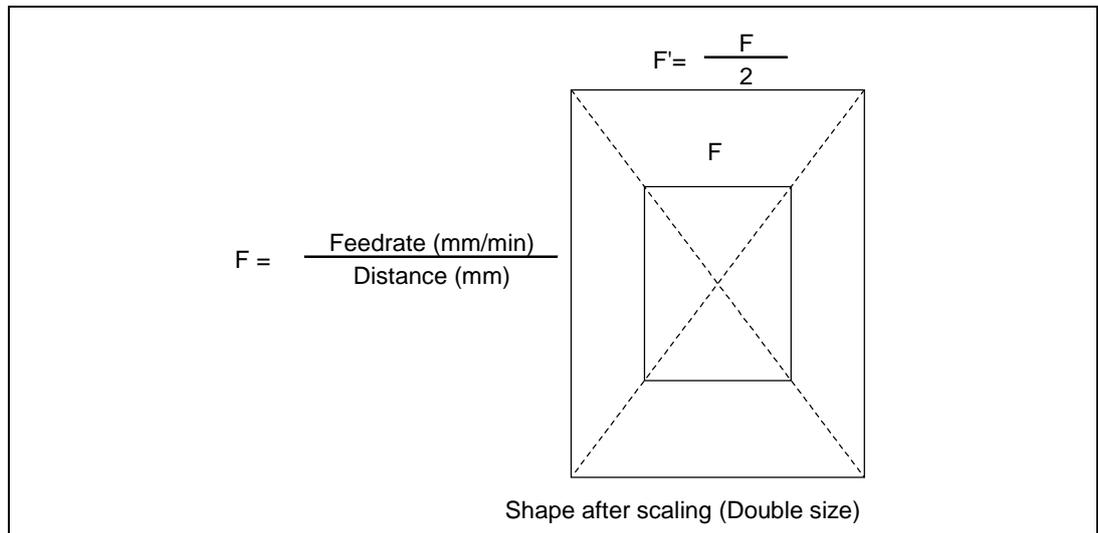
The feedrate follows the command regardless of the tool radius.



## Relation with other functions

## (1) Scaling (G51)

When using a scaling function, issue a F command for the shape after scaling. For example, if a double-size scaling is carried out, the machining distance will be twice. Thus, if executing a cutting at the same speed as that of before scaling, command the value (F') calculated by dividing F value by the multiples of scaling.



## (2) High-speed machining mode II (G05P2)

With the inverse time feed (G93) modal, high-speed machining mode II (G05P2) is operated in the inverse time feed mode, instead of high-speed machining mode. High-speed machining mode will be valid when the inverse time feed mode is canceled.

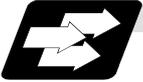
- (3) If the speed calculated in the G93 mode exceeds the speed range at the feed per minute, clamping is performed at the clamp speed set with parameters.
- (4) The program error (P125) will occur when the commands below are issued in the inverse time feed (G93) mode.

G code	Function
G02.3, G03.3	Exponential interpolation
G06.2	NURBS interpolation
G12	Circular cutting CW
G13	Circular cutting CCW
G31~G31.3	Skip
G33	Thread cutting
G34~G36, G37	Special fixed cycle
G37.1	Automatic tool length measurement
G73~G89	Fixed cycle
G96	Constant surface speed control ON

- (5) The program error (P125) will occur if inverse time feed (G93) is commanded in the following modes.

G code	Function
G02.3, G03.3	Exponential interpolation
G33	Thread cutting
G73~G89	Fixed cycle
G96	Constant surface speed control ON

## 7.6 Feedrate Designation and Effects on Control Axes



## Function and purpose

It has already been mentioned that a machine has a number of control axes. These control axes can be divided into linear axes which control linear movement and rotary axes which control rotary movement. The feedrate is designed to assign the displacement speed of these axes, and the effect exerted on the tool movement speed which poses problems during cutting differs according to when control is exercised over the linear axes or when it is exercised over the rotary axes.

The displacement amount for each axis is assigned separately for each axis by a value corresponding to the respective axis. The feedrate is not assigned for each axis but assigned as a single value. Therefore, when two or more axes are to be controlled simultaneously, it is necessary to understand how this will work for each of the axes involved.

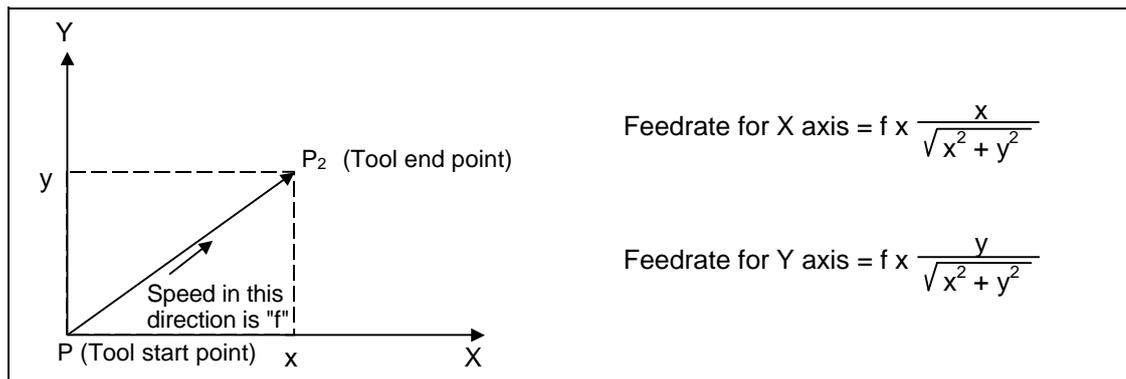
The assignment of the feedrate is described with the following related items.



## When controlling linear axes

Even when only one machine axis is to be controlled or there are two or more axes to be controlled simultaneously, the feed rate which is assigned by the F code functions as a linear speed in the tool advance direction.

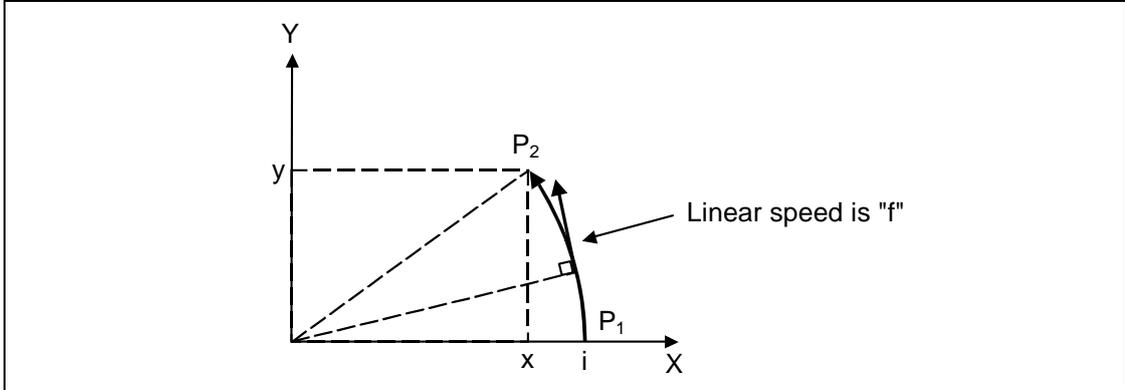
**(Example)** When the feedrate is designated as "f" and linear axes (X and Y) are to be controlled:



When only linear axes are to be controlled, it is sufficient to designate the cutting feed in the program. The feedrate for each axis is such that the designated rate is broken down into the components corresponding to the movement amounts.

**(Example)** When the feedrate is designated as "f" and the linear axes (X and Y) are to be controlled using the circular interpolation function:

The rate in the tool advance direction, or in other words the tangential direction, will be the feedrate designated in the program.



In this case, the feed rate of the X and Z axes will change along with the tool movement. However, the combined speed will always be maintained at the constant value "f".

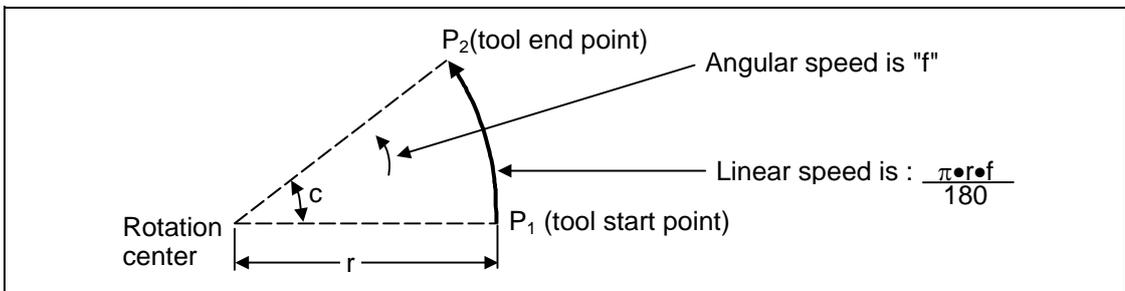


**When controlling rotary axes**

When rotary axes are to be controlled, the designated feedrate functions as the rotary speed of the rotary axes or, in other words, as an angular speed.

Consequently, the cutting feed in the tool advance direction, or in other words the linear speed, varies according to the distance between the center of rotation and the tool. This distance must be borne in mind when designating the feedrate in the program.

**(Example)** When the feedrate is designated as "f" and rotary axis (CA) is to be controlled ("f" units = °/min)



In this case, in order to make the cutting feed (linear feed) in the tool advance direction "fc" :

$$fc = f \times \frac{\pi \cdot r}{180}$$

Therefore, the feedrate to be designated in the program must be :

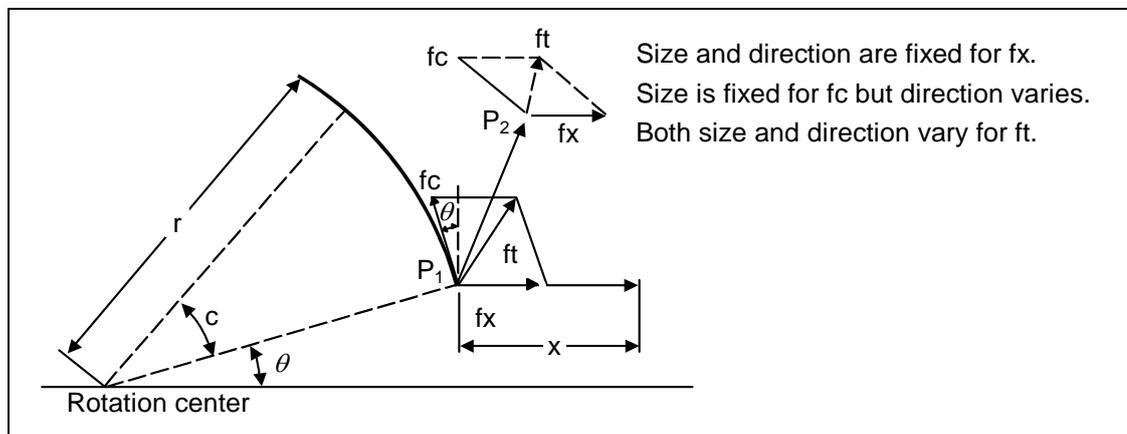
$$f = fc \times \frac{180}{\pi \cdot r}$$



### When linear and rotary axes are to be controlled at the same time

The controller proceeds in exactly the same way whether linear or rotary axes are to be controlled. When a rotary axis is to be controlled, the numerical value assigned by the coordinate word (A, B, C) is the angle and the numerical values assigned by the feedrate (F) are all handled as linear speeds. In other words,  $1^\circ$  of the rotary axis is treated as being equivalent to 1mm of the linear axis. Consequently, when both linear and rotary axes are to be controlled simultaneously, the components for each axis of the numerical values assigned by F will be the same as previously described "When controlling linear axes". However, although in this case both the size and direction of the speed components based on linear axis control do not vary, the direction of the speed components based on rotary axis control will change along with the tool movement (their size will not change). This means, as a result, that the combined tool advance direction feedrate will vary along with the tool movement.

**(Example)** When the feed rate is designated as "f" and Linear (X) and rotary (C) axes are to be controlled simultaneously. In the X-axis incremental command value is "x" and the C-axis incremental command value is "c":



X-axis feedrate (linear speed) "fx" and C-axis feedrate (angular speed) "ω" are expressed as:

$$f_x = f \times \frac{X}{\sqrt{X^2 + C^2}} \dots\dots\dots (1)$$

$$\omega = f \times \frac{C}{\sqrt{X^2 + C^2}} \dots\dots\dots (2)$$

Linear speed "fc" based on C-axis control is expressed as:

$$f_c = \omega \times \frac{\pi \times r}{180} \dots\dots\dots (3)$$

If the speed in the tool advance direction at start point P1 is "ft" and the component speeds in the X-axis and Y-axis directions are "ftx" and "fty", respectively, then these can be expressed as:

$$f_{tx} = -\sin\left(\frac{\pi}{180} \theta\right) \times \frac{\pi}{180} \omega + f_x \dots\dots\dots (4)$$

$$f_{ty} = -\cos\left(\frac{\pi}{180} \theta\right) \times \frac{\pi}{180} \omega \dots\dots\dots (5)$$

Where r is the distance between center of rotation and tool (in mm units), and θ is the angle between the P1 point and the X axis at the center of rotation (in units °).

The combined speed "ft" according to (1), (2), (3), (4) and (5) is:

$$\begin{aligned} f_t &= \sqrt{f_{tx}^2 + f_{ty}^2} \\ &= f \times \frac{\sqrt{X^2 - X \cdot C \cdot \sin\left(\frac{\pi}{180} \theta\right) \frac{\pi}{90} + \left(\frac{\pi \cdot r \cdot C}{180}\right)^2}}{\sqrt{X^2 + C^2}} \dots\dots\dots (6) \end{aligned}$$

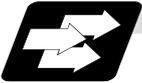
Consequently, feedrate "f" designated by the program must be as follows:

$$f = f_t \times \frac{\sqrt{X^2 + C^2}}{\sqrt{X^2 - X \cdot C \cdot \sin\left(\frac{\pi}{180} \theta\right) \frac{\pi}{90} + \left(\frac{\pi \cdot r \cdot C}{180}\right)^2}} \dots\dots\dots (7)$$

"ft" in formula (6) is the speed at the P1 point and the value of θ changes as the C axis rotates, which means that the value of "ft" will also change.

Consequently, in order to keep the cutting feed "ft" as constant as possible the angle of rotation which is designated in one block must be reduced to as low as possible and the extent of the change in the θ value must be minimized.

7.7 Rapid Traverse Constant Inclination Acceleration/Deceleration



Function and purpose

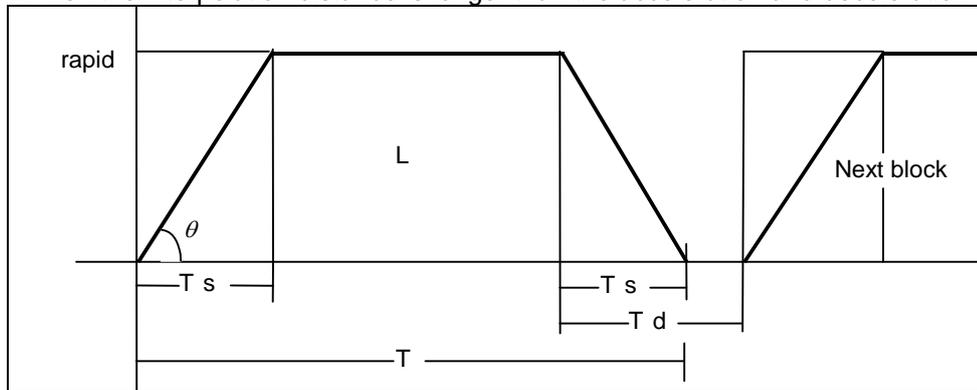
This function performs acceleration and deceleration at a constant inclination during linear acceleration/deceleration in the rapid traverse mode. Compared to the method of acceleration/deceleration after interpolation, the constant inclination acceleration/deceleration method makes for improved cycle time.



Detailed description

- (1) Rapid traverse constant inclination acceleration/deceleration are valid only for a rapid traverse command.  
Also, this function is effective only when the rapid traverse command acceleration/deceleration mode is linear acceleration and linear deceleration.
- (2) The acceleration/deceleration patterns in the case where rapid traverse constant inclination acceleration/deceleration are performed are as follows.

<When the interpolation distance is longer than the acceleration and deceleration distance>



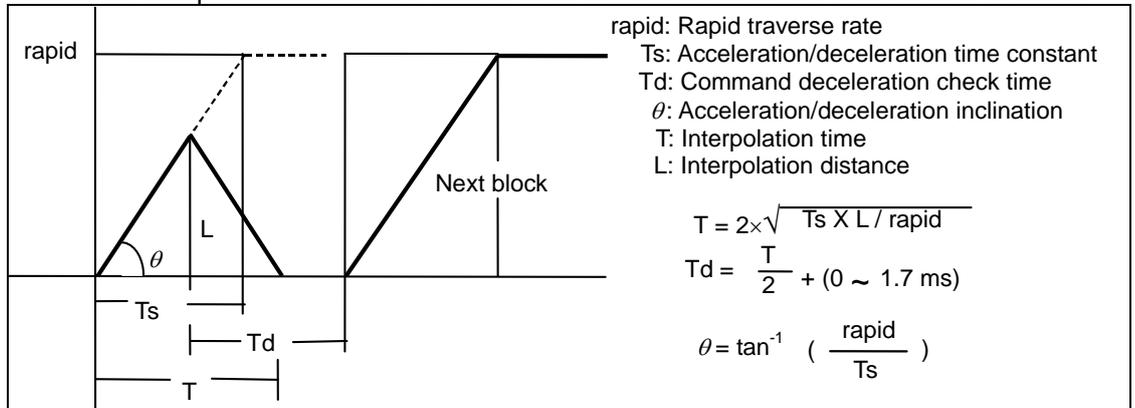
rapid : Rapid traverse rate  
 Ts : Acceleration/deceleration time constant  
 Td : Command deceleration check time  
 θ : Acceleration/deceleration inclination  
 T : Interpolation time  
 L : Interpolation distance

$$T = \frac{L}{\text{rapid}} + Ts$$

$$Td = Ts + (0 \sim 1.7 \text{ ms})$$

$$\theta = \tan^{-1} \left( \frac{\text{rapid}}{Ts} \right)$$

<When the interpolation distance is shorter than the acceleration and deceleration distance>



rapid : Rapid traverse rate  
 Ts : Acceleration/deceleration time constant  
 Td : Command deceleration check time  
 θ : Acceleration/deceleration inclination  
 T : Interpolation time  
 L : Interpolation distance

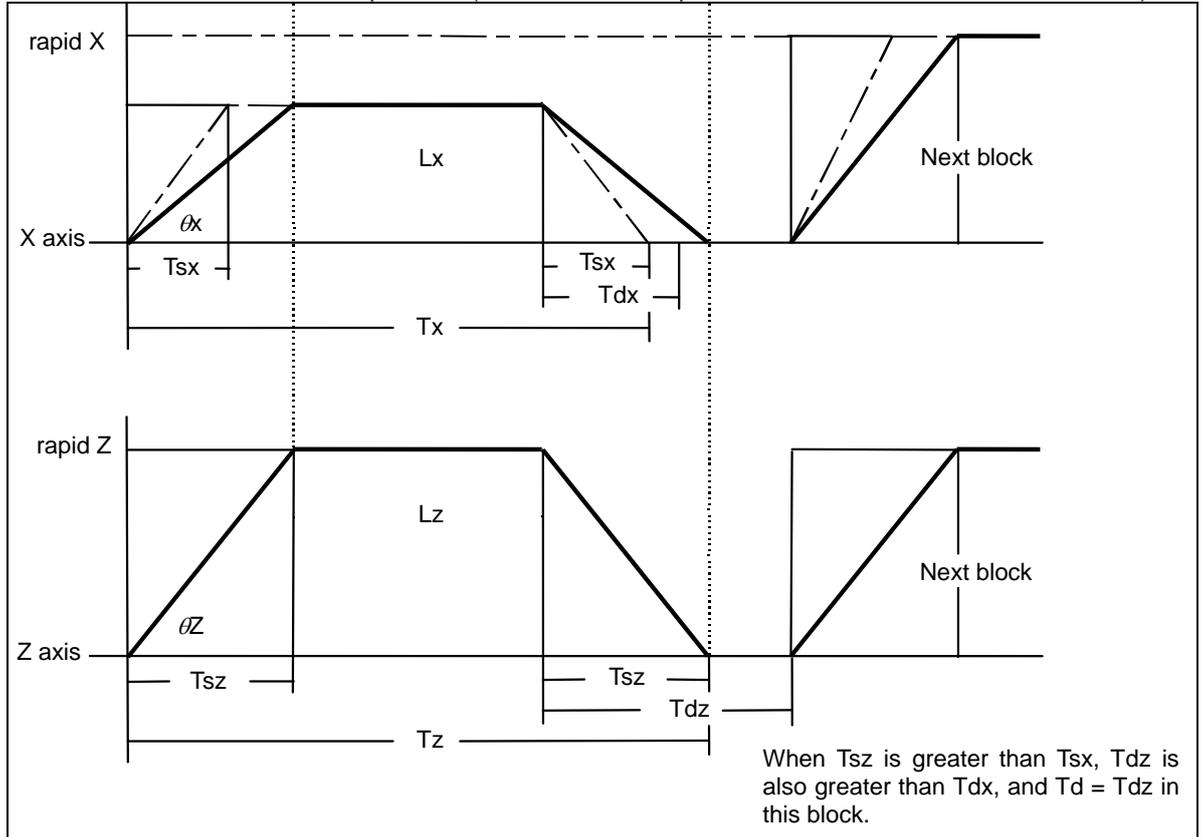
$$T = 2 \times \sqrt{Ts \times L / \text{rapid}}$$

$$Td = \frac{T}{2} + (0 \sim 1.7 \text{ ms})$$

$$\theta = \tan^{-1} \left( \frac{\text{rapid}}{Ts} \right)$$

- (3) When 2-axis simultaneous interpolation (linear interpolations) is performed during rapid traverse constant inclination acceleration and deceleration, the acceleration (deceleration) time is the longest value of the acceleration (deceleration) times determined for each axis by the rapid traverse rate of commands executed simultaneously, the rapid traverse acceleration and deceleration time constant, and the interpolation distance, respectively. Consequently, linear interpolation is performed even when the axes have different acceleration and deceleration time constants.

<2-axis simultaneous interpolation (When linear interpolation is used,  $T_{sx} < T_{sz}$ , and  $L_x \neq L_z$ )>

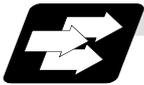


- (4) The program format of G0 (rapid traverse command) when rapid traverse constant inclination acceleration/deceleration are executed is the same as when this function is invalid (time constant acceleration/deceleration).
- (5) This function is valid only for G0 (rapid traverse).

## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration

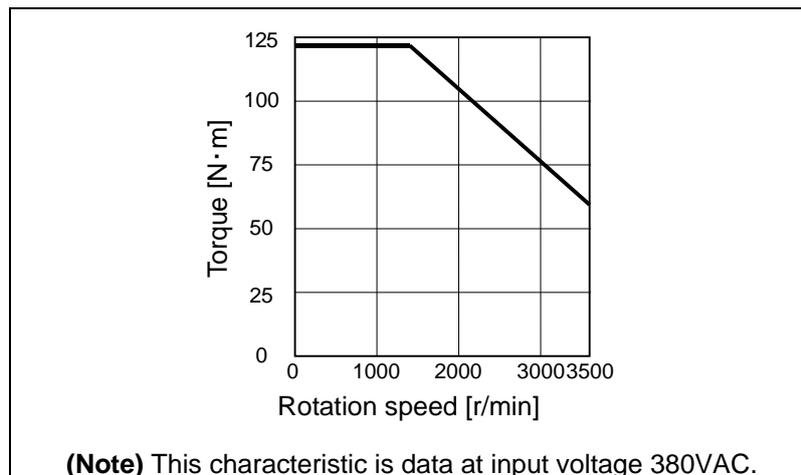
#### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration



##### Function and purpose

This function carries out the acceleration/deceleration according to the torque characteristic of the motor in the rapid traverse mode during automatic operation. (This function is not available in manual operation.) The rapid traverse constant inclination multi-step acceleration/deceleration method makes for improved cycle time because the positioning time is shortened by using the motor ability to its maximum.

In general, the servomotor has the characteristic that the torque falls in the high-speed rotation range.



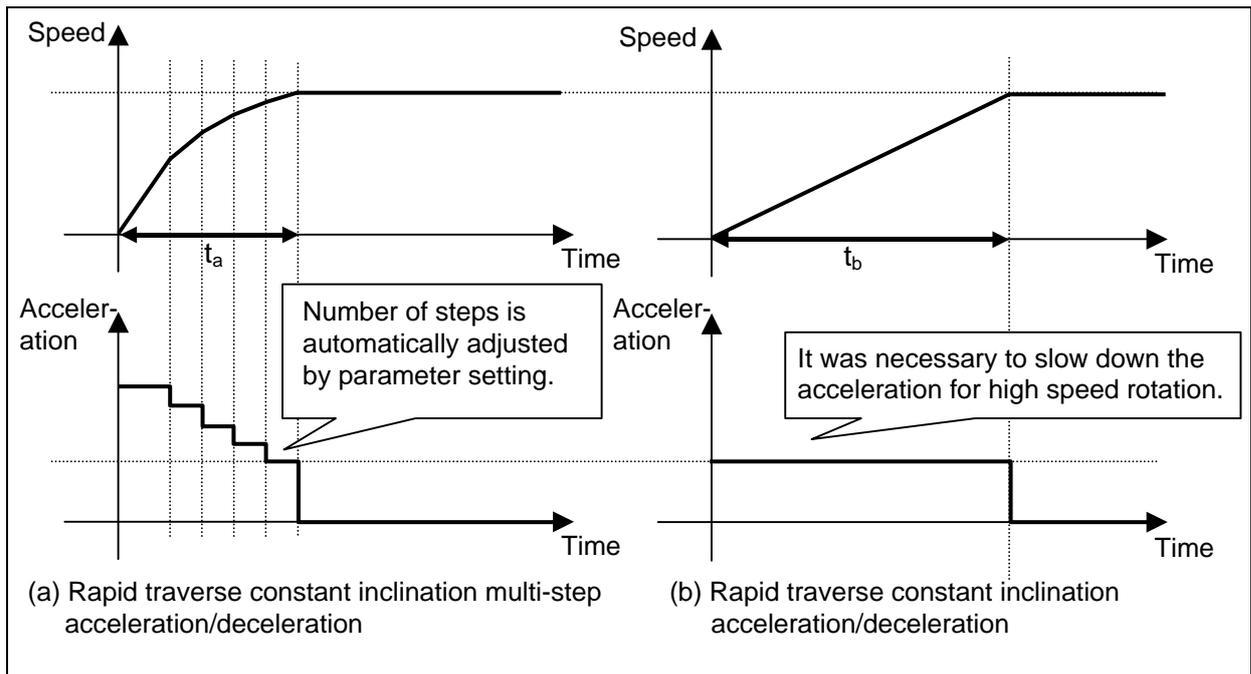
In the rapid traverse constant inclination acceleration/deceleration method, the acceleration has been treated constantly because this torque characteristic is not considered. So, It is necessary to use a minimum acceleration within the used speed range. Therefore, the margin of acceleration must be had in a low-speed range. Or if the acceleration is used to its maximum, the upper limit of the rotation speed must be slowed.

Then, to use the servomotor ability to its maximum, acceleration/deceleration to which the torque characteristic is considered is carried out by the rapid traverse constant inclination multi-step acceleration/deceleration method.

The acceleration/deceleration patterns in the case where rapid traverse constant inclination multi-step acceleration/deceleration are performed are as follows.

## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration



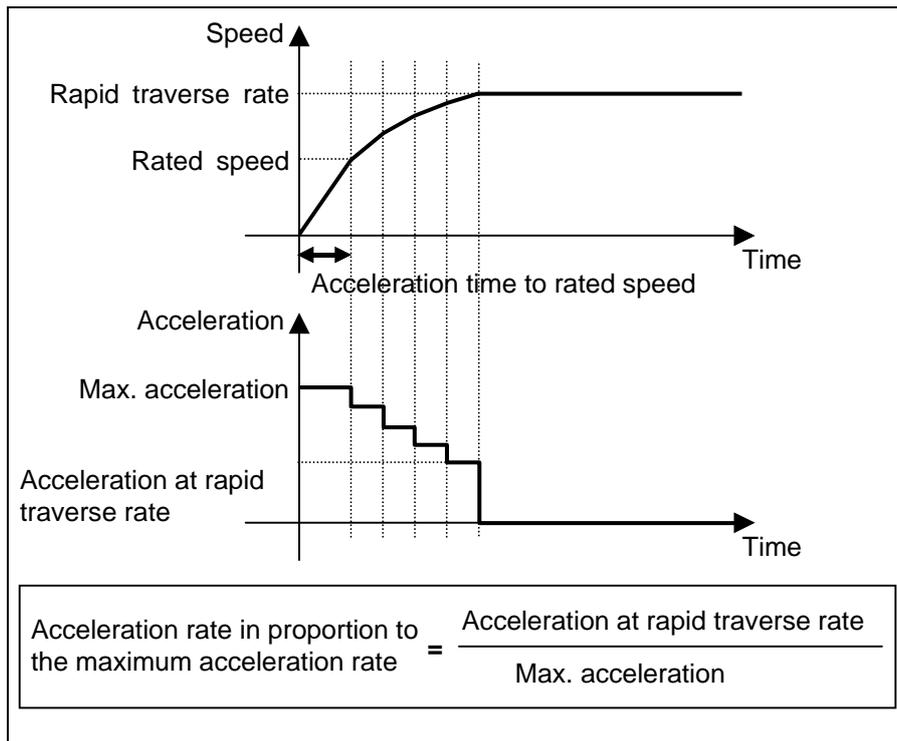
#### Detailed description

- (1) It is necessary to enable this function by set "2" to the parameter "#1205 G0bdcc". However, note the following conditions.
  - (a) "2" cannot be set to parameter "#1205 G0bdcc" besides the 1st part system. When "2" is set for besides 1st part system, "Y51 parameter error 17" will occur.
  - (b) When there is no specification for the rapid traverse constant inclination acceleration/deceleration, "2" cannot be set to parameter "#1205 G0bdcc". Even if the parameter is set to "2", this function is invalid. A normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.
  - (c) Even if "2" is set to "#1205 G0bdcc" when G00 non-interpolation type ("#1086 G00Intp" = "1"), this function is invalid. In this case, a normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.
- (2) To use this function, the following parameters must be set for each axis.
 

#2001 rapid	Rapid traverse [mm/min]
#2151 rated_spd	Rated speed [mm/min]
#2153 G0t_rated	Acceleration time to rated speed [ms]
#2152 acc_rate	Acceleration at rapid traverse in ratio to the maximum acceleration [%]

## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration



- (3) When either of the following conditions applies, this function is invalid and operates as "rapid traverse constant inclination acceleration/deceleration". For the axis which the rapid traverse constant inclination multi-step acceleration/deceleration is not necessary for, set "0" to "#2151 rated\_spd", "#2152 acc\_rate" and "#2153 G0t\_rated".
- (a) When "#2151 rated\_spd" (rated speed) is "0" or larger than "#2001 rapid" (rapid traverse)
  - (b) When "#2152 acc\_rate" (Acceleration rate in proportion to the maximum acceleration rate) is "0" or "100"
  - (c) Even if "2" is set to "#1205 G0bdcc" when G00 non-interpolation type ("#1086 G00Intp" = "1"), this function is invalid. In this case, a normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.

## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration

- (4) The comparison of the acceleration/deceleration patterns by the parameter setting is in the table below.

Mode	Rapid traverse constant inclination multi-step acceleration/deceleration	#1086 G00Intp	#1205 G0bdcc	Operation
G00 command	ON	0	0	Time constant acceleration/deceleration (interpolation type)
			1	Constant inclination acceleration/deceleration (acceleration/deceleration before interpolation)
			2	Constant inclination multi-step acceleration/deceleration
		1	Arbitrary	Time constant acceleration/deceleration (non-interpolation type)
	OFF	0	0	Time constant acceleration/deceleration (interpolation type)
			1	Constant inclination acceleration/deceleration (acceleration/deceleration before interpolation)
			2	Time constant acceleration/deceleration (interpolation type)
		1	Arbitrary	Time constant acceleration/deceleration (non-interpolation type)
Manual rapid traverse	Arbitrary	Arbitrary	Arbitrary	Time constant acceleration/deceleration (non-interpolation type)

## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration



#### Detailed description (decision method of steps)

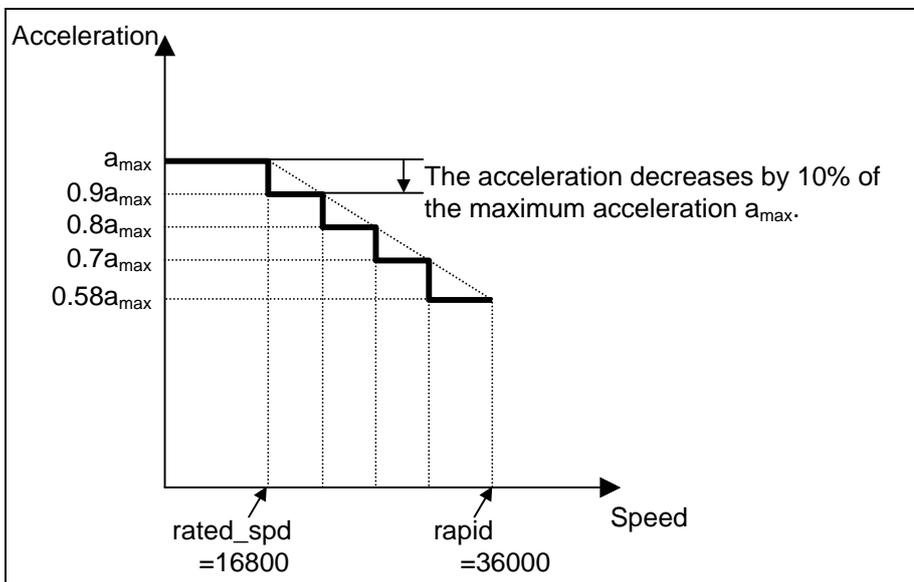
For rapid traverse constant inclination multi-step acceleration/deceleration, the number of steps is automatically adjusted by set parameter.

The acceleration per step is assumed to be a decrease by 10% of the maximum acceleration per step. Therefore, the number of steps is decided as follows.

$$\text{"Step"} = (100 - \text{"\#2152 acc\_rate"}) / 10 + 1 \text{ (Discard fractions less than 1)}$$

The acceleration/deceleration pattern when the parameter setting value is as follows is shown below.

No.	Item		Setting value
2001	rapid	Rapid traverse rate	36000 [mm/min]
2151	rated_spd	Rated speed	16800 [mm/min]
2152	acc_rate	Acceleration rate in proportion to the maximum acceleration rate	58 [%]



## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration

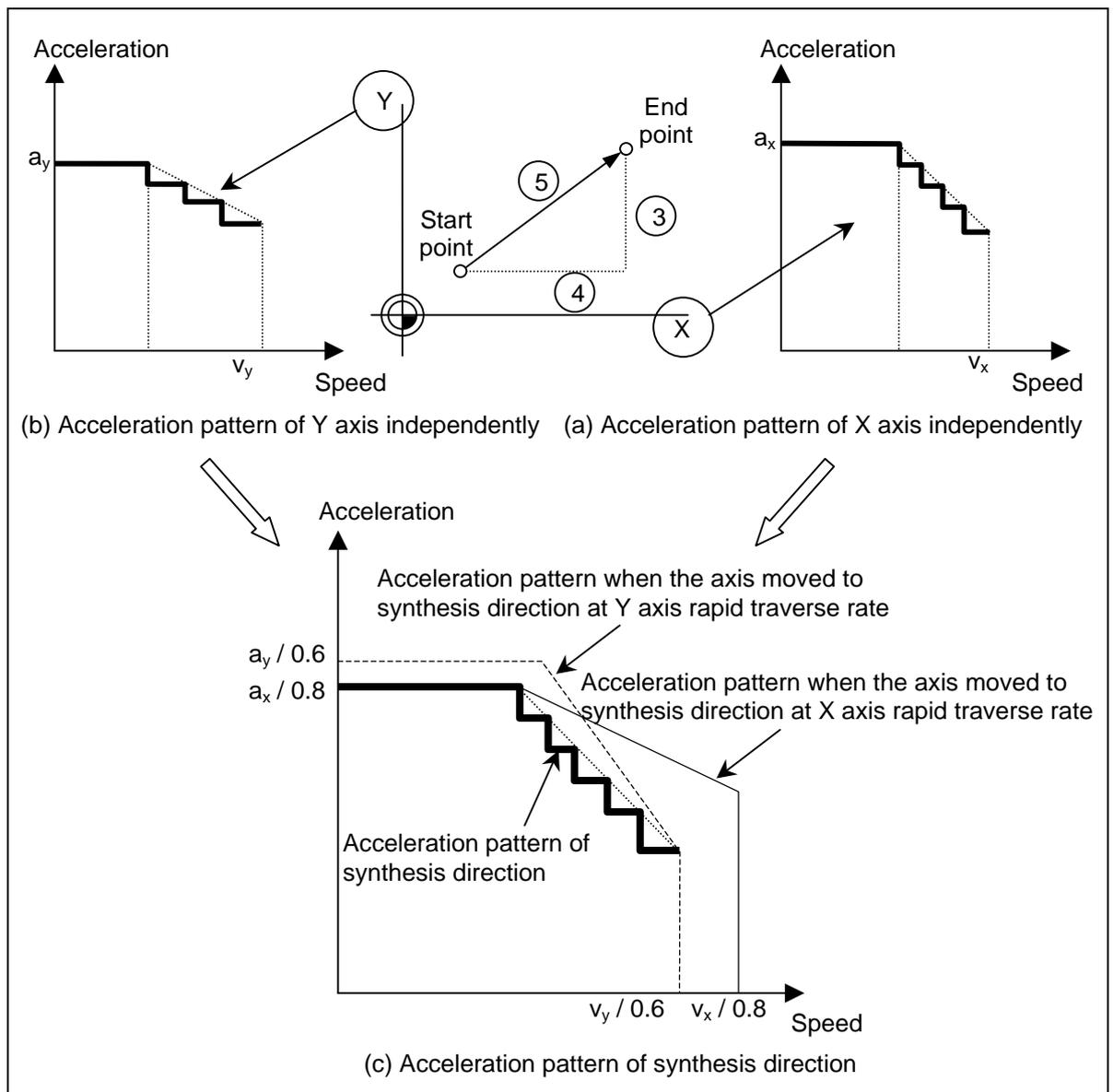


#### Detailed description (Acceleration pattern at two or more axis interpolation)

When there are two or more rapid traverse axes with a different acceleration pattern, there are the following two operation methods.

- Interpolation type (#1086 G0Intp = 0) : Moves from the start point to the end point by straight line
- Non-interpolation type (#1086 G0Intp = 1) : Each axis moves severally at the speed of the parameter

Rapid traverse constant inclination multi-step acceleration/deceleration are valid only for an interpolation type. For the interpolation type, the acceleration pattern operates to the maximum acceleration within the range where tolerable acceleration of each axis is not exceeded.



## 7. Feed Functions

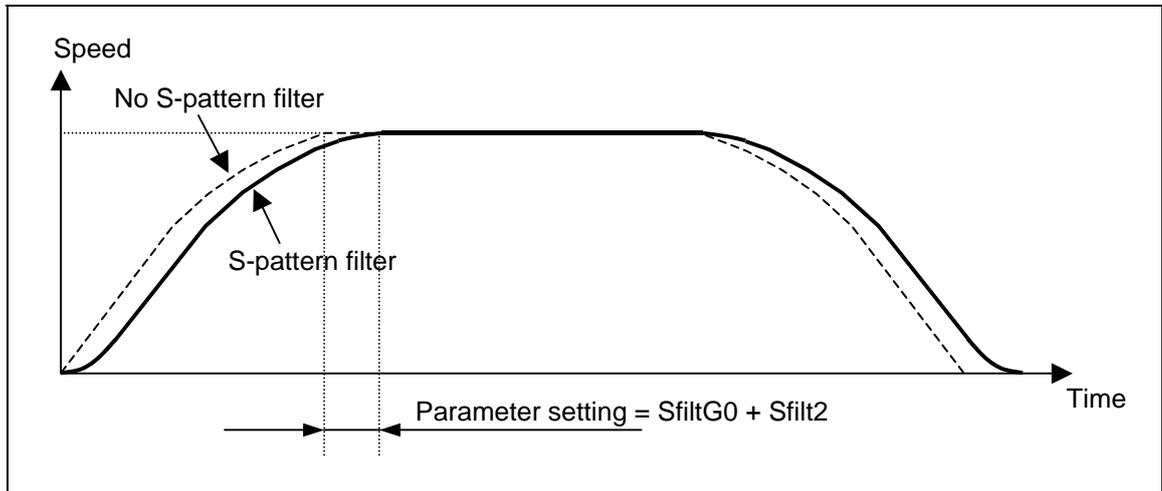
### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration



#### Detailed description (S-pattern filter control)

With S-pattern filter control, this enables the rapid traverse inclination multi-step acceleration/deceleration fluctuation to further smoothen.

This can be set in the range of 0 to 200 (ms) with the basic specification parameter "#1569 SfiltG0" (G00 soft acceleration/deceleration filter). With "#1570 Sfilt2" (Soft acceleration/deceleration filter 2), this also enables the acceleration/deceleration fluctuation to further smoothen.

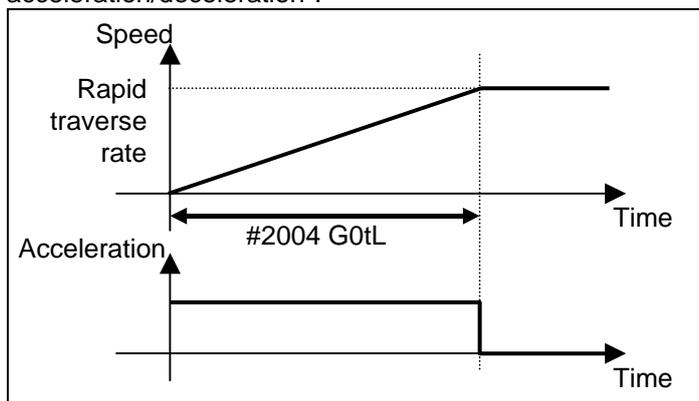


#### Detailed description (Rapid traverse rate for the high-accuracy control mode)

The high-accuracy control mode's rapid traverse rate ("#2109 Rapid (H-precision)") can be set besides rapid traverse rate ("#2001 rapid") during high-accuracy control, high-speed high-accuracy control I/II or high-accuracy spline control.

Operation when the value is set at the high-accuracy control mode's rapid traverse rate is as follows.

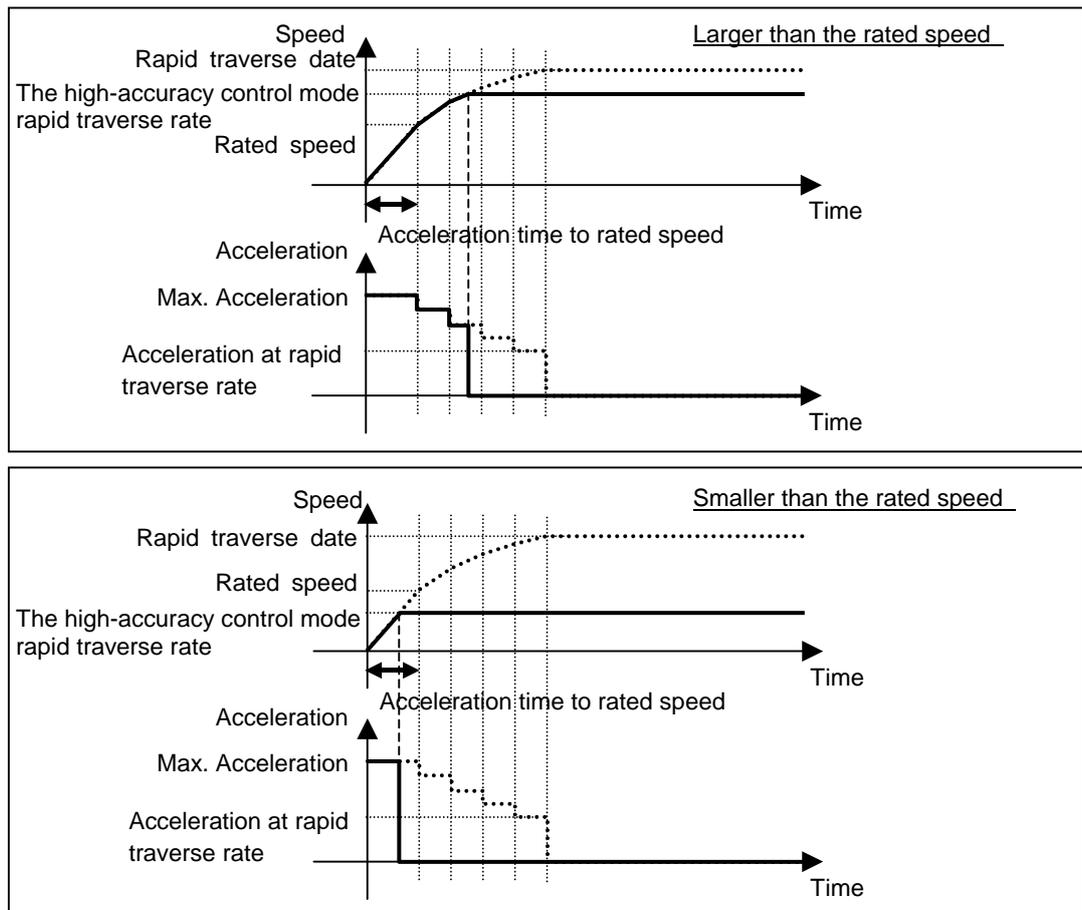
- (1) When "The high-accuracy control mode rapid traverse rate" > "rapid traverse rate"  
This function is invalid and operates as "rapid traverse constant inclination acceleration/deceleration".



- (2) When "The high-accuracy control mode rapid traverse rate" < "rapid traverse rate"  
"The high-accuracy control mode rapid traverse rate" is applied according to acceleration pattern calculated from acceleration rate to "rapid traverse", "rated speed", "G0 time constant to rated speed" and "maximum acceleration".

## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration

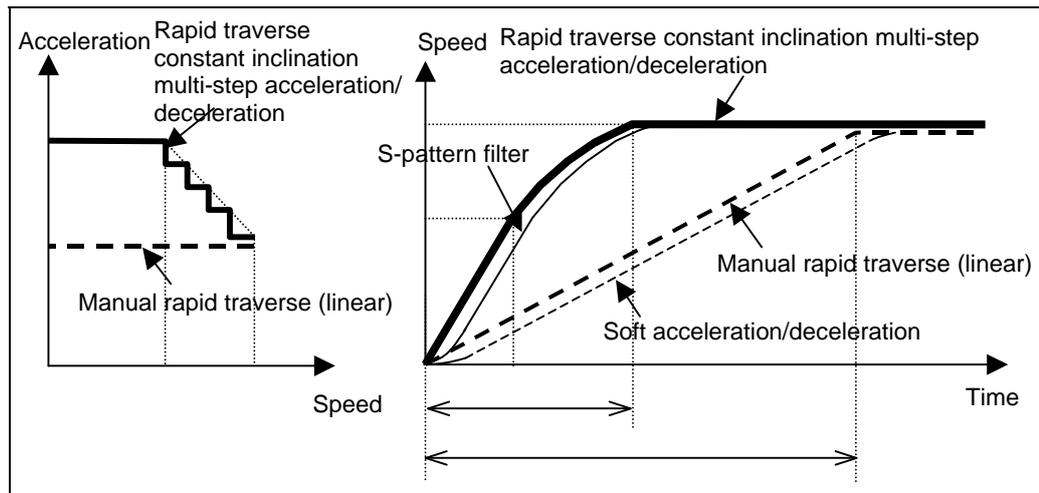


#### Precautions

- (1) Rapid traverse constant inclination multi-step acceleration/deceleration are valid only for a rapid traverse command. Note that when the manual rapid traverse, rapid traverse constant inclination multi-step acceleration/deceleration cannot be used. In this case, a time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied. So, acceleration/deceleration is decided by the following parameters.
- #2001 rapid : Rapid traverse rate
  - #2003 smgst : Acceleration/deceleration mode
  - #2004 G0tL : G0 time constant (linear)
  - #2005 G0t1 : G0 time constant (primary delay)
- The acceleration time (time constant) is different to the rapid traverse constant inclination multi-step acceleration/deceleration and the manual rapid traverse as shown in figure.

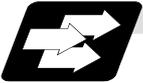
## 7. Feed Functions

### 7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration



- (2) Rapid traverse constant inclination multi-step acceleration/deceleration cannot be used in part system excluding 1st part system. However, even if two or more part system is used, it is possible to use this function in case of the 1st part system.
- (3) When there is no specification for the rapid traverse constant inclination acceleration/deceleration, this function is invalid even if "2" is set to parameter "#1205 G0bdcc". In this case, a normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.
- (4) When G00 non-interpolation type ("#1086 G00Intp" = "1"), rapid traverse constant inclination multi-step acceleration/deceleration cannot be used. It is valid at interpolation mode only.
- (5) When the rapid traverse constant inclination multi-step acceleration/deceleration is applied, rapid traverse acceleration/deceleration types ("#2003 smgst" bit0 to bit3) are ignored.
- (6) When the rapid traverse constant inclination multi-step acceleration/deceleration is valid, G0 constant inclination ("#1200 G0\_acc") cannot be used. Even if G0 constant inclination is valid ("#1200 G0\_acc" = 1), the setting is ignored.
- (7) When the rapid traverse constant inclination multi-step acceleration/deceleration is valid, programmable in-position check cannot be used. The in-position width will be ignored even if commanded.
- (8) This function cannot be used during the tool center point control.
- (9) For rapid traverse constant inclination multi-step acceleration/deceleration, feedforward control is invalid.

## 7.9 Exact Stop Check; G09



## Function and purpose

In order to prevent roundness during corner cutting and machine shock when the tool feedrate changes suddenly, there are times when it is desirable to start the commands in the following block once the in-position state after the machine has decelerated and stopped or the elapsing of the deceleration check time has been checked. The exact stop check function is designed to accomplish this purpose.

Either the deceleration check time or in-position state is selected with parameter "#1193 inpos". In-position check is valid when "#1193 inpos" is set to 1.

The in-position width is set with parameter "#2224 SV024" on the servo parameter screen by the machine manufacturer.



## Command format

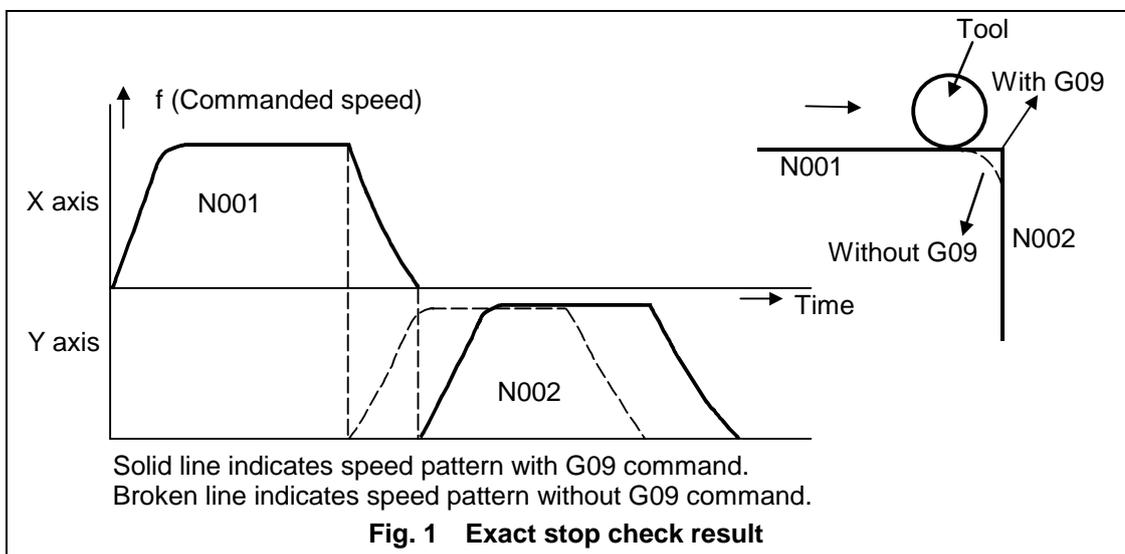
G09 ;

The exact stop check command G09 has an effect only with the cutting command (G01 - G03) in its particular block.



## Example of program

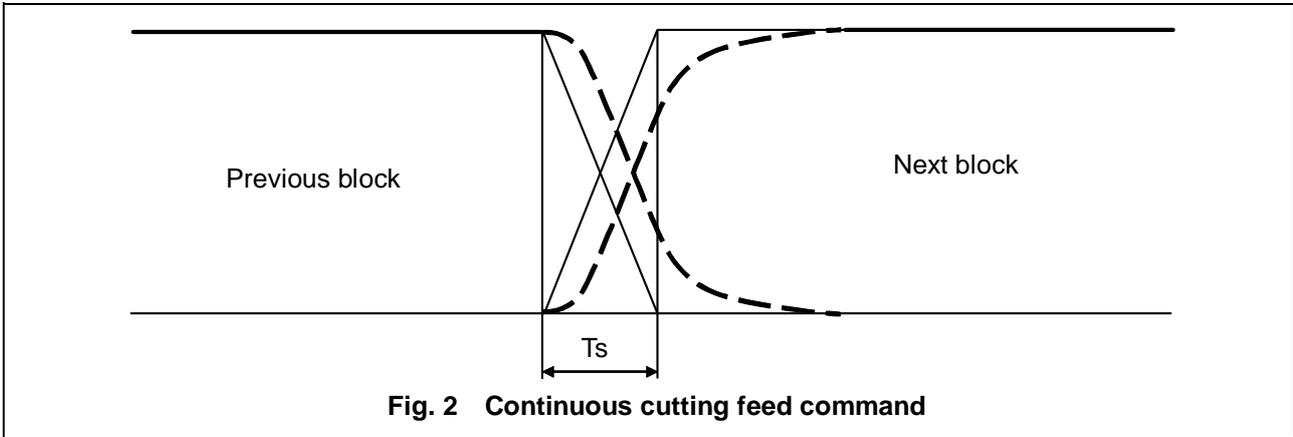
N001 G09 G01 X100.000 F150 ;	The following block is started once the deceleration check time or in-position state has been checked after the machine has decelerated and stopped.
N002 Y100.000 ;	



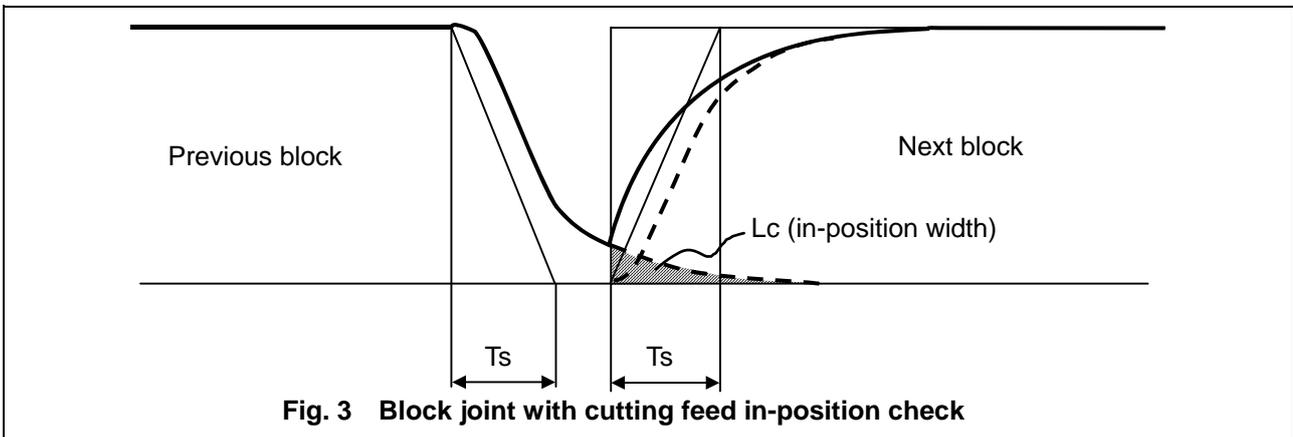


Detailed description

(1) With continuous cutting feed



(2) With cutting feed in-position check

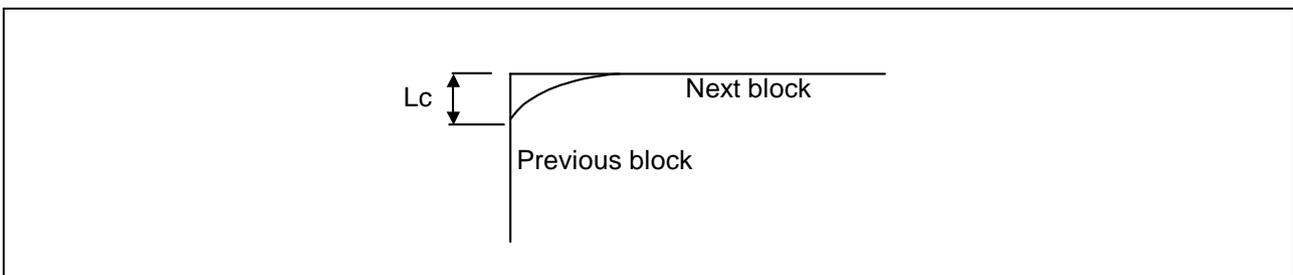


In Figs. 2 and 3:

- Ts = Cutting feed acceleration/deceleration time constant
- Lc = In-position width

As shown in Fig. 3, the remaining distance (shaded area in Fig. 3) of the previous block when the next block is started can be set into the servo parameter "#2224 SV024" as the in-position width "Lc".

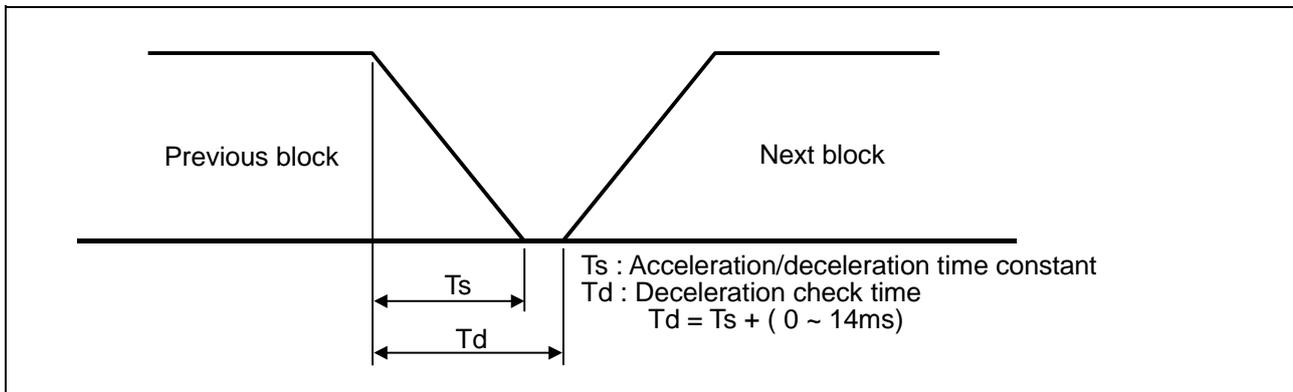
The in-position width is designed to reduce the roundness at the workpiece corners to below the constant value.



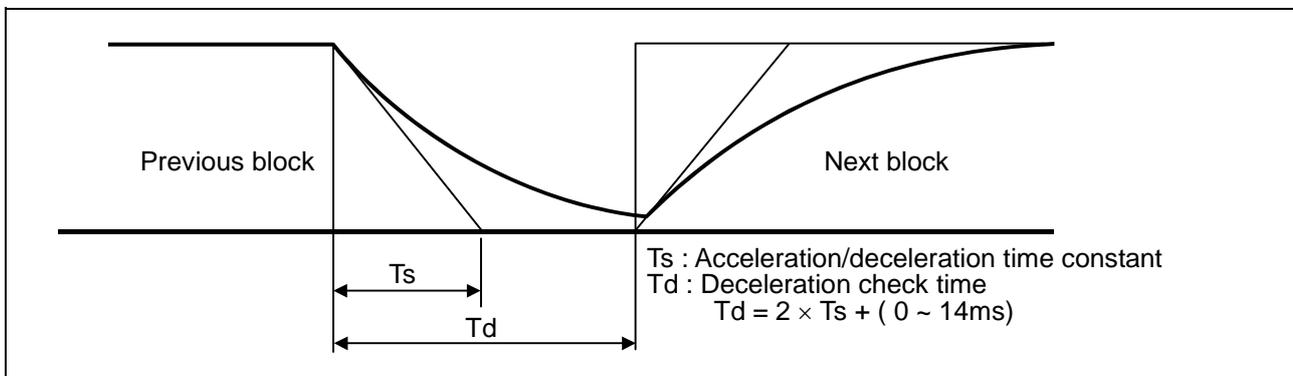
To eliminate corner roundness, set the value as small as possible to servo parameter "#2224 SV024" and perform an in-position check or assign the dwell command (G04) between blocks.

## (3) With deceleration check

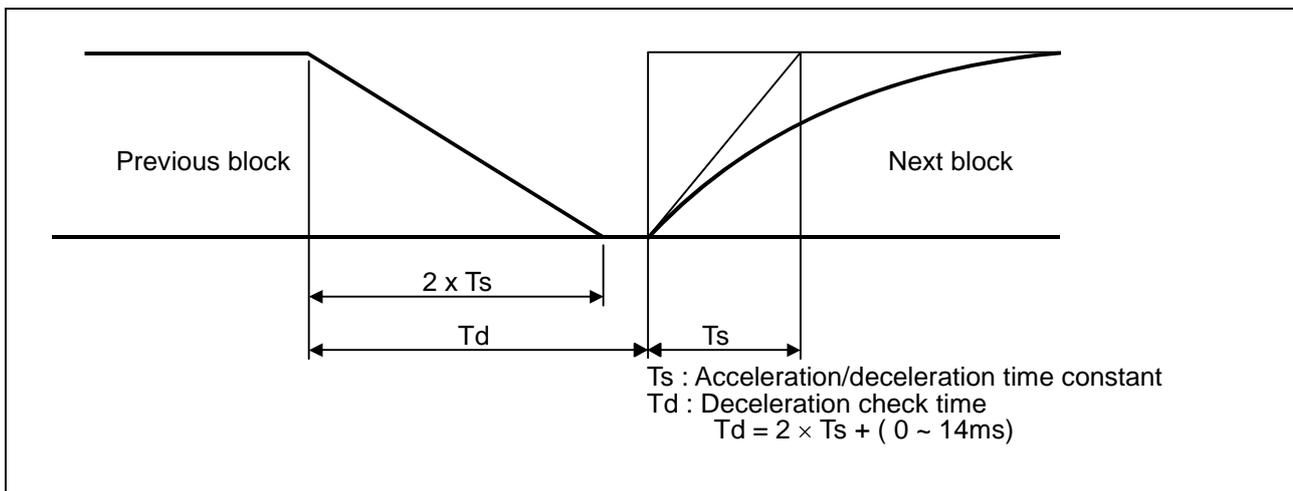
## (a) With linear acceleration/deceleration



## (b) With exponential acceleration/deceleration



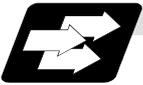
## (c) With exponential acceleration/linear deceleration



The time required for the deceleration check during cutting feed is the longest among the cutting feed deceleration check times of each axis determined by the cutting feed acceleration/deceleration time constants and by the cutting feed acceleration/deceleration mode of the axes commanded simultaneously.

**(Note 1)** To execute exact stop check in a fixed cycle cutting block, insert command G09 into the fixed cycle subprogram.

7.10 Exact Stop Check Mode; G61



Function and purpose

Whereas the G09 exact stop check command checks the in-position status only for the block in which the command has been assigned, the G61 command functions as a modal. This means that deceleration will apply at the end points of each block to all the cutting commands (G01 to G03) subsequent to G61 and that the in-position status will be checked. G61 is released by high-accuracy control (G61.1), automatic corner override (G62), tapping mode (G63), or cutting mode (G64).



Command format

```
G61 ;
```

In-position check is executed in the G61 block, and thereafter, the in-position check is executed at the end of the cutting command block is executed until the check mode is canceled.

7.11 Deceleration Check



Function and purpose

The deceleration check is a function that determines the method of the check at the completion of the axis movement block's movement.

The deceleration check includes the in-position check and commanded speed check method.

The G0 and G1 deceleration check method combination can be selected.

(Refer to section "Deceleration check combination".)

With this function, the deceleration check in the reverse direction of G1 → G0 or G1 → G1 can be changed by changing the parameter setting.

(1) Types of deceleration check

<p>Commanded speed check</p> <p>With the commanded speed check, the completion of deceleration is judged when the command to the motor is completed</p>	
<p>In-position check</p> <p>With the in-position check, the completion of deceleration is judged when the motor moves to the in-position width designated with the parameter.</p>	

**(2) Designating deceleration check**

The deceleration check by designating a parameter includes "deceleration check specification type 1" and "deceleration check specification type 2". The setting is selected with the parameter "#1306 InpsTyp".

**(a) Deceleration check specification type 1 ("#1306 InpsTyp" = 0)**

The G0 and G1 deceleration check method can be selected with the base specification parameter deceleration check method 1 (#1193 inpos) and "deceleration check method 2" (#1223 aux07/bit1).

Parameter	Rapid traverse command	Parameter	Other than rapid traverse command (G1 : other than G0 command)	
<b>inpos (#1193)</b>	G0→XX (G0+G9→XX)	<b>AUX07/BIT-1 (#1223/BIT-1)</b>	G1+G9→XX	G0→XX
<b>0</b>	Command deceleration check	<b>0</b>	Command deceleration check	No deceleration check
<b>1</b>	In-position check	<b>1</b>	In-position check	

**(Note 1)** XX expresses all commands.

**(Note 2)** "#1223 aux07" is the part system common parameter.

**(b) Deceleration check specification type 2 ("#1306 InpsTyp" = 1)**

Rapid traverse and cutting in-position are designated with the "#1193 inpos" parameter.

Parameter	Command block		
#1193 Inpos	G0	G1+G9	G1
<b>0</b>	Command deceleration check	Command deceleration check	No deceleration check
<b>1</b>	In-position check	In-position check	No deceleration check

**(Note 1)** "#1193 inpos" is the parameter per part system.

**(Note 2)** "G0" means the rapid traverse, and "G1" means the cutting feed.

## 7.11.1 G1 → G0 Deceleration Check



## Detailed operations

- (1) In G1 → G0 continuous blocks, the parameter "#1502 G0lpfg" can be changed to change the deceleration check in the reverse direction.

	Same direction	Reverse direction
G0lpfg: 0		
G0lpfg: 1		



## Example of program

When there is a deceleration check in the movement of several axes:

- |     |   |  |
|-----|---|--|
| (1) | G91 G1 X100. Y100. F4000 ;<br>G0 X-100. Y120. ; | A deceleration check is carried out, because the X axis moves in the reverse direction in the program above.   |
| (2) | G91 G1 X100. Y-100. F4000 ;<br>G0 X80. Y100. ;  | A deceleration check is carried out, because the Y axis moves in the reverse direction in the program above.   |
| (3) | G90 G1 X100. Y100. F4000 ;<br>G0 X80. Y120. ;   | A deceleration check is carried out, because the X axis moves in the reverse direction in the program above.<br>(When the program start position is X0 Y0) |
| (4) | G91 G1 X100. Y100. F4000 ;<br>G0 X100. Y100. ;  | A deceleration check is not carried out, because both the X axis and the Y axis move in the same direction in the program above.                           |
| (5) | G91 G1 X100. Y80. F4000 ;<br>G0 X80. ;          | A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Y axis movement command in the program above.     |

## 7.11.2 G1 → G1 Deceleration Check



## Detailed operations

- (1) In G1 → G1 continuous blocks, the parameter "#1503 G1lpg" can be changed to change the deceleration check of the reverse direction.

	Same direction	Reverse direction
G1lpg: 0		
G1lpg: 1		



## Example of program

When there is a deceleration check in the movement of several axes:

- |   |  |
|---|--|
| (1) G91 G1 X100. Y100. F4000 ;<br>G1 X-100. Y120. ; | A deceleration check is carried out, because the X axis moves in the reverse direction in the program above.   |
| (2) G91 G1 X100. Y-100. F4000 ;<br>G1 X80. Y100. ;  | A deceleration check is carried out, because the Y axis moves in the reverse direction in the program above.   |
| (3) G90 G1 X100. Y100. F4000 ;<br>G1 X80. Y120. ;   | A deceleration check is carried out, because the X axis moves in the reverse direction in the program above.<br>(When the program start position is X0 Y0) |
| (4) G91 G1 X100. Y100. F4000 ;<br>G1 X100. Y100. ;  | A deceleration check is not carried out, because both the X axis and the Y axis move in the same direction in the program above.                           |
| (5) G91 G1 X100. Y80. F4000 ;<br>G1 X80. ;          | A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Y axis movement command in the program above.     |

## 7.12 Automatic Corner Override



## Function and purpose

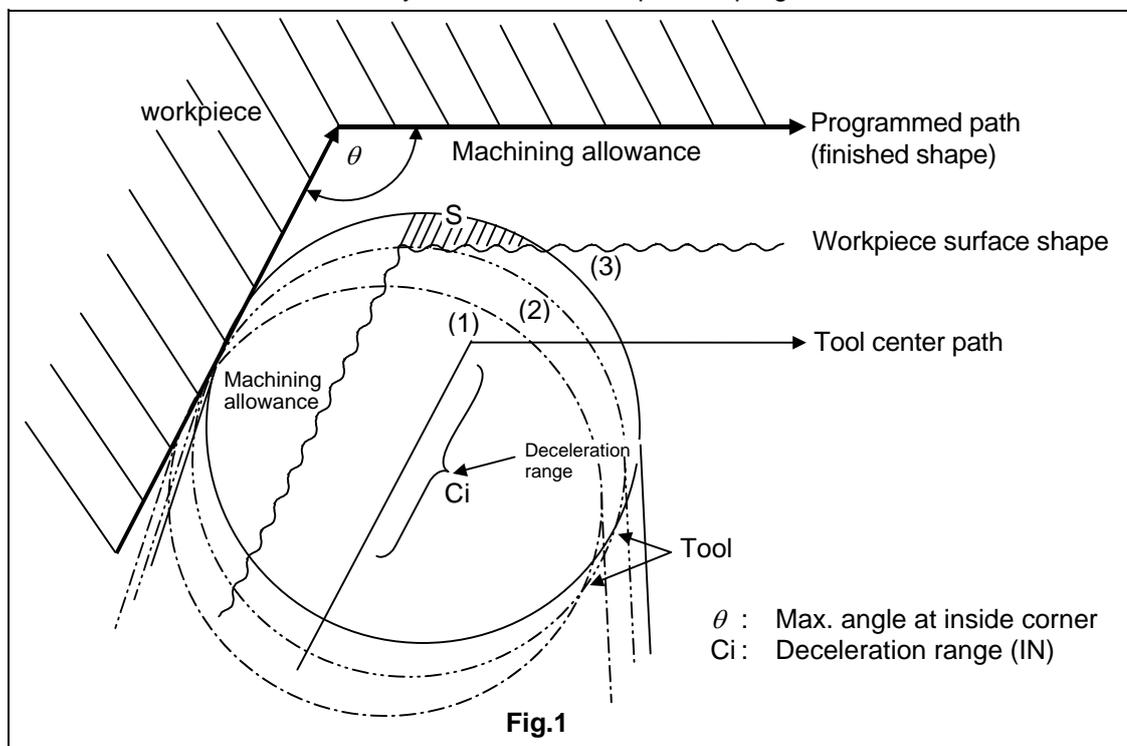
With tool radius compensation, this function reduces the load during inside cutting of automatic corner R, or during inside corner cutting, by automatically applying override to the feed rate. Automatic corner override is valid until the tool radius compensation cancel (G40), exact stop check mode (G61), high-accuracy control mode (G61.1), tapping mode (G63), or cutting mode (G64) command is issued. This function is always valid in the tool radius compensation (G41/G42).



## Machining inside corners

When cutting an inside corner as in Fig. 1, the machining allowance amount increases and a greater load is applied to the tool. To remedy this, override is applied automatically within the corner set range, the feedrate is reduced, the increase in the load is reduced and cutting is performed effectively.

However, this function is valid only when finished shapes are programmed.



**(1) Operation**

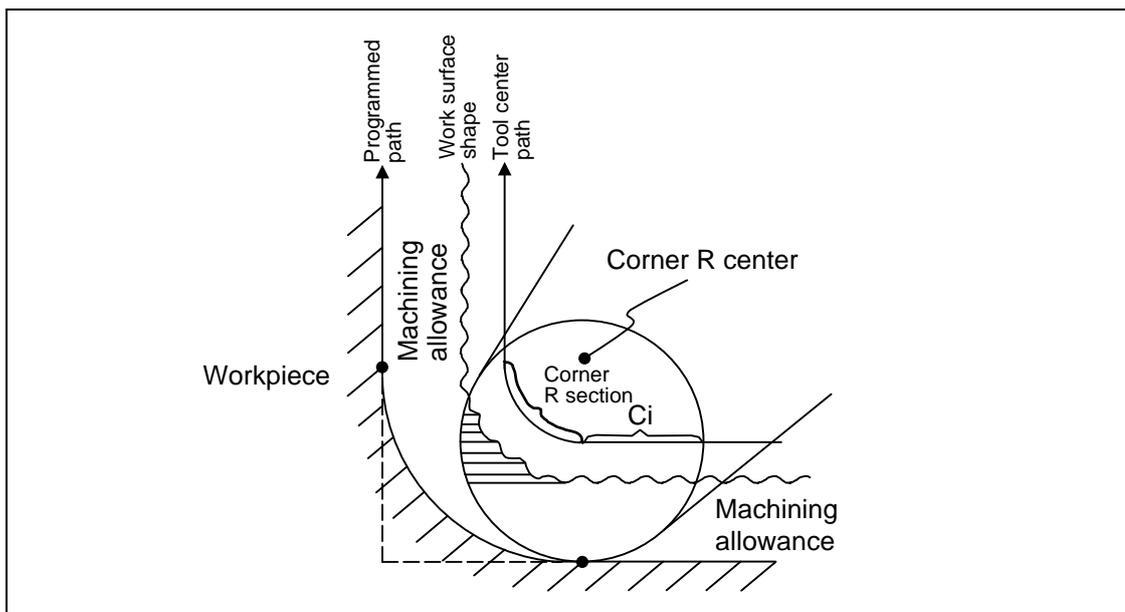
- (a) When automatic corner override is not to be applied :  
When the tool moves in the order of (1) → (2) → (3) in Fig. 1, the machining allowance at (3) increases by an amount equivalent to the area of shaded section S and so the tool load increases.
- (b) When automatic corner override is to be applied :  
When the inside corner angle  $\theta$  in Fig. 1 is less than the angle set in the parameter, the override set into the parameter is automatically applied in the deceleration range  $C_i$ .

**(2) Parameter setting**

The following parameters are set into the machining parameters :

#	Parameter	Setting range
#8007	OVERRIDE	0 to 100%
#8008	MAX ANGLE	0 to 180°
#8009	DSC. ZONE	0 to 99999.999mm or 0 to 3937.000 inches

Refer to the Instruction Manual for details on the setting method.

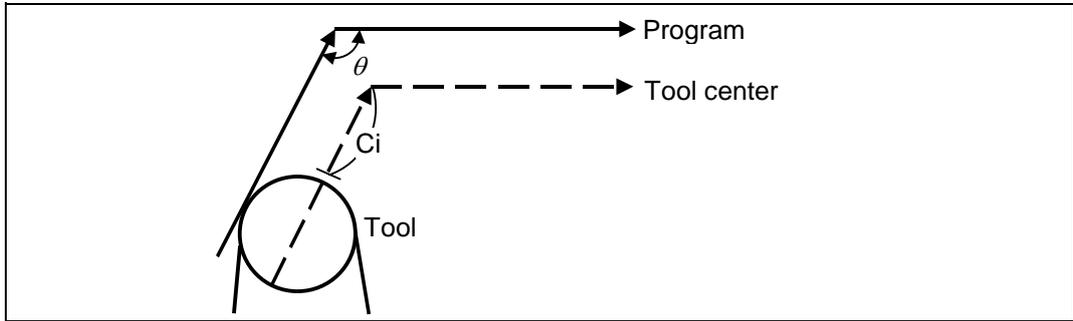
**Automatic corner R**

- (1) The override set in the parameter is automatically applied at the deceleration range  $C_i$  and corner R section for inside offset with automatic corner R. (There is no angle check.)



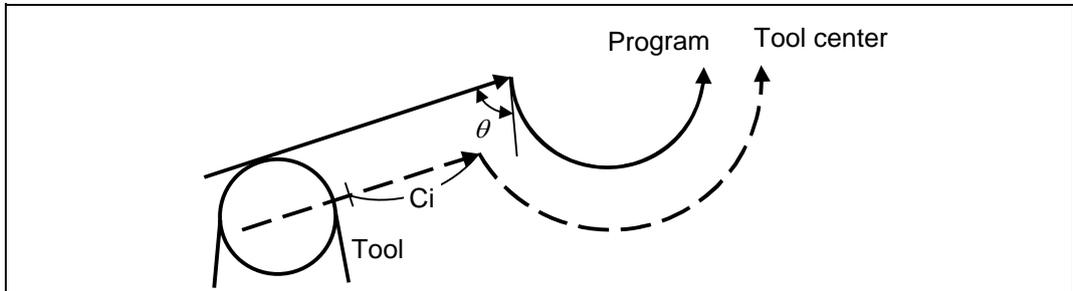
Example of operations

(1) Line - line corner



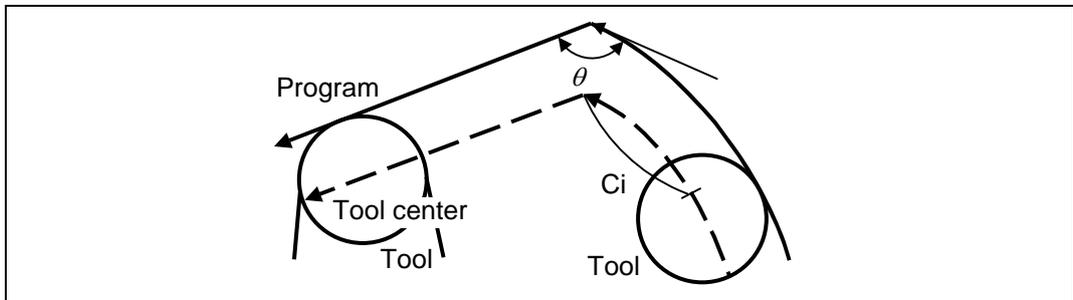
The override set in the parameter is applied at Ci.

(2) Line - arc (outside) corner



The override set in the parameter is applied at Ci.

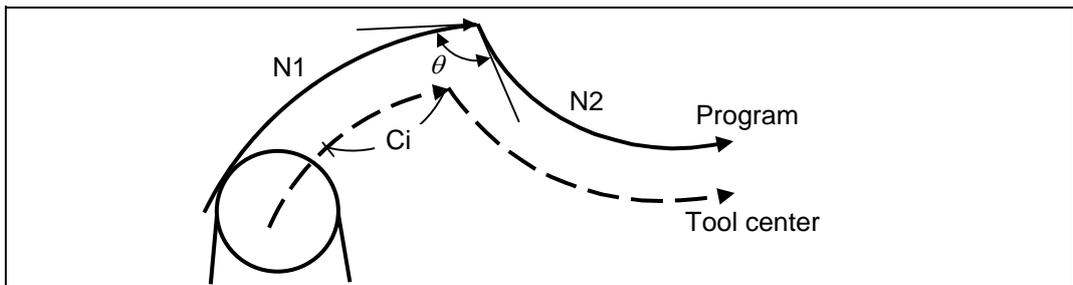
(3) Arc (inside compensation) - line corner



The override set in the parameter is applied at Ci.

(Note) The deceleration range Ci where the override is applied is the length of the arc with an arc command.

(4) Arc (inside compensation) - arc (outside compensation) corner



The override set in the parameter is applied at Ci.



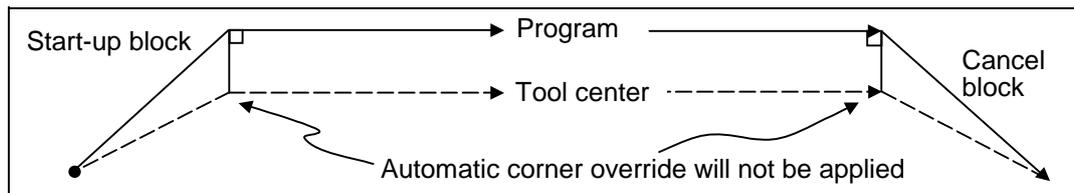
## Relation with other functions

Function	Override at corner
Cutting feed override	Automatic corner override is applied after cutting feed override has been applied.
Override cancel	Automatic corner override is not canceled by override cancel.
Speed clamp	Valid after automatic corner override
Dry run	Automatic corner override is invalid.
Synchronous feed	Automatic corner override is applied to the synchronous feedrate.
Thread cutting	Automatic corner override is invalid.
G31 skip	Program error results with G31 command during tool radius compensation.
Machine lock	Valid
Machine lock high speed	Automatic corner override is invalid.
G00	Invalid
G01	Valid
G02, G03	Valid

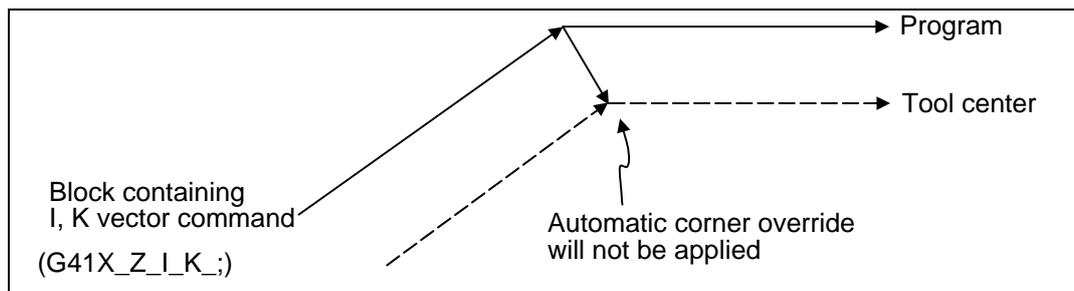


## Precautions

- (1) Automatic corner override is valid only in the G01, G02, and G03 modes; it is not effective in the G00 mode. When switching from the G00 mode to the G01 (or G02 or G03) mode at a corner (or vice versa), automatic corner override will not be applied at that corner in the G00 block.
- (2) The automatic corner override will not be applied until the tool radius compensation mode is entered.
- (3) Automatic corner override will not be applied on a corner where the tool radius compensation is started or canceled.



- (4) Automatic corner override will not be applied on a corner where the tool radius compensation I, K vector command is issued.



- (5) Automatic corner override will not be applied when intersection calculation cannot be executed.  
Intersection calculation cannot be executed in the following case.
  - (a) When the movement command block does not continue for four or more times.
- (6) The deceleration range with an arc command is the length of the arc.
- (7) The inside corner angle, as set by parameter, is the angle on the programmed path.
- (8) Automatic corner override will not be applied when the maximum angle in the parameter is set to 0 or 180.
- (9) Automatic corner override will not be applied when the override in the parameter is set to 0 or 100.

### 7.13 Tapping Mode; G63



#### Function and purpose

The G63 command allows the control mode best suited for tapping to be entered, as indicated below :

- (1) Cutting override is fixed at 100%.
- (2) Deceleration commands at joints between blocks are invalid.
- (3) Feed hold is invalid.
- (4) Single block is invalid.
- (5) In-tapping mode signal is output.

G63 is released by the exact stop check mode (G61), high-accuracy control mode (G61.1), or cutting mode (G64).



#### Command format

G63 ;

### 7.14 Cutting Mode; G64



#### Function and purpose

The G64 command allows the cutting mode in which smooth cutting surfaces are obtained to be established. Unlike the exact stop check mode (G61), the next block is executed continuously with the machine not decelerating and stopping between cutting feed blocks in this mode.

G64 is released by the exact stop mode (G61), high-accuracy control mode (G61.1), or tapping mode (G63).

This cutting mode is established in the initialized status.



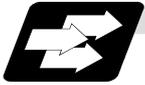
#### Command format

G64 ;

## 8. Dwell

The G04 command can delay the start of the next block.

### 8.1 Per-second Dwell ; G04



#### Function and purpose

The machine movement is temporarily stopped by the program command to make the waiting time state. Therefore, the start of the next block can be delayed. The waiting time state can be canceled by inputting the skip signal.



#### Command format

**G04 X\_\_ ; or G04 P\_\_ ;**  
X, P..... Dwell time

The input command unit for the dwell time depends on the parameter.

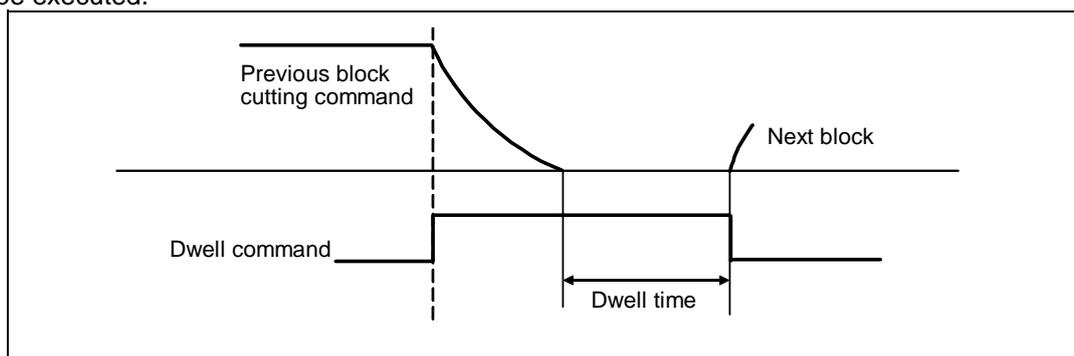


#### Detailed description

- (1) When designating the dwell time with X, the decimal point command is valid.
- (2) When designating the dwell time with P, the availability of the decimal point command can be selected with the parameter (#8112). When the decimal point command is invalid in the parameter setting, the command below the decimal point issued with P is ignored.
- (3) When the decimal point command is valid or invalid, the dwell time command range is as follows.

Command range when the decimal point command is valid	Command range when the decimal point command is invalid
0 ~ 99999.999 (s)	0 ~ 99999999 (ms)

- (4) The dwell time setting unit applied when there is no decimal point can be made 1s by setting 1 in the parameter #1078 Decpt2. This is effect only for X and P for which the decimal command is valid.
- (5) When a cutting command is in the previous block, the dwell command starts calculating the dwell time after the machine has decelerated and stopped. When it is commanded in the same block as an M, S, T or B command, the calculation starts simultaneously.
- (6) The dwell is valid during the interlock.
- (7) The dwell is valid even for the machine lock.
- (8) The dwell can be canceled by setting the parameter #1173 dwlskp beforehand. If the set skip signal is input during the dwell time, the remaining time is discarded, and the following block will be executed.





## Example of program

Command	Dwell time [sec]			
	#1078 Decpt2 = 0		#1078 Decpt2 = 1	
	DECIMAL PNT-N	DECIMAL PNT-P	DECIMAL PNT-N	DECIMAL PNT-P
G04 X500 ;	0.5		500	
G04 X5000 ;	5		5000	
G04 X5. ;	5		5	
G04 X#100 ;	1000		1000	
G04 P5000 ;	5		5	5000
G04 P12.345 ;	0.012	12.345	0.012	12.345
G04 P#100 ;	1	1000	1	1000

**(Note 1)** The above examples are the results under the following conditions.

- Input setting unit 0.001mm or 0.0001inch
- #100 = 1000 ;

**(Note 2)** "DECIMAL PNT-P" is a control parameter (#8112).

**(Note 3)** If the input setting unit is 0.0001inch, the X before G04 will be multiplied by 10. For example for "X5. G04 ;", the dwell time will be 50 sec.

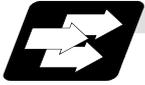


## Precautions and restrictions

- (1) When using this function, command X after G04 in order to make sure that the dwell is based on X.

## 9. Miscellaneous Functions

### 9.1 Miscellaneous Functions (M8-digits BCD)



#### Function and purpose

The miscellaneous (M) functions are also known as auxiliary functions, and they include such numerically controlled machine functions as spindle forward and reverse rotation, operation stop and coolant ON/OFF.

These functions are designated by an 8-digit number (0 to 99999999) following the address M with this controller, and up to 4 groups can be commanded in a single block.

**(Example)** G00 Xx Mm<sub>1</sub> Mm<sub>2</sub> Mm<sub>3</sub> Mm<sub>4</sub> ;

When five or more commands are issued, only the last four will be valid.

The output signal is an 8-digit BCD code and start signal.

The four commands of M00, M01, M02, and M30 are used as auxiliary commands for specific objectives and so they cannot be used as general auxiliary commands. This therefore leaves 96 miscellaneous functions which are usable as such commands.

Reference should be made to the instructions issued by the machine manufacturer for the actual correspondence between the functions and numerical values.

When the M00, M01, M02, and M30 functions are used, the next block is not read into the pre-read buffer due to pre-read inhibiting.

If the M function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. Which of these sequences actually applies depends on the machine specifications.

(1) The M function is executed after the movement command.

(2) The M function is executed at the same time as the movement command.

Processing and completion sequences are required in each case for all M commands.

The four M functions used for specific purposes will now be described.



#### Program stop : M00

When the tape reader has read this function, it stops reading the next block. As far as the NC system's functions are concerned, only the tape reading is stopped. Whether such machine functions as the spindle rotation and coolant supply are stopped or not differs according to the machine in question.

Re-start is enabled by pressing the automatic start button on the machine operation board.

Whether resetting can be initiated by M00 depends on the machine specifications.

**Optional stop ; M01**

If the tape reader reads the M01 command when the optional stop switch on the machine operation board is ON, it will stop and the same effect as with the M00 function will apply. If the optional stop switch is OFF, the M01 command is ignored.

**(Example)**

:	Optional stop switch status and operation
N10 G00 X1000 ;	• Stops at N11 when switch is ON
N11 M01 ;	• Next command (N12) is executed without stopping at N11 when switch is OFF
N12 G01 X2000 Z3000 F600 ;	
:	

**Program end ; M02 or M30**

This command is normally used in the final block for completing the machining, and so it is primarily used for tape rewinding. Whether the tape is actually rewound or not depends on the machine specifications.

Depending on the machine specifications, the system is reset by the M02 or M30 command upon completion of tape rewinding and any other commands issued in the same block.

(Although the contents of the command position display counter are not cleared by this reset action, the modal commands and compensation amounts are canceled.)

The next operation stops when the rewinding operation is completed (the in-automatic operation lamp goes off). To restart the unit, the automatic start button must be pressed or similar steps must be taken.

When the program is restarted after M02 and M30 are completed, if the first movement command is designated only with a coordinate word, the interpolation mode will function when the program ends. It is recommended that a G function always be designated for the movement command designated first.

**(Note 1)** Independent signals are also output respectively for the M00, M01, M02 and M30 commands and these outputs are each reset by pressing the reset key.

**(Note 2)** M02 or M30 can be assigned by manual data input (MDI). At this time, commands can be issued simultaneously with other commands just as with the tape.

**Internal processing with M00/M01/M02/M30 commands**

Internal processing suspends pre-reading when the M00, M01, M02 or M30 command has been read. Other tape rewinding operations and the initialization of modals by resetting differ according to the machine specifications.

9.2 Secondary Miscellaneous Functions (B8-digits, A8 or C8-digits)



Function and purpose

These serve to assign the indexing table positioning and other such functions. In this controller, they are assigned by an 8-digit number from 0 to 99999999 following address A, B or C. The machine maker determines which codes correspond to which positions.

If the A, B or C function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. The machine specifications determine which sequence applies.

- (1) The A, B or C function is executed after the movement command.
- (2) The A, B or C function is executed simultaneously with the movement command.

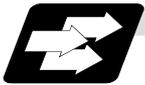
Processing and completion sequences are required for all secondary miscellaneous functions. The table below given the various address combinations. It is not possible to use an address which is the same for the axis name of an additional axis and secondary miscellaneous function.

Additional axis name	A	B	C
2nd miscellaneous function			
A	×	○	○
B	○	×	○
C	○	○	×

**(Note)** When A has been assigned as the secondary miscellaneous function address, the following commands cannot be used.

- (1) Linear angle commands (,A can be used.)
- (2) Geometric commands

## 9.3 Index Table Indexing



## Function and purpose

Index table indexing can be carried out by setting the index axis. The indexing command only issues the indexing angle to the axis set for indexing. It is not necessary to command special M codes for table clamping and unclamping, thus simplifying the program.



## Detailed description

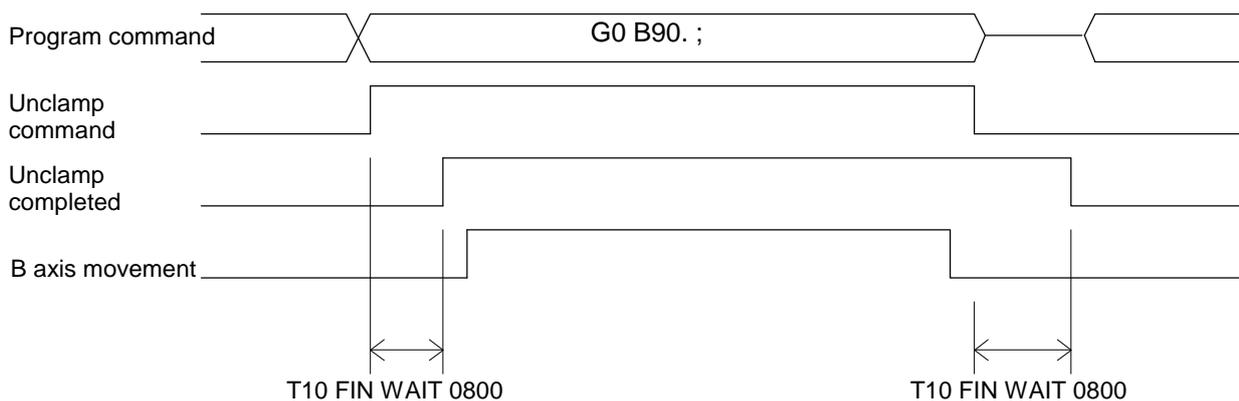
The index table index function carries out operations as follows.

**(Example)** G00\_B90 ;

↳ The axis that was designated as the index axis with parameter "#2076 index\_x".

- (1) Set the "index\_x" parameter (#2076) for the axis in which index table indexing will be carried out to "1".
- (2) The movement command (either absolute or incremental) for the selected axis is executed with the program command.
- (3) An unclamp process are carried out before the axis movement.
- (4) The commanded axis movement starts after the unclamp process completes.
- (5) The clamp process is carried out after the movement is completed.
- (6) The next block is processed after the unclamp process completes.

## &lt;Operation time chart&gt;

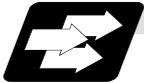


**Precautions**

- (1) Several axes can be set as index table indexing axes.
- (2) The movement speed of index table indexing axes follows the feedrate of the modal (G0/G1) at that time.
- (3) The unclamp process for the indexing axes is also issued when the index table indexing axes are commanded in the same block as other axes. Thus, the movement of other axes commanded in the same block is not carried out until the unclamp process completes.  
Note that the movement of other axes commanded in the same block is carried out for non-interpolation commands.
- (4) Index table indexing axes are used as normal rotation axes, but this function performs an unclamp process even for linear axes.
- (5) If some error that makes unclamp command OFF occurs during indexing axis movement in automatic operation, the unclamp state will be remained, and the indexing axis will execute a deceleration stop.  
Other axes commanded in the same block will also execute a deceleration stop, except for non-interpolation commands.
- (6) If the axis movement is interrupted by an interlock, etc., during indexing axis movement, the unclamp state will be remained.
- (7) The clamp and unclamp process are not executed when the movement commands of the index table indexing axis are continuous.  
Note that the clamp and unclamp process are executed even when the movement commands are continued during single block operation.
- (8) Make sure that the command position is at a position where clamping is possible.

## 10. Spindle Functions

### 10.1 Spindle Functions (S6-digits Analog)



#### Function and purpose

When the S6-digits function is added, a 6-digit value (0 to 999999) can be designated after the S code.

Always select S command binary output when using this function. If the S function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. The machine specifications determine which one is applied.

- (1) The S function is executed after the movement command.
- (2) The S function is executed simultaneously with the movement command.

By assigning a 6-digit number following the S code, these functions enable the appropriate gear signals, voltages corresponding to the commanded spindle speed (r/min) and start signals to be output.

If the gear step is changed manually other than when the S command is being executed, the voltage will be obtained from the set speed at that gear step and the previously commanded speed, and then will be output.

The analog signal specifications are given below.

- (1) Output voltage ..... 0 to 10V
- (2) Resolution.....  $1/4096 (2^{-12})$
- (3) Load conditions .....  $10k\Omega$
- (4) Output impedance .....  $220\Omega$

If the parameters for up to 4 gear stages are set in advance, the gear stage corresponding to the S command will be selected and the gear signal will be output. The analog voltage is calculated in accordance with the input gear signal.

- (1) Parameters corresponding to individual gears .....Limit rotation speed, maximum rotation speed, shift rotation speed, tap rotation speed.
- (2) Parameters corresponding to all gears.....Minimum rotation speed, orientation rotation speed

### 10.2 Spindle Functions (S8-digits)



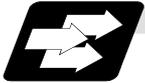
#### Function and purpose

These functions are assigned with an 8-digit (0 to 99999999) number following the address S, and one group can be assigned in one block.

The output signal is a 32-bit binary data with sign and start signal. Processing and completion sequences are required for all S commands.

## 10.3 Constant Surface Speed Control; G96, G97

### 10.3.1 Constant Surface Speed Control



#### Function and purpose

These cinommands automatically control the spindle speed in line with the changes in the radius coordinate values as cutting proceeds in the diametrical direction, and they serve to keep the cutting pot speed constant during the cutting.



#### Command format

**G96 S\_\_ P\_\_ ; Constant surface speed ON**

S : Peripheral speed

P : Constant surface speed control axis

**G97 ; Constant surface speed cancel**



#### Detailed description

- (1) The constant surface speed control axis is set by parameter "#1181 G96\_ax".
  - 0 : Fixed at 1st axis (P command invalid)
  - 1 : 1st axis
  - 2 : 2nd axis
  - 3 : 3rd axis
- (2) When the above-mentioned parameter is not zero, the constant surface speed control axis can be assigned by address P.
 

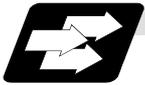
**(Example)** G96\_ax : 1

Program	Constant surface speed control axis
G96 S100 ;	1st axis
G96 S100 P3 ;	3rd axis

- (3) Example of selection program and operation

G90 G96 G01 X50. Z100. S200 ;	} The spindle speed is controlled so that the peripheral speed is 200m/min.
}	
G97 G01 X50. Z100. F300 S500 ;	} The spindle speed is controlled to 500r/min.
}	
M02 ;	The modal returns to the initial setting.

10.4 Spindle Clamp Speed Setting; G92



Function and purpose

The maximum clamp speed of the spindle can be assigned by address S following G92 and the minimum clamp speed by address Q.



Command format

<b>G92 S__ Q__;</b>	
S	: Maximum clamp speed
Q	: Minimum clamp speed



Detailed description

- Besides this command, parameters can be used to set the rotational speed range up to 4 stages in 1 r/min units to accommodate gear selection between the spindle and spindle motor. The lowest upper limit and highest lower limit are valid among the rotational speed ranges based on the parameters and based on G92 Ss Qq ;
- Set in the parameter "#1146 Sclamp" or "#1227 aux11/bit5" whether to carry out rotation speed clamp only in the constant surface speed mode or even when the constant surface speed is canceled.

(Note1) G92S command and speed clamp operation

		Sclamp = 0		Sclamp = 1	
		aux11/bit5 = 0	aux11/bit5 = 1	aux11/bit5 = 0	aux11/bit5 = 1
<b>Command</b>	In G96	SPEED CLAMP COMMAND		SPEED CLAMP COMMAND	
	In G97	SPINDLE SPEED COMMAND		SPEED CLAMP COMMAND	
<b>Operation</b>	In G96	SPEED CLAMP EXECUTION		SPEED CLAMP EXECUTION	
	In G97	NO SPEED CLAMP		SPEED CLAMP EXECUTION	NO SPEED CLAMP

(Note2) The address Q following the G92 command is handled as the spindle speed clamp speed regardless of the constant surface mode.

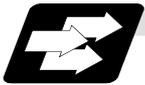
- The command value of spindle clamp speed will be cleared by modal reset (reset2 or reset & rewind).  
Note that the modal is retained if the parameter "#1210 RstGmd / bit19" is ON.



Precautions

- Once the maximum clamp speed and the minimum clamp speed are set, the maximum clamp speed will not be canceled even if the command such as "G92 S0" is issued. Even when "G92 S0" is commanded, the value of Qq is kept valid and S value (S0) falls below Q value (Qq). Thus, Qq will be handled as the maximum clamp speed and S0 as the minimum clamp speed.

### 10.5 Spindle/C Axis Control



#### Function and purpose

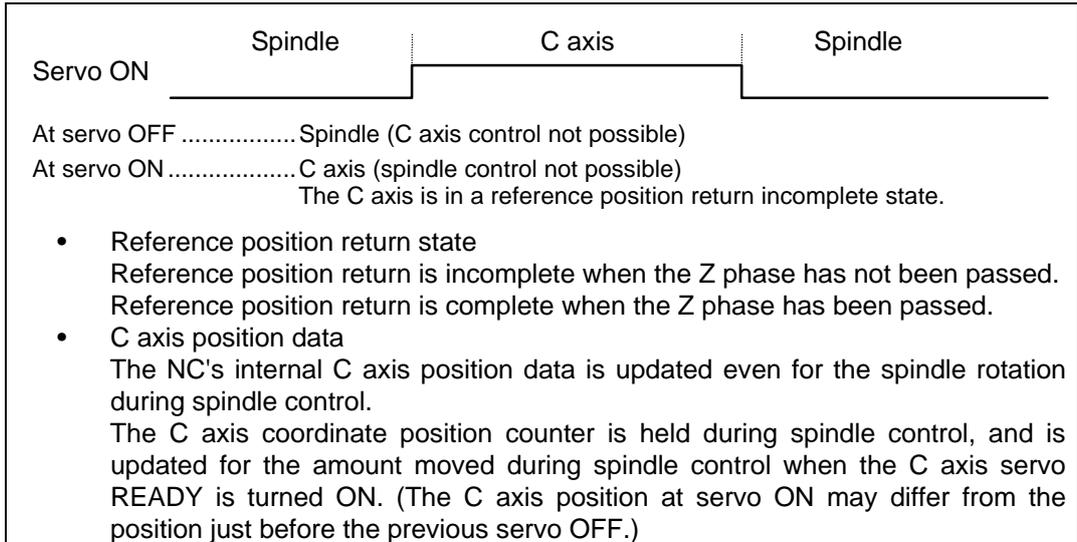
This function enables one spindle (MDS-A/B-SP and later) to also be used as a C axis (rotation axis) by an external signal.



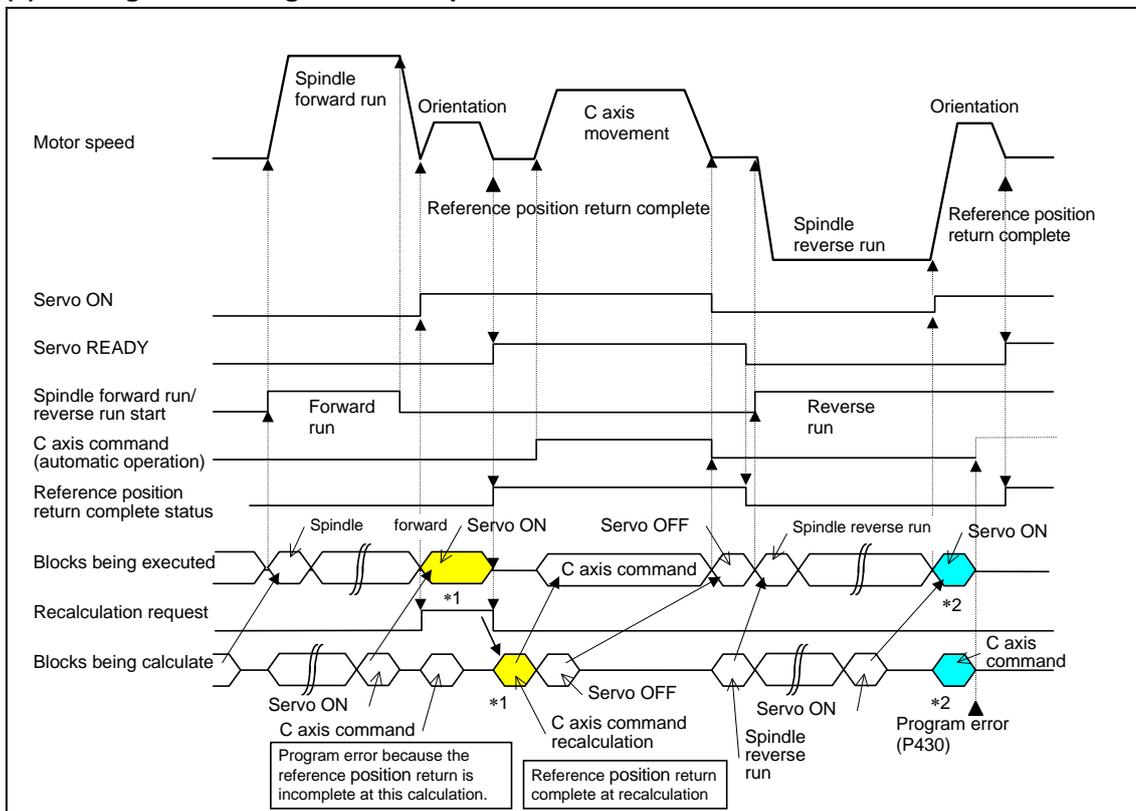
#### Detailed description

##### (1) Spindle/C axis changeover

Changeover between the spindle and C axis is done by the C axis SERVO ON signal.



##### (2) Changeover timing chart example



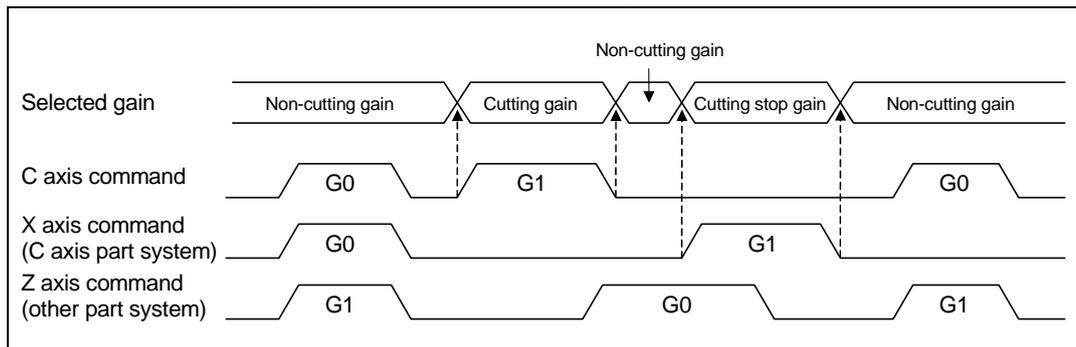
**(Note)** For axis commands, the reference position return complete is checked at calculation. Thus, when the C axis servo ON command and C axis command are continuous, the program error (P430) occur as shown above in \*2.

In response to this kind of situation, the following two processes must be carried out on user PLC, as shown above in \*1.

- Input the recalculation request signal with a servo ON command.
- Wait for the completion of the servo ON command until the C axis enters a servo READY state.

**(3) C axis gain**

The C axis gain is changed over (the optimum gain is selected) by the C axis cutting condition. During C axis cutting feed, cutting gain is applied. During other axis' cutting feed (C axis face turning), non-cutting stop gain is applied. Non-cutting gain is applied in all other cases.



**(Note 1)** The cutting feed of other part systems does not affect the C axis gain selection.

**(Note 2)** There are 1st to 3rd cutting gains, which are selected with the ladder.

**(4) Deceleration check in movement including spindle/C-axis**

The deceleration check in a movement command including the spindle/C-axis is as the table described below when the following condition is fulfilled.

When the different values are set for the position loop gain in non-cutting mode (spindle parameter #3203 PGCO) and the position loop gain in cutting mode (spindle parameter #3330 PGC1 to #3333 PGC4).

That is because a vibration and so on occurs in the machine when the gain is changed during the axis movement.

Parameter	Rapid traverse command	Parameter	Other than rapid traverse command (G1 : other than G0 command)	
<b>Inpos (#1193)</b>	G0→XX (G0+G9→XX)	<b>AUX07/BIT-1 (#1223/BIT-1)</b>	G1+G9→XX (G1+G9→XX)	G1→G1
<b>0</b>	Command deceleration check	<b>0</b>	In-position check (Applicable only to SV024)	No deceleration check
<b>1</b>	In-position check	<b>1</b>		

**(Note 1)** When G1 command is issued, the in-position check is performed regardless of the deceleration check parameter.

**(Note 2)** XX expresses all commands.

**Precautions and Restrictions**

- (1) A reference position return cannot be executed by the orientation when there is no Z phase in the detector (PLG, ENC, other).

Replace the detector with one having a Z phase, or if using the detector as it is, set the position control changeover to "After deceleration stop" in the parameters (Spindle parameters, SP129 bitE: 1), and set the axis to "Axis without zero point" (Zero point return parameters, noref: 1).

- (2) The program error (P430) will occur if a C axis command is issued during servo OFF or during orientation.

- (3) Do not execute a servo OFF during a C axis command.

The remaining C axis commands will be cleared at servo ON.

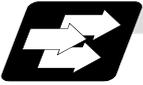
(If servo OFF is executed during C axis control, the feed will stop and spindle control will occur.)

- (4) If servo ON is executed during spindle rotation, the rotation will stop and C axis control will occur.

- (5) Dog-type reference position return are not possible for the C axis.

Set the reference position return to the orientation method in the parameters (Spindle parameters, SP129 bitE: 0), or set the axis to "Axis without zero point" (Zero point return parameters, noref: 1).

## 10.6 Multiple Spindle Control



### Function and purpose

Multiple spindle control is a function used to control the sub-spindle in a machine tool that has a main spindle (1st spindle) and a sub-spindle (2nd spindle to 4th spindle).

Multiple spindle control II:  
(ext36/bit0 = 1)

Control following the external signal (spindle command selection signal, spindle selection signal) and spindle control command ([S\*\*\*\*\* ;] only), etc.  
The spindle selection command [SO=\*\*\*\*\* ;] cannot be used.

10.6.1 Multiple Spindle Control II



Function and purpose

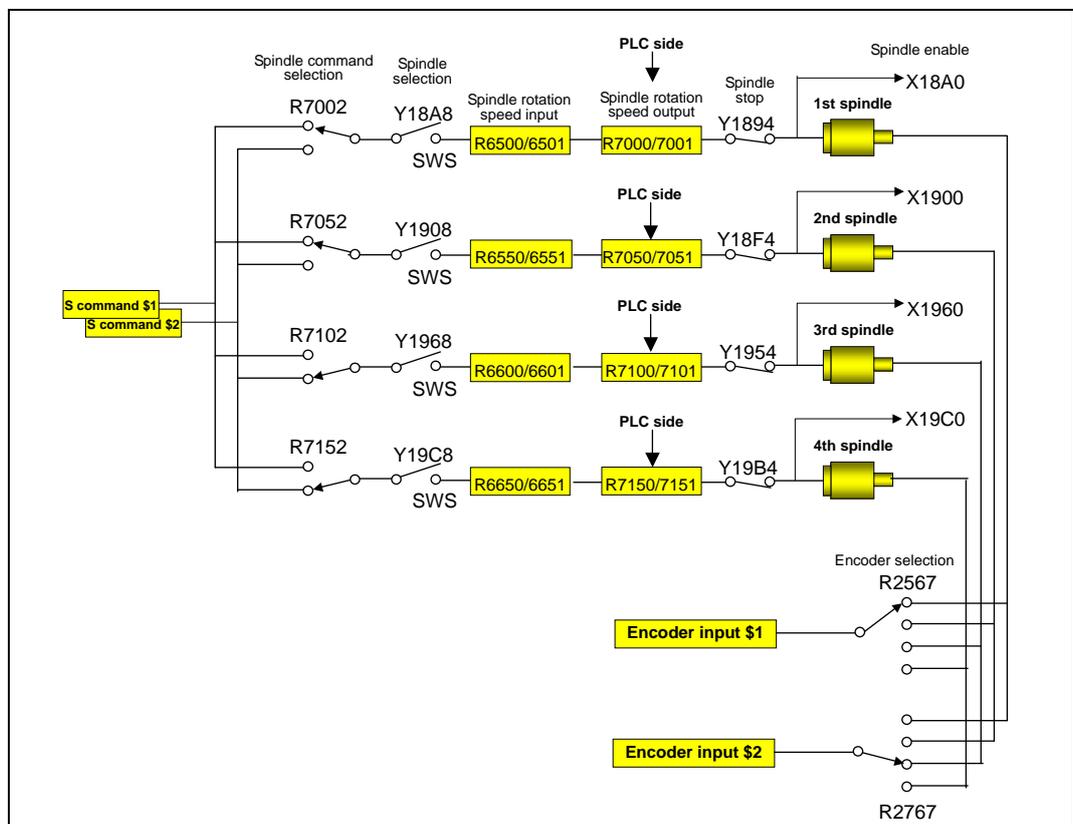
Multiple spindle control II is a function that designates which spindle to select with the signals from PLC. The command is issued to the spindle with one S command.



Detailed description

(1) Spindle command selection, spindle selection

The S command to the spindle is output as the rotation speed command to the selected spindle when the spindle selection signal (SWS) from the PLC turns ON. The selected spindle rotates at the output rotation speed. The spindle whose selection is canceled when the spindle selection signal (SWS) turns OFF maintains the speed at which it was rotating at before being canceled. This allows each axis to be simultaneously rotated at differing rotation speeds. The spindle command selection signal is used to determine which part system each spindle receives the S command from.



(Note) Refer to the PLC Interface Manual for details on each signal.



### Relation with other functions

- (1) Spindle clamp speed setting (G92)  
 This is valid only on the spindle selected with the spindle selection signal (SWS).  
 The spindle not selected with the spindle selection signal (SWS) maintains the speed at which it was rotating at before being canceled.  
 (The spindle clamp speed is maintained with the G92 command.)
- (2) Constant surface speed control  
 Constant surface speed control can be applied on all spindles.  
 The spindle rotation speed is automatically controlled during constant surface speed control, so when machining with constant surface speed, the spindle selection signal (SWS) for that spindle must be left ON.  
 The spindle not selected with the spindle selection signal (SWS) maintains the speed at which it was rotating at before being canceled.
- (3) Thread cutting/synchronous feed  
 The threads are cut with the spindle selected with the spindle selection signal (SWS). The encoder feedback selected with the encoder selection signal is used.
- (4) Synchronous tap  
 The synchronous tap spindle is selected with the spindle selection signal (SWS).  
 Select the synchronous tap spindle before issuing the synchronous tap command. Do not change the synchronous tap spindle selection signal during the synchronous tapping mode.  
 If a C axis mode command is issued to the synchronous tap spindle, the "M01 operation error 1026" will occur. When the C axis command is canceled, the error will be canceled and machining will resume.  
 If a polygon machining command is issued to the synchronous tap spindle, the "M01 operation error 1026" will occur. When the polygon machining command is canceled, the error will be canceled and machining will resume.
- (5) Asynchronous tap  
 The asynchronous tap spindle is selected with the spindle selection signal (SWS).  
 Select the asynchronous tap spindle before issuing the tap command. Input a calculation request to change the asynchronous tap spindle selection. Do not change the asynchronous tap spindle selection signal during the asynchronous tapping mode.
- (6) Tap return  
 The tap return spindle is selected with the spindle selection signal (SWS).  
 Select the spindle for which the tap cycle execution is stopped before turning the tap return signal ON.  
 If tap return is executed when a different spindle is selected, the "M01 operation error 1032" will occur. Do not change the spindle selection signal during tap return.



### Restrictions

- (1) The S manual value command is invalid when multiple spindle control II is valid.
- (2) Setup parameter "#1199 Sselect" is invalid when multiple spindle control II is valid.
- (3) The spindle control mode changeover G code cannot be used when multiple spindle control II is valid. A program error (P34) will occur.
- (4) The "S1=\*\*\*" and "S2=\*\*\*" commands are invalid when multiple spindle control II is valid. A program error (P33) will occur.
- (5) The spindle gear shift command output signal (GR1/GR2) is not output when multiple spindle control II is valid.

## 11. Tool Functions (T command)

### 11.1 Tool Functions (T8-digit BCD)



#### Function and purpose

The tool functions are also known simply as T functions and they assign the tool numbers and tool offset number. They are designated with a 8-digit number following the address T, and one set can be commanded in one block. The output signal is an 8-digit BCD signal and start signal.

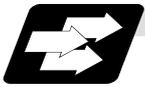
If the T function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. The machine specifications determine which sequence applies.

- (1) The T function is executed after the movement command.
- (2) The T function is executed simultaneously with the movement command.

Processing and completion sequences are required for all T commands.

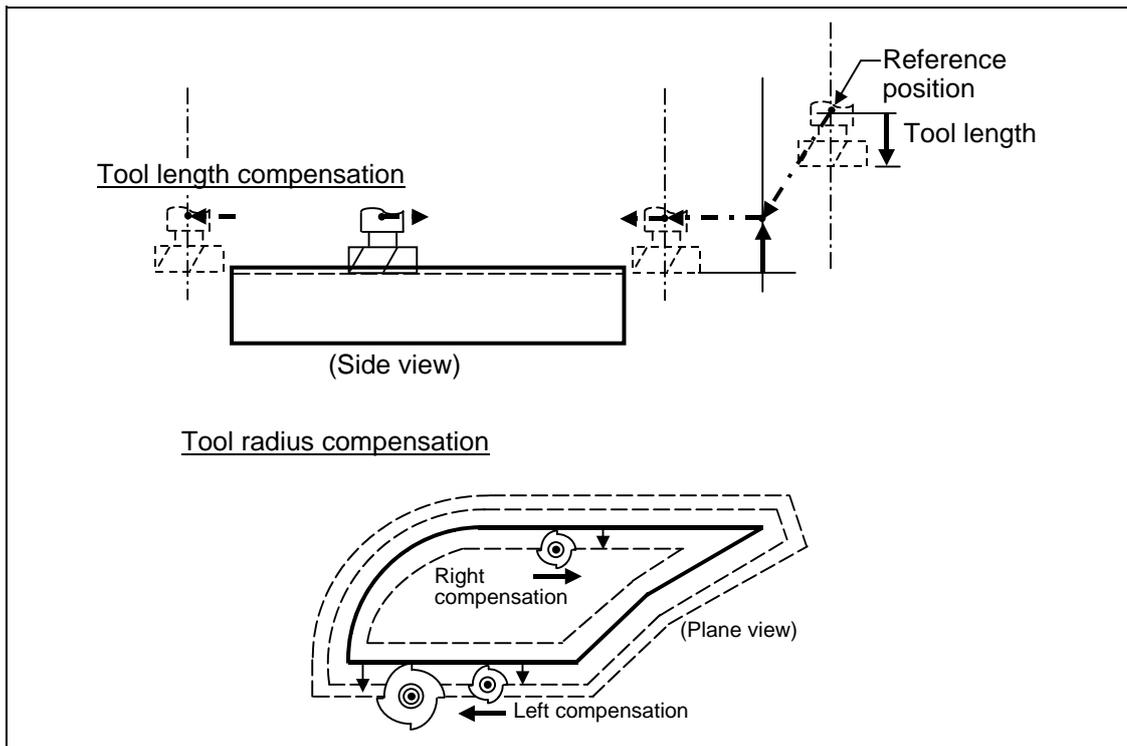
## 12. Tool Compensation Functions

### 12.1 Tool Compensation



#### Function and purpose

The basic tool compensation function includes the tool length compensation and tool radius compensation. Each compensation amount is designated with the tool compensation No. Each compensation amount is input from the setting and display unit or the program.

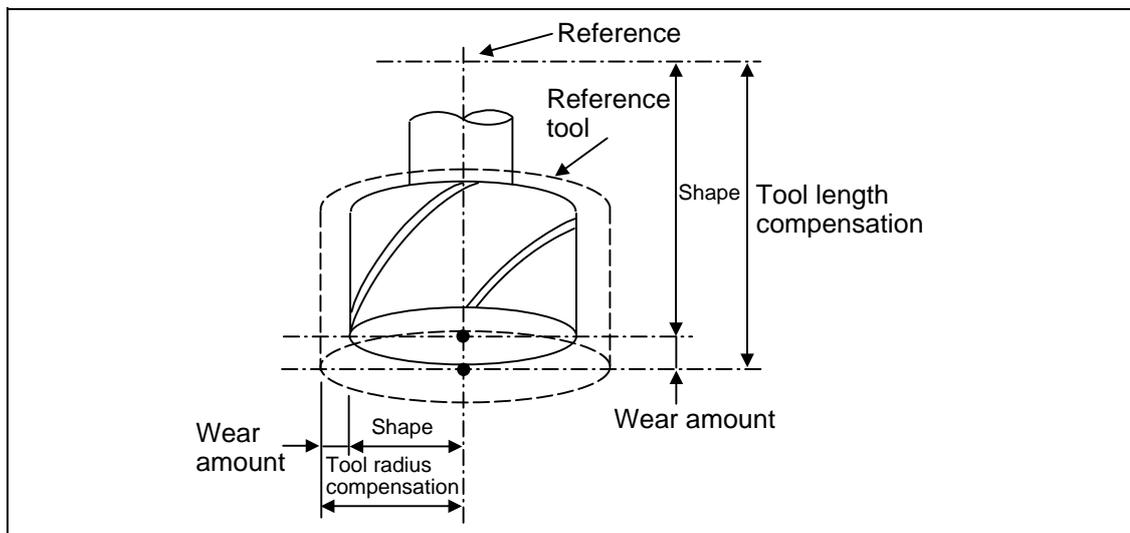




**Tool compensation memory**

There are two types of tool compensation memories for setting and selecting the tool compensation amount. (The type used is determined by the machine maker specifications.) The compensation amount settings are preset with the setting and display unit. Type 1 is selected when parameter "#1037 cmdtyp" is set to "1", and type 2 is selected when set to "2".

Type of tool compensation memory	Classification of length compensation, radius compensation	Classification of shape compensation, wear compensation
Type 1	Not applied	Not applied
Type 2	Applied	Applied



Type 1

One compensation amount corresponds to one compensation No. as shown on the right. Thus, these can be used commonly regardless of the tool length compensation amount, tool radius compensation amount, shape compensation amount and wear compensation amount.

$$(D1) = a_1, (H1) = a_1$$

$$(D2) = a_2, (H2) = a_2$$

$$\vdots$$

$$(Dn) = a_n, (Hn) = a_n$$

Compensation No.	Compensation amount
1	$a_1$
2	$a_2$
3	$a_3$
•	•
•	•
n	$a_n$

Type 2

The shape compensation amount related to the tool length, wear compensation amount, shape compensation related to the tool radius and the wear compensation amount can be set independently for one compensation No. as shown below.

The tool length compensation amount is set with H, and the tool radius compensation amount with D.

$$(H1) = b1 + c1, (D1) = d1 + e1$$

$$(H2) = b2 + c2, (D2) = d2 + e2$$

$$\vdots$$

$$(Hn) = bn + cn, (Dn) = dn + en$$

Compensation No.	Tool length (H)		Tool radius (D)/ (Position compensation)	
	Shape compensation amount	Wear compensation amount	Shape compensation amount	Wear compensation amount
1	b1	c1	d1	e1
2	b2	c2	d2	e2
3	b3	c3	d3	e3
•	•	•	•	•
•	•	•	•	•
n	bn	cn	dn	en

 **CAUTION**

 If the tool compensation amount is changed during automatic operation (including during single block stop), it will be validated from the next block or blocks onwards.

**Tool compensation No. (H/D)**

This address designates the tool compensation No.

- (1) H is used for the tool length compensation, and D is used for the tool position offset and tool radius compensation.
- (2) The tool compensation No. that is designated once does not change until a new H or D is designated.
- (3) The compensation No. can be commanded once in each block. (If two or more Nos. are commanded, the latter one will be valid.)
- (4) The No. of compensation sets that can be used will differ according to the machine.  
For 40 sets: Designate with the H01 to H40 (D01 to D40) numbers.
- (5) If a value larger than this is set, the program error (P170) will occur.
- (6) The setting value ranges are as follows for each No.  
The compensation amount for each compensation No. is preset with the setting and display unit.

Setting	Shape compensation amount		Wear compensation amount	
	Metric system	Inch system	Metric system	Inch system
#1003=B	±99999.999 (mm)	±9999.9999 (inch)	±99999.999 (mm)	±9999.9999 (inch)
#1003=C	±99999.9999 (mm)	±9999.99999 (inch)	±99999.9999 (mm)	±9999.99999 (inch)
#1003=D	±99999.99999 (mm)	±9999.999999 (inch)	±99999.99999 (mm)	±9999.999999 (inch)
#1003=E	±99999.999999 (mm)	±9999.9999999 (inch)	±99999.999999 (mm)	±9999.9999999 (inch)

## 12.2 Tool Length Compensation/Cancel; G43/G44



## Function and purpose

The end position of the movement command can be compensated by the preset amount when this command is used. A continuity can be applied to the program by setting the actual deviation from the tool length value decided during programming as the compensation amount using this function.



## Command format

<b>G43 Zz Hh ;</b>	<b>Tool length compensation start</b>
:	
<b>G44 Zz ;</b>	<b>Tool length compensation cancel</b>



## Detailed description

## (1) Tool length compensation movement amount

The movement amount is calculated with the following expressions when the G43 tool length compensation command or G44 tool length compensation cancel command is issued.

	Z axis movement amount	
G43 Zz Hh1 ;	$z + (h_1)$	Compensation in + direction by tool compensation amount
G43 Zz Hh1 ;	$z - (h_1)$	Compensation in - direction by tool compensation amount
G44 Zz ;	$z - (+) (h_1)$	Compensation amount cancel

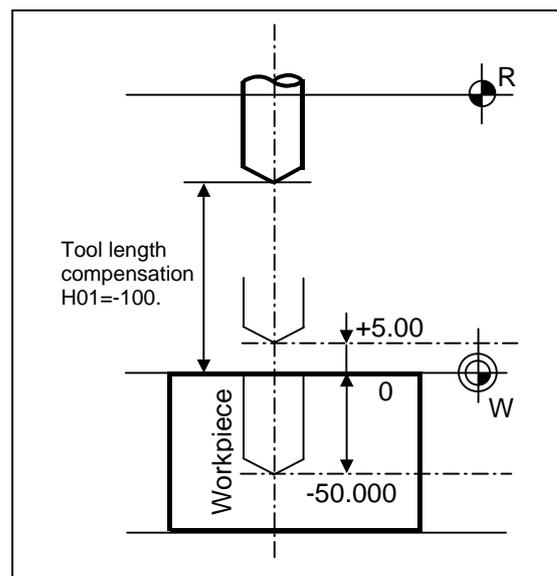
(Note)  $h_1$  : Compensation amount for compensation No.  $h_1$

Regardless of the absolute value command or incremental value command, the actual end point will be the point compensated by the compensation amount designated for the programmed movement command end point coordinate value.

The G44 (tool length compensation cancel) mode is entered when the power is turned ON or when M02 has been executed.

**(Example 1)** For absolute value command  
 H01 = -100000  
 N1 G28 Z0 T01 M06 ;  
 N2 G90 G92 Z0 ;  
 N3 G43 Z5000 H01 ;  
 N4 G01 Z-50000 F500 ;

**(Example 2)** For incremental value command  
 H01 = -100000  
 N1 G28 Z0 T01 M06 ;  
 N2 G91 G92 Z0 ;  
 N3 G43 Z5000 H01 ;  
 N4 G01 Z-55000 F500 ;



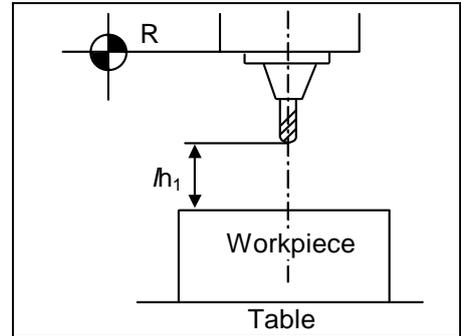
**(2) Compensation No.**

(a) The compensation amount differs according to the compensation type.

**Type 1**

G43 Hh<sub>1</sub> ;

When the above is commanded, the compensation amount  $h_1$  commanded with compensation No.  $h_1$  will be applied commonly regardless of the tool length compensation amount, tool radius compensation amount, shape compensation amount or wear compensation amount.

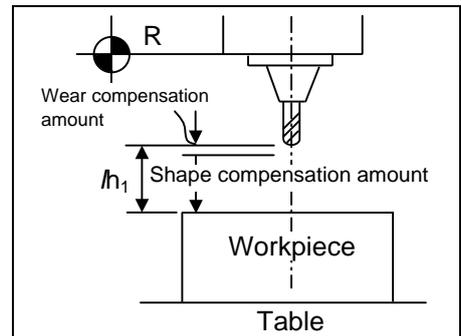


**Type 2**

G43 Hh<sub>1</sub> ;

When the above is commanded, the compensation amount  $h_1$  commanded with compensation No.  $h_1$  will be as follows.

$h_1$ : Shape compensation + wear compensation amount



- (b) The valid range of the compensation No. will differ according to the specifications (No. of compensation sets).
- (c) If the commanded compensation No. exceeds the specification range, the program error (P170) will occur.
- (d) Tool length cancel will be applied when H0 is designated.
- (e) The compensation No. commanded in the same block as G43 will be valid for the following modals.

**(Example 3)**

```
G43 Zz1 Hh1 ; ..... Tool length compensation is executed with h1.
:
G45 Xx1 Yy1 Hh6 ;
:
G44 Zz2 ; ..... The tool length compensation is canceled.
:
G43 Zz2 ; ..... Tool length compensation is executed again with h1.
:
```

(f) If G43 is commanded in the G43 modal, a compensation of the difference between the compensation No. data will be executed.

**(Example 4)**

```
G43 Zz1 Hh1 ; ..... Becomes the z1 + (lh1) movement.
:
G43 Zz2 Hh2 ; ..... Becomes the z2 + (lh2 - lh1) movement.
:
```

**(3) Axis valid for tool length compensation**

- (a) When parameter "#1080 Dril\_Z" is set to "1", the tool length compensation is always applied on the Z axis.
- (b) When parameter "#1080 Dril\_Z" is set to "0", the axis will depend on the axis address commanded in the same block as G43. The order of priority is shown below.  
Zp > Yp > Xp

**(Example 5)**

```

G43 Xx1 Hh1 ; .....+ compensation to X axis
:
G44 Xx2 ;
:
G43 Yy1 H-h2 ; .....-compensation to Y axis
:
G44 Yy2 ;
:
G43 αα1 Hh3 ;.....+ compensation to additional axis
:
G44 αα1 ;
:
G43 Xx3 Yy3 Zz3 ; .....Compensation is applied on Z axis
:
G44 ;

```

The handling of the additional axis will follow the parameters "#1029 to 1031 aux\_I, J and K" settings.

If the tool length compensation is commanded for the rotary axis, set the rotary axis name for one of the parallel axes.

- (c) If H (compensation No.) is not designated in the same block as G43, the Z axis will be valid.

**(Example 6)**

```

G43 Hh1 ; .....Compensation and cancel to X axis
:
44 ;

```

**(4) Movement during other commands in tool length compensation modal**

- (a) If reference position return is executed with G28 and manual operation, the tool length compensation will be canceled when the reference position return is completed.

**(Example 7)**

```

G43 Zz1 Hh1 ;
:
G28 Zz2 ; .....Canceled when reference position is reached.
:
G43 Zz2 Hh2 ; (Same as G44)
:
G44 G28 Zz2 ; .....After the Z axis is canceled, reference position
return is executed.

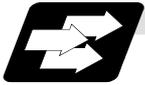
```

- (b) The movement is commanded to the G53 machine coordinate system, the axis will move to the machine position when the tool compensation amount is canceled. When the G54 to G59 workpiece coordinate system is returned to, the position returned to will be the coordinates shifted by the tool compensation amount.

## 12. Tool Compensation Functions

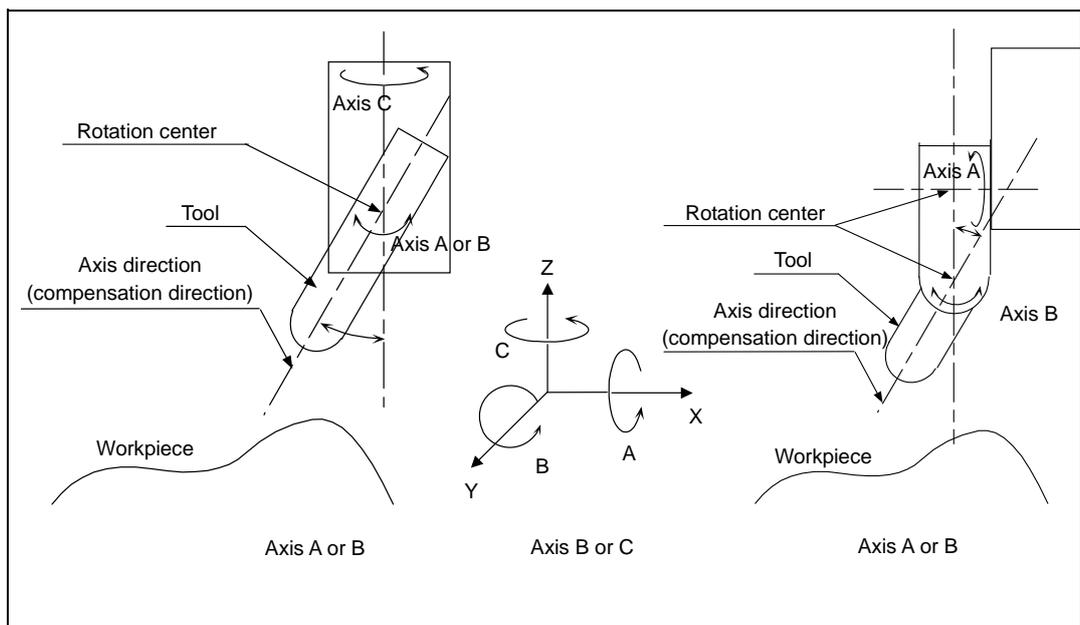
### 12.3 Tool Length Compensation in the Tool Axis Direction ; G43.1/G44

#### 12.3 Tool Length Compensation in the Tool Axis Direction ; G43.1/G44



##### Function and purpose

- (1) Changes in the tool length compensation in the tool axis direction and compensation amount  
The tool length can be compensated in the tool axis direction even when the rotation axis rotates and the tool axis direction becomes other than the Z axis direction. By using this function, and setting the deviation between the tool length amount set in the program and the actual tool length as the compensation amount, a more flexible program can be created. This is especially valid for programs in which many rotation axis movement commands are present. The tool length compensation amount in the tool axis direction can be changed by rotating the manual pulse generator when the tool length compensation amount in the tool axis direction is being changed during the tool length compensation in the tool axis direction mode.
- (2) Machine configuration  
The compensation using the tool length compensation in the tool axis direction function is applied to the direction of the tool tip axis (rotary axis).  
As for the axes that determine the compensation direction, a combination of the C axis (spindle) for Z axis rotation and the A axis for X axis rotation or B axis for Y axis rotation is designated using a parameter.



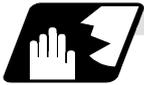
##### Command format

**G43.1 X\_\_ Y\_\_ Z\_\_ H\_\_ ; Tool length compensation in the tool axis direction**  
**G44 X\_\_ Y\_\_ Z\_\_ ; Tool length compensation cancel**

X, Y, Z : Movement data  
H : Tool length compensation No.  
(If the compensation No. exceeds the specification range, a program error (P170) will occur.)

## 12. Tool Compensation Functions

### 12.3 Tool Length Compensation in the Tool Axis Direction ; G43.1/G44



#### Detailed description

- (1) G43 and G43.1 are all G codes in the same group. Therefore, it is not possible to designate more than one of these commands simultaneously for compensation. G44 is used to cancel the G43 and G43.1 commands.
- (2) If the G43.1 command is designated when the option for the tool length compensation in the tool axis direction is not provided, the program error (P930) will occur.
- (3) If reference position has not been completed for any of the X, Y, Z, A or B and C axes in the G43.1 block, the program error (P430) will occur. However, the error does not apply to the following cases.
  - When mechanical axes have been selected:  
The error does not apply to the A, B and C axes.
  - When "1" has been set for the "#2031 nofef" zero point return parameter:  
The error does not apply to the axis for which "nofef" is set to "1" because it is considered that the reference position return of the axis has already completed.



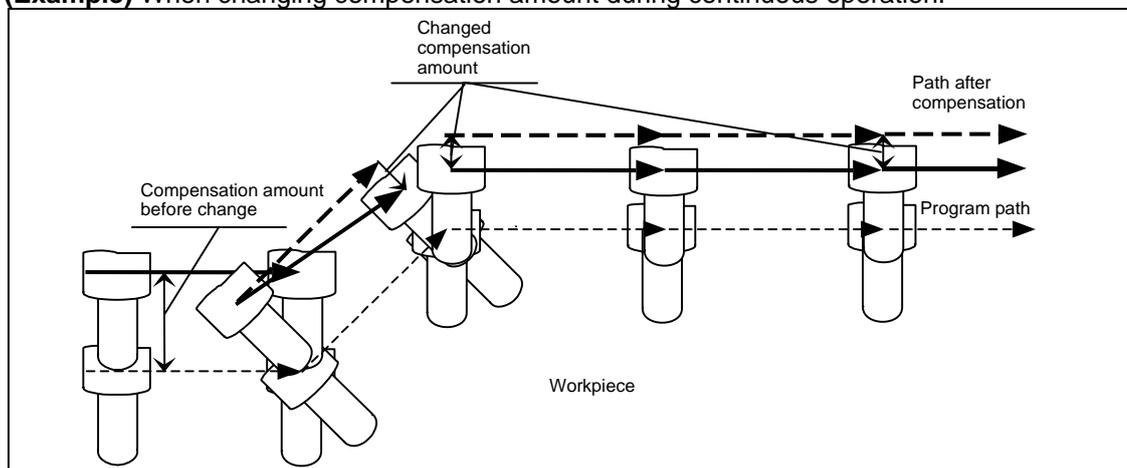
#### Changing the amount of tool length compensation in the tool axis direction

- (1) When the following conditions have been met, the handle movement amount is added to the tool length compensation amount in the tool axis direction by rotating the manual pulse generator.
  - When the operation mode is MDI, memory or tape operation mode and the state is "during single block stop", "during feed hold" or "during cutting feed movement".  
Note that compensation amount cannot be changed during error or warning.
  - During tool length compensation in the tool axis direction (G43.1).
  - In the tool length compensation amount in the tool axis direction changing mode (YC92/1).
  - In the tool handle feed & interruption mode (YC5E/1).
  - The 3rd axis (tool axis) is selected for the handle selection axis.
- (2) The change amount is canceled when the compensation No. is changed.

**(Note 1)** The coordinate value in the tool length compensation amount in the tool axis direction change mode operates in the same manner as that when the manual ABS is ON, regardless of manual ABS switch (YC28) or base axis specification parameter "#1061 intabs".

**(Note 2)** If compensation amount is changed during continuous operation, single block stop, or feed hold, the compensation amount will be effective immediately in the next block.

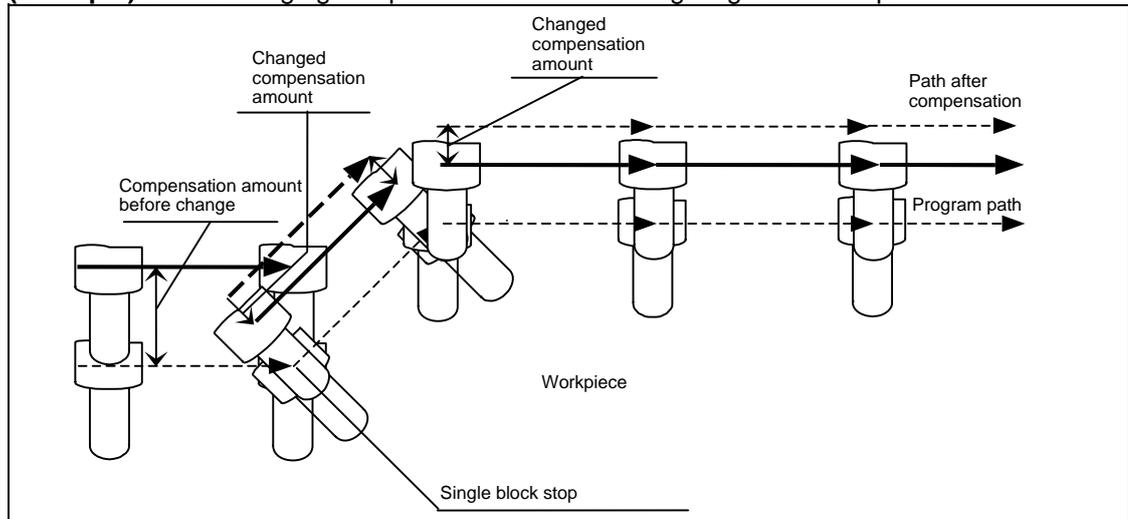
**(Example)** When changing compensation amount during continuous operation.



## 12. Tool Compensation Functions

### 12.3 Tool Length Compensation in the Tool Axis Direction ; G43.1/G44

**(Example)** When changing compensation amount during single block stop.



**(Note 3)** When changing compensation amount, the compensation amount corresponding to the actual compensation No. will be changed. However, when executing the NC reset or tool length compensation in the direction of tool axis cancel (G44), the compensation amount will be returned to the original.



#### Tool length compensation in the tool axis direction vector

The vectors representing the tool length compensation in the tool axis direction are as follows.

(1) When the A and C axes are set as the rotary axes:

$$\begin{aligned} V_x &= L \times \sin(A) \times \sin(C) \\ V_y &= -L \times \sin(A) \times \cos(C) \\ V_z &= L \times \cos(A) \end{aligned}$$

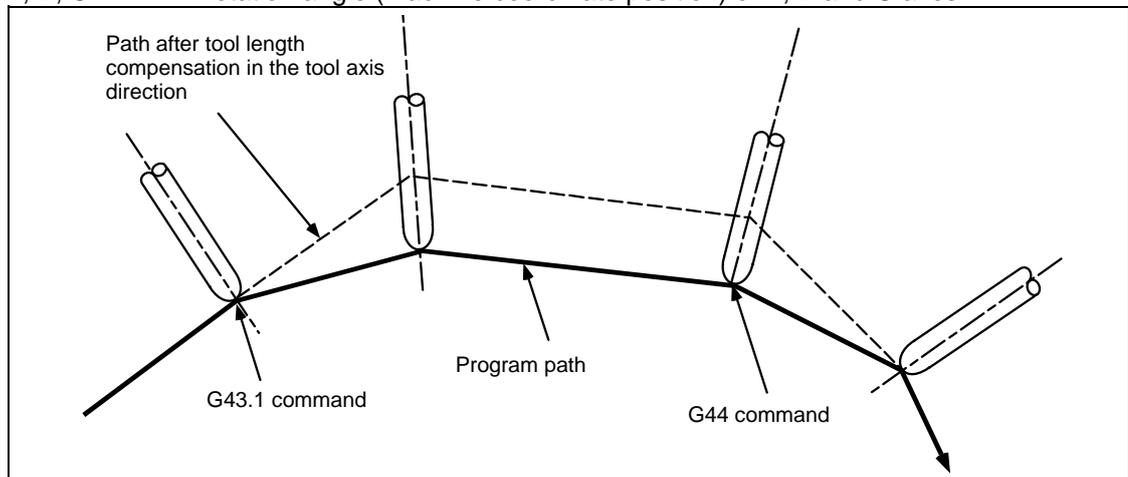
(2) When the B and C axes are set as the rotary axes:

$$\begin{aligned} V_x &= L \times \sin(B) \times \cos(C) \\ V_y &= L \times \sin(B) \times \sin(C) \\ V_z &= L \times \cos(B) \end{aligned}$$

$V_x, V_y, V_z$  : Tool length compensation in the tool axis direction vectors for X, Y and Z axes

$L$  : Tool length compensation amount (1h)

$A, B, C$  : Rotation angle (machine coordinate position) of A, B and C axes



## 12. Tool Compensation Functions

### 12.3 Tool Length Compensation in the Tool Axis Direction ; G43.1/G44

(3) Rotary axis angle command

The value used for the angle of the rotary axis (tool tip axis) differs according to the type of rotary axis involved.

When servo axes are used:

The machine coordinate position is used for the rotation angles of the A, B and C axes.

When mechanical axes are used:

Instead of the machine coordinate position of the axes, the values read out from the R registers (R2628 to R2631) are used for the rotation angles of the A, B and C axes.



#### Compensation amount resetting

Tool length compensation in the tool axis direction is cleared in the following cases.

- (1) When manual reference position return is completed.
- (2) When reset 1, reset 2 or reset & rewind has been executed.
- (3) When the G44 command has been designated.
- (4) When the compensation No. 0 command has been executed.
- (5) When NC reset has been executed with "1" set for the basic system parameter "#1151 rstint".
- (6) When the G53 command is designated while the compensation status is still established, the compensation is temporarily canceled, and the tool moves to the machine position designated by G53.

## 12. Tool Compensation Functions

### 12.3 Tool Length Compensation in the Tool Axis Direction ; G43.1/G44



#### Example of program

##### (1) Example of arc machining

Shown below is an example of a program for linear → arc → arc → linear machining using the B and C rotary axes on the ZX plane.

<p>Machining program  N01 G91 G28 X0 Y0 Z0 ; Compensation amount H01 = 50 mm  N02 G28 B0 C0 ;  N03 G90 G54 G00 X400. Y0 ;  N04 Z-150. ;  N05 B90. ; B axis: 90 degrees  N06 G18 ;  N07 <u>G43.1</u> X250. H01 ; Tool length compensation in the tool axis direction ON    N08 G01 Z0 F200 ;  N09 G02 X0 Z250. I-250. K0 B0 ; Top right arc, B axis: 0 degrees  N10 G02 X-250. Z0 I0 K-250. B-90. ; Bottom right arc, B axis: -90 degrees  N11 G01 Z-150. ;  N12 G00 G44 X-400. ; Tool length compensation in the tool axis direction OFF    N13 G91 G28 B0 C0 ;  N14 G28 X0 Y0 Z0 ;  N15 M02 ;</p>	<p>Example of program</p> <p>(Reference)  Example of tool length compensation (G43)</p>
--	--



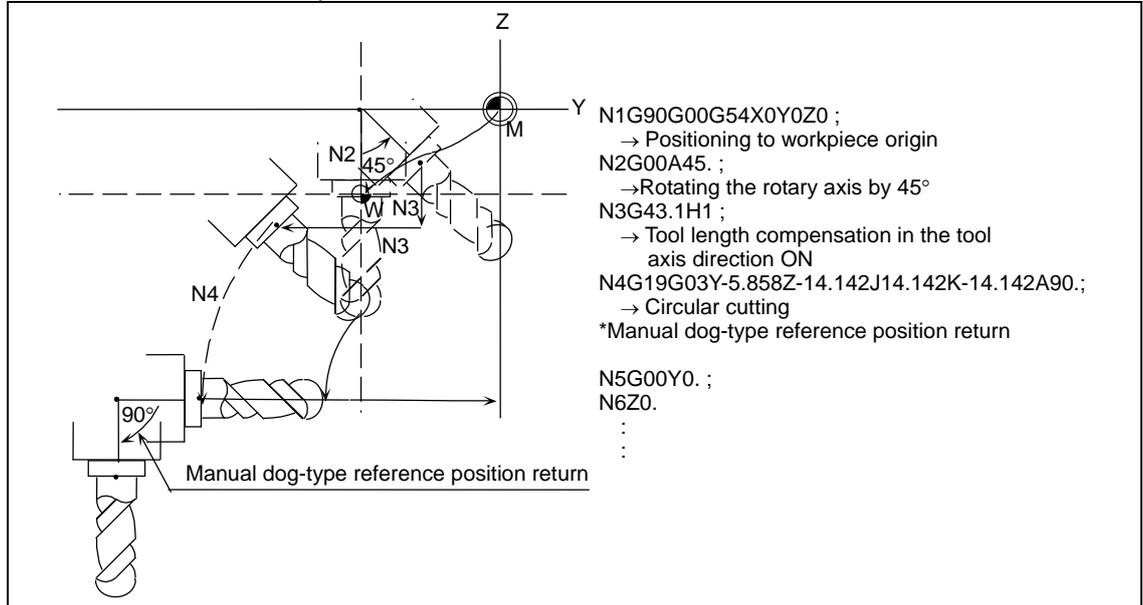
## 12. Tool Compensation Functions

### 12.3 Tool Length Compensation in the Tool Axis Direction ; G43.1/G44

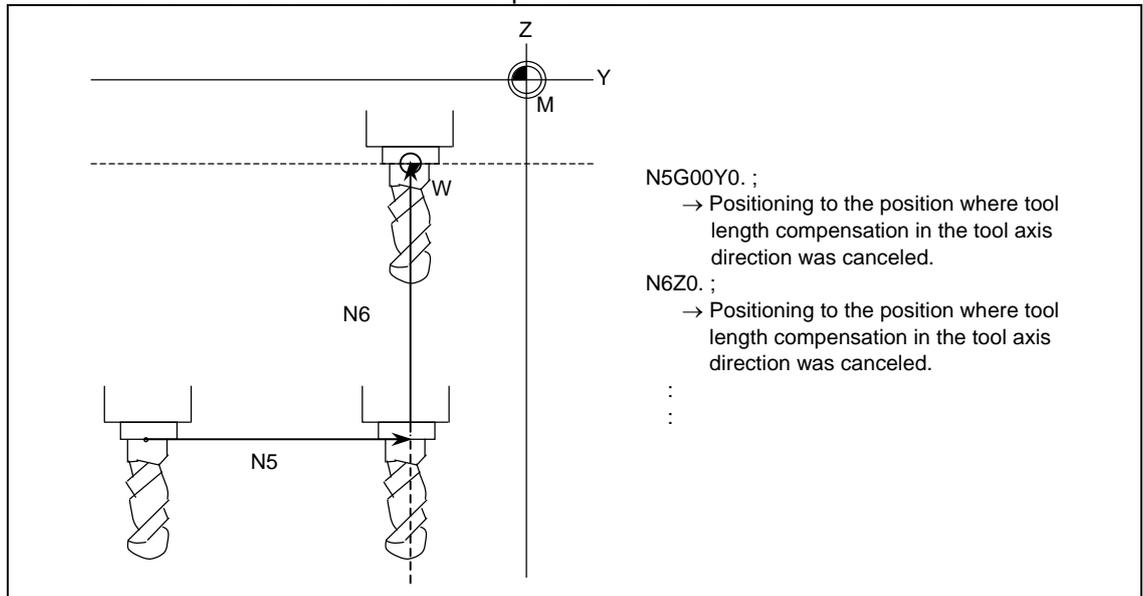
#### (b) Reference position return for the rotary axis

Tool length compensation in the tool axis direction will be canceled, as well as the dog-type reference position return and the high-speed reference position return.

#### <A axis Manual reference position return>



#### <Movement after A axis Manual reference position return>



#### (4) Relation with graphic check

(a) Graphic check draws a path after compensation.

## 12.4 Tool Radius Compensation; G38, G39/G40/G41,G42



## Function and purpose

This function compensates the radius of the tool. The compensation can be done in the random vector direction by the radius amount of the tool selected with the G command (G38 to G42) and the D command.



## Command format

<b>G40X__Y__ ;</b>	<b>: Tool radius compensation cancel</b>	} Can be commanded only during the radius compensation mode.
<b>G41X__Y__ ;</b>	<b>: Tool radius compensation (left)</b>	
<b>G42X__Y__ ;</b>	<b>: Tool radius compensation (right)</b>	
<b>G38I__J__ ;</b>	<b>: Change or hold of compensation vector</b>	
<b>G39X__Y__ ;</b>	<b>: Corner changeover</b>	



## Detailed description

The No. of compensation sets will differ according to the machine model.

(The No. of sets is the total of the tool length offset, tool position offset and tool radius compensation sets.)

The H command is ignored during the tool radius compensation, and only the D command is valid. The compensation will be executed within the plane designated with the plane selection G code or axis address 2 axis, and axes other than those included in the designated plane and the axes parallel to the designated plane will not be affected. Refer to the section on plane selection for details on selecting the plane with the G code.

## 12.4.1 Tool Radius Compensation Operation

**Tool radius compensation cancel mode**

The tool radius compensation cancel mode is established by any of the following conditions.

- (1) After the power has been switched on
- (2) After the reset button on the setting and display unit has been pressed
- (3) After the M02 or M30 command with reset function has been executed
- (4) After the tool radius compensation cancel command (G40) has been executed

The offset vectors are zero in the compensation cancel mode, and the tool nose point path coincides with the programmed path.

Programs including tool radius compensation must be terminated in the compensation cancel mode.

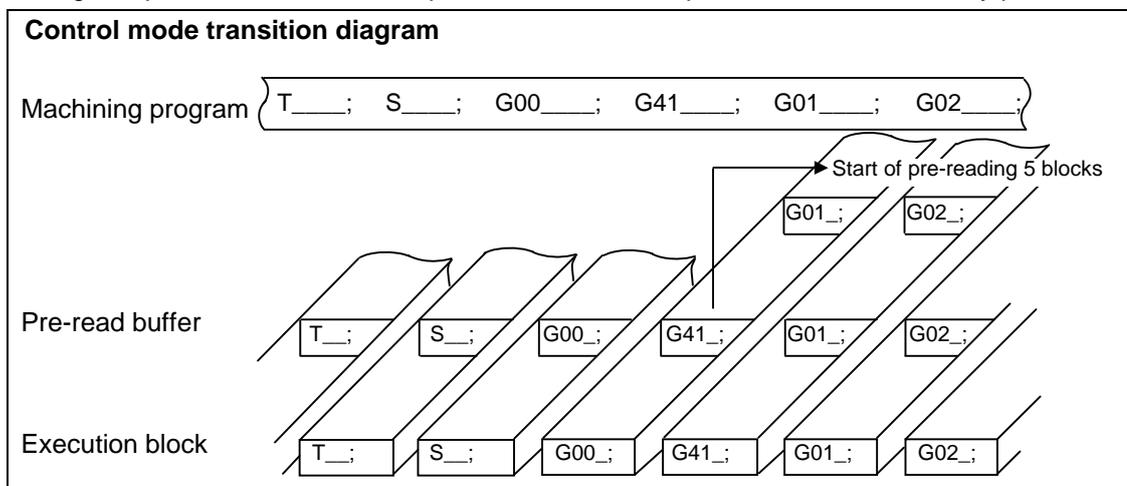
**Tool radius compensation start (start-up)**

Tool radius compensation starts when all the following conditions are met in the compensation cancel mode.

- (1) The movement command is issued after G41 or G42.
- (2) The tool radius compensation offset No. is  $0 < D \leq \text{max. offset No.}$
- (3) The movement command of positioning (G00) or linear interpolation (G01) is issued.

At the start of compensation, the operation is executed after at least three movement command blocks (if three movement command blocks are not available, after five movement command blocks) have been read regardless of the continuous operation or single block operation.

During compensation, 5 blocks are pre-read and the compensation is arithmetically processed.



There are two ways of starting the compensation operation: type A and type B.

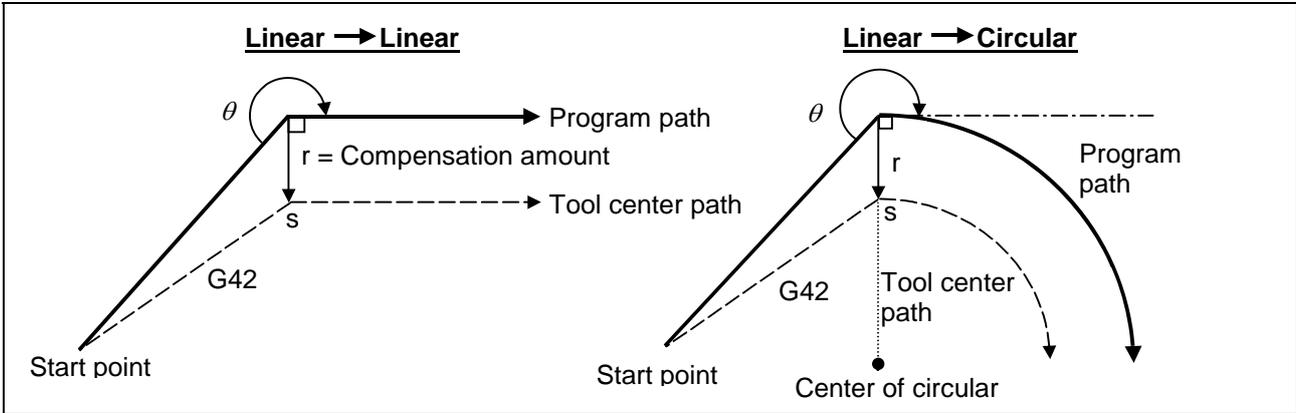
The type can be selected with bit 2 of parameter "#1229 set 01". This type is used in common with the compensation cancel type.

In the following explanatory figure, "S" denotes the single block stop point.

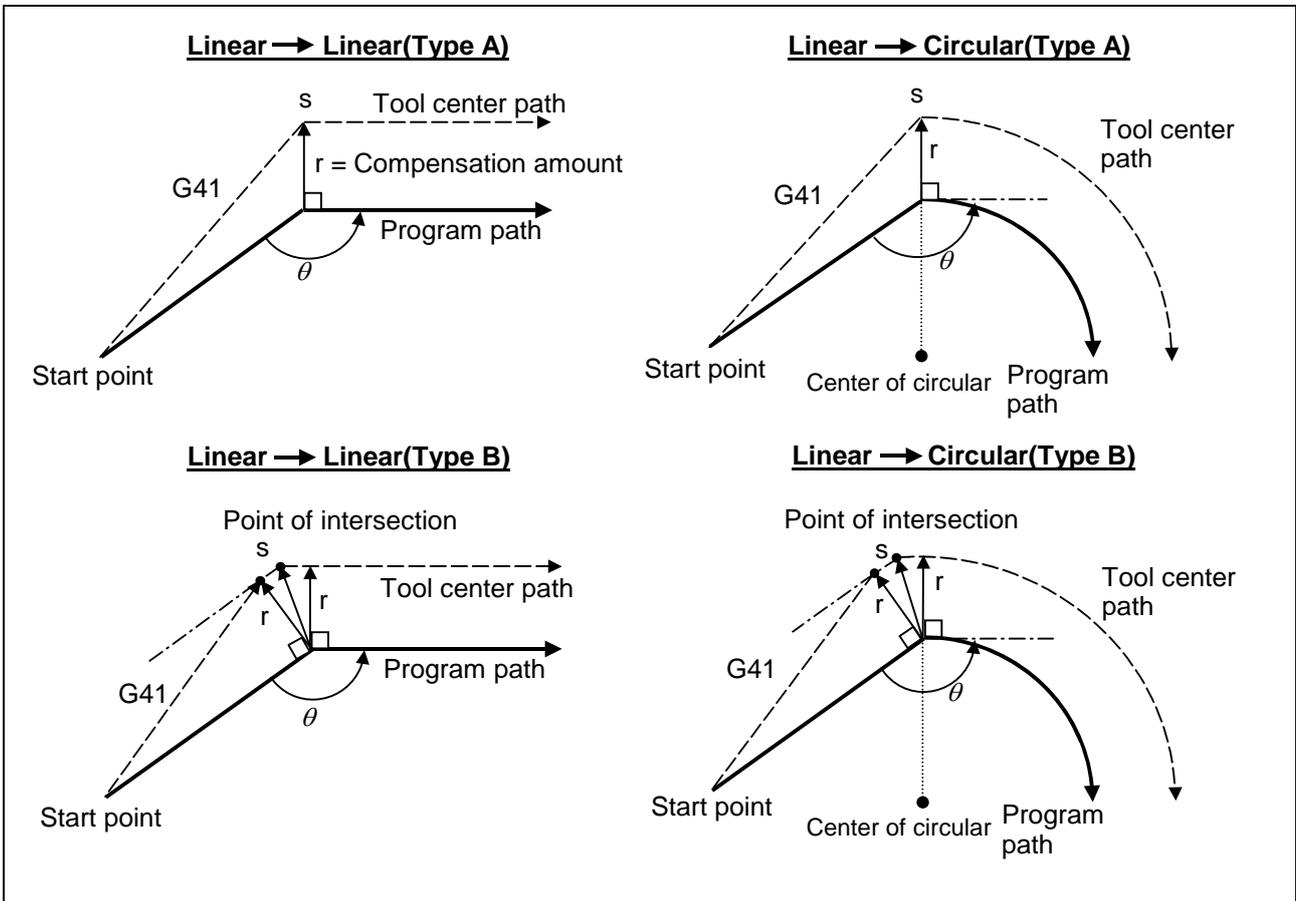


Start of movement for tool radius compensation

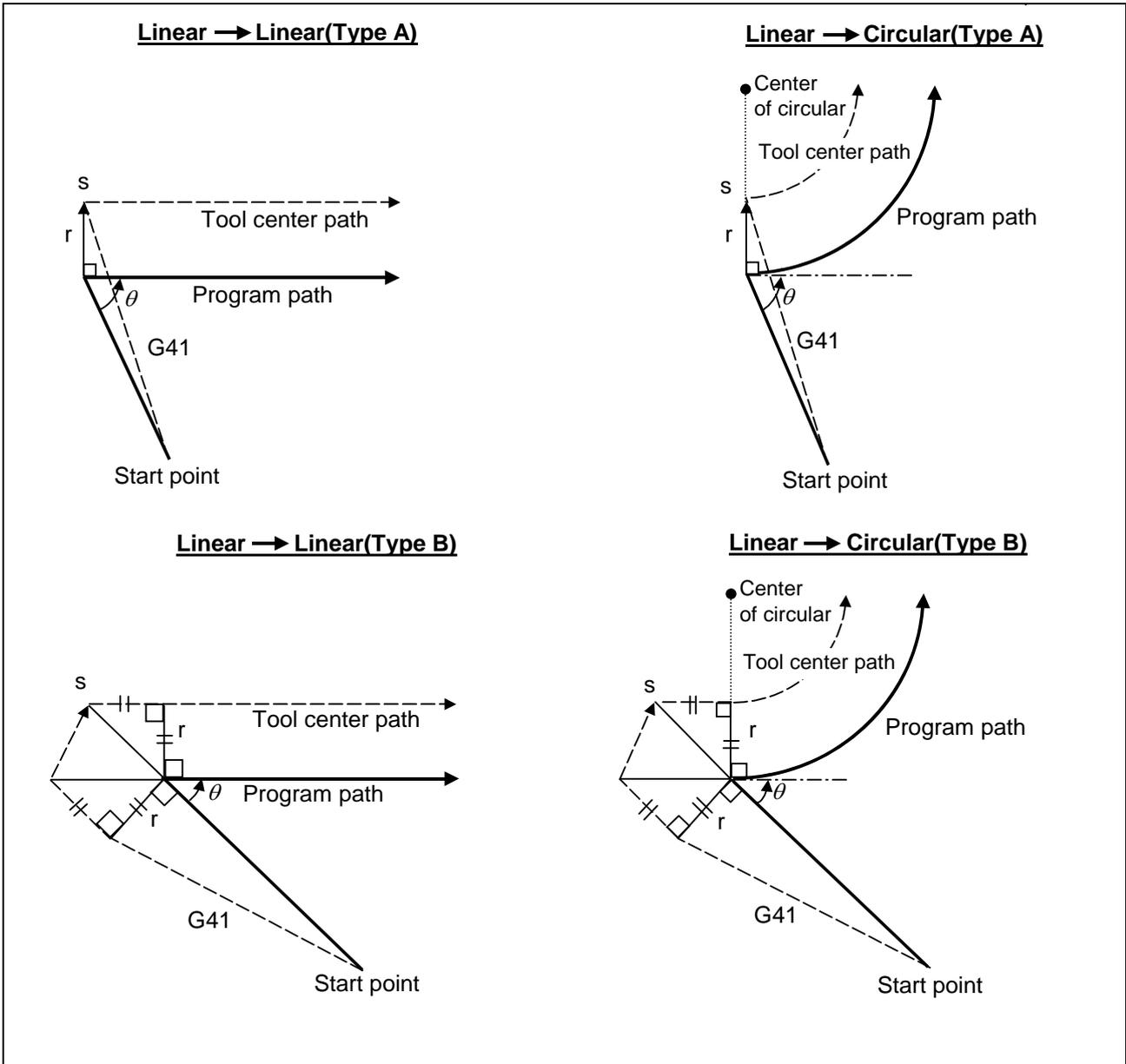
(1) For inner side of corner



(2) For outer side of corner (obtuse angle) [ $90^\circ \leq \theta < 180^\circ$ ]



(3) For outer side of corner (obtuse angle)  $[0 < 90^\circ]$



**(Note)** If there is axis movement command in the same block as G41 or G42, compensation is performed perpendicularly to the next block's direction.



**Operation in compensation mode**

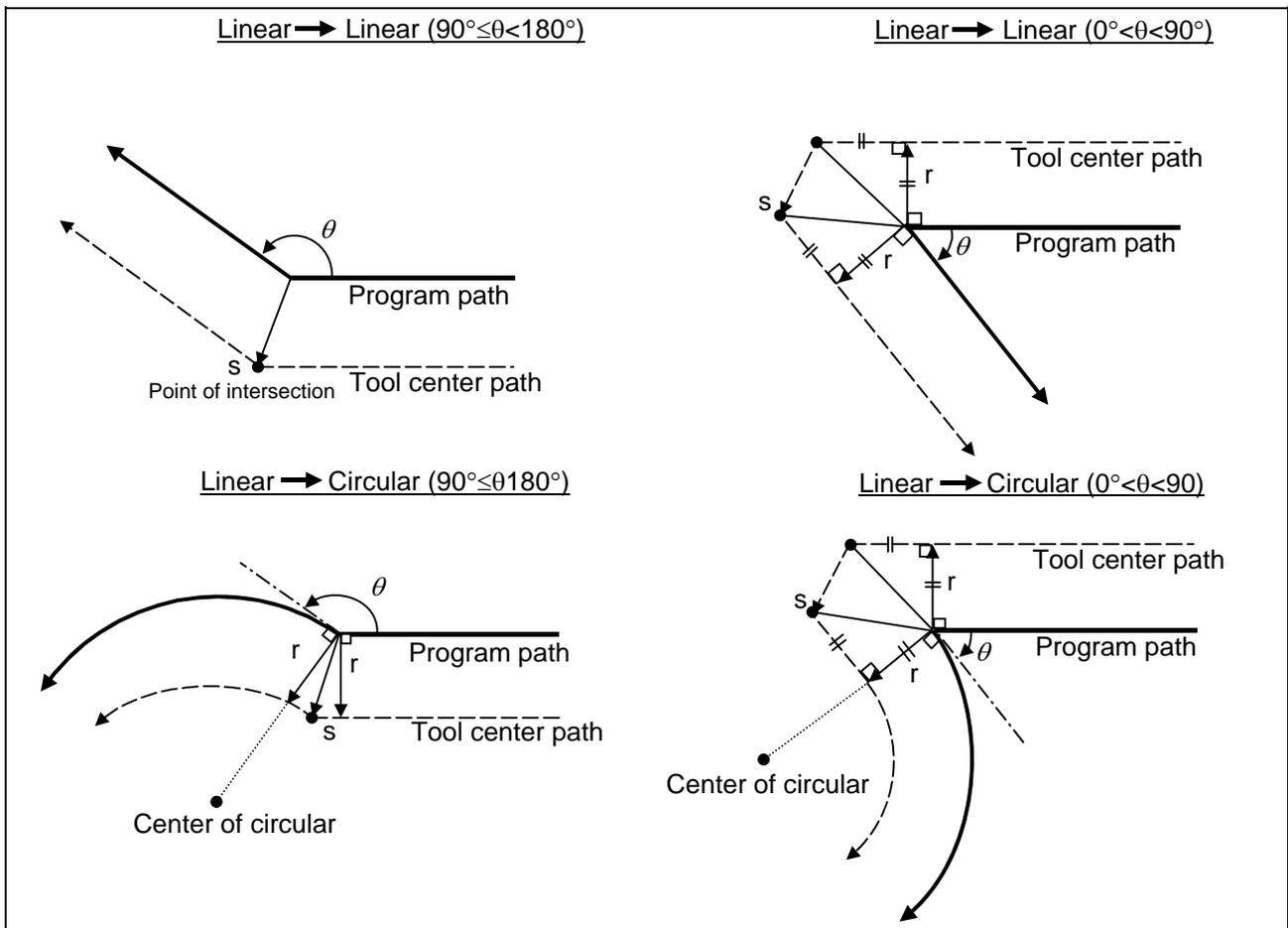
Relative to the program path (G00, G01, G02, G03), the tool center path is found from the straight line/circular arc to make compensation.

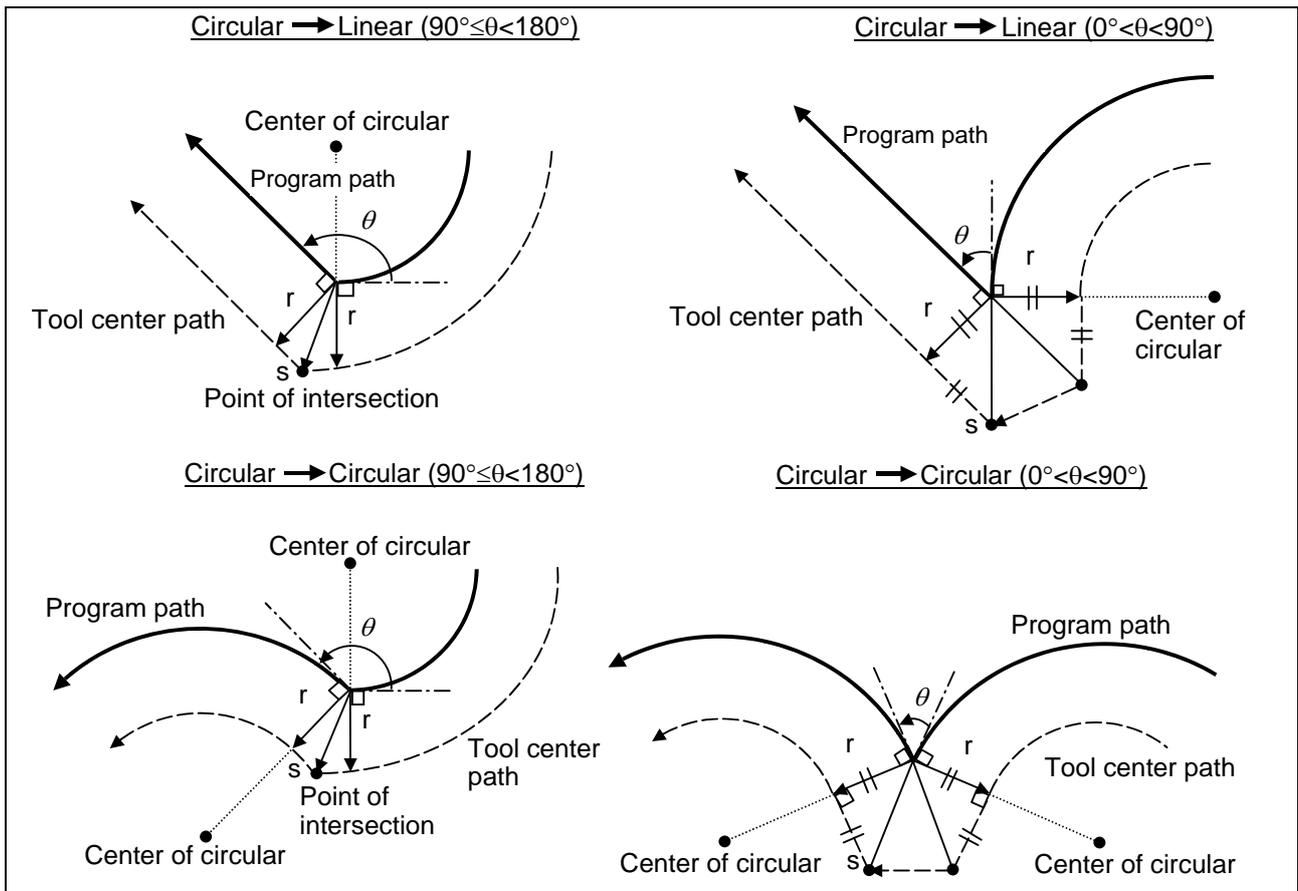
Even if the same compensation command (G41, G42) is issued in the compensation mode, the command will be ignored.

When 4 or more blocks not accompanying movement are commanded continuously in the compensation mode, overcutting or undercutting will result.

When the M00 command has been issued during tool radius compensation, pre-reading is prohibited.

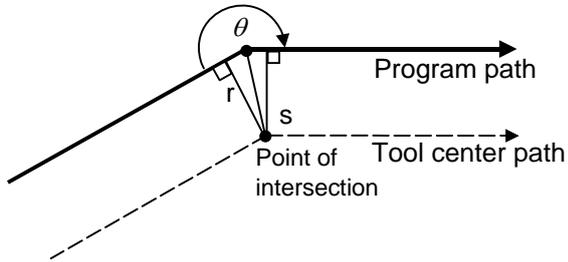
**(1) Machining an outer wall**



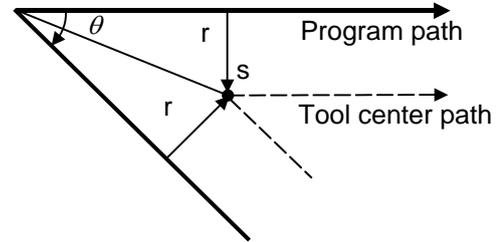


(2) Machining an inner wall

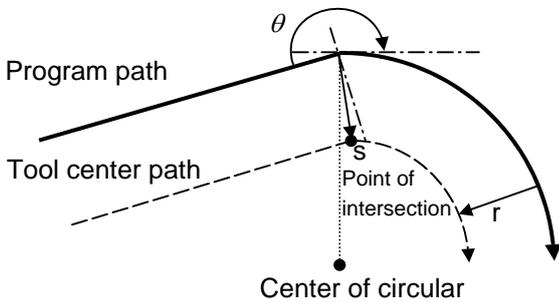
Linear → Linear (Obtuse angle)



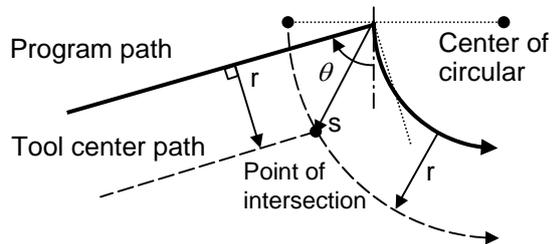
Linear → Linear (Acute angle)



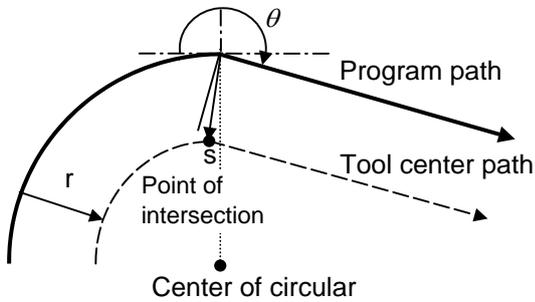
Linear → Circular (Obtuse angle)



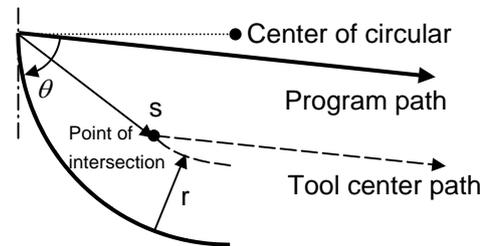
Linear → Circular (Acute angle)



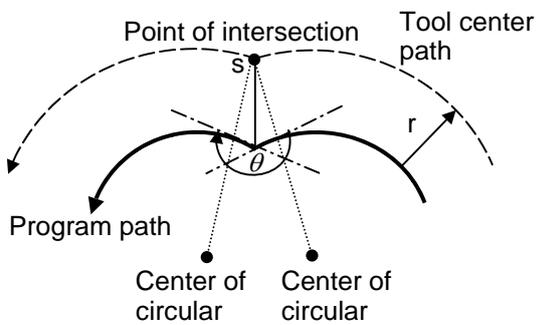
Circular → Linear (Obtuse angle)



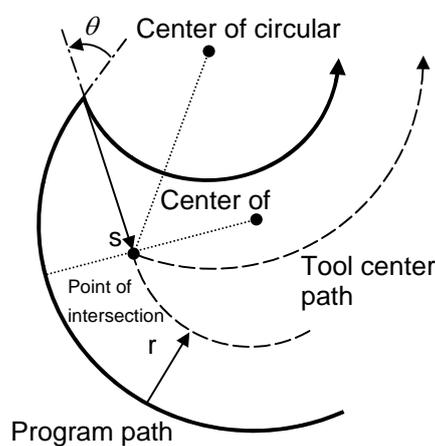
Circular → Linear (Acute angle)



Circular → Linear (Obtuse angle)



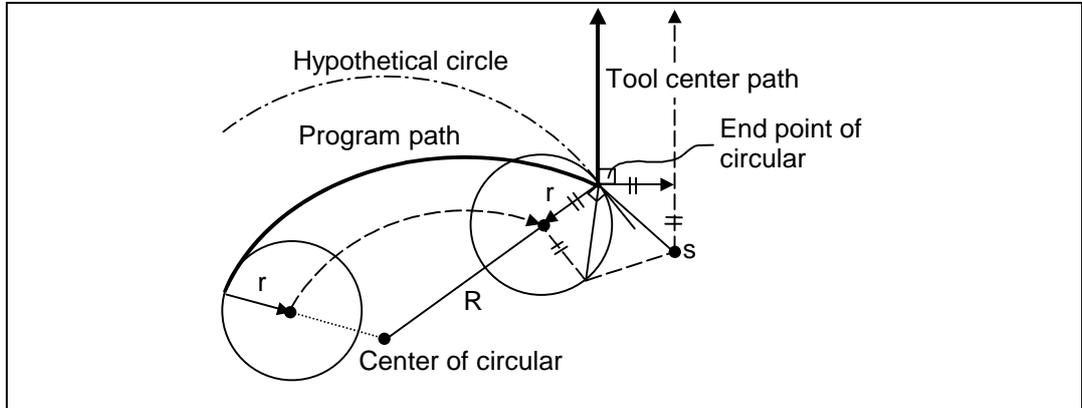
Circular → Linear (Acute angle)



**(3) When the arc end point is not on the arc**

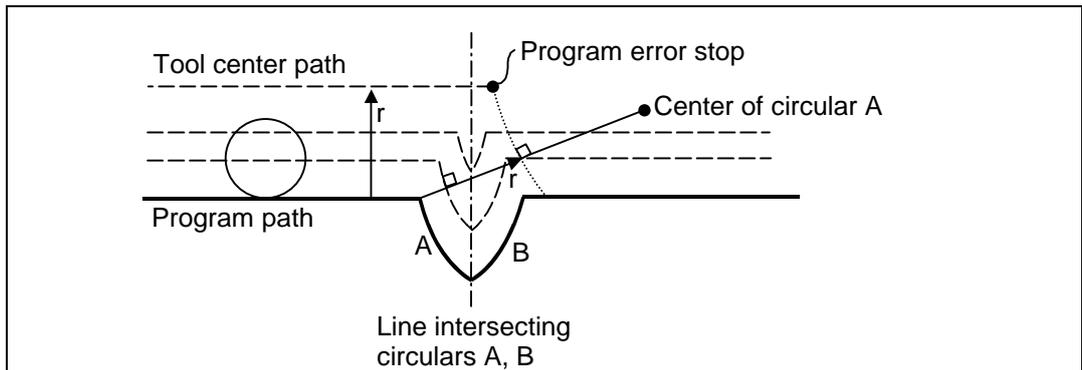
For spiral arc ..... A spiral arc will be interpolated from the start to end point of the arc.

For normal arc command..... If the error after compensation is within parameter "#1084 RadErr", the area from the arc start point to the end point is interpolated as a spiral arc.



**(4) When the inner intersection point does not exist**

In an instance such as that shown in the figure below, the intersection point of arcs A and B may cease to exist due to the offset amount. In such cases, program error (P152) appears and the tool stops at the end point of the previous block.



**Tool radius compensation cancel**

If either of the following conditions is met in the tool radius compensation mode, the compensation will be canceled. However, the movement command must be a command which is not a circular command.

If the compensation is canceled by a circular command, program error (P151) results.

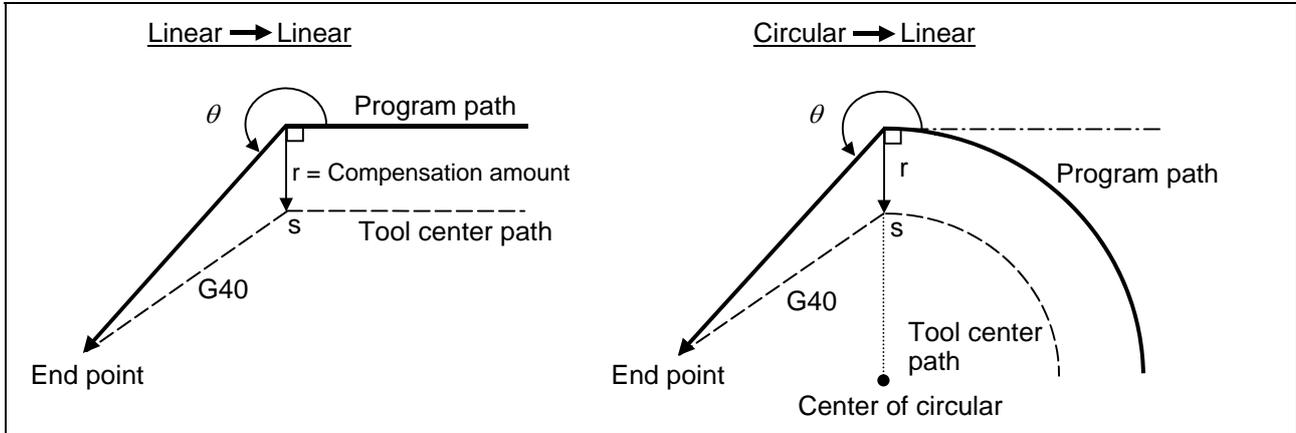
- (1) The G40 command has been executed.
- (2) The D00 tool number has been executed.

The cancel mode is established once the compensation cancel command has been read, 5-block pre-reading is suspended and 1-block pre-reading is made operational.

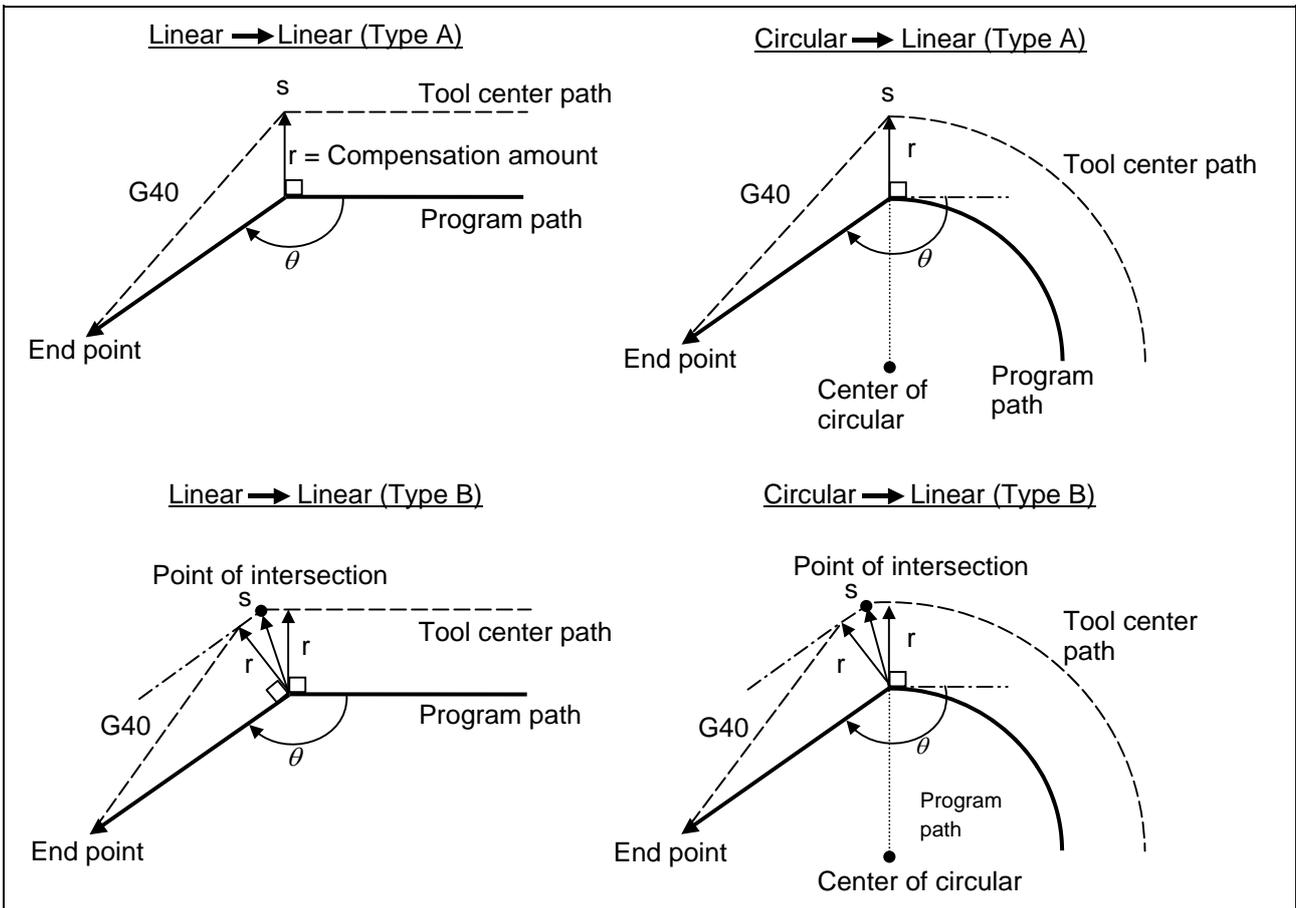


Tool radius compensation cancel operation

(1) For inner side of corner

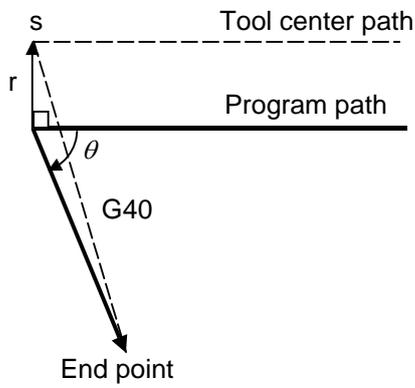


(2) For outer side of corner (obtuse angle)

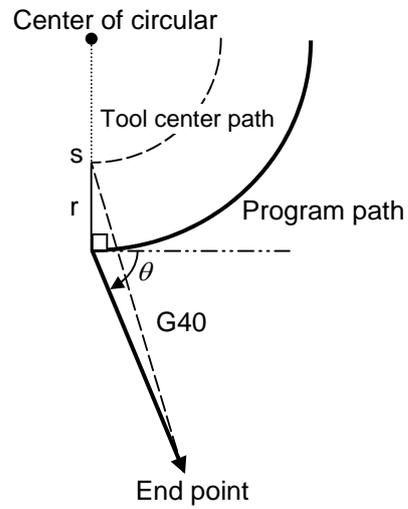


(3) For outer side of corner (acute angle)

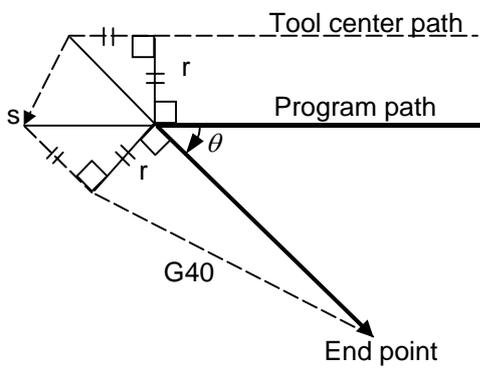
Linear → Linear (Type A)



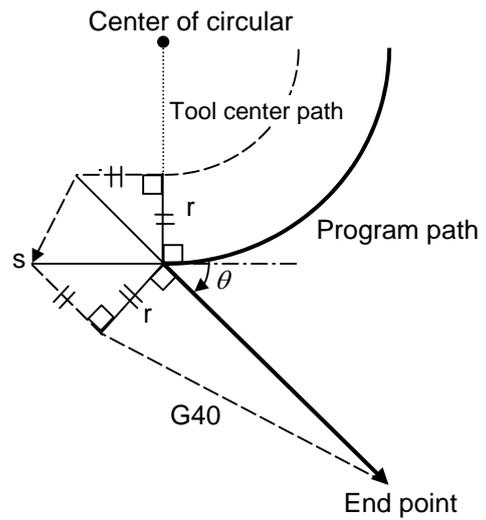
Circular → Linear (Type A)



Linear → Linear (Type B)



Circular → Linear (Type B)

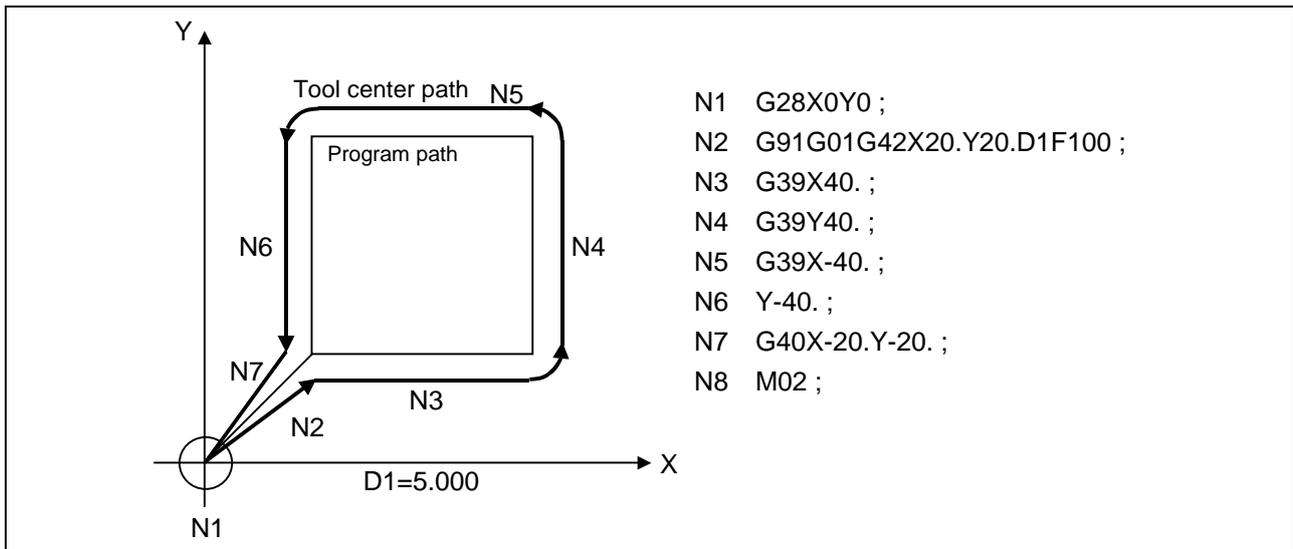
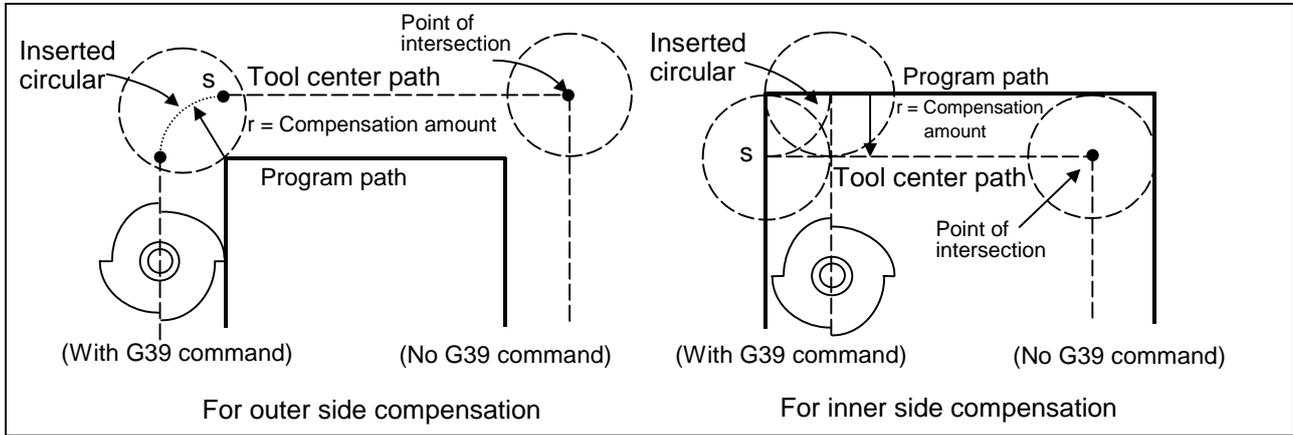


12.4.2 Other Commands and Operations During Tool Radius Compensation



Insertion of corner arc

An arc that uses the compensation amount as the radius is inserted without calculating the point of intersection at the workpiece corner when G39 (corner arc) is commanded.





**Changing and holding of compensation vector**

The compensation vector can be changed or held during tool radius compensation by using the G38 command.

(1) Holding of vector: When G38 is commanded in a block having a movement command, the point of intersection will not be calculated at the program end point, and instead the vector of the previous block will be held.

G38 Xx Yy ;

This can be used for pick feed, etc.

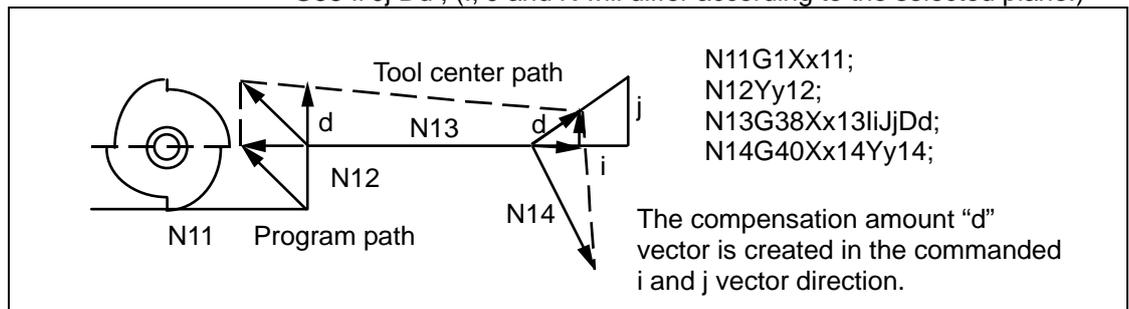
<p>(1) Holding the inside compensation vector</p> <p>N11 Program path N12 N13 Tool center path r1</p> <p>N11G1Xx11; N12G38Xx12Yy12; N13G40Xx13;</p> <p>r1: Vector at N11-N12 block intersection calculation</p>	<p>(2) Holding the outside compensation acute angle</p> <p>Tool center path N11 N12 N13 Program path r1</p> <p>N11G1Xx11Yy11; N13 Program path N12G38Xx12Yy12; N13G40Xx13;</p> <p>r1: Vector at N11-N12 block intersection calculation</p>
---	--

<p>(3) Holding the outside compensation obtuse angle</p> <p>Tool center path N11 N12 N13 Program path r1</p> <p>N11G1Xx11Yy11; N13 Program path N12G38Xx12Yy12; N13G40Xx13;</p> <p>r1: Vector at N11-N12 block intersection calculation</p>
---

(2) Changing of vector: A new compensation vector direction can be commanded with I, J and K, and a new offset amount with D.

(These can be commanded in the same block as the movement command.)

G38 Ii Jj Dd ; (I, J and K will differ according to the selected plane.)



(Note) If G38 is commanded in the same block as the circular block (G02/G03) I and J commands, I and J will be handled as the G38 vector, and an error will occur.



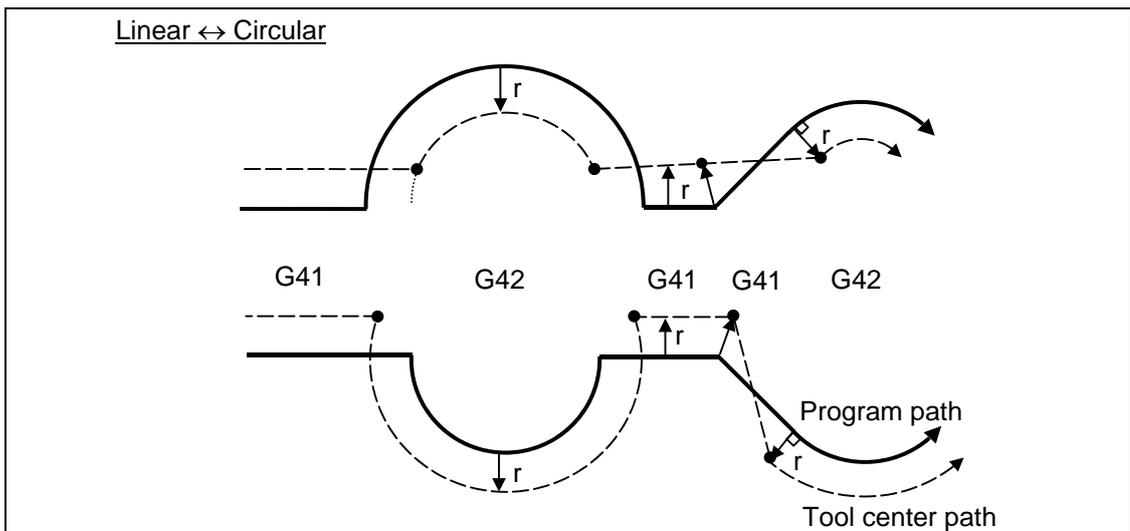
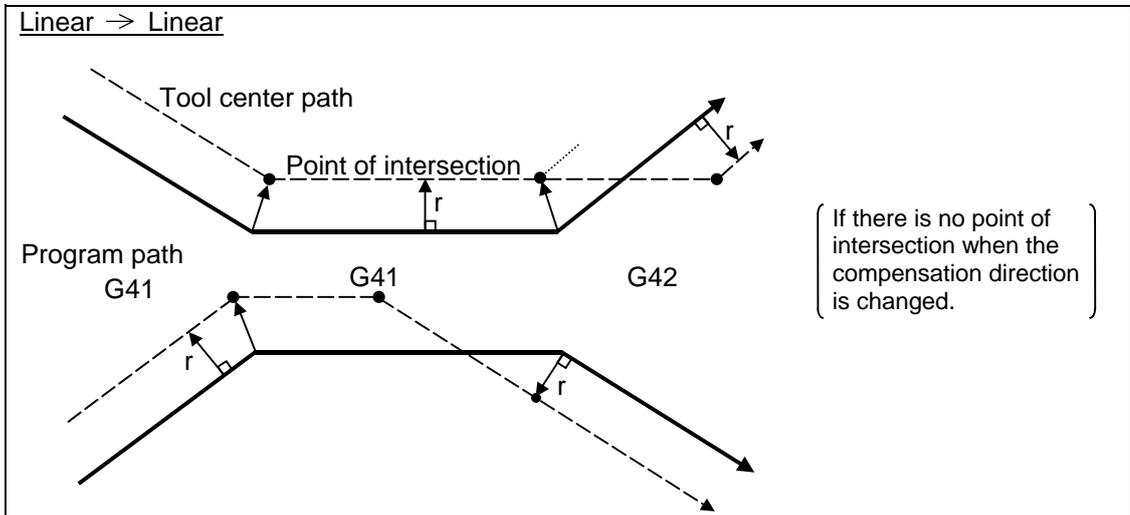
**Changing the compensation direction during tool radius compensation**

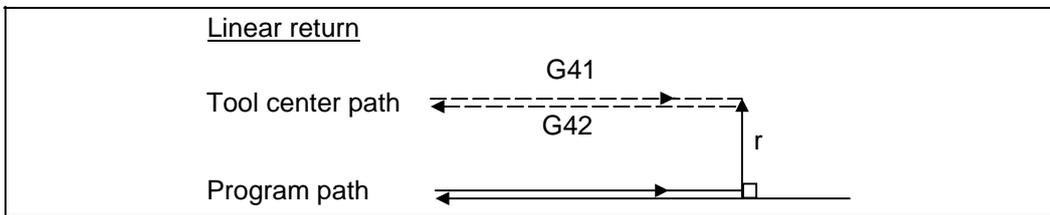
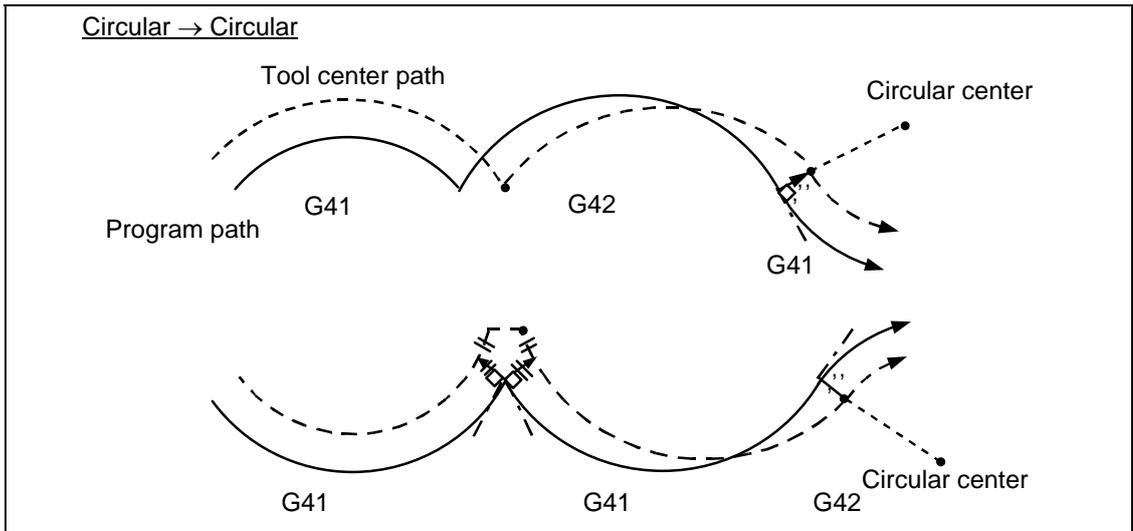
The compensation direction is determined by the tool radius compensation commands (G41, G42) and compensation amount sign.

G code	Compensation amount sign	
	+	-
G41	Left-hand compensation	Right-hand compensation
G42	Right-hand compensation	Left-hand compensation

The compensation direction can be changed by changing the compensation command in the compensation mode without the compensation having to be first canceled. However, no change is possible in the compensation start block and the following block.

Refer to section "12.4.5 General precautions for tool radius compensation" for the movement when the symbol is changed.





Arc exceeding 360° due to compensation

The diagram shows a circular path where the tool center path (dashed line) overlaps itself due to compensation. The program path (solid line) is a single circle with a G41 command. The tool center path starts with a G42 command, follows the program path, and then overlaps itself, creating an 'Uncut section' where the tool has already cut the material. The diagram is labeled with 'G42', 'Tool center path', 'Program path', 'G41', and 'Uncut section'.

In the case below, it is possible that the arc may exceed 360°

- With offset direction selection based on G41/G42
- I, J, K was commanded in G40.

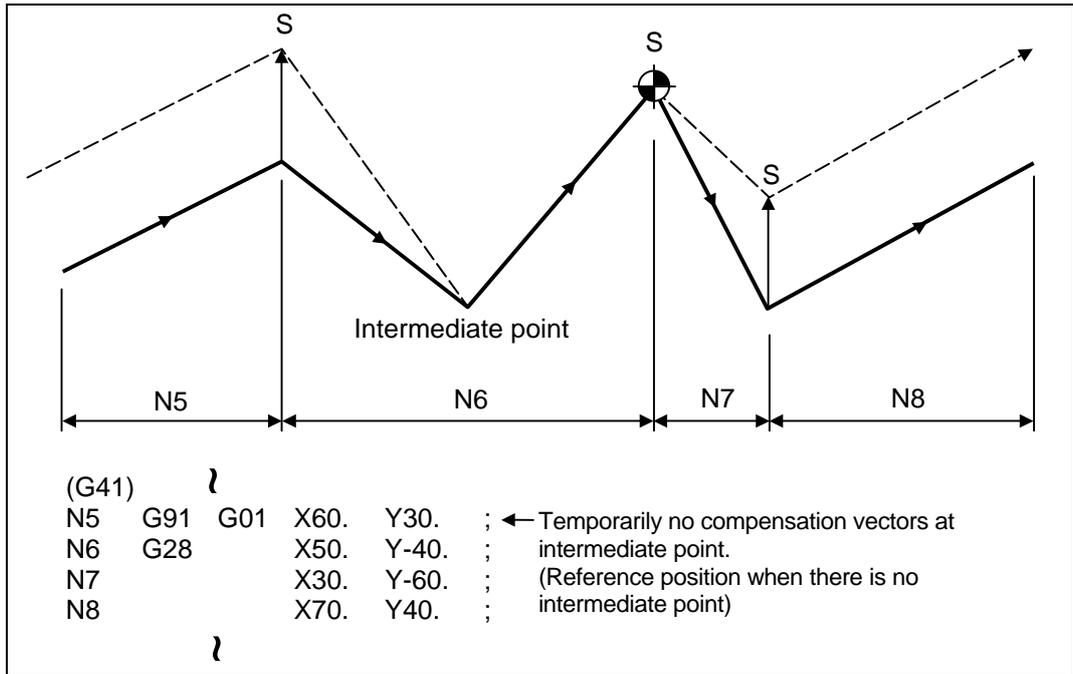
In cases like this the tool center path will pass through a section where the arc is doubled due to the compensation and a section will be left uncut.



**Command for eliminating compensation vectors temporarily**

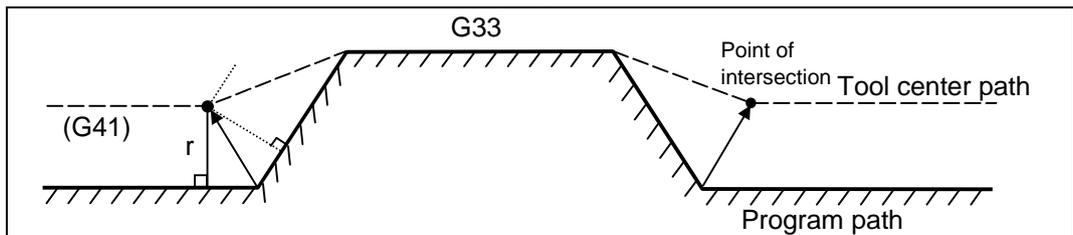
When the following command is issued in the compensation mode, the offset vectors are temporarily eliminated and a return is then made automatically to the compensation mode. In this case, the compensation is not canceled, and the tool goes directly from the intersection point vector to the point without vectors or, in other words, to the programmed command point. When a return is made to the compensation mode, it goes directly to the intersection point.

**(1) Reference position return command**



**(2) G33 thread cutting command**

Tool nose radius compensation does not apply to the G33 block.



**(3) The compensation vector will be eliminated temporarily with the G53 command (basic machine coordinate system selection).**

**(Note 1)** The compensation vectors do not change with the coordinate system setting (G92) command.



**Blocks without movement and pre-read inhibit M command**

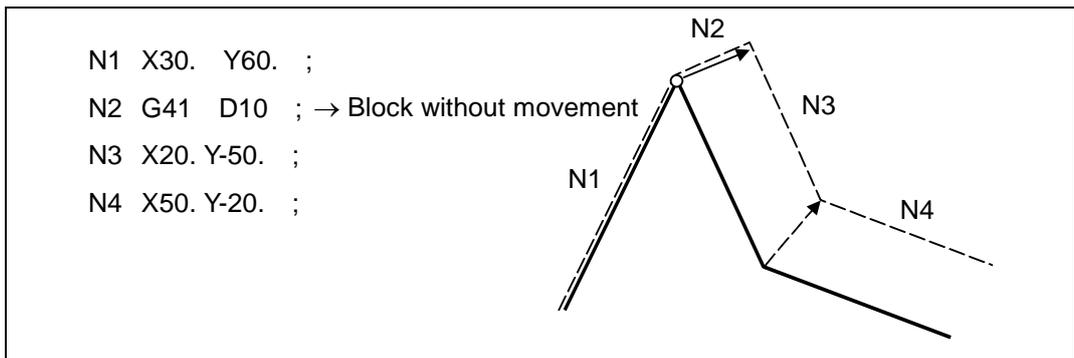
The following blocks are known as blocks without movement.

- a. M03 ; .....M command
  - b. S12 ; .....S command
  - c. T45 ; .....T command
  - d. G04 X500 ; .....Dwell
  - e. G22 X200. Y150. Z100 ; ....Machining inhibit region setting
  - f. G10 L10 P01 R50 ; .....Offset amount setting
  - g. G92 X600. Y400. Z500. ; ...Coordinate system setting
  - h. (G17) Z40. ; .....Movement but not on offset plane
  - i. G90 ; .....G code only
  - j. G91 X0 ; .....Zero movement amount ..... Movement amount is zero
- } No movement

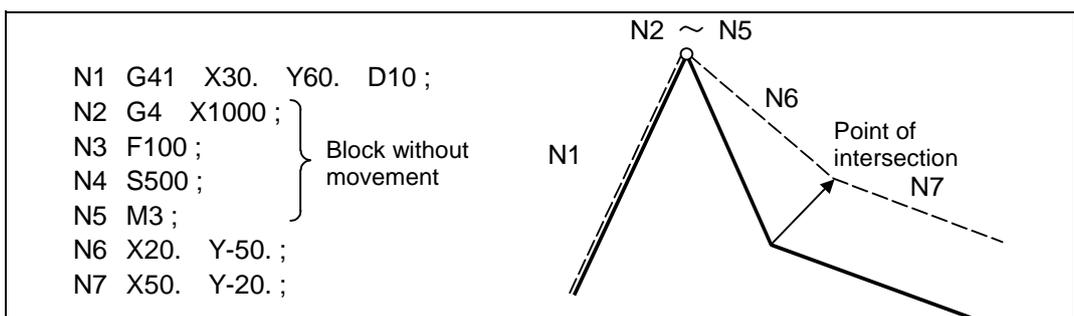
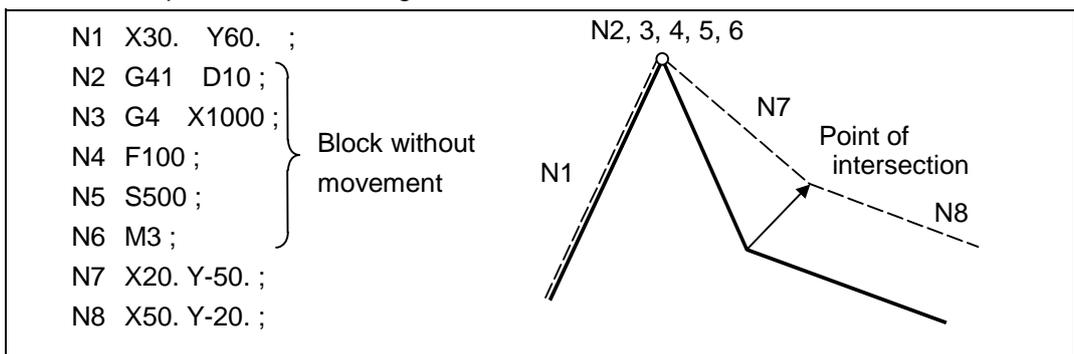
M00, M01, M02 and M30 are handled as pre-read inhibit M codes.

**(1) When command is assigned at start of the compensation**

Perpendicular compensation will be applied on the next movement block.

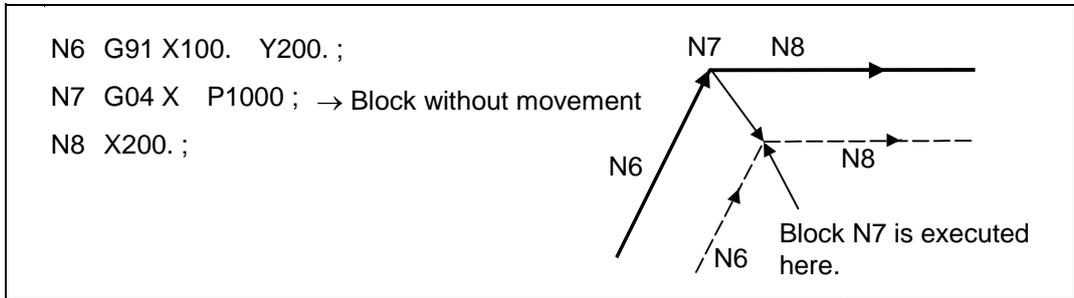


If four or more blocks containing no move command continue or if there is a pre-read inhibit M code, no compensation vector is generated.

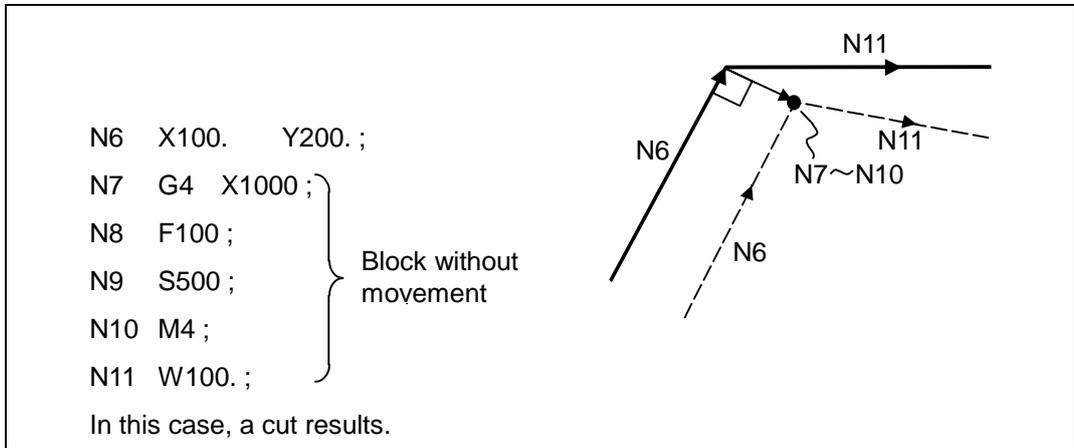


**(2) When command is assigned in the compensation mode**

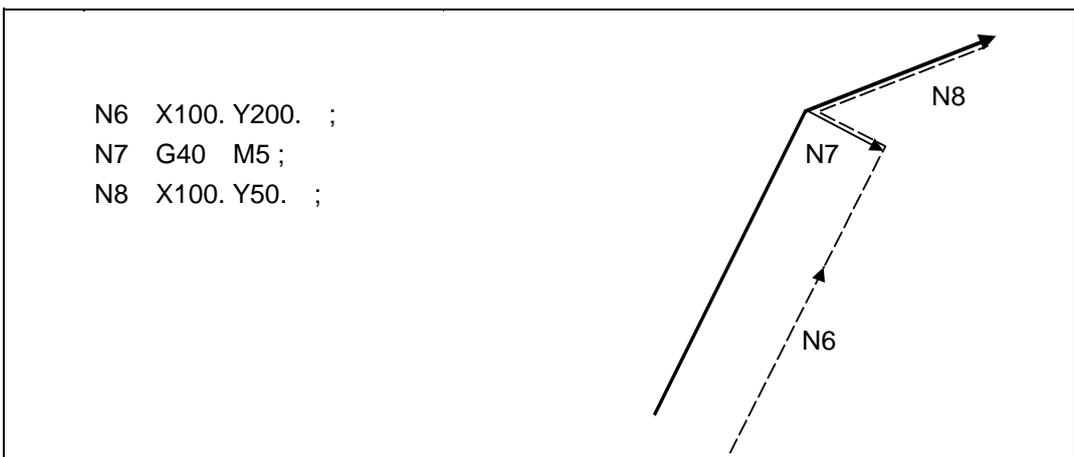
When 4 or more blocks without movement follow in succession in the compensation mode or when there is no pre-read inhibit M code, the intersection point vectors will be created as usual.



When 4 or more blocks without movement follow in succession or if there is a pre-read inhibit M code, the offset vectors are created perpendicularly at the end point of the previous block.



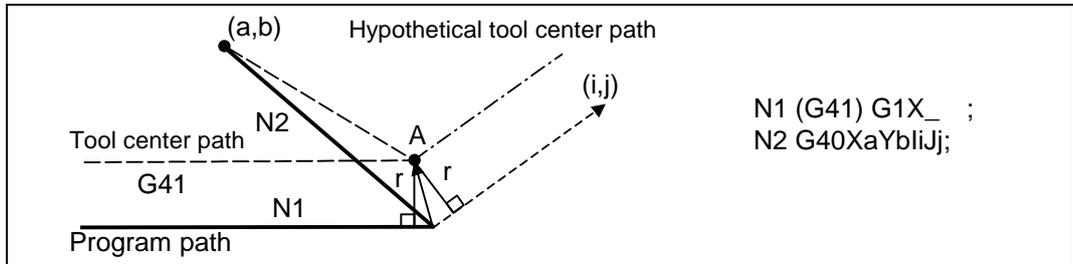
**(3) When commanded together with compensation cancel**



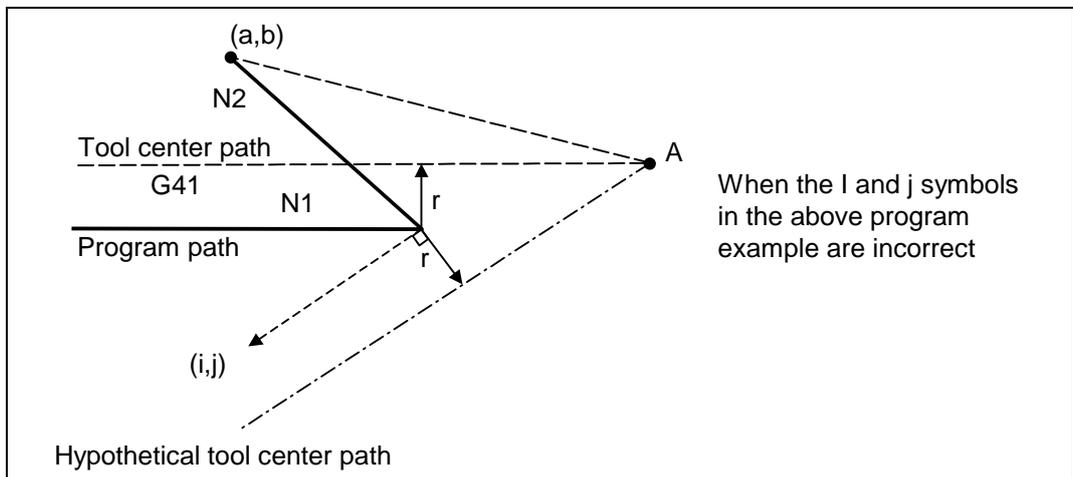


When I, J, K are commanded in G40

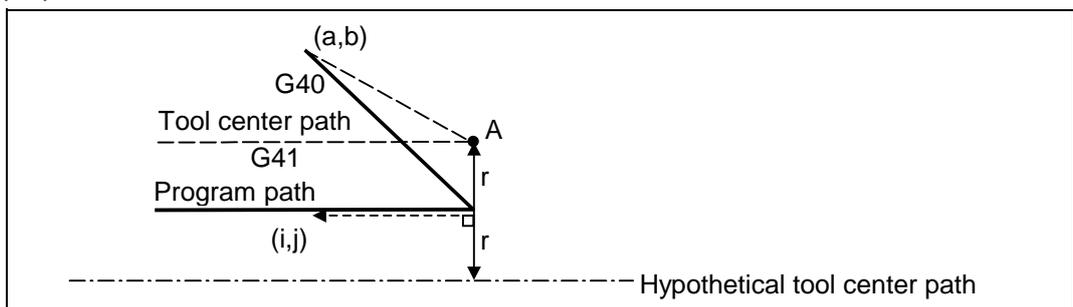
- (1) If the final movement command block in the four blocks before the G40 block is the G41 or G42 mode, it will be assumed that the movement is commanded in the vector I, J or K direction from the end point of the final movement command. After interpolating between the hypothetical tool center path and point of intersection, it will be canceled. The compensation direction will not change.



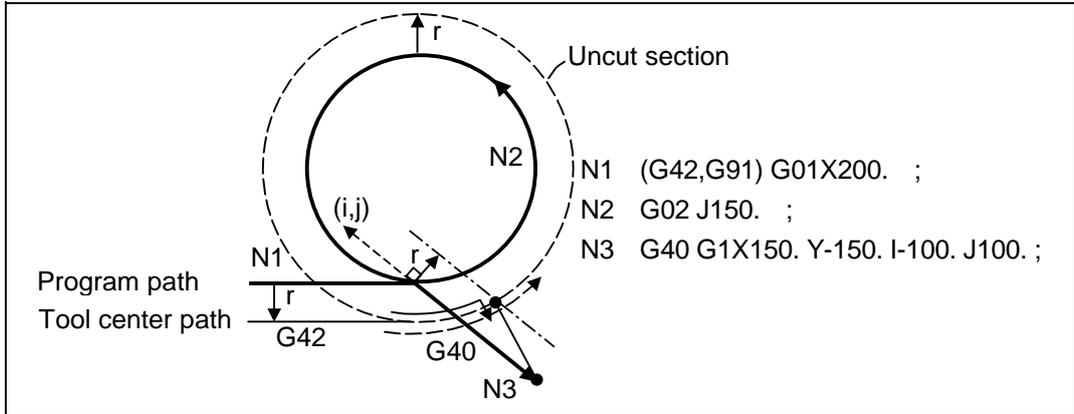
In this case, the point of intersection will always be obtained, regardless of the compensation direction, even when the commanded vector is incorrect as shown below.



If the compensation vector obtained with point of intersection calculation is extremely large, a perpendicular vector will be created in the block before G40.



- (2) If the arc is 360° or more due to the details of I, J and K at G40 after the arc command, an uncut section will occur.

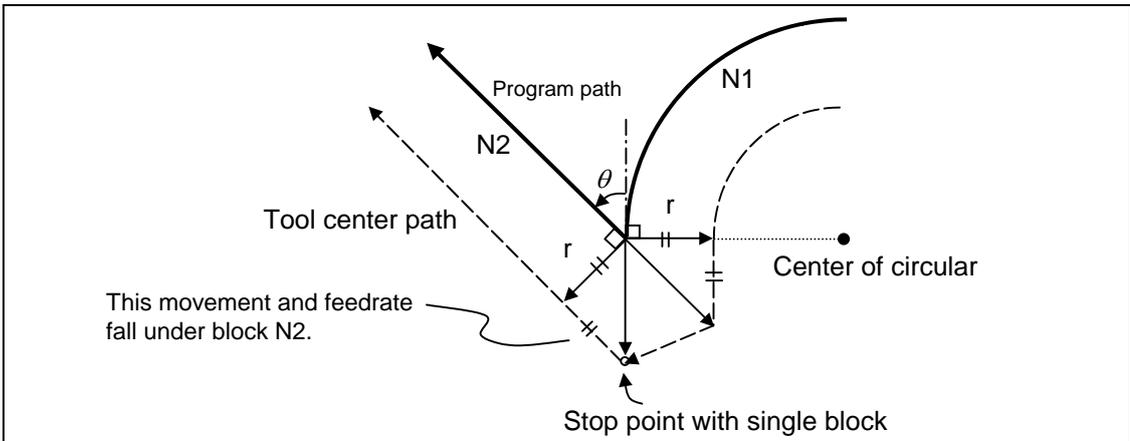


**Corner movement**

When a multiple number of offset vectors are created at the joints between movement command blocks, the tool will move in a straight line between those vectors.

This action is called corner movement.

When the vectors do not coincide, the tool moves in order to machine the corner although this movement is part and parcel of the joint block. Consequently, operation in the single block mode will execute the previous block + corner movement as a single block and the remaining joining movement + following block will be executed as a single block in the following operation.



12.4.3 G41/G42 Commands and I, J, K Designation



Function and purpose

The compensation direction can be intentionally changed by issuing the G41/G42 command and I, J, K in the same block.



Command format

G17 (XY plane) G41/G42 X\_\_ Y\_\_ I\_\_ J\_\_ ;  
 G18 (ZX plane) G41/G42 X\_\_ Z\_\_ I\_\_ K\_\_ ;  
 G19 (YZ plane) G41/G42 Y\_\_ Z\_\_ J\_\_ K\_\_ ;

Assign an linear command (G00, G01) in a movement mode.

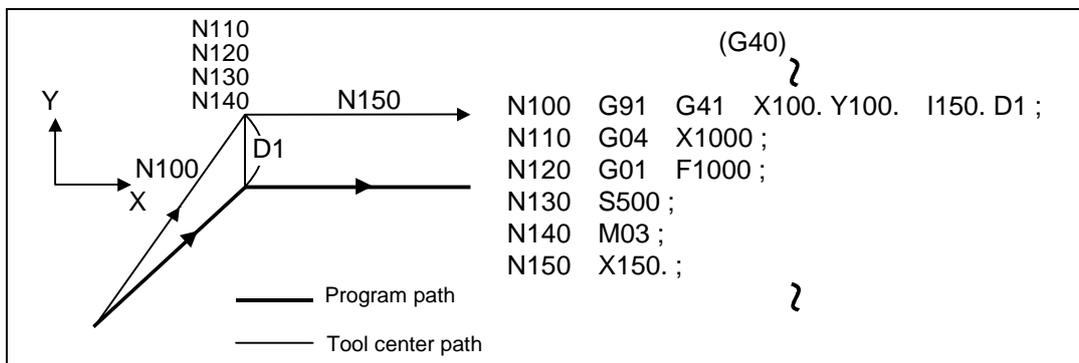


I, J type vectors (G17 XY plane selection)

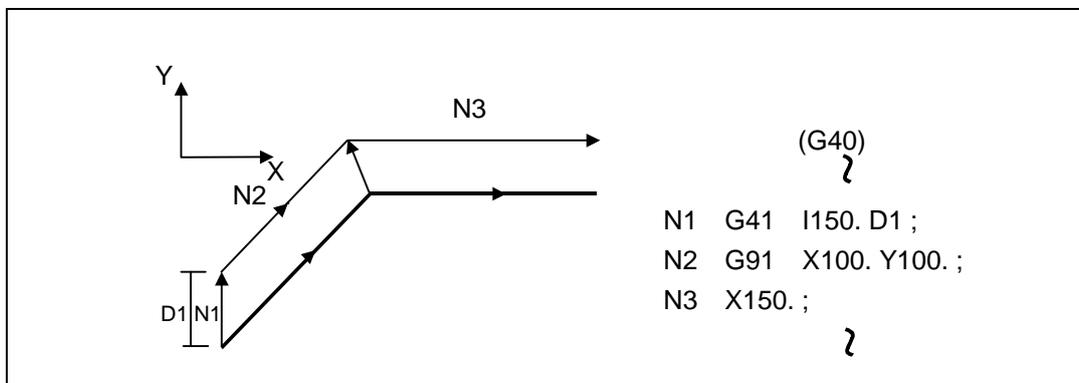
The new I, J type vector (G17 plane) created by this command is now described. (Similar descriptions apply to vector I, K for the G18 plane and to J, K for the G19 plane.)

As shown in the figures, the vectors with a size equivalent to the offset amount are made to serve as the I, J type compensation vector perpendicularly to the direction designated by I, J without the intersection point of the programmed path being calculated. the I, J vector can be commanded even in the mode (G41/G42 mode in the block before) and even at the compensation start (G40 mode in the block before).

(1) When I, J is commanded at compensation start



(2) When there are no movement commands at the compensation start.



(3) When I, J has been commanded in the G41/G42 mode (G17 plane)

?

```

(G17 G41 G91)
N100 G41 G00X150. J50. ;
N110 G02 I150. ;
N120 G00 X-150. ;
    
```

?

(1) I, J type vector  
 (2) Intersection point calculation type vector

————— Program path  
 - - - - - Tool center path  
 ······ Tool path after interrupt

(Reference)

(a) G18 plane

?

```

(G18 G41 G91)
N100 G41 G00 Z150. I50. ;
N110 G02 K50. ;
N120 G00 Z-150. ;
    
```

?

(b) G19 plane

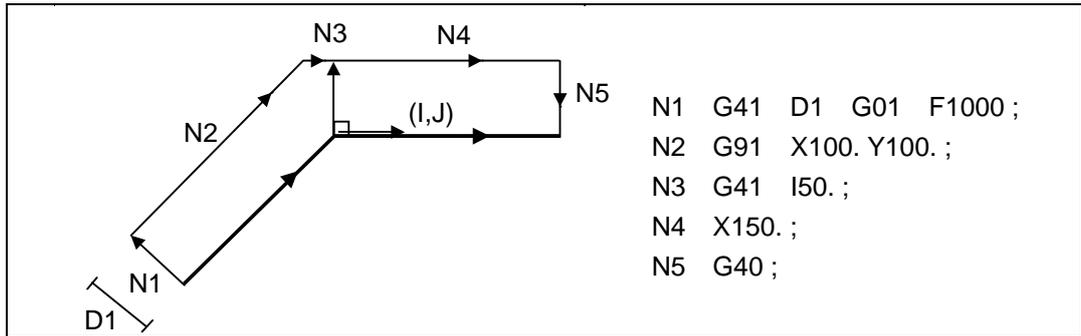
?

```

(G19 G41 G91)
N100 G41 G00 Y150. K50. ;
N110 G02 J50. ;
N120 G00 Y-150. ;
    
```

?

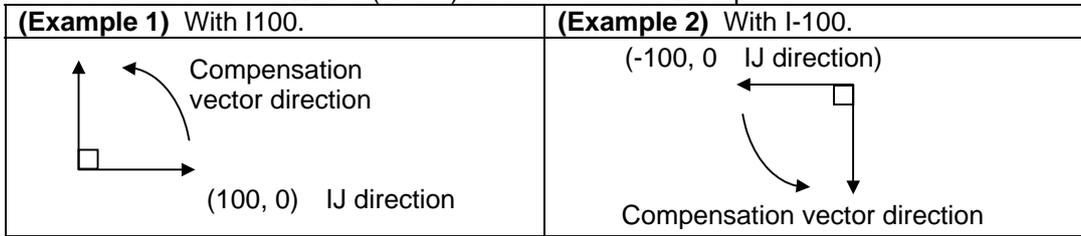
(4) When I, J has been commanded in a block without movement



Direction of compensation vectors

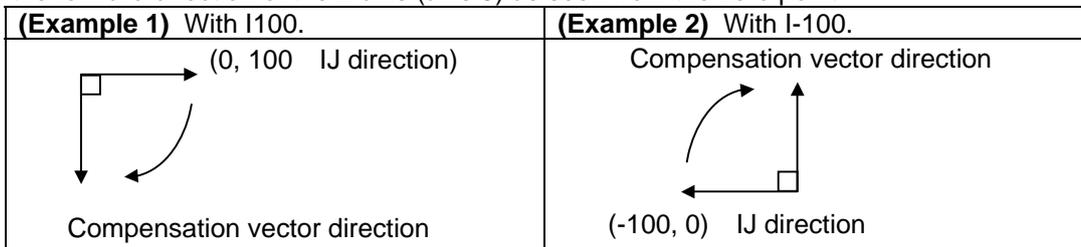
(1) In G41 mode

Direction produced by rotating the direction commanded by I, J through 90° to the left from the forward direction of the Z axis (axis 3) as seen from the zero point



(2) In G42 mode

Direction produced by rotating the direction commanded by I, J through 90° to the right from the forward direction of the Z axis (axis 3) as seen from the zero point





**Selection of compensation modal**

The G41 or G42 modal can be selected at any time.

```

N1 G28 X0 Y0 ;
N2 G41 D1 F1000 ;
N3 G01 G91 X100. Y100. ;
N4 G42 X100. I100. J-100. D2 ;
N5 X100. Y-100. ;
N6 G40 ;
N7 M02 ;
%
```



**Compensation amount for compensation vectors**

The offset amounts are determined by the offset number (modal) in the block with the I, J designation.

< Example 1 >

```

(G41 D1 G91)
?
N100 G41 X150. I50. ;
N110 X100. Y-100. ;
?
```

Vector A is the offset amount entered in offset number modal D1 in the N200 block.

< Example 2 >

```

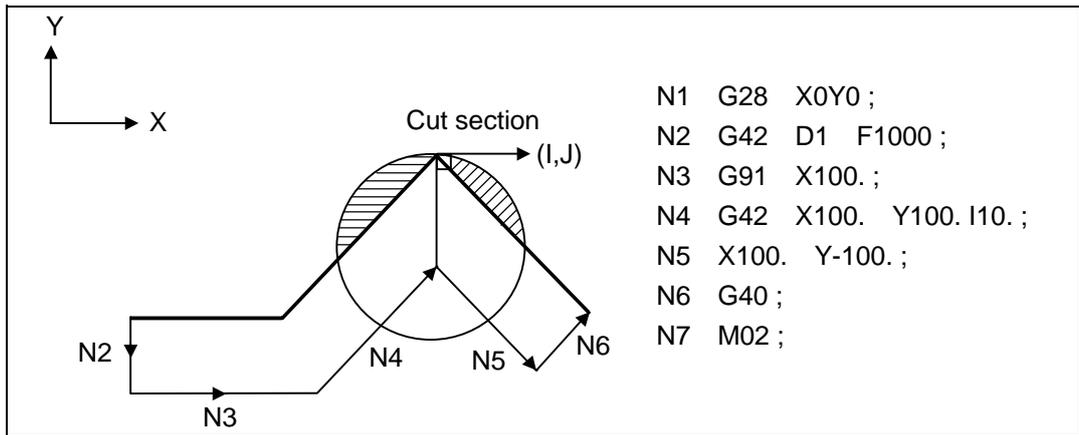
(G41 D1 G91)
?
N200 G41 X150. I50. D2 ;
N210 X100. Y-100. ;
?
```

Vector B is the offset amount entered in offset number modal D2 in the N200 block.



**Precautions**

- (1) Issue the I, J type vector in a linear mode (G0, G1). If it is issued in an arc mode at the start of compensation, program error (P151) will occur.  
An IJ designation in an arc mode functions as an arc center designation in the compensation mode.
- (2) When the I, J type vector has been designated, it is not deleted (avoidance of interference) even if there is interference.  
Consequently, overcutting may arise in such a case.

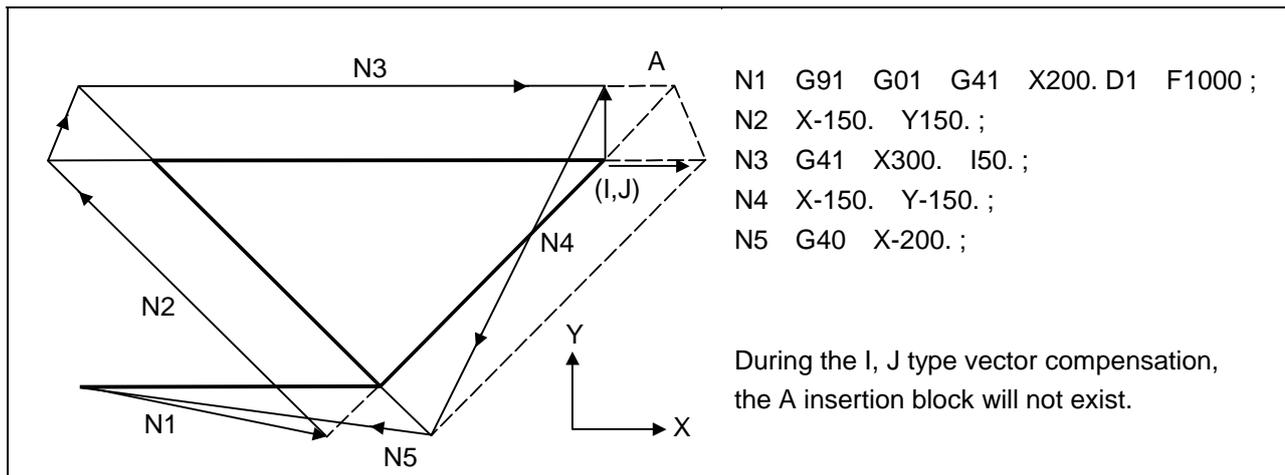


- (3) The vectors differ for the G38 I \_J\_ (K\_) command and the G41/G42 I \_J\_(K\_) command.

	<b>G38</b>	<b>G41/G42</b>
Example	} (G41) } G38 G91 X100. I50. J50. ; }	} (G41) } G41 G91 X100. I50. J50. ; }
	Vector in IJ direction having a compensation amount size	Vector perpendicular in IJ direction and having a compensation amount size

(4) Refer to the following table for the offset methods based on the presence and/or absence of the G41 and G42 commands and I, J, (K) command.

G41/G42	I, J (K)	Offset method
No	No	Intersection point calculation type vector
No	Yes	Intersection point calculation type vector
Yes	No	Intersection point calculation type vector
Yes	Yes	I, J, type vector No insertion block



12.4.4 Interrupts During Tool Radius Compensation

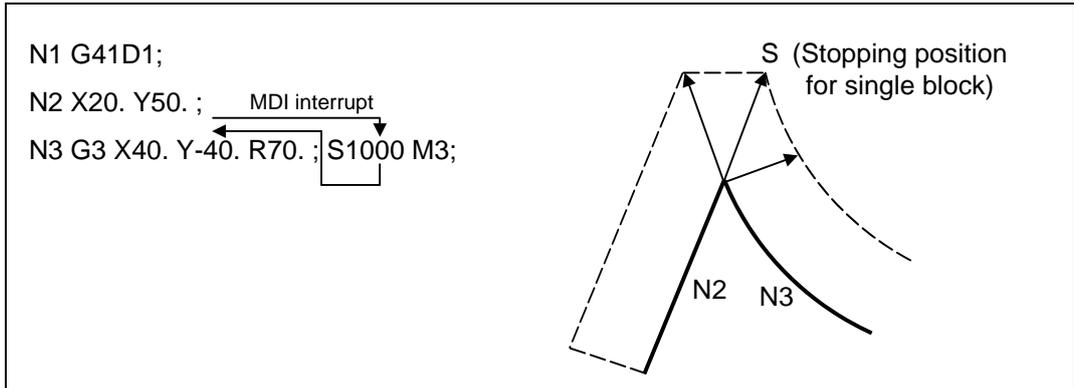


MDI interrupt

Tool radius compensation is valid in any automatic operation mode-whether tape, memory or MDI operation.

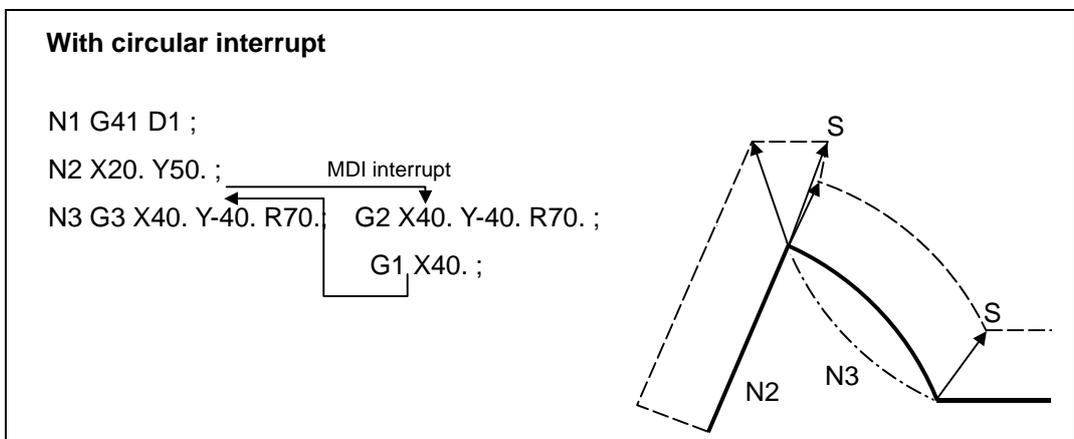
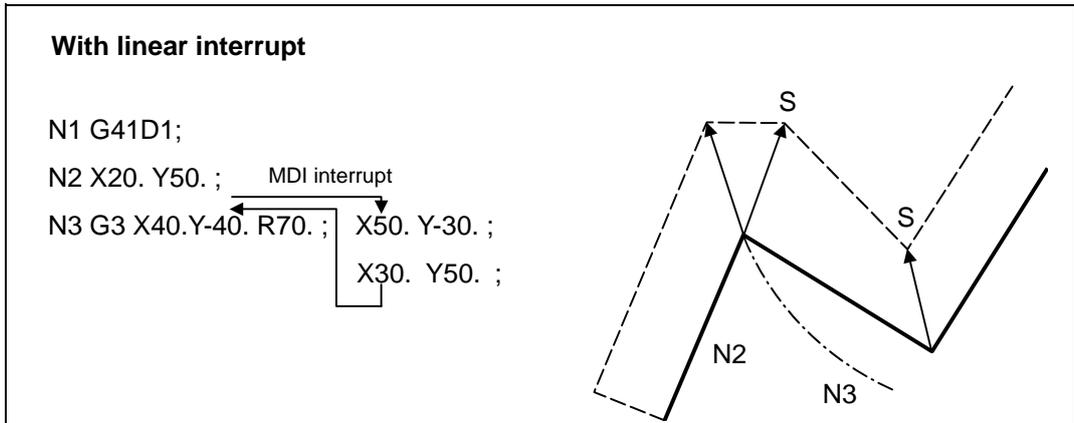
An interrupt based on MDI will give the result as in the figure below after block stop during tape or memory operation.

(1) Interrupt without movement (tool path does not change)



(2) Interrupt with movement

The offset vectors are automatically re-calculated at the movement block after interrupt.

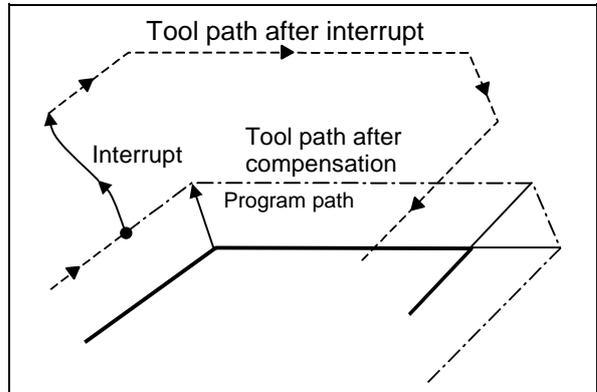




**Manual interrupt**

**(1) Interrupt with manual absolute OFF.**

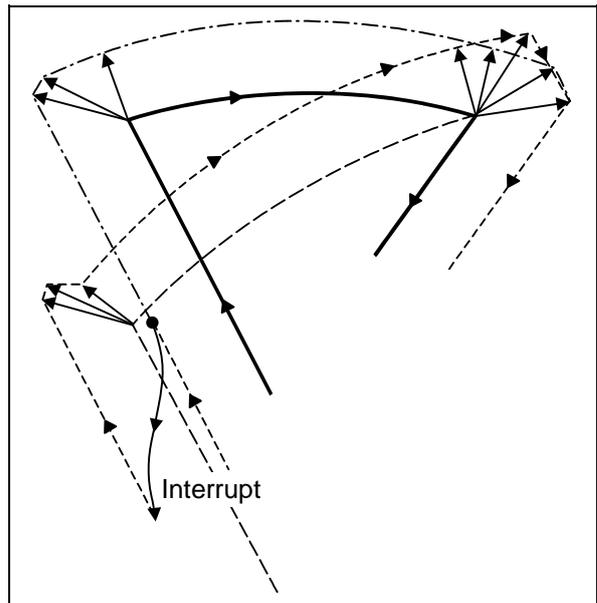
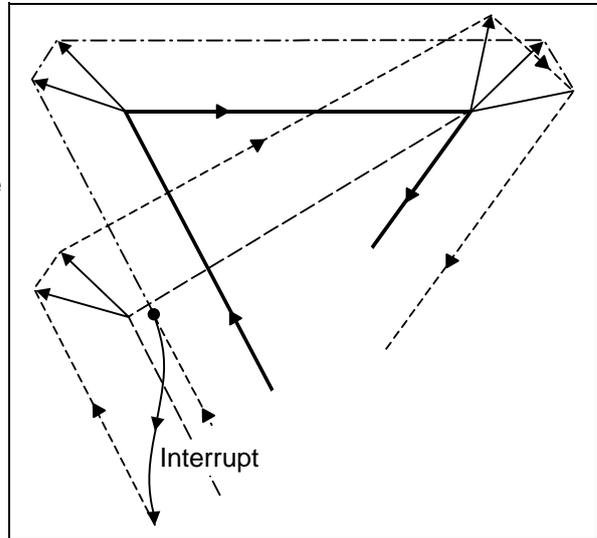
The tool path is shifted by an amount equivalent to the interrupt amount.



**(2) Interrupt with manual absolute ON.**

In the incremental value mode, the same operation results as with manual absolute OFF.

In the absolute value mode, however, the tool returns to its original path at the end point of the block following the interrupted block, as shown in the figure.



## 12.4.5 General Precautions for Tool Radius Compensation



## Precautions

**(1) Designating the offset amounts**

The offset amounts can be designated with the D code by designating an offset amount No. Once designated, the D code is valid until another D code is commanded. If an H code is designated, the program error (P170) No COMP No will occur.

Besides being used to designate the compensation amounts for tool radius compensation, the D codes are also used to designate the compensation amounts for tool position compensation.

**(2) Changing the offset amounts**

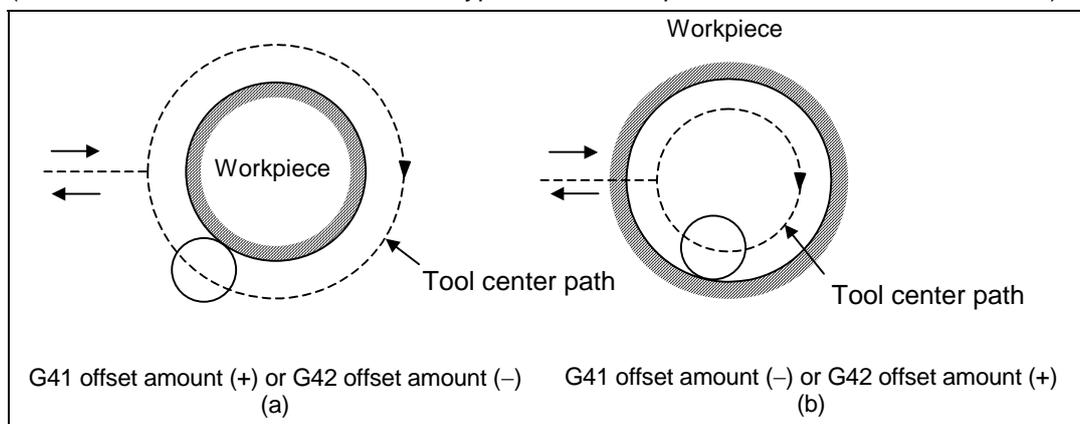
Offset amounts are normally changed when a different tool has been selected in the compensation cancel mode. However, when an amount is changed in the compensation mode, the vectors at the end point of the block are calculated using the offset amount designated in that block.

**(3) Offset amount symbols and tool center path**

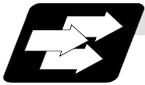
If the offset amount is negative (-), the figure will be the same as if G41 and G42 are interchanged. Thus, the axis that was rotating around the outer side of the workpiece will rotate around the inner side, and vice versa.

An example is shown below. Normally, the offset amount is programmed as positive (+). However, if the tool path center is programmed as shown in (a) and the offset amount is set to be negative (-), the movement will be as shown in (b). On the other hand, if the program is created as shown in (b) and the offset amount is set to be negative (-), the movement will be as shown in (a). Thus, only one program is required to execute machining of both male and female shapes. The tolerance for each shape can be randomly determined by adequately selecting the offset amount.

(Note that a circle will be divided with type A when compensation is started or canceled.)



12.4.6 Changing of Compensation No. During Compensation Mode

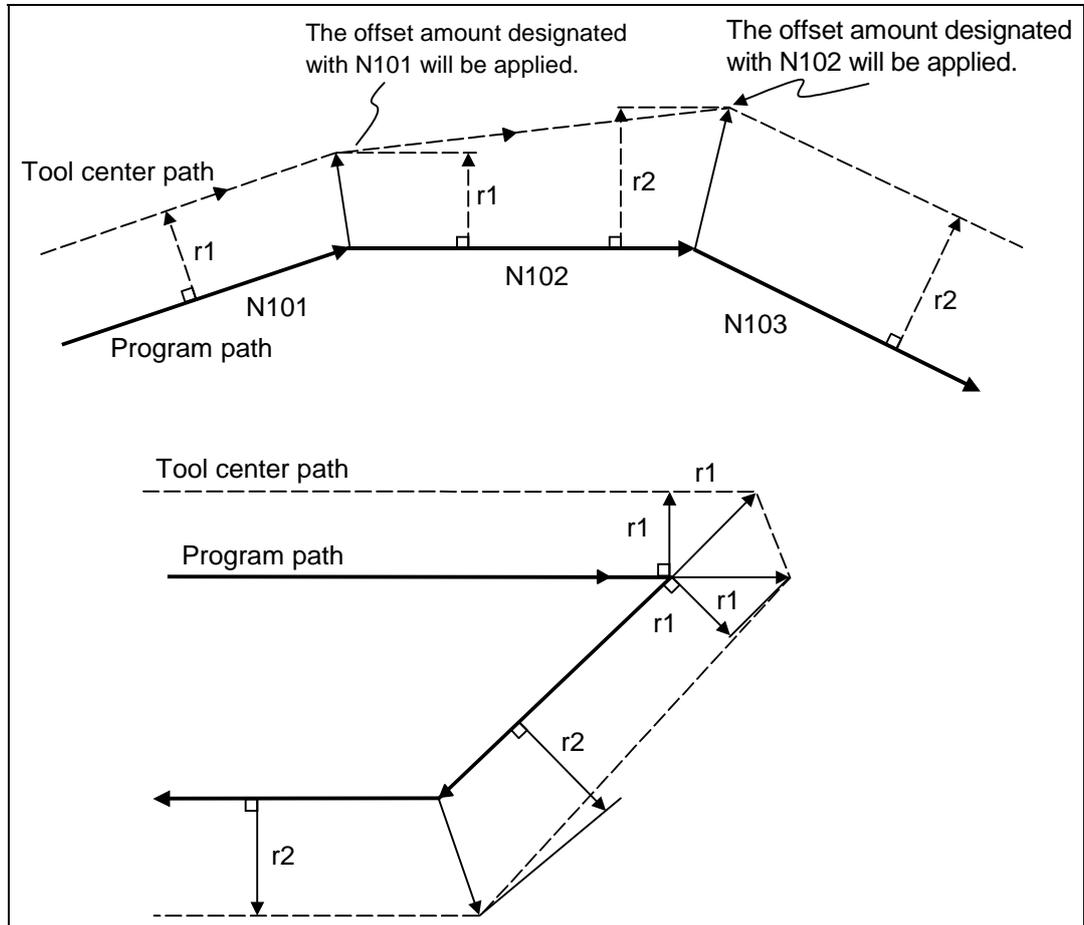


Function and purpose

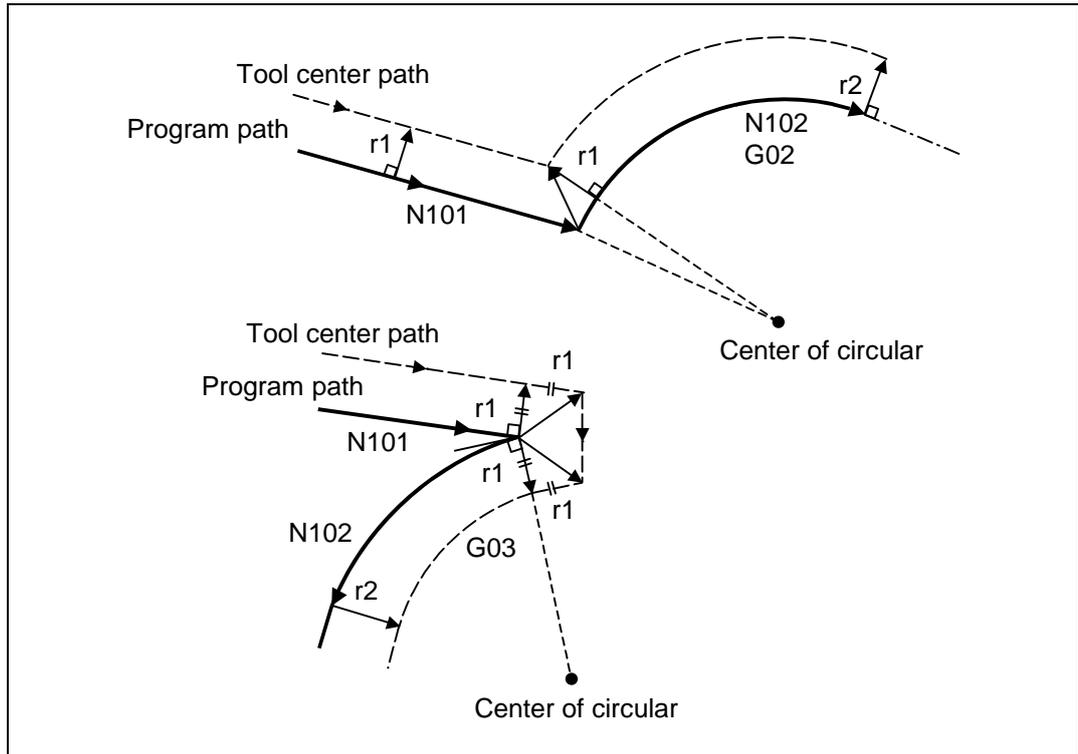
As a principle, the compensation No. must not be changed during the compensation mode. If changed, the movement will be as shown below.  
When offset No. (compensation amount) is changed:

```
G41 G01 ..... Dr1 ;
    (α = 0, 1, 2, 3)
N101      G0α   Xx1   Yy1   ;
N102      G0α   Xx2   Yy2   Dr2 ; ..... Offset No. changed
N103      G0α   Xx3   Yy3   ;
```

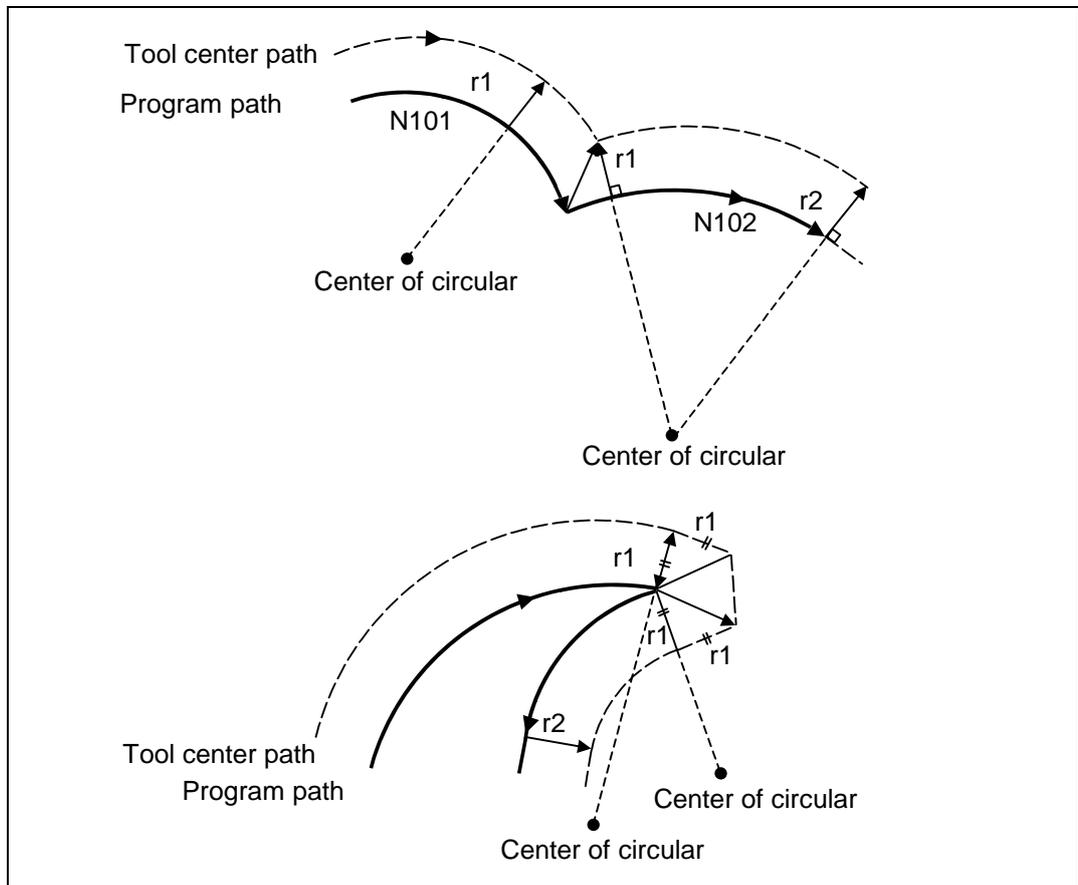
(1) During linear → linear



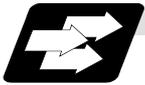
(2) Linear  $\Rightarrow$  circular



(3) Circular  $\Rightarrow$  circular



12.4.7 Start of Tool Radius Compensation and Z Axis Cut in Operation



Function and purpose

Often when starting cutting, a method of applying a radius compensation (normally the XY plane) beforehand at a position separated for the workpiece, and then cutting in with the Z axis is often used. When using this method, create the program so that the Z axis movement is divided into the two steps of rapid traverse and cutting feed after nearing the workpiece.



Example of program

When the following type of program is created:

```

N1 G91 G00 G41 X 500. Y 500. D1 ;
N2 S1000 ;
N3 M3 ;
N4 G01 Z-300. F1 ;
N6     Y 100. F2 ;
      .
      .
      .
    
```

With this program, at the start of the N1 compensation the program will be read to the N6 block. The relation of N1 and N6 can be judged, and correct compensation can be executed as shown above.

If the above program's N4 block is divided into two

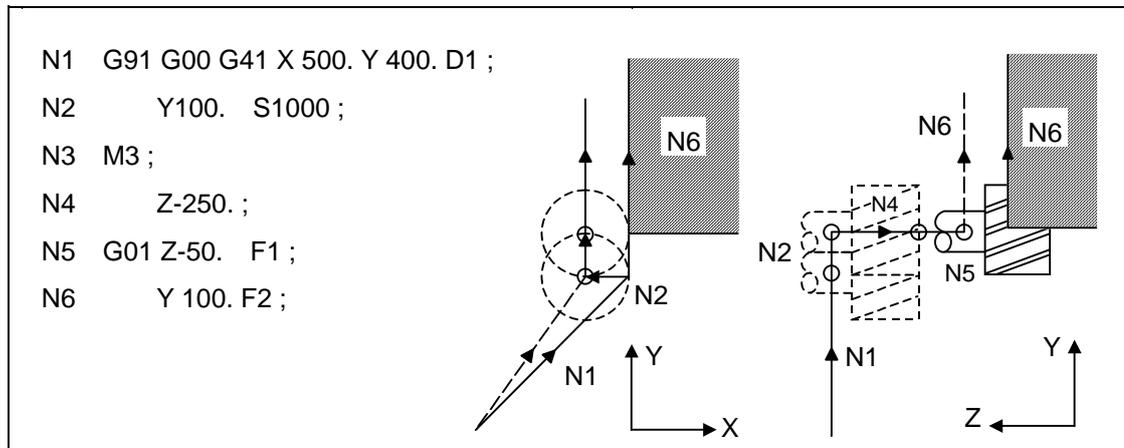
```

N1 G91 G00 G41 X 500. Y 500. D1;
N2 S1000 ;
N3 M3 ;
N4     Z-250. ;
N5 G01 Z-50. F1 ;
N6     Y 100. F2 ;
    
```

In this case, the four blocks N2 to N5 do not have a command in the XY plane, so when the N1 compensation is started, the program cannot be read to the N6 block.

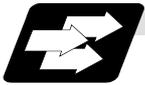
As a result, the compensation is done based only on the information in the N1 block, and the compensation vector is not created at the start of compensation. Thus, an excessive cut in occurs as shown above.

In this case, consider the calculation of the inner side, and before the Z axis cutting, issue a command in the same direction as the direction that the Z axis advances in after lowering, to prevent excessive cutting.



The movement is correctly compensated as the same direction as the N6 advance direction is commanded in N2.

## 12.4.8 Interference Check



## Function and purpose

## (1) Outline

A tool, whose radius has been compensated with the tool radius compensation function by the usual 2-block pre-read, may sometimes cut into the workpiece. This is known as interference, and interference check is the function which prevents this from occurring.

There are three types of interference check, as indicated below, and each can be selected for use by parameter.

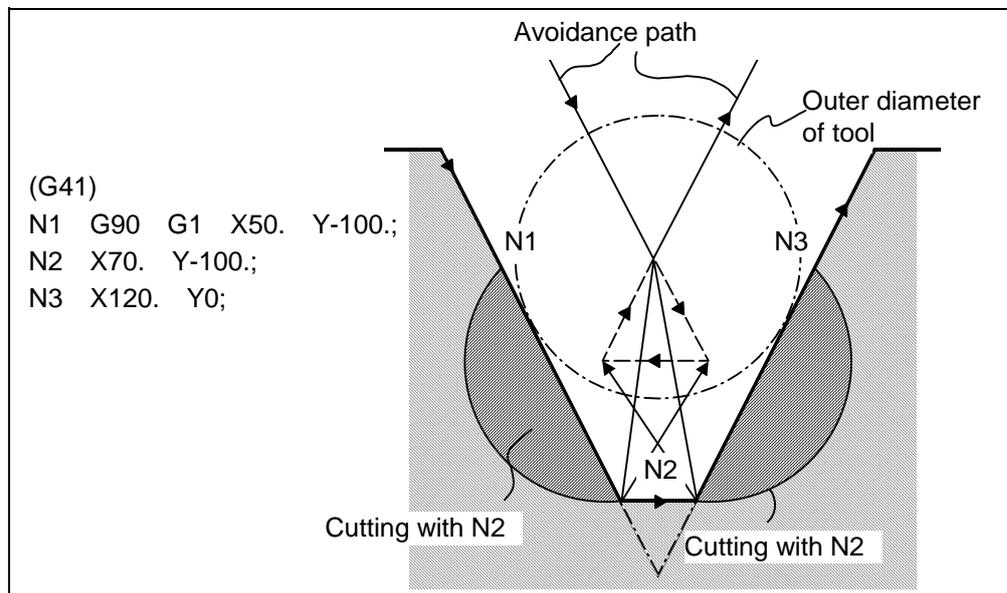
Function	Parameter	Operation
Interference check alarm function	#8102 : OFF #8103 : OFF	A program error results before the execution of the block in which the cut arises, and operation stops.
Interference check avoidance function	#8102 : ON #8103 : OFF	The tool path is changed so that workpiece is not cut into.
Interference check invalid function	#8103 : ON	Cutting proceeds unchanged even when it occurs. Use this for microscopic segment programs.

(Note) #8102 COLL. ALM OFF (interference check avoidance)  
#8103 COLL. CHK OFF (interference check invalid)



## Detailed description

## (Example)



## (1) With alarm function

The alarm occurs before N1 is executed and so, using the edit function, N1 can be changed as below and machining can be continued :

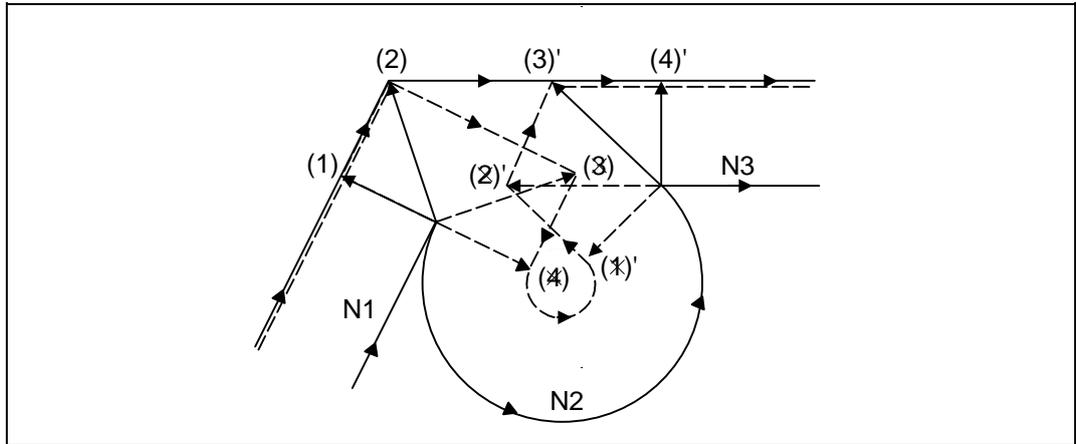
```
N1 G90 G1 X20. Y-40. ;
```

## (2) With avoidance function

The intersection point of N1 and N3 is calculated and the interference avoidance vectors are created.

**(3) With interference check invalid function**

The tool passes while cutting the N1 and N3 line.



Example of interference check

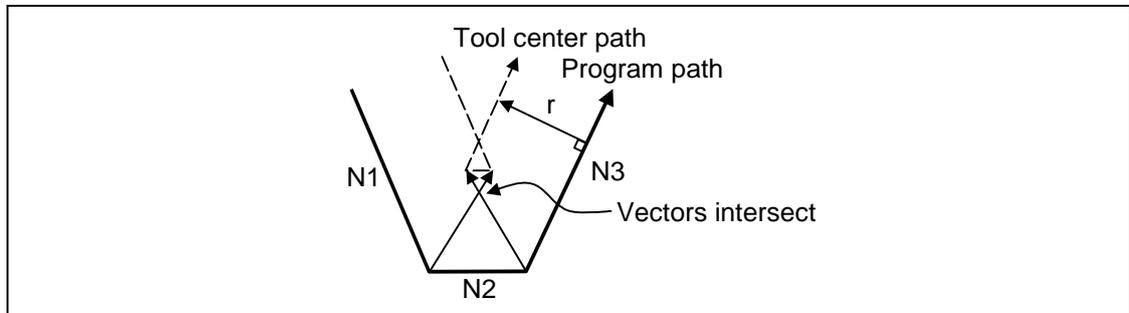
- |                        |                   |                        |
|------------------------|-------------------|------------------------|
| Vectors (1) (4)' check | → No interference |                        |
| ↓                      |                   |                        |
| Vectors (2) (3)' check | → No interference |                        |
| ↓                      |                   |                        |
| Vectors (3) (2)' check | → Interference →  | Erase vectors (3) (2)' |
|                        |                   | ↓                      |
|                        |                   | Erase vectors (4) (1)' |

With the above process, the vectors (1), (2), (3)' and (4)' will remain as the valid vectors, and the path that connects these vectors will be executed as the interference avoidance path.



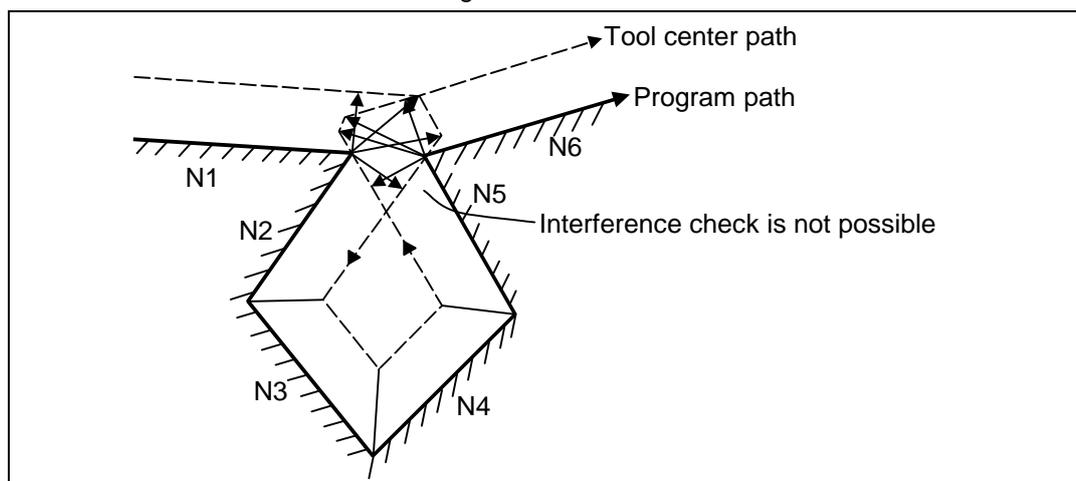
### Conditions viewed as interference

If there is a movement command in three of the five pre-read blocks, and if the compensation calculation vectors created at the contacts of each movement command intersect, it will be viewed as interference.



### When interference check cannot be executed

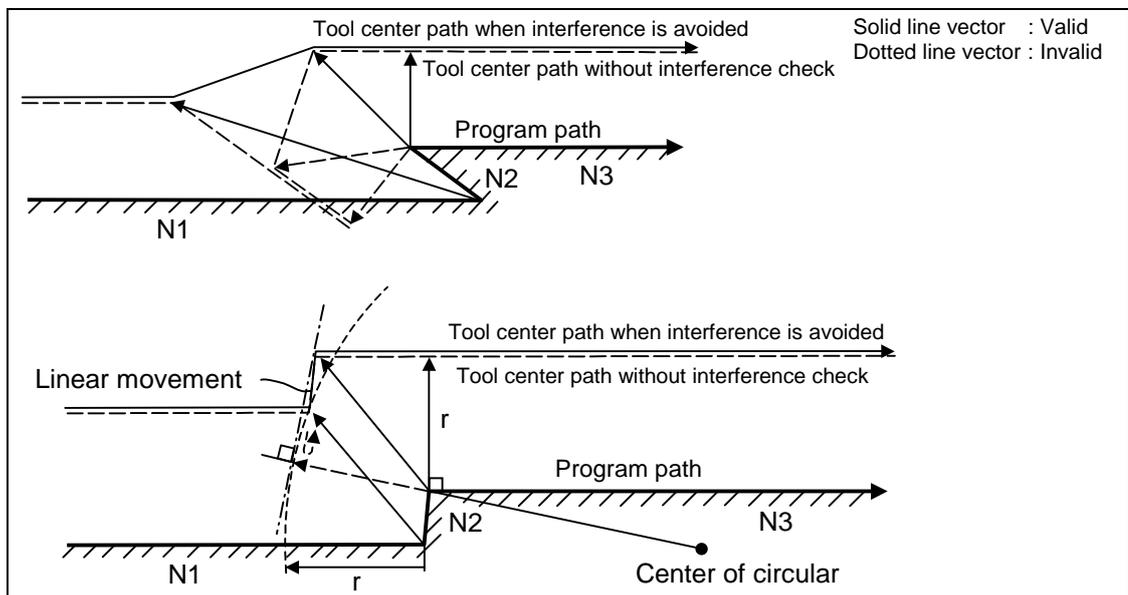
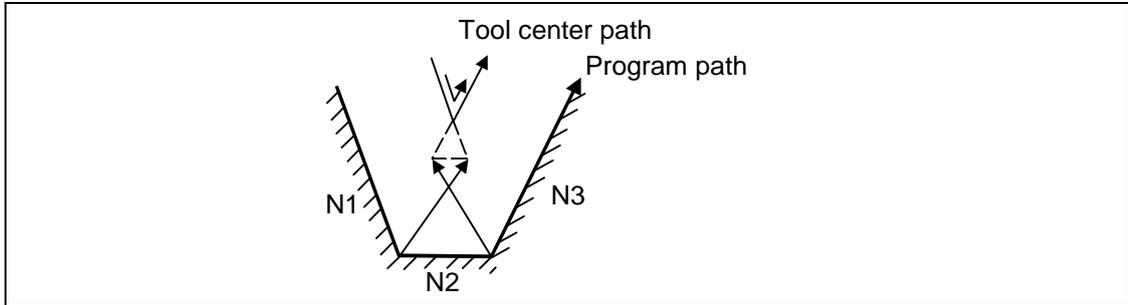
- (1) When three of the movement command blocks cannot be pre-read  
(When there are three or more blocks in the five pre-read blocks that do not have movement)
- (2) When there is an interference following the fourth movement block

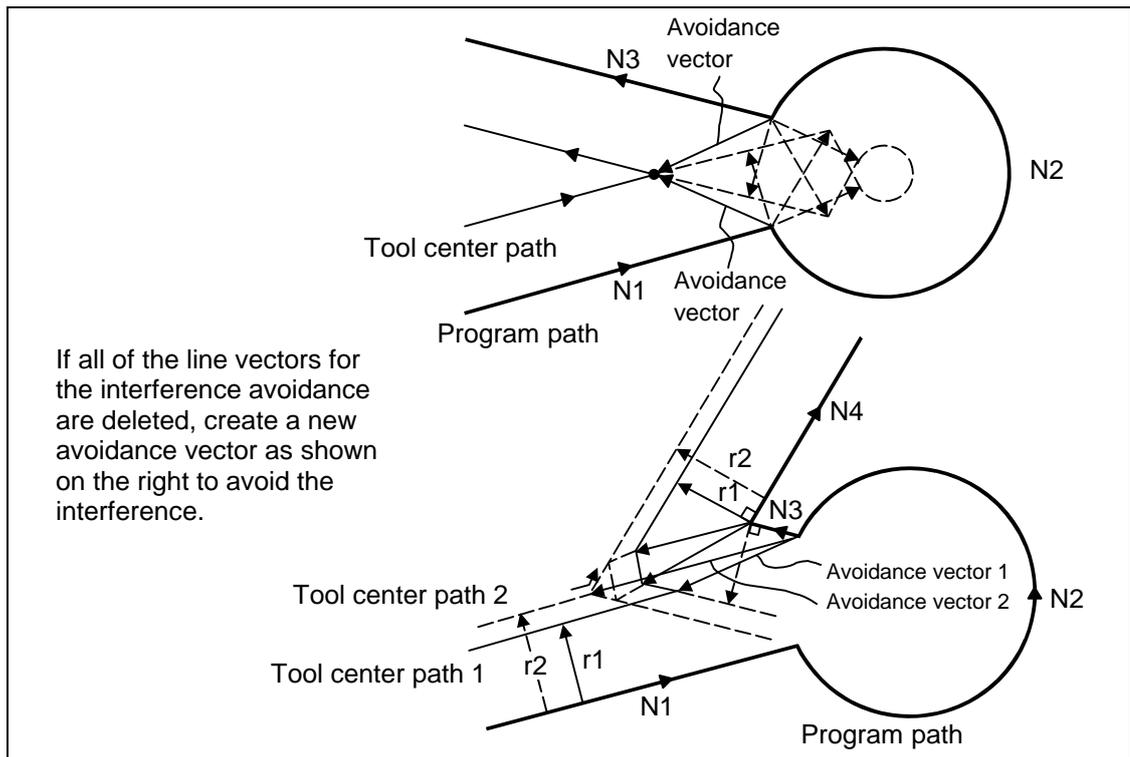




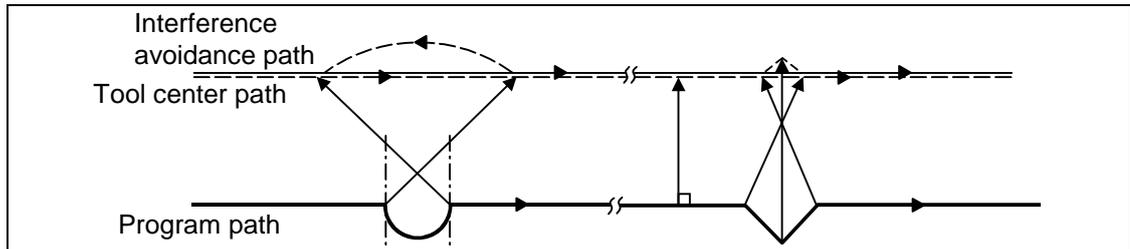
Operation during interference avoidance

The movement will be as shown below when the interference avoidance check is used.





In the case of the figure below, the groove will be left uncut.





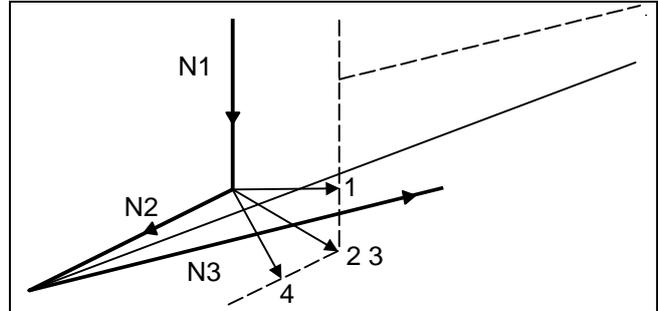
**Interference check alarm**

The interference check alarm occurs under the following conditions.

**(1) When the interference check alarm function has been selected**

- (a) When all the vectors at the end block of its own block have been deleted.

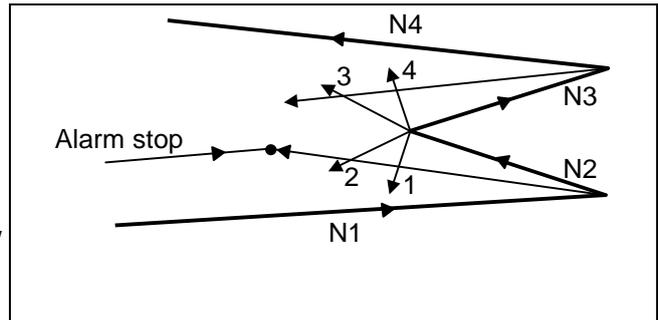
When, as shown in the figure, vectors 1 through 4 at the end point of the N1 block have all been deleted, program error (P153) results prior to N1 execution.



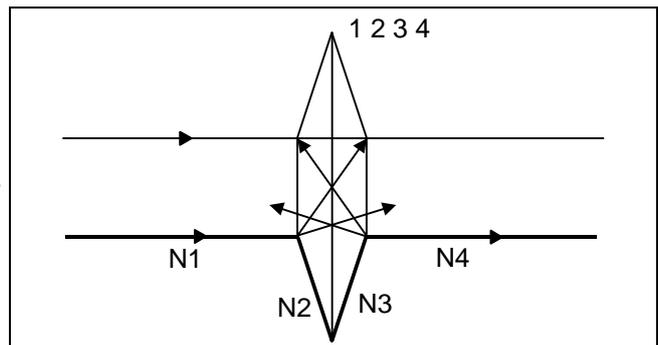
**(2) When the interference check avoidance function has been selected**

- (a) When there are valid vectors at the end point of the following block even when all the vectors at the end point of its own block have been deleted.

- (i) When, in the figure, the N2 interference check is conducted, the N2 end point vectors are all deleted but the N3 end point vectors are regarded as valid. Program error (P153) now occurs at the N1 end point.

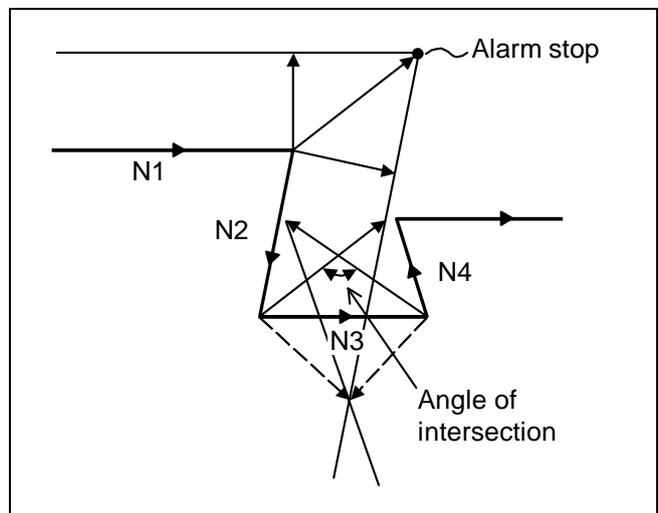
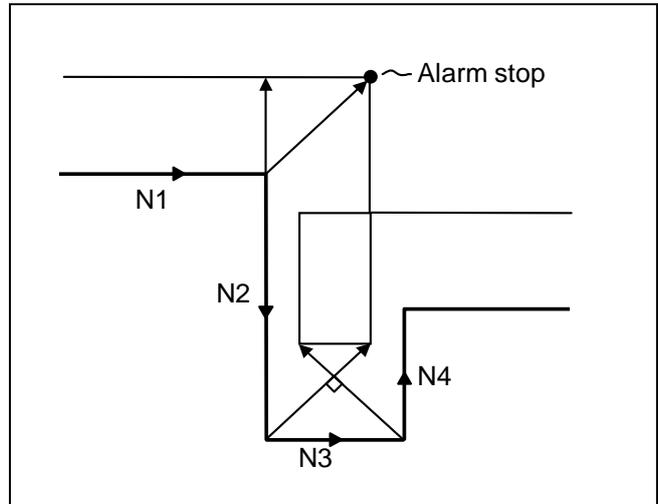


- (ii) In a case such as that shown in the figure, the tool will move in the reverse direction at N2. Program error (P153) occurs after N1 execution.



(b) When avoidance vectors cannot be created

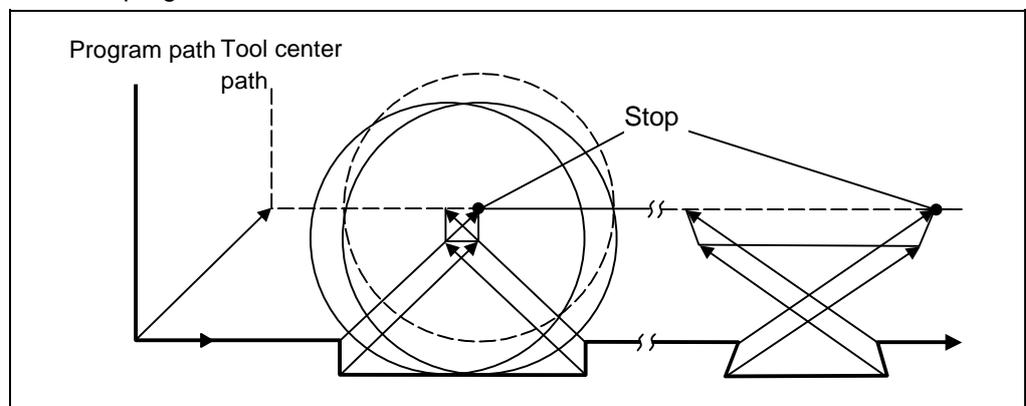
(i) Even when, as in the figure, the conditions for creating the avoidance vectors are met, it may still be impossible to create these vectors or the interference vectors may interfere with N3. Program error (P153) will occur at the N1 end point when the vector intersecting angle is more than 90°.



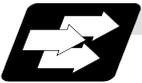
(c) When the program advance direction and the advance direction after compensation are reversed

In the following case, interference is still regarded as occurring even when there is actually no interference.

When grooves which are narrower than the tool radius or which have parallel or widening walls are programmed



12.4.9 Diameter Designation of Compensation Amount



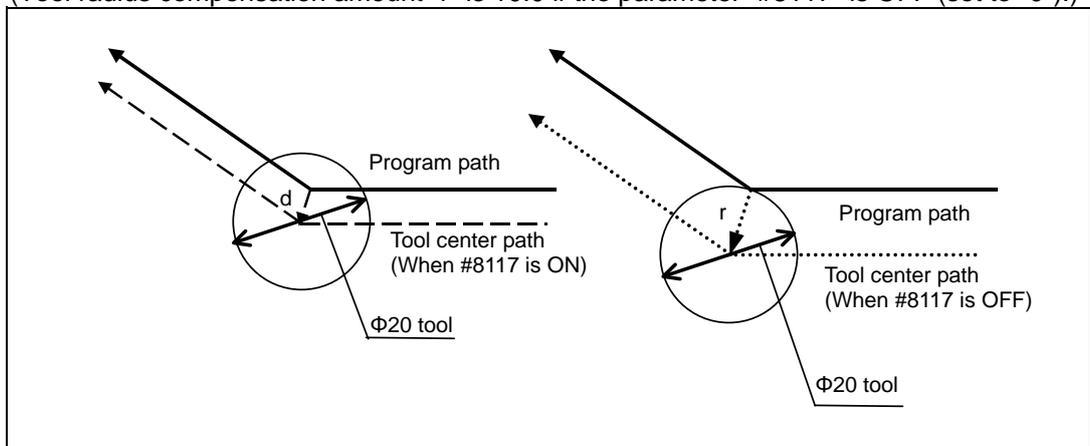
Function and purpose

With this function, the tool radius compensation amount can be designated by tool diameter. When the control parameter "#8117 OFS Diam DESIGN" is ON, the compensation amount specified to the commanded tool No. will be recognized as the diameter compensation amount, and the amount will be converted to the radius compensation amount when executing the compensation.

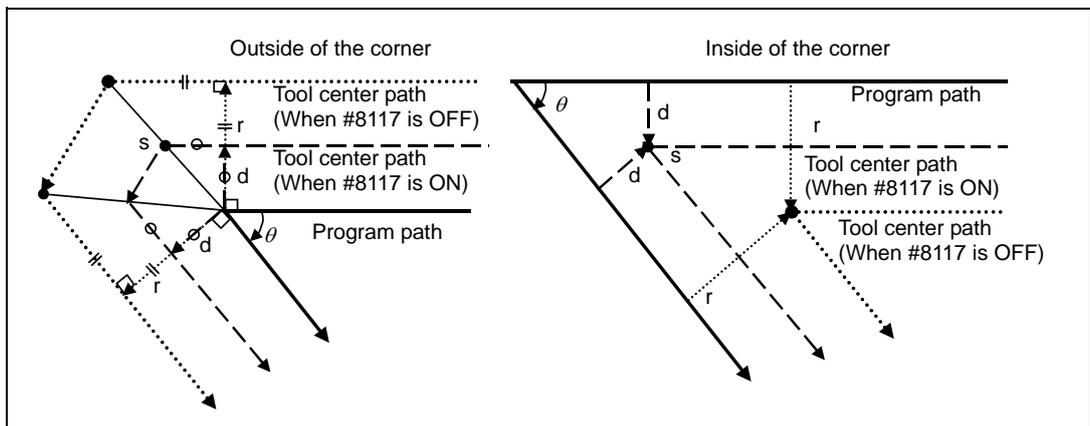


Operations when designating the compensation amount with diameter

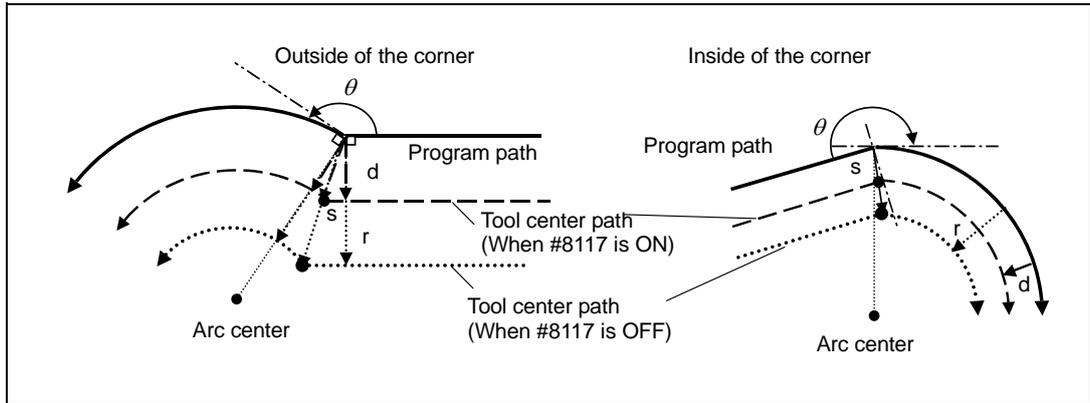
- (1) When the tool radius compensation amount  $D=10.0$  is commanded, tool radius compensation amount "d" is 5.0 if the parameter "#8117" is ON (set to "1").  
(Tool radius compensation amount "r" is 10.0 if the parameter "#8117" is OFF (set to "0").)



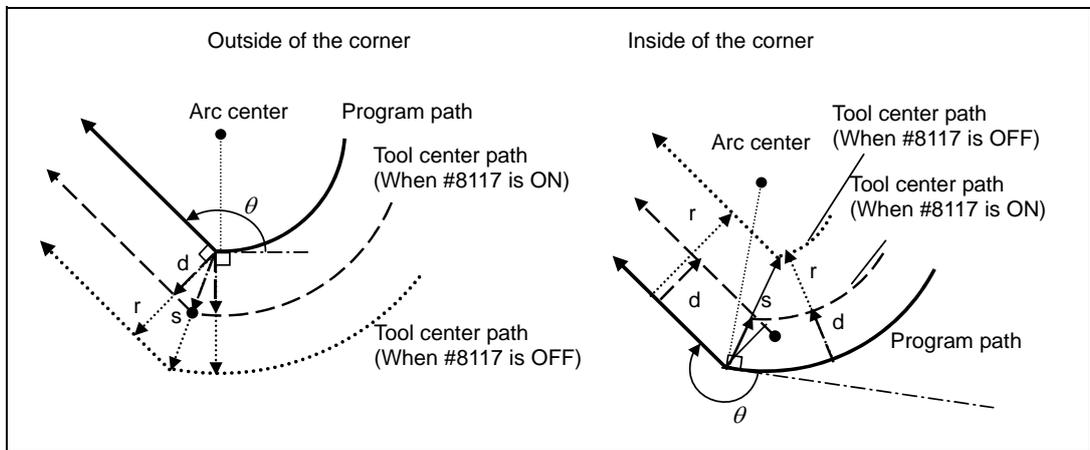
(a) Linear to linear (acute angle)



(b) Linear to arc (obtuse angle)



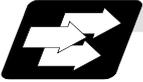
(c) Arc to linear (obtuse angle)



**Restrictions**

- (1) If tool radius compensation amount has already been set, the compensation amount is not be changed even if the parameter "8117" is changed.
- (2) Make sure not to change the parameter #8117 during the compensation.  
When the parameter is changed using parameter input by program function, the program error (P241) will occur.
- (3) If the parameter #8117 is set to ON with the parameter "#1037 cmdtyp" set to "2", the tool radius wear data is also regarded as the diameter compensation amount, thus, it will be converted to the radius value and compensation will be performed.
- (4) Diameter designation of tool radius compensation amount can be used for the tool life management data.
- (5) There is no effect by #8117 on the tool radius measurement function.

12.4.10 Workpiece Coordinate Changing During Radius Compensation



Function and purpose

When the tool radius compensation is executed, the tool center path is calculated based on the position on the coordinate system. The based coordinate system can be changed by the parameter.

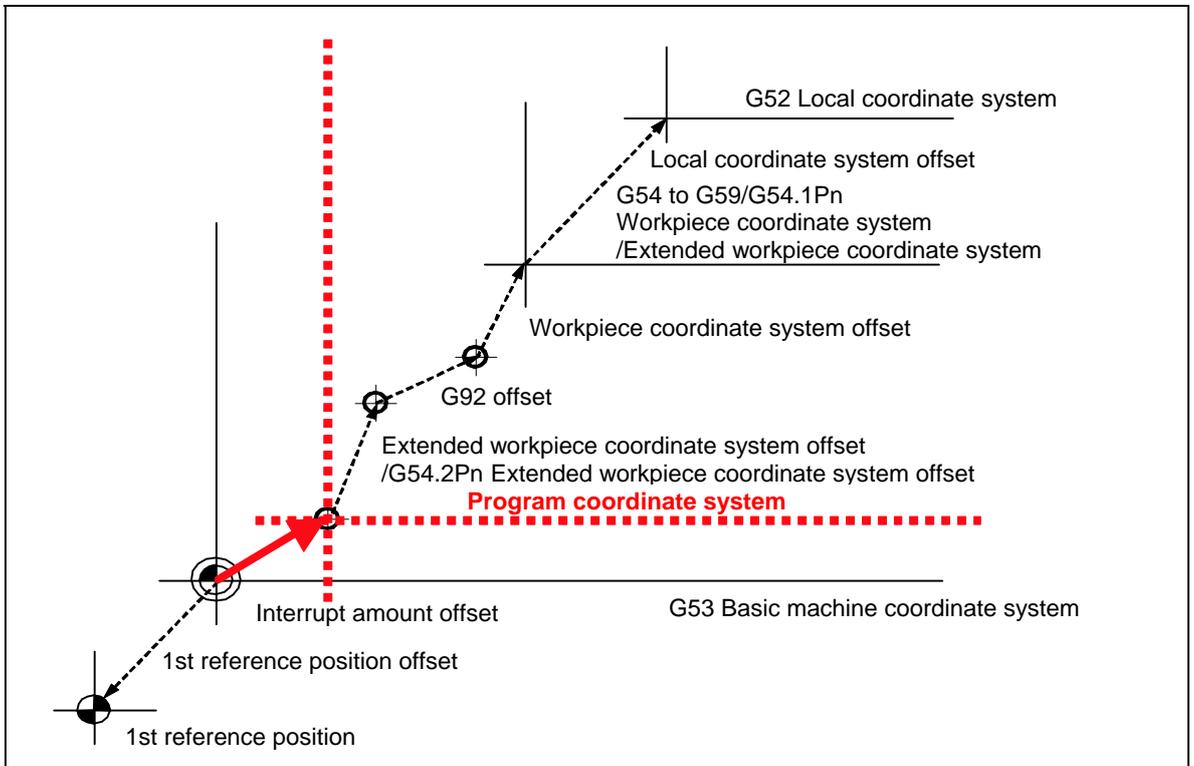


Detailed description

When the parameter is "0", the tool radius compensation is calculated based on the position on the workpiece coordinate system.

When the parameter is "1", the tool radius compensation is calculated based on the position on the program coordinate system.

The program coordinate systems are defined as shown in the figure below.



The coordinate system changed by parameter is as follows.

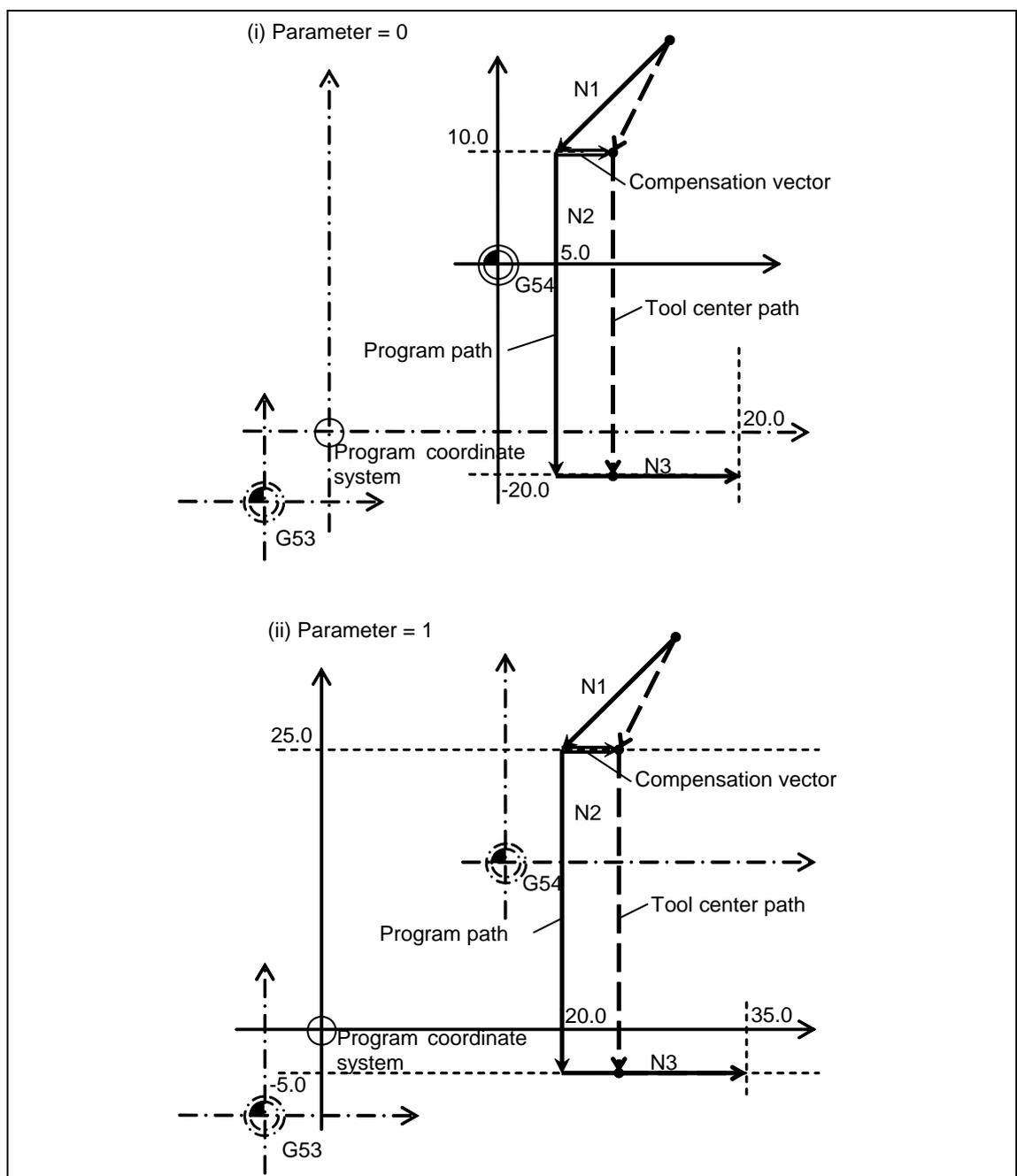
G90 G54 G00 X15. Y20.	
N1 G41 D3 X5. Y10.;	
N2 G01 Y-20. F1000;	
N3 G40 X30.;	
M30;	

D3 = 5.000

G54 offset

X15.000

Y15.000



## 12. Tool Compensation Functions

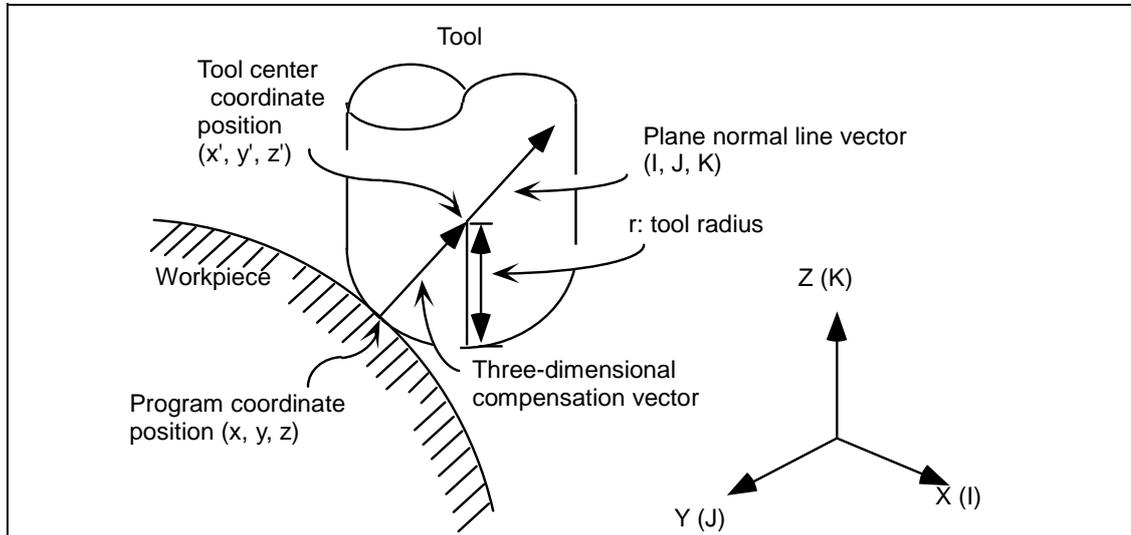
### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

#### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42



##### Function and purpose

The three-dimensional tool radius compensation compensates the tool in a three-dimensional space following the commanded three-dimensional vectors.



As shown above, the tool is moved to the tool center coordinate position  $(x', y', z')$  which is offset by the tool radius "r" in respect to the program coordinate position  $(x, y, z)$  following the plane normal line vector  $(I, J, K)$ .

Though two-dimensional tool radius compensation creates the vectors at a right angle to the  $(I, J, K)$  direction, three-dimensional tool radius compensation creates the vector in the  $(I, J, K)$  direction. (The vector is created at the end point of the block.)

The three-dimensional compensation vector (offset) axis elements are as the right.

$$H_x = \frac{I}{\sqrt{I^2 + J^2 + K^2}} * r$$

$$H_y = \frac{J}{\sqrt{I^2 + J^2 + K^2}} * r$$

$$H_z = \frac{K}{\sqrt{I^2 + J^2 + K^2}} * r$$

Thus, the tool center coordinate position  $(x', y', z')$  is each expressed as the right.

Note that  $(x, y, z)$  are the program coordinate position.

$$x' = x + H_x$$

$$y' = y + H_y$$

$$z' = z + H_z$$

**(Note 1)** Three-dimensional compensation vector  $(H_x, H_y, H_z)$  refers to the plane normal line vector such as follows;

- The direction is same as the plane normal line vector  $(I, J, K)$ .
- The size equals to the tool radius "r".

**(Note 2)** When the machining parameter "#8071 3-D CMP" is set to a value other than "0", the value of "#8071 3-D CMP" will be used as the  $\sqrt{I^2 + J^2 + K^2}$  value. (Refer to the Setup Manual for details.)

**(Note 3)** This function is an additional specification. If commanded when the function is not provided, an error will occur.

## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42



#### Command format

Command the compensation No. D and plane normal line vector (I, J, K) in the same block as the three-dimensional tool radius compensation command G41 (G42).

If only one or two axes are commanded, the normal tool radius compensation mode will be applied.

(When setting "0" to the axes, this command is valid.)

<b>G41(G42) X__ Y__ Z__ I__ J__ K__ D__ ;</b>	<b>Three-dimensional tool radius compensation starts.</b> - Refer to (Example 1) and (Example 2).
<b>X__ Y__ Z__ I__ J__ K__ ;</b> :	<b>New plane normal line vector are commanded in the compensation mode.</b> - Refer to (Example 3) to (Example 7).
<b>G40; (or D00;)</b> <b>G40 X__ Y__ Z__ ;</b> <b>(or X__ Y__ Z__ D00;)</b>	<b>Three-dimensional tool radius compensation is canceled.</b> - Refer to (Example 8) and (Example 9).
G41 : Three-dimensional tool radius compensation command (+ direction) G42 : Three-dimensional tool radius compensation command (- direction) G40 : Three-dimensional tool radius compensation cancel command X, Y, Z : Movement axis command compensation space I, J, K : Plane normal line vector D : Compensation No. (Note that when "D00" is issued, three dimensional tool radius compensation will be canceled even if G40 is not commanded.)	

Compensation G code	Compensation amount:		D00
	+	-	
G40	Cancel	Cancel	Cancel
G41	I, J, K direction	Reverse direction of I, J, K	Cancel
G42	Reverse direction of I, J, K	I, J, K direction	Cancel



#### Detailed description

The compensation space is determined by the axis address commands (X, Y, Z, U, V, W) of the block where the three-dimensional tool radius compensation starts.

Here, U, V and W are each the additional axes for the X, Y and Z axis.

If the X axis and U axis (Y and V, Z and W) are commanded simultaneously in the three-dimensional tool radius compensation start block, the currently commanded plane selection axis will have the priority.

If the axis address is not commanded, it will be interpreted that the X, Y and Z axes are commanded for the coordinate axes.

#### (Example)

G17	XYZ space
G41 Xx Yy Zz Ii Jj Kk ;	
G17 ;	XYZ space
G41 Yy Ii Jj Kk ;	
G17 V ;	XVZ space
G41 Xx Vv Zz Ii Jj Kk ;	
G17 W ;	XYW space
G41 Ww Ii Jj Kk ;	
G17 ;	XYZ space
G41 Xx Yy Zz Ww Ii Jj Kk ;	
G17 W ;	XYW space
G41 Xx Yy Zz Ww Ii Jj Kk ;	
G17 ;	XYZ space
G41 Ii Jj Kk ;	
G17 U ;	UYZ space
G41 Ii Jj Kk ;	

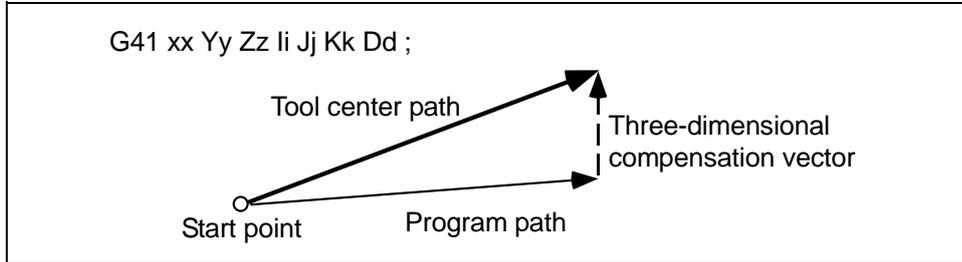
## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

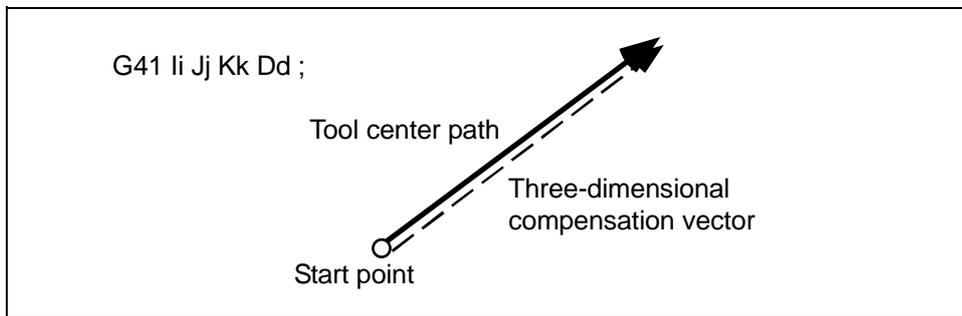


#### Example of operation

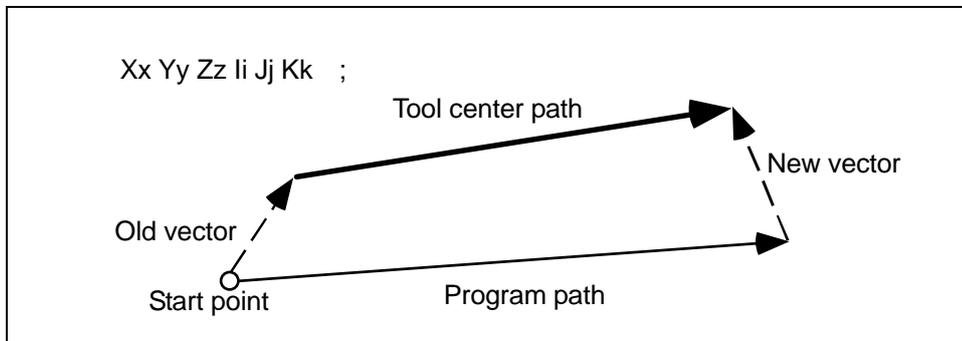
(1) Compensation start: When there is a movement command



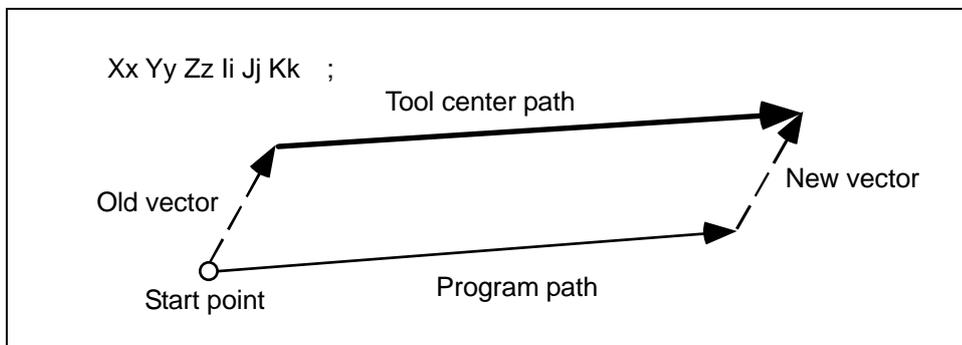
(2) Compensation start: When there is no movement command



(3) Movement during the compensation: When there is a movement command and a plane normal line vector command



(4) Movement during the compensation: When there is no plane normal line vector command



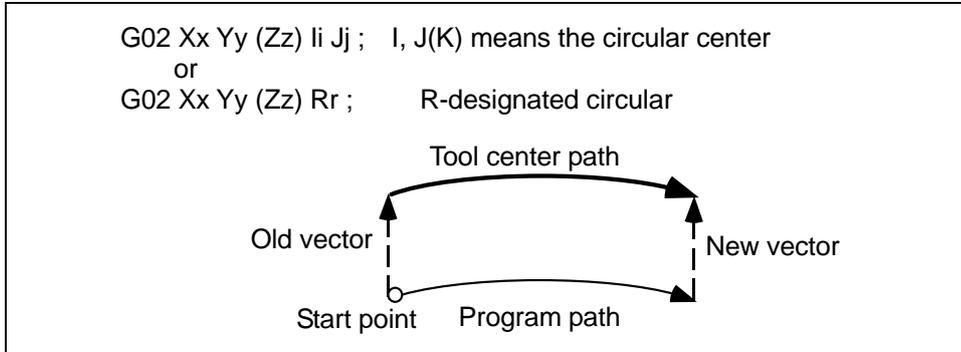
## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

(5) Movement during the compensation: For arc or helical cutting

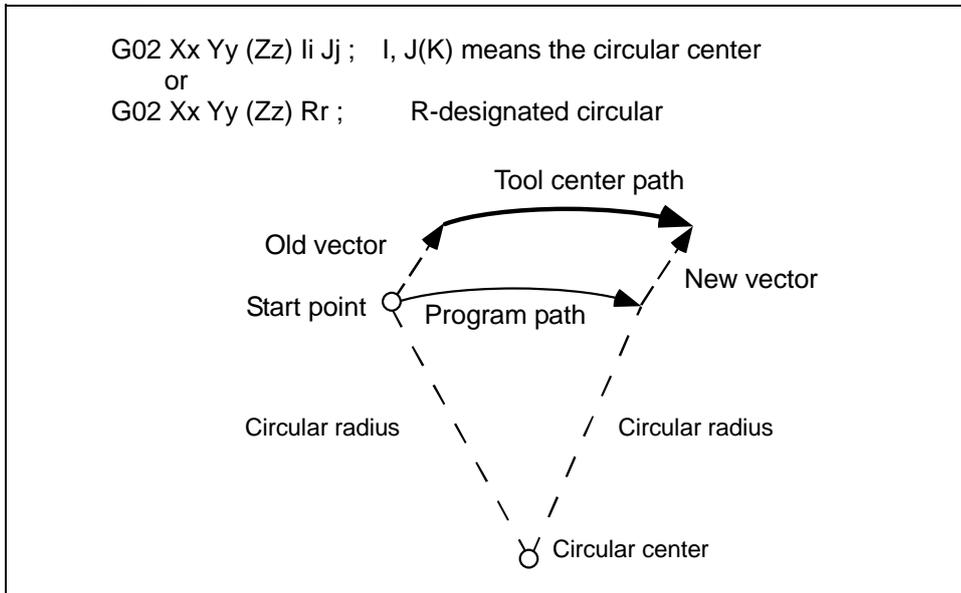
The I, J, K commands for a circular or helical cutting are regarded as the circular center commands, thus, the new vector is equivalent to the old vector.

Even for the R-designation method, commanded I, J, K addresses will be ignored, then the new vector will be equivalent to the old vector.

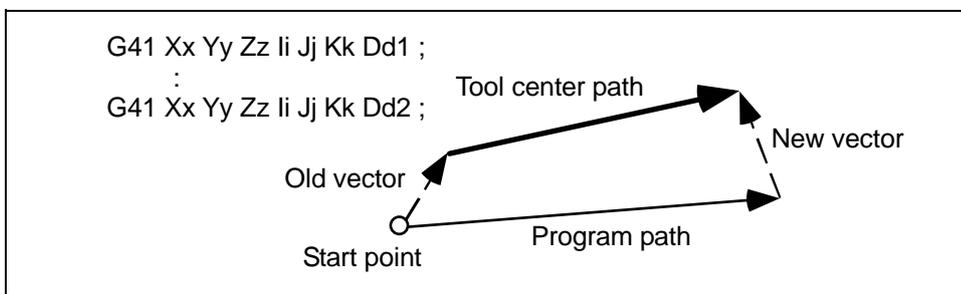


**(Note)** The center coordinate will not shift during the circular or helical cutting.

Thus, when I, J, K are commanded with the vector as below, the program error (P70) will occur.



(6) Movement during the compensation: When compensation amount is to be changed

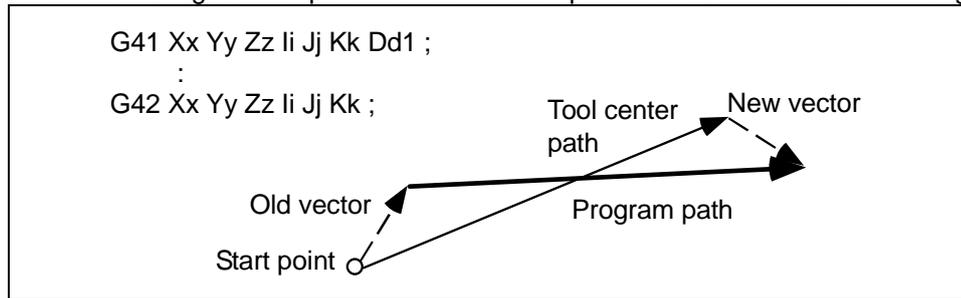


**(Note 1)** If I, J, K are not commanded in a block where the compensation amount is to be changed, the vector will be equivalent to the old vector. In this case, the modal will change, however, the compensation amount will change when I, J, K are commanded.

## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

(7) Movement during the compensation: When compensation direction is to be changed

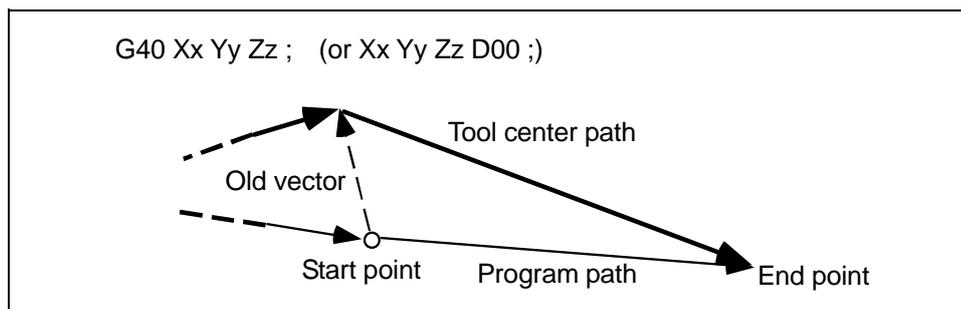


**(Note 1)** If I, J and K are not commanded in a block where the compensation direction is to be changed, the vector will be equivalent to the old vector and the compensation direction will not be changed.

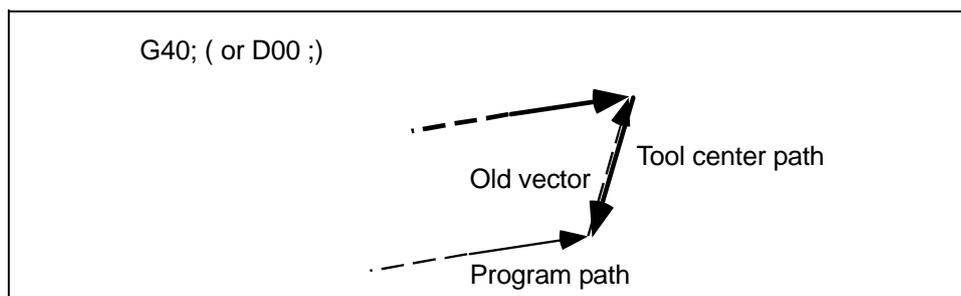
In this case, the modal will change, however, the compensation direction will change when I, J and K are commanded.

**(Note 2)** If the compensation direction is changed in an arc (G02/G03) block, I, J will be the center of arc, thus, the compensation direction will not change. Even for the R-designation method, commanded I, J and K will be ignored, and the compensation direction cannot be changed.

(8) Compensation cancel: When there is a movement command



(9) Compensation cancel: When there is no movement command



## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42



#### Relation with other functions

- (1) Normal tool radius compensation
 

If the plane normal line vector (I, J, K) is not commanded for all three axes in the three-dimensional tool radius compensation start block, the normal tool radius compensation mode will take place.

If G41 (G42) is commanded without commanding the plane normal line vector during three-dimensional tool radius compensation, the modal will change, however, the old vector will be used.

If G41 (G42) with the plane normal line vector is commanded during tool radius compensation, this command will be ignored and the normal tool radius compensation will take place.
- (2) Tool length offset
 

Tool length offset is applied on the coordinate after three-dimensional tool radius compensation.
- (3) Tool position offset
 

Tool position offset is applied on the coordinate after three-dimensional tool radius compensation.
- (4) Fixed cycle
 

The program error (P155) will occur.
- (5) Scaling
 

Scaling is applied on the coordinate before three-dimensional tool radius compensation. Scaling is not applied on the plane normal line vector (I, J, K).

```

D1=10.
G90 ;
G51 X0 Y0 P0.5 ;
N1 G41 D1 X-10. Y-20. Z-10. I-5.
  J-5. K-5. ;
N2 X-30. Y-30. Z-20. ;
N3 X-50. Y-20. Z-10. ;
N4 Y0. ;

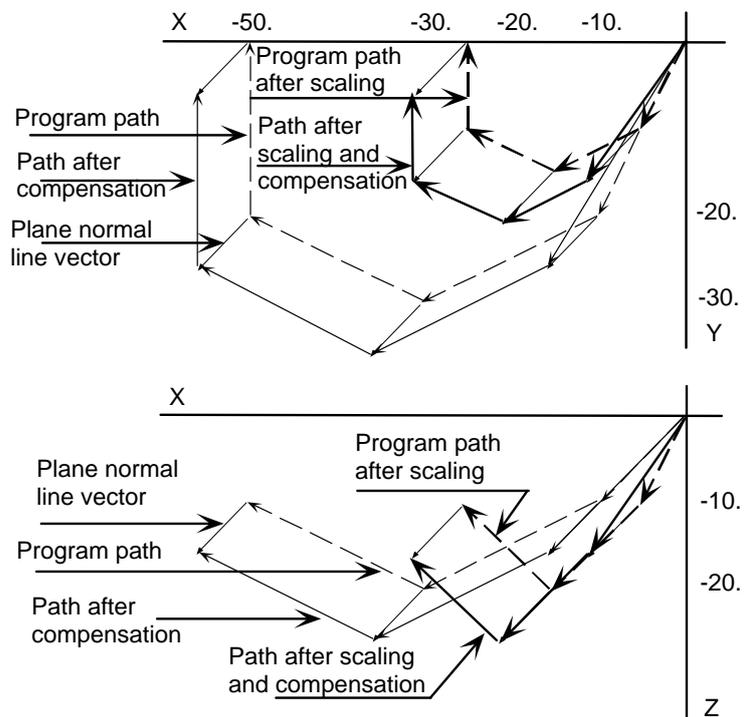
N1( -5.000, -10.000, -10.000 )
N1( -10.773, -15.773, -15.773 )

N2( -15.000, -15.000, -20.000 )
N2( -20.773, -20.773, -25.773 )

N3( -25.000, -10.000, -10.000 )
N3( -30.773, -15.773, -15.773 )

N4( -25.000,  0.000, -10.000 )
N4( -30.773,  -5.773, -15.773 )

*Upper:
Program position after scaling
Lower:
Position after scaling and
compensation
    
```



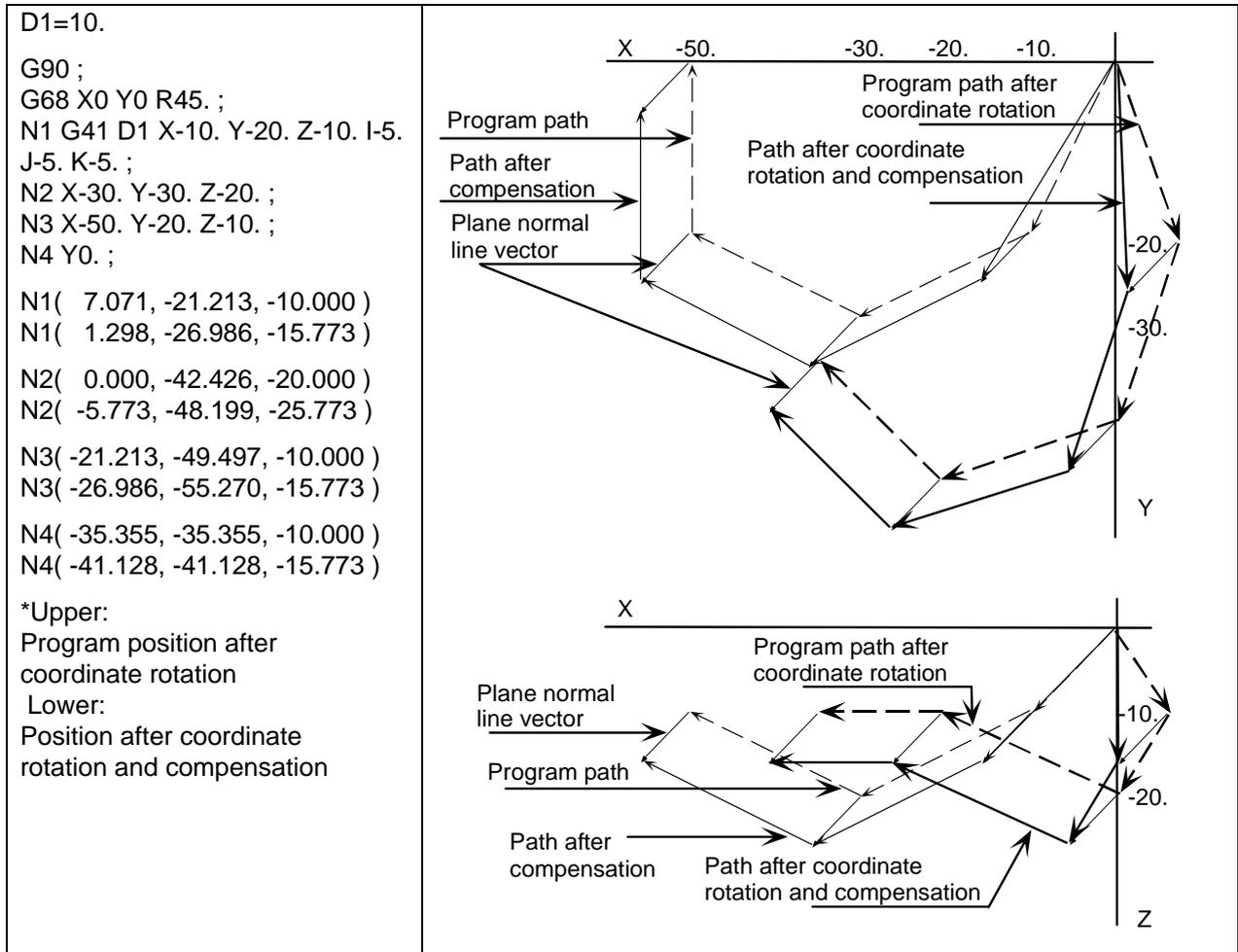
## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

#### (6) Program coordinate rotation

Program coordinate rotation is executed in respect to the coordinates before three-dimensional tool radius compensation.

The plane normal line vector (I, J, K) dose not rotate.



#### (7) Parameter coordinate rotation

Parameter coordinate rotation is applied on the coordinates after three-dimensional tool radius compensation.

The plane normal line vector (I, J, K) rotates.

#### (8) Mirror image

Mirror image is applied on the coordinates after three-dimensional tool radius compensation.

Mirror image is applied on the plane normal line vector (I, J, K).

#### (9) Skip

The program error (P608) will occur.

#### (10) Reference position check

The compensation amount will not be canceled. Thus, if this is commanded during three-dimensional tool radius compensation, the path will be deviated by the compensation amount, so the program error (P434) will occur.

#### (11) Automatic corner override

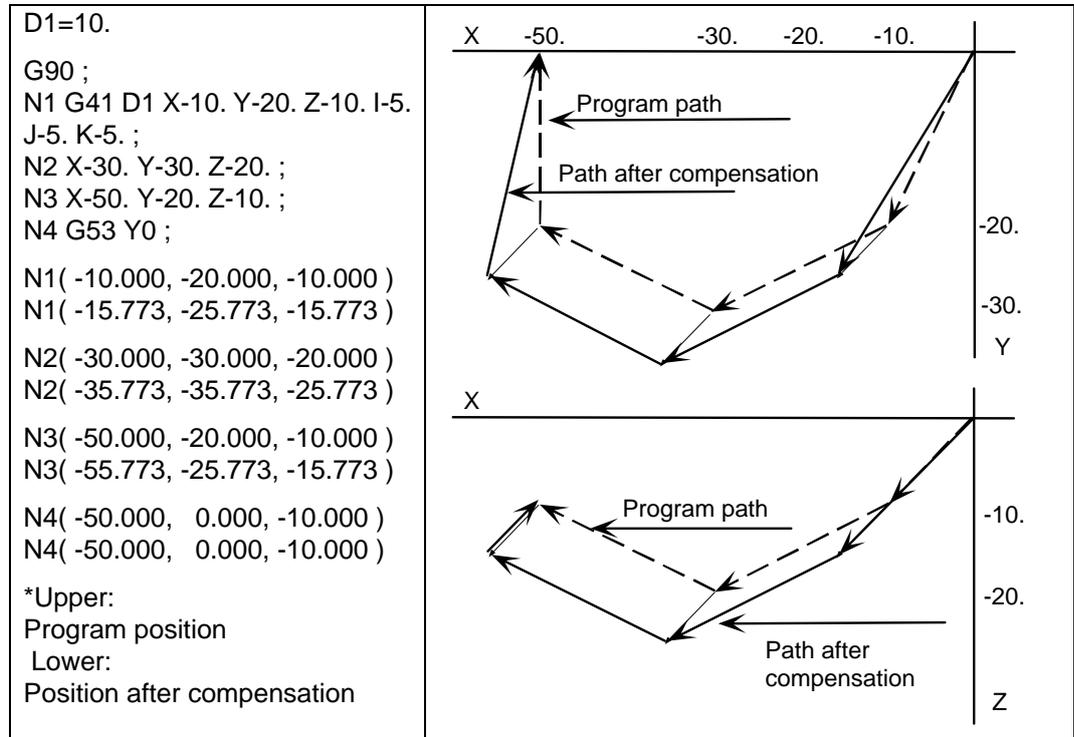
Automatic corner override is invalid during three-dimensional tool radius compensation.

## 12. Tool Compensation Functions

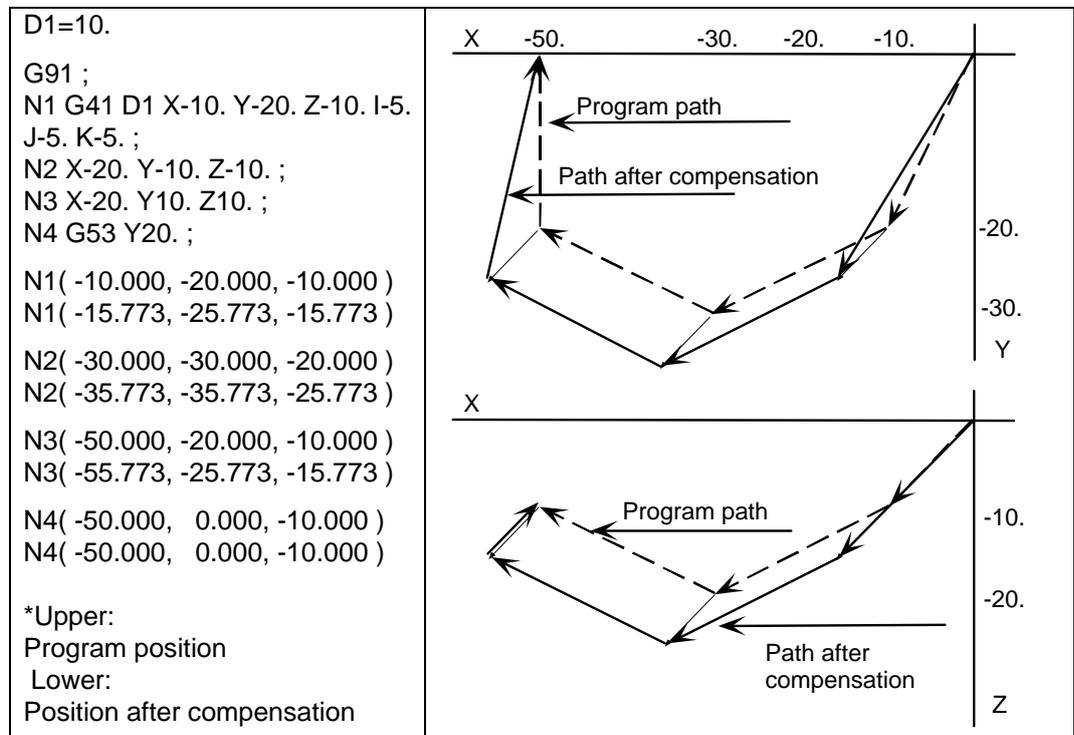
### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

#### (12) Machine coordinate system selection

- (a) For the absolute command, all axes will be temporarily canceled at the commanded coordinate position.



- (b) For the incremental command, the axis will move by the amount obtained by subtracting each axis vector from the incremental movement amount.  
(The compensation amount is temporarily canceled.)



## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

#### (13) Coordinate system setting

When commanded in the same block as the coordinate system setting, the coordinate system will be set, and operation will start up independently with the plane normal line vector (I, J, K).

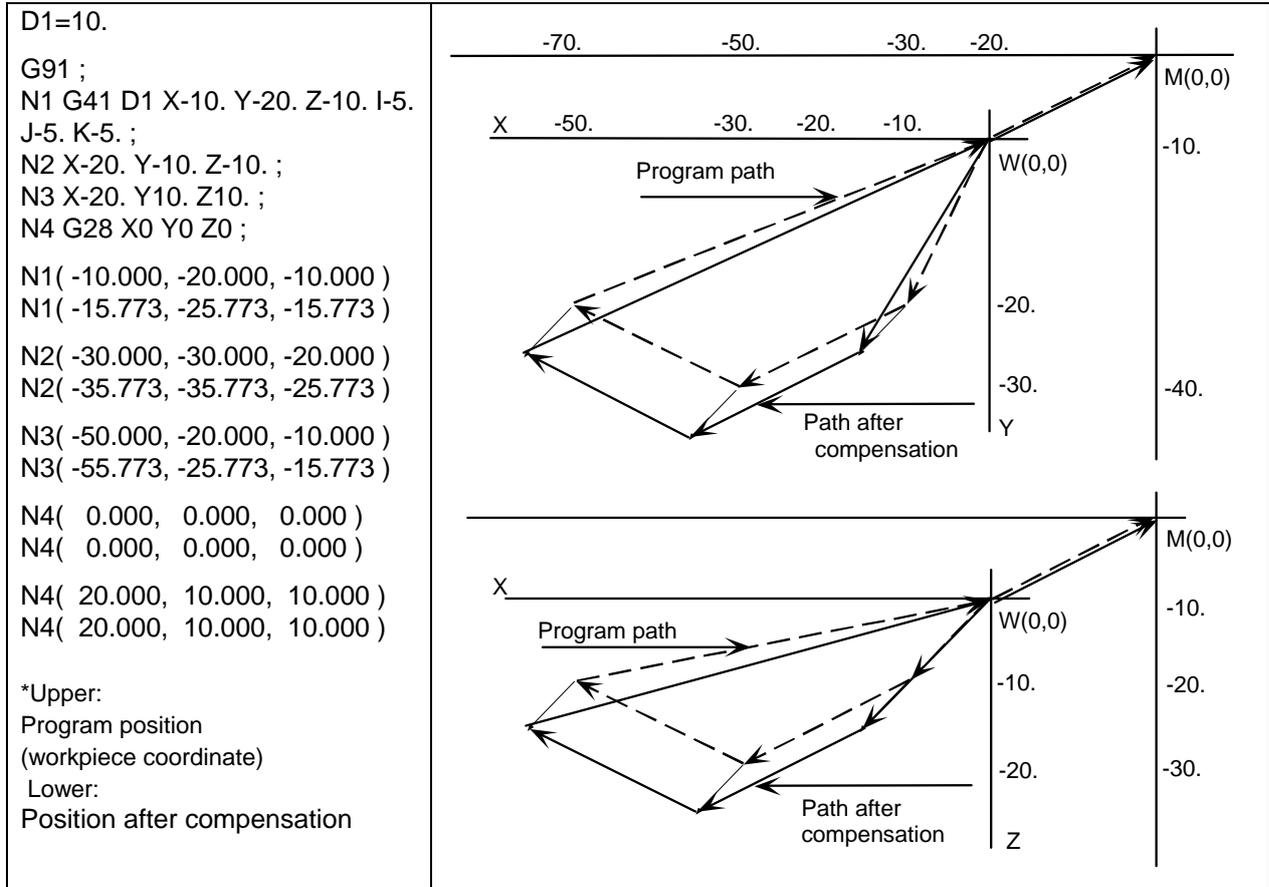
<p>D1=10.            G91 ;            N1 G92 G41 D1 X-10. Y-20.            Z-10. I-5. J-5. K-5. ;            N2 X-20. Y-10. Z-10. ;            N3 X-30. Y-10. Z10. ;            N4 Y20. ;</p> <p>N1( -10.000, -20.000, -10.000 )            N1( -15.773, -25.773, -15.773 )            N2( -30.000, -30.000, -20.000 )            N2( -35.773, -35.773, -25.773 )            N3( -50.000, -20.000, -10.000 )            N3( -55.773, -25.773, -15.773 )            N4( -50.000, 0.000, -10.000 )            N4( -55.773, -5.773, -15.773 )</p> <p>*Upper:            Program position            Lower:            Position after compensation</p>	<p>The figure consists of two diagrams illustrating 3D tool radius compensation. The top diagram shows the X-Y plane. The X-axis is horizontal, and the Y-axis is vertical. The program path (solid line) starts at X=-10, Y=0 and moves to X=-10, Y=-20. The path after compensation (dashed line) starts at X=-15.773, Y=0 and moves to X=-15.773, Y=-25.773. The bottom diagram shows the X-Z plane. The X-axis is horizontal, and the Z-axis is vertical. The program path (solid line) starts at X=-10, Z=0 and moves to X=-10, Z=-10. The path after compensation (dashed line) starts at X=-15.773, Z=0 and moves to X=-15.773, Z=-15.773. Both diagrams include a G92 command at the end of the path.</p>
--	---

## 12. Tool Compensation Functions

### 12.5 Three-dimensional Tool Radius Compensation ; G40/G41,G42

#### (14) Reference position return

All the axes will be temporarily canceled at the intermediate point.



#### (15) NC reset

Three-dimensional tool radius compensation will be canceled if NC reset is executed during three-dimensional tool radius compensation.

#### (16) Emergency stop

Three-dimensional tool radius compensation will be canceled by the emergency stop or emergency stop cancel during three-dimensional tool radius compensation.



## Restrictions

- (1) The compensation No. is selected with the D address, however, the D address is valid only when G41 or G42 is commanded. If D is not commanded, the number of the previous D address will be valid.
- (2) Switch the mode to the compensation mode in the G00 or G01 mode. When changed during the arc mode, the program error (P150) will occur.  
The compensation direction and compensation amount after the mode change will become valid from the block where I, J and K are commanded in the G00 or G01 mode.  
If three-dimensional tool radius compensation is commanded in a block not containing the plane normal line vector (I, J, K) during the arc mode, only the modal information will be changed.  
The plane normal line vector will be validated from the block where I, J and K are commanded next.
- (3) During the three-dimensional tool radius compensation mode in a certain space, it is not possible to switch the space to another one and to execute three-dimensional tool radius compensation. To switch the compensation space, always cancel the compensation mode with G40 or D00 first.

**(Example)**

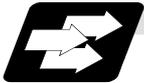
G41 Xx Yy Zz Ii Jj Kk;      Compensation starts in X, Y, Z space.

⋮

G41 Uu Yy Zz Ii Jj Kk;      Compensation is carried out in X, Y, Z space, and U axis moves by commanded value.

- (4) If the compensation No. D is other than the range of 1 to 40 with the standard specifications or 1 to 800 (max.) with the additional specifications, the program error (P170) will occur.
- (5) Only the G40 and D00 commands can be used to cancel three-dimensional tool radius compensation.
- (6) If the size ( $I^2+J^2+K^2$ ) of the vector commanded with I, J and K overflows, the program error (P35) will occur.

12.6 Tool Position Offset; G45 to G48



Function and purpose

Using the G45 to G46 commands, the movement distance of the axes specified in the same block can be extended or reduced by a preset compensation length. Furthermore, the compensation amount can be similarly doubled (x 2 expansion) or halved (x 2 reduction) with commands G47 and G48.

The number of sets for the compensation differ according to machine specification. Refer to Specifications Manual.

D01 to Dn

(The numbers given are the total number of sets for tool length compensation, tool position offset and Tool radius compensation.)

G45 command	G46 command	G47 command	G48 command
Expansion by compensation amount only	Reduction by compensation amount only	× 2 expansion by compensation amount	× 2 reduction by compensation amount

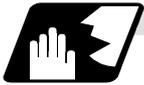
± =

(Program command value)      (compensation amount)      (Movement amount after compensation)



Command format

<b>G45 X__ Y__ Z__ H__ (D__);</b>	<b>Expansion of movement amount by compensation amount set in compensation memory</b>
<b>G46 X__ Y__ Z__ H__ (D__);</b>	<b>Reduction of movement amount by compensation amount set in compensation memory</b>
<b>G47 X__ Y__ Z__ H__ (D__);</b>	<b>Expansion of movement amount by double the compensation amount set in compensation memory</b>
<b>G48 X__ Y__ Z__ H__ (D__);</b>	<b>Reduction of movement amount by double the compensation amount set in compensation memory</b>
X, Y, Z	: Movement amount of each axis
H (D)	: Tool compensation No.
Tool compensation type I	Tool length compensation and tool radius compensation is not distinguished. Thus, the same compensation amount will be obtained whichever H or D is used.
Tool compensation type II	The compensation amount will be obtained as follows. H: Compensation amount designated with tool length compensation No. D: Compensation amount designated with tool radius compensation No.



Detailed description

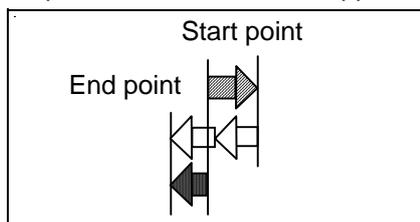
Details for incremental values are given below.

Command	Movement amount of equivalent command (assigned compensation amount = I)	Example (when X = 1000)
G45Xx Dd	$X (x + I)$	I = 10    X = 1010 I = -10   X = 990
G45X-x Dd	$X - (x + I)$	I = 10    X = -1010 I = -10   X = -990
G46Xx Dd	$X (x - I)$	I = 10    X = 990 I = -10   X = 1010
G46X-x Dd	$X - (x - I)$	I = 10    X = -990 I = -10   X = -1010
G47Xx Dd	$X (x + 2 * I)$	I = 10    X = 1020 I = -10   X = 980
G47X-x Dd	$X - (x + 2 * I)$	I = 10    X = -1020 I = -10   X = -980
G48Xx Dd	$X (x - 2 * I)$	I = 10    X = 980 I = -10   X = 1020
G48X-x Dd	$X - (x - 2 * I)$	I = 10    X = -980 I = -10   X = -1020



Precautions

- (1) These commands should be used when operation is not in a canned cycle mode. (They are ignored even if they are assigned during a canned cycle.)
- (2) As a result of the internal arithmetic processing based on the expansion or reduction, the tool will proceed to move in the opposite direction when the command direction is reversed.



Programmed command    G48X20.000D01;  
 Compensation            Compensation amount = +15.000  
 Tool movement          Actual movement = X - 10.000

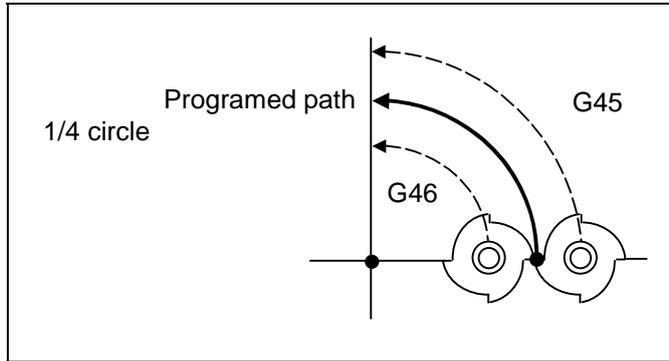
- (3) When a zero movement amount has been specified in the incremental value command (G91) mode, the result is as follows.

Compensation number : D01  
 Compensation amount corresponding to D01: 1234

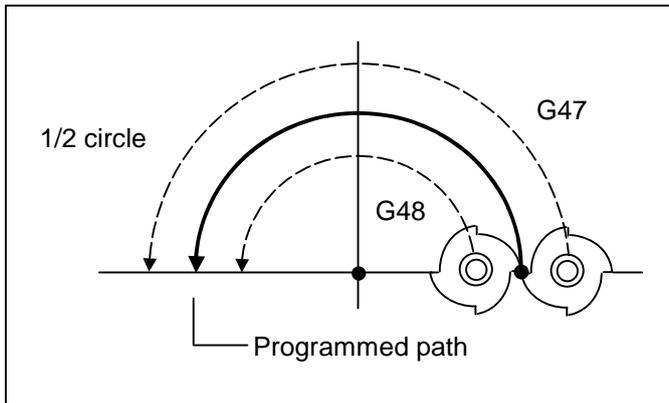
NC command	G45X0 D01;	G45X - 0 D01;	G46X0 D01;	G46X - 0 D01;
Equivalent command	X1234;	X - 1234;	X - 1234;	X1234;

When a zero movement amount has been specified with an absolute value command, the operation is completed immediately and the tool does not move for the compensation amount.

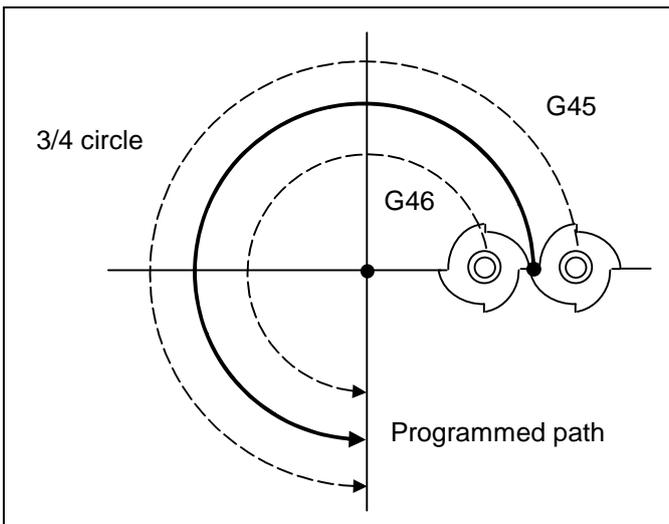
- (4) In the case of circular interpolation, cutter compensation is possible using the G45 to G48 commands only for one quadrant, two quadrants (semi-sphere) or three quadrants when the start and end points are on the axis. The commands are assigned as follows depending on whether the compensation is applied for outside or inside the arc programmed path. However, in this case, compensation must already be provided in the desired direction at the arc start point. (If a compensation command is assigned for the arc independently, the arc start point and end point radii will shift by an amount equivalent to the compensation amount.)



G45 for compensation outside the circle  
G46 for compensation inside the circle



G47 for compensation outside the circle  
G48 for compensation inside the circle

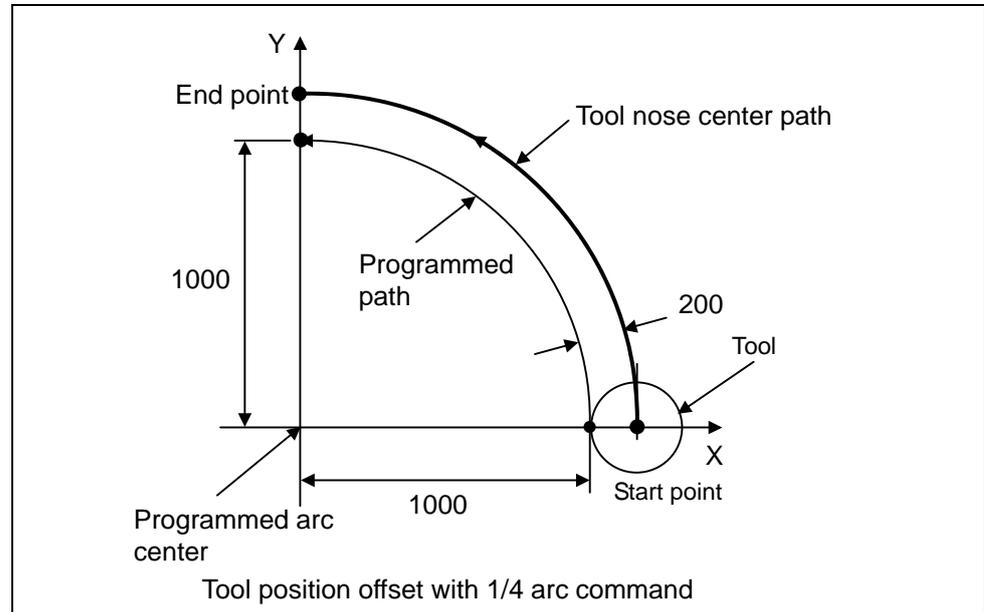


G45 for compensation outside the circle  
G46 for compensation inside the circle



## Example of program

(Example 1)



It is assumed that compensation has already been provided in the + X direction by  $D01 = 200$ .

```
G91 G45 G03 X-1000 Y1000 I-1000 F1000 D01;
```

Even if the compensation numbers are not assigned in the same block as the G45 to G48 commands, compensation is provided with the tool position compensation number previously stored in the memory.

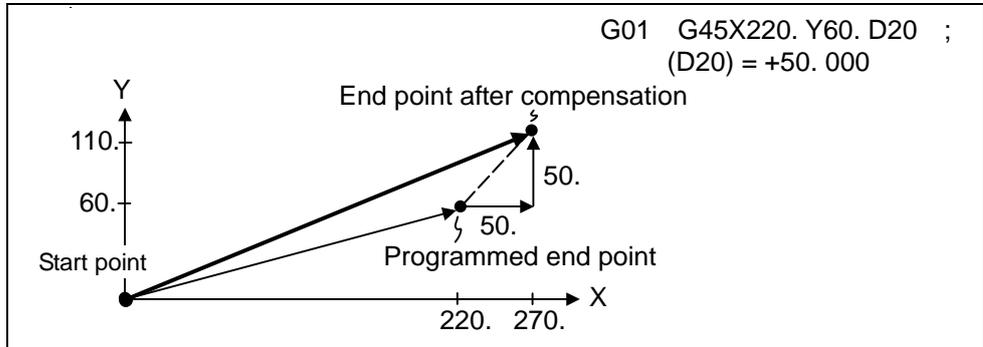
Program error "P170" results when the specified compensation number has exceeded the specification range.

These G codes are unmodal and are effective only in the command block.

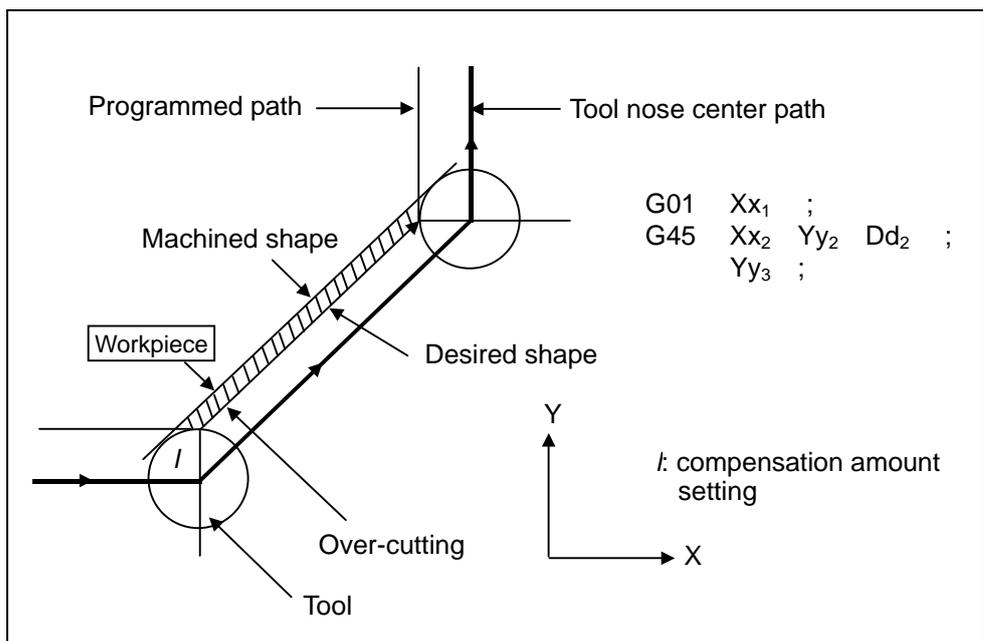
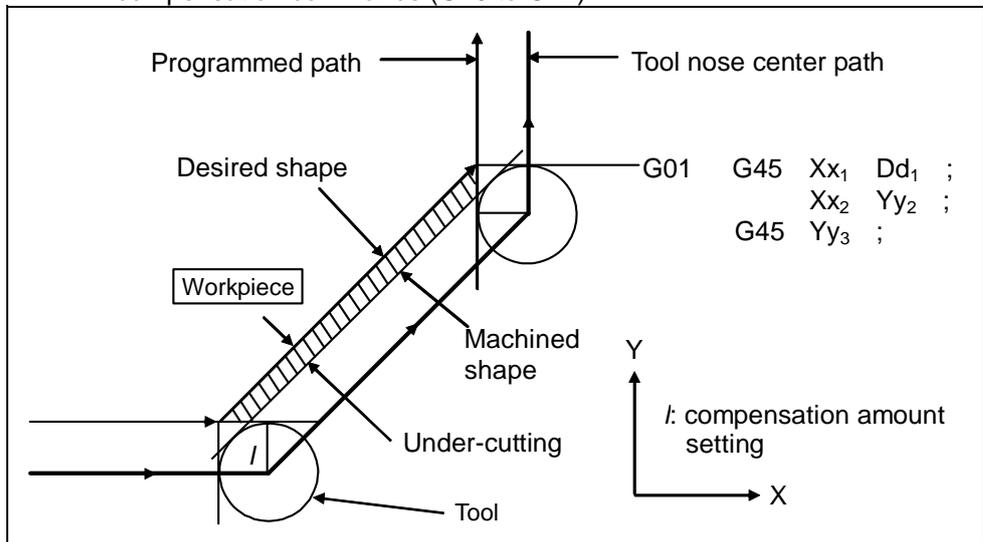
Even with an absolute value command, the amount of the movement is extended or reduced for each axis with respect to the direction of movement from the end point of the preceding block to the position assigned by the G45 to G48 block.

In other words, even for an absolute value command, compensation can be applied to movement amounts (incremental values) in the same block.

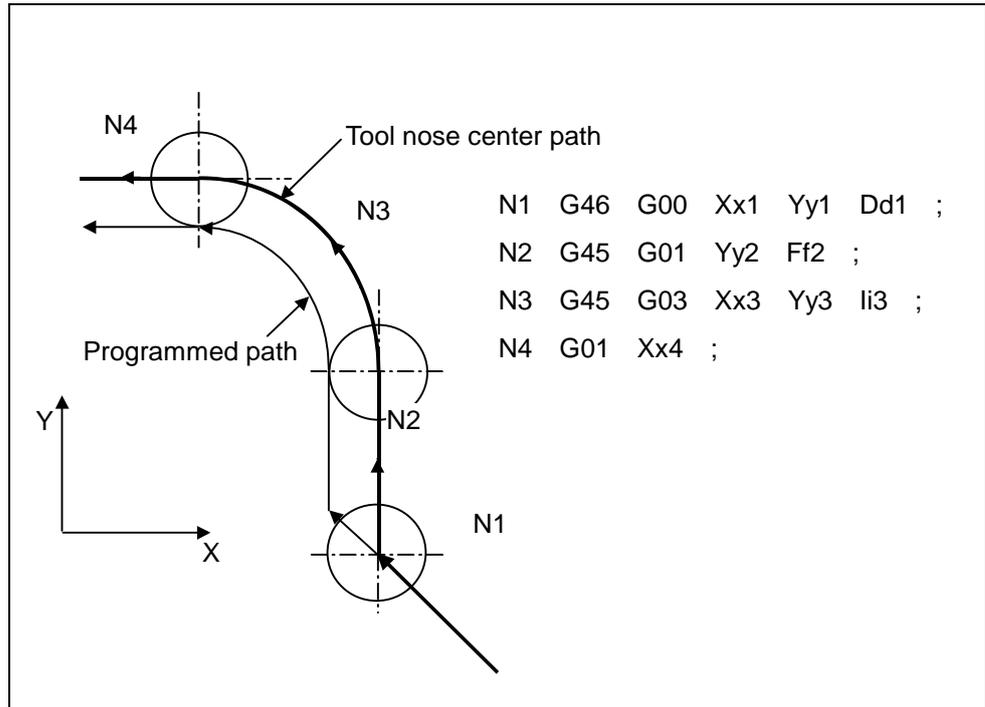
When a command for "n" number of simultaneous axes is given, the same compensation will be applied to all axes. It is valid even for the additional axes (but it must be within the range of the number of axes which can be controlled simultaneously.)



**(Note 1)** If compensation is applied to two axes, over-cutting or under-cutting will result, as shown in the figures below. In cases like this, use the cutter compensation commands (G40 to G42).

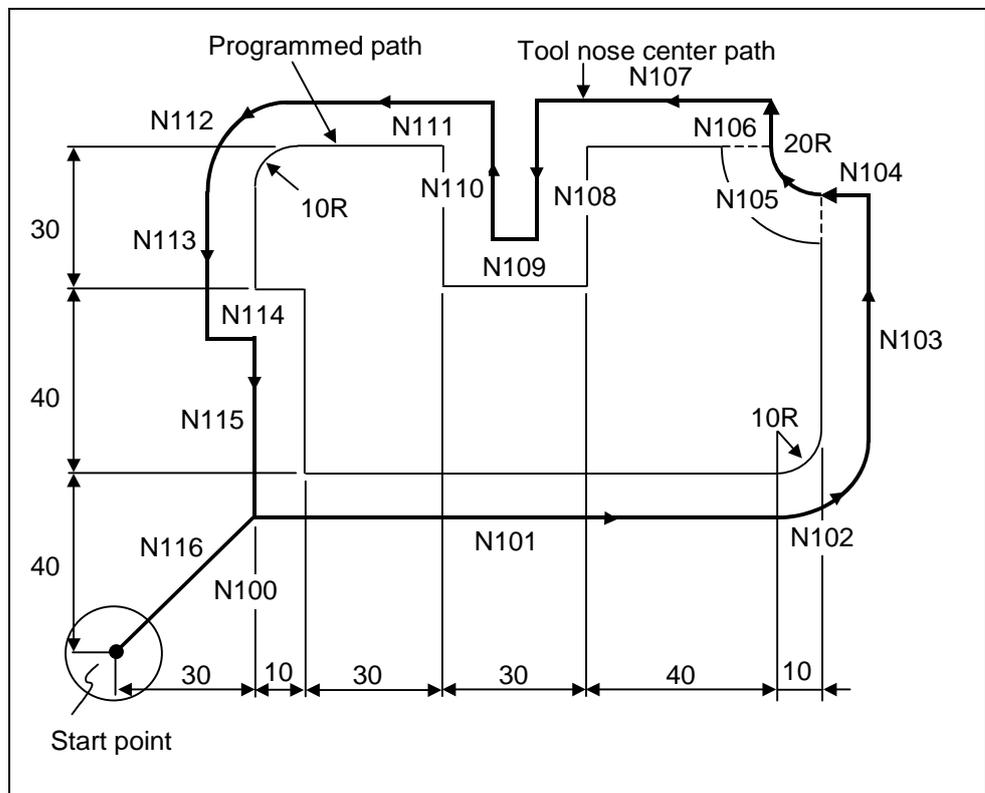


(Example 2)



(Example 3)

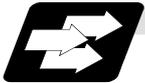
When the G45 to G48 command is assigned, the compensation amount for each pass is the amount assigned by the compensation number, and the tool does not move for the difference from the previous compensation as it would do with the tool length compensation command (G43).



Compensation amount D01 = 10.000mm (Offset amount of tool radius)

```
N100 G91 G46 G00 X40.0 Y40.0 D01 ;
N101 G45 G01 X100.0 F200 ;
N102 G45 G03 X10.0 Y10.0 J10.0 ;
N103 G45 G01 Y40.0 ;
N104 G46 X0 ;
N105 G46 G02 X-20.0 Y20.0 J20.0 ;
N106 G45 G01 Y0 ;
N107 G47 X-30.0 ;
N108 Y-30.0 ;
N109 G48 X-30.0 ;
N110 Y 30.0 ;
N111 G45 X-30.0 ;
N112 G45 G03 X-10.0 Y-10.0 J-10.0 ;
N113 G45 G01 Y-20.0 ;
N114 X10.0 ;
N115 Y-40.0 ;
N116 G46 X-40.0 Y-40.0 ;
N117 M02 ;
%
```

## 12.7 Programmed Compensation Input; G10, G11.1



## Function and purpose

The tool compensation and workpiece offset can be set or changed on the tape using the G10 command. During the absolute value (G90) mode, the commanded compensation amount will become the new compensation amount, and during the incremental value (G91) mode, the commanded compensation amount will be added to the currently set compensation amount to create the new compensation amount.



## Command format

## (1) Workpiece offset input

```
G90 G10 L2 P_X_Y_Z_;
G91
P          : 0 External workpiece
            1 G54
            2 G55
            3 G56
            4 G57
            5 G58
            6 G59
```

**(Note)** The offset amount in the G91 will be an incremental amount and will be cumulated each time the program is executed. Command G90 or G91 before the G10 as a cautionary means to prevent this type of error.

## (2) Tool compensation input

(a) For tool compensation memory I

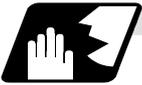
```
G10 L10 P_R_ ;
P          : Compensation No.
R          : Compensation amount
```

(b) For tool compensation memory II

```
G10 L10 P_R_ ;      Tool length compensation shape compensation
G10 L11 P_R_ ;      Tool length compensation wear compensation
G10 L12 P_R_ ;      Tool radius shape compensation
G10 L13 P_R_ ;      Tool radius wear compensation
```

## (3) Offset input cancel

```
G11.1
```



**Detailed description**

- (1) Program error (P171) will occur if this command is input when the specifications are not available.
- (2) G10 is an unmodal command and is valid only in the commanded block.
- (3) The G10 command does not contain movement, but must not be used with G commands other than G54 to G59, G90 or G91.
- (4) Do not command G10 in the same block as the fixed cycle and sub-program call command. This will cause malfunctioning and program errors.
- (5) The workpiece offset input command (L2 or L20) should not issued in the same block as the tool compensation input command (L10).
- (6) If an illegal L No. or compensation No. is commanded, the program errors (P172 and P170) will occur respectively.  
If the offset amount exceeds the maximum command value, the program error (P35) will occur.
- (7) Decimal point inputs can be used for the offset amount.
- (8) The offset amounts for the external workpiece coordinate system and the workpiece coordinate system are commanded as distances from the basic machine coordinate system zero point.
- (9) The workpiece coordinate system updated by inputting the workpiece coordinate system will follow the previous modal (G54 to G59) or the modal (G54 to G59) in the same block.
- (10) L2 (or L20) can be omitted when the workpiece offset is input.
- (11) If the P command is omitted at the workpiece offset input, the currently selected workpiece offset will be handled as the input.
- (12) When the "P" to designate the compensation No. is commanded in the same block as G22.1 or G23.1, the tool compensation data will not be input. "P" will be regarded as the number of repetition of subprogram call, that will cause an illegal operation.



**Example of program**

**(1) Input the compensation amount**

```
..... ; G10L10P10R-12345 ; G10L10P05R98765 ; G10L10P30R2468 ; ...
```

H10=-12345 H05=98765 H30=2468

**(2) Updating of compensation amount**

**(Example 1)** Assume that H10 = -1000 is already set.

N1 G01 G90 G43 Z - 100000 H10;	(Z = -101000)
N2 G28 Z0;	
N3 G91 G10 L10 P10R - 500 ;	(The mode is the G91 mode, so -500 is added.)
N4 G01 G90 G43 Z - 100000 H10 ;	(Z = -101500)

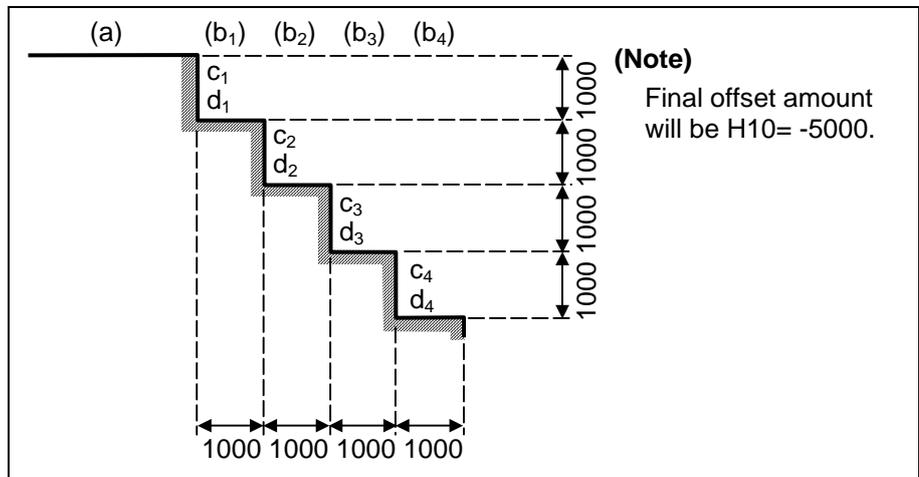
(Example 2) Assume that H10 = -1000 is already set.

**Main program**

N1	G00 X100000 ;	a
N2	#1 = -1000 ;	
N3	G22 L1111 L4 ;	b <sub>1</sub> , b <sub>2</sub> , b <sub>3</sub> , b <sub>4</sub>

**Subprogram L1111**

N1	G01 G91 G43 Z0 H10 F100 ;	c <sub>1</sub> , c <sub>2</sub> , c <sub>3</sub> , c <sub>4</sub>
	G01 X1000 ;	d <sub>1</sub> , d <sub>2</sub> , d <sub>3</sub> , d <sub>4</sub>
	#1 = #1 - 1000 ;	
	G90 G10 L10 P10 R#1 ;	
	G23 ;	



(Example 3) The program for Example 2 can also be written as follows.

**Main program**

N1	G00 X100000 ;
N2	G22 L1111 L4 ;

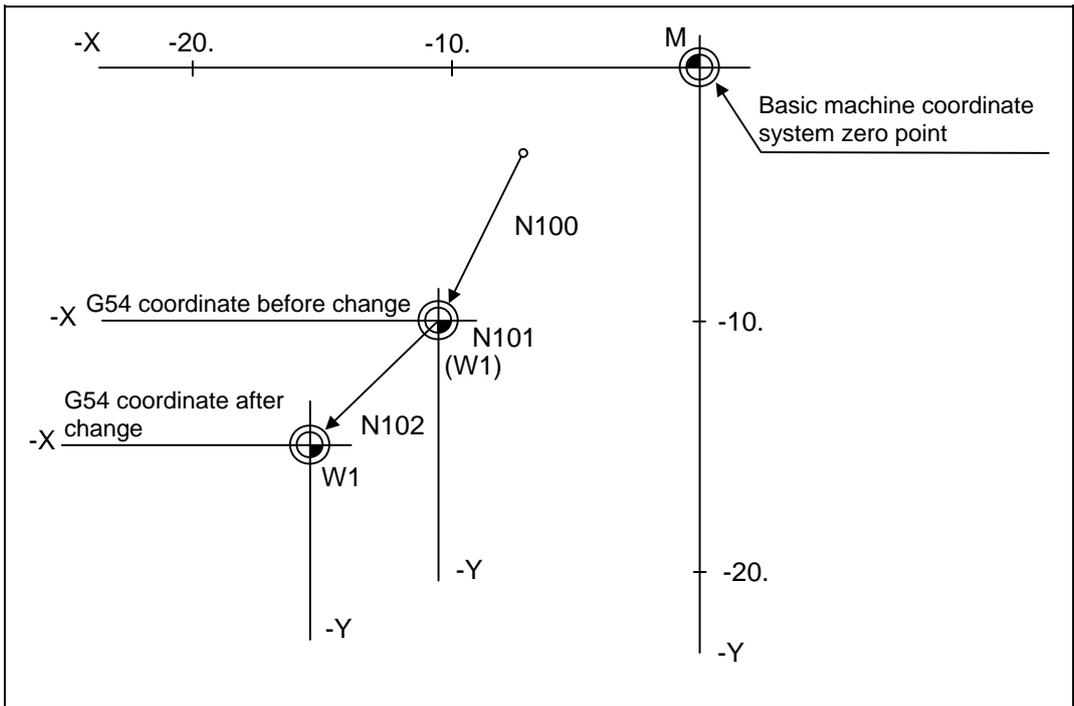
**Subprogram L1111**

N1	G01 G91 G43 Z0 H10 F100 ;
N2	G01 X1000 ;
N3	G10 L10 P10 R-1000 ;
N4	G23 ;

**(3) When updating the workpiece coordinate system offset amount**

Assume that the previous workpiece coordinate system offset amount is as follows.  
 $X = -10.000$     $Y = -10.000$

N100	G00 G90 G54 X0 Y0 ;
N101	G90 G10 L2 P1 X-15.000 Y-15.000 ;
N102	X0 Y0 ;
M02	;



**(Note 1)** Changes of workpiece position display at N101  
 At N101, the G54 workpiece position display data will change before and after the workpiece coordinate system is changed with G10.

$$\begin{array}{l} X = 0 \qquad X = +5.000 \\ \qquad \qquad \rightarrow \\ Y = 0 \qquad Y = +5.000 \end{array}$$

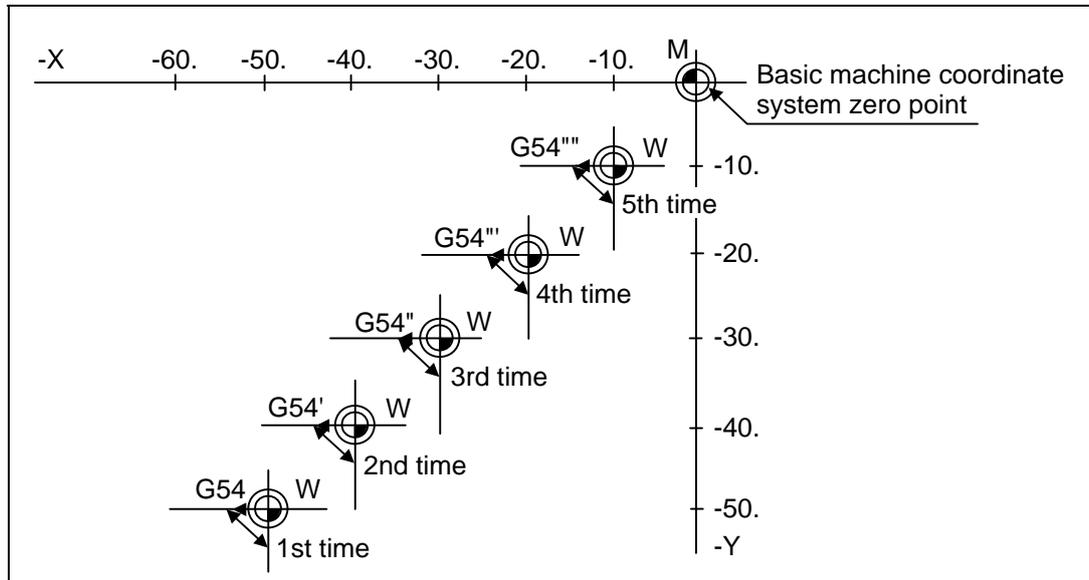
When workpiece coordinate system offset amount is set in G54 to G59

G90 G10 L2 P1 X-10.000 Y-10.000 ;
G90 G10 L2 P2 X-20.000 Y-20.000 ;
G90 G10 L2 P3 X-30.000 Y-30.000 ;
G90 G10 L2 P4 X-40.000 Y-40.000 ;
G90 G10 L2 P5 X-50.000 Y-50.000 ;
G90 G10 L2 P6 X-60.000 Y-60.000 ;

(4) When using one workpiece coordinate system as multiple workpiece coordinate systems

```

      ⋮
Main program { #1 = -50. #2 = 10.;
               G22 L200 P5;
               M02;
               %
Subprogram   { N1 G90 G54 G10 L2 P1 X#1 Y#1;
               L200 { N2 G00 X0 Y0;
                     N3 X-5. F100;
                     N4 X0 Y-5.;
                     N5 Y0;
                     N6 #1 = #1 + #2;
                     N7 G23;
                     %
    
```



**Precautions**

- (1) Even if this command is displayed on the screen, the offset No. and variable details will not be updated until actually executed.
  - N1 G90 G10 L10 P10R-100 ;
  - N2 G43 Z-10000 H10 ;
  - N3 G0 X-10000 Y-10000 ;
  - N4 G90 G10 L10 P10 R-200 ;...The H10 offset amount is updated when the N4 block is executed.

## 12. Tool Compensation Functions

### 12.8 Compensation Data Input to Variable by Program; G11

#### 12.8 Compensation Data Input to Variable by Program; G11



##### Function and purpose

Using G11, the compensation amount of the No. commanded as the transmission source can be set into the arbitrary variable.



##### Command format

**G11 P\_\_ Q\_\_;**      **Compensation data input to variable by program**

P        : Compensation No. of transmission source

Q        : Variable No. of transmission destination

Tool compensation memory type	Function	
Type A	Tool compensation amount substituted	
Type B	Tool length shape compensation amount	Substituted
	Tool length wear compensation amount	--
	Tool radius shape compensation amount	--
	Tool radius wear compensation amount	--

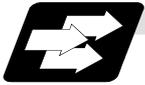


##### Detailed description

- (1) G11 command is unmodal, thus, it is valid only in the block where it is commanded.
- (2) G11 must not be used with G codes other than G54 to G59, G90 or G91. If G11 is used with unsuitable G codes, an illegal operation will result.
- (3) When G11 is commanded in the same block as G22 or G23, the tool compensation data will not be input. ("P" will be regarded as the number of repetition of subprogram call.)

12.9 Inputting the Tool Life Management Data; G10, G11

12.9.1 Inputting the Tool Life Management Data by G10 L3 Command



Function and purpose

Using the G10 command (unmodal command), the tool life management data can be registered, changed and added to, and preregistered groups can be deleted.



Command format

(1) Registering data

<b>G10 L3 ;</b>	<b>Start of life management data registration</b>	
<b>P_L_Q_;</b>	<b>Registration of the group No., the life, the control method</b>	} <b>First group</b>
<b>T_H_D_;</b>	<b>Registration of the tool No., the length compensation No., the radius compensation No.</b>	
<b>T_H_D_;</b>		} <b>Next group</b>
<b>P_L_Q_;</b>	<b>Registration of next group No., the life, the control method</b>	
<b>T_H_D_;</b>	<b>Registration of the tool No., the length compensation No., the radius compensation No.</b>	
<b>G11 ;</b>	<b>End of life management data registration</b>	
<b>P</b>	: Group No.	
<b>L</b>	: Life	
<b>Q</b>	: Control method	
<b>T</b>	: Tool No., The spare tools are selected in the order of the tool numbers registered here.	
<b>H</b>	: Length compensation No.	
<b>D</b>	: Radius compensation No.	

(2) Changing and adding groups

<b>G10 L3 P1;</b>	<b>Start of life management data change or addition</b>	
<b>P_L_Q_;</b>	<b>Change or addition of the group No., the life, the control method</b>	} <b>First group</b>
<b>T_H_D_;</b>	<b>Change or addition of the tool No., the length compensation No., the radius compensation No.</b>	
<b>T_H_D_;</b>		} <b>Next group</b>
<b>P_L_Q_;</b>	<b>Change or addition of next group No., the life, the control method</b>	
<b>T_H_D_;</b>	<b>Change or addition of the tool No., the length compensation No., the radius compensation No.</b>	
<b>G11 ;</b>	<b>End of life management data change or addition</b>	
<b>P</b>	: Group No.	
<b>L</b>	: Life	
<b>Q</b>	: Control method	
<b>T</b>	: Tool No.	
<b>H</b>	: Length compensation No.	
<b>D</b>	: Radius compensation No.	

(3) Deleting a group

<b>G10 L3 P2;</b>	<b>Start of life management data deletion</b>
<b>P_;</b>	<b>Delete the group No.</b>
<b>P_;</b>	<b>Delete next group No.</b>
<b>G11 ;</b>	<b>End life management data deletion</b>
<b>P</b>	: Group No.

## 12. Tool Compensation Functions

### 12.9 Inputting the Tool Life Management Data; G10, G11



#### Example of operation

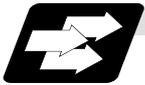
	Program example	Operation
Data registration	G10 L3; P10 L10 Q1; T10 H10 D10; G11; M02 ;	<ol style="list-style-type: none"> <li>1. After deleting all group data, the registration starts.</li> <li>2. Group No. 10 is registered.</li> <li>3. Tool No. 10 is registered in group No. 10.</li> <li>4. The registration ends.</li> <li>5. The program ends.</li> </ol>
Group change, addition	G10 L3 P1; P10 L10 Q1; T10 H10 D10; G11; M02 ;	<ol style="list-style-type: none"> <li>1. Changing and addition of the group and tool starts.</li> <li>2. The change and addition operation takes place in the following manner. <ol style="list-style-type: none"> <li>(1) When group No. 10 has not been registered. <ul style="list-style-type: none"> <li>• Group No. 10 is additionally registered.</li> <li>• Tool No. 10 is registered in group No. 10.</li> </ul> </li> <li>(2) When group No. 10 has been registered, but tool No.10 has not been registered. <ul style="list-style-type: none"> <li>• Tool No. 10 is additionally registered in group No. 10.</li> </ul> </li> <li>(3) When group No. 10 and tool No. 10 have been both registered. <ul style="list-style-type: none"> <li>• The tool No. 10 data is changed.</li> </ul> </li> </ol> </li> <li>3. The group and tool change and addition ends.</li> <li>4. The program ends.</li> </ol>
Group deletion	G1 L3 P2; P10; G11; M02 ;	<ol style="list-style-type: none"> <li>1. The group deletion starts.</li> <li>2. The group No. 10 data is deleted.</li> <li>3. The group deletion ends.</li> <li>4. The program ends.</li> </ol>



#### Command range

Item	Command range
Group No. (Pn)	1 to 99999999
Life (Ln)	0 to 9999 times (No. of times control method) 0 to 4000 minutes (time control method)
Control method (Qn)	1 to 3 1: Number of mounts control 2: Time control 3: Number of cutting times control
Tool No. (Tn)	1 to 99999999
Length compensation No. (Hn)	0 to 999
Radius compensation No. (Dn)	0 to 999

12.9.2 Inputting the Tool Life Management Data by G10 L30 Command



Function and purpose

Using the G10 command (unmodal command), the tool life management data can be registered, changed and added to, and preregistered groups can be deleted. Registering data To specify absolute value compensation amount or increment value compensation amount by control method, the length compensation and diameter compensation can be registered/changed with the tool compensation amount format.



Command format

(1) Registering data

<b>G10 L30 ;</b>	<b>Start of life management data registration</b>	
<b>P_L_Q_;</b>	<b>Registration of the group No., the life, the control method</b>	} <b>First group</b>
<b>T_H_R_;</b>	<b>Registration of the tool No., the length compensation (No./amount), the radius compensation (No./amount)</b>	
<b>T_H_R_;</b>		
<b>P_L_Q_;</b>	<b>Registration of next the group No., the life, the control method</b>	} <b>Next group</b>
<b>T_H_R_;</b>	<b>Registration of the tool No., the length compensation (No./amount), the radius compensation (No./amount)</b>	
<b>G11 ;</b>	<b>End of data registration for life management</b>	
<b>P</b>	: Group No.	
<b>L</b>	: Life	
<b>Q</b>	: Control method	
<b>T</b>	: Tool No., the spare tools are selected in the order of the tool Nos. registered here.	
<b>H</b>	: Length compensation No. or length compensation amount	
<b>R</b>	: Length diameter compensation No. or diameter compensation amount	

(2) Changing and adding groups

<b>G10 L30 P1;</b>	<b>Start of life management data change or addition</b>	
<b>P_L_Q_;</b>	<b>Change or addition of the group No., the life, the control method</b>	} <b>First group</b>
<b>T_H_R_;</b>	<b>Change or addition of the tool No., the length compensation (No./amount), the radius compensation (No./amount)</b>	
<b>T_H_R_;</b>		
<b>P_L_Q_;</b>	<b>Change or addition of the group No., the life, the control method</b>	} <b>Next group</b>
<b>T_H_R_;</b>	<b>Change or addition of the tool No., the length compensation (No./amount), the radius compensation (No./amount)</b>	
<b>G11 ;</b>	<b>End of life management data change or addition</b>	
<b>P</b>	: Group No.	
<b>L</b>	: Life	
<b>Q</b>	: Length compensation data format, radius compensation data format, control method	
<b>T</b>	: Tool No.	
<b>H</b>	: Length compensation No. or length compensation amount	
<b>R</b>	: Length diameter compensation No. or diameter compensation amount	

## 12. Tool Compensation Functions

### 12.9 Inputting the Tool Life Management Data; G10, G11

#### (3) Deleting a group

<b>G10 L30 P2;</b>	<b>Start of life management data deletion</b>
<b>P_;</b>	<b>Delete the group No.</b>
<b>P_;</b>	<b>Delete next group No.</b>
<b>G11 ;</b>	<b>End life management data deletion</b>
<b>P</b>	<b>: Group No.</b>

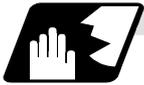


#### Example of operation

	Program example	Operation
Data registration	G10 L30 ; P10 L10 Q001 ; T10 H10 R10 ; G11 ; M02 ;	<ol style="list-style-type: none"> <li>After deleting all group data, the registration starts.</li> <li>Group No. 10 is registered. Tool management method is number of mounts Compensation No. method is applied to tool length compensation and tool radius compensation.</li> <li>Tool No. 10 is registered in group No. 10.</li> <li>The registration ends.</li> <li>The program ends.</li> </ol>
Group change, addition	G10 L30 P1 ; P10 L10 Q122 ; T10 H10 R0.25 ; G11 ; M02 ;	<ol style="list-style-type: none"> <li>Changing and addition of the group and tool starts.</li> <li>The change and addition operation takes place in the following manner. <ol style="list-style-type: none"> <li>When group No. 10 has not been registered. <ol style="list-style-type: none"> <li>Group No. 10 is additionally registered. The change and addition tool <ul style="list-style-type: none"> <li>Tool management method is number of usages</li> <li>Tool length compensation is incremental value compensation amount method</li> <li>Tool radius compensation is absolute value compensation amount method</li> </ul> </li> <li>Tool No. 10 is registered in group No. 10. The absolute value compensation amount "0.25" is set to tool radius compensation.</li> </ol> </li> <li>When group No. 10 has been registered, but tool No. 10 has not been registered. <ul style="list-style-type: none"> <li>Tool No. 10 is additionally registered in group No. 10.</li> </ul> </li> <li>When group No. 10 and tool No. 10 have been both registered. <ul style="list-style-type: none"> <li>The tool No. 10 data is changed.</li> </ul> </li> </ol> </li> <li>The group and tool change and addition ends.</li> <li>The program ends.</li> </ol>
Group deletion	G10 L30 P2 ; P10 ; G11 ; M02 ;	<ol style="list-style-type: none"> <li>The group deletion starts.</li> <li>The group No. 10 data is deleted.</li> <li>The group deletion ends.</li> <li>The program ends.</li> </ol>

## 12. Tool Compensation Functions

### 12.9 Inputting the Tool Life Management Data; G10, G11



#### Command range

Item		Command range
Group No.	(Pn)	1 to 99999999
Tool No.	(Tn)	1 to 99999999
Control method	(Qabc)	abc : Three integer digits a. Tool length compensation data format 0: Compensation No. 1: Incremental value compensation amount 2: Absolute value compensation amount b. Tool radius compensation data format 0: Compensation No. 1: Incremental value compensation amount 2: Absolute value compensation amount c. Tool management method 0: Usage time 1: Number of mounts 2: Number of usages
Life	(Ln)	0 to 4000 minutes (usage time) 0 to 9999 times (number of mounts) 0 to 9999 times (number of usages)
Length compensation (No./amount)	(Hn)	0 to 999 (compensation No.) ±9999.999 (incremental value compensation amount) ±9999.999 (absolute value compensation amount)
Radius compensation (No./amount)	(Dn)	0 to 999 (compensation No.) ±9999.999 (incremental value compensation amount) ±9999.999 (absolute value compensation amount)

## 12.9.3 Precautions for Inputting the Tool Life Management Data



## Precautions

- (1) The tool life data is registered, changed, added to or deleted by executing the program in the memory or MDI mode.
- (2) The group No. and tool No. cannot be commanded in duplicate.
- (3) When two or more addresses are commanded in one block, the latter address will be valid.
- (4) If the life data (L\_) is omitted, the life data for that group will be "0".
- (5) If the control method (Q\_) is omitted, the control method for that group will follow the base specification parameter "#1106 Tcount".  
Note that when carrying out the No. of cutting times control method, command the method from the program.
- (6) If the control method (Q\_) is not designated with 3-digit by G10 L30 command, the omitted high-order are equivalent to "0".  
Therefore, "Q1" is equivalent to "Q001", and "Q12" is equivalent to "Q012".
- (7) If the length compensation No. (H\_) is omitted, the length compensation No. for that group will be "0".
- (8) If the radius compensation No. (D\_) is omitted, the radius compensation No. for that group will be "0".
- (9) Programming with a sequence No. is not possible between G10 L3 or G10 L30 and G11.  
Program error (P33) will occur.
- (10) If the usage data count valid signal (YC8A) is ON, G10 L3 or G10 L30 cannot be commanded.
- (11) The registered data is held even if the power is turned OFF.
- (12) When G10 L3 or G10 L30 is commanded, the commanded group and tool will be registered after all of the registered data is erased.
- (13) The change and addition conditions in the G10L3P1 or G10 L30 P1 command are as follows.
  - (a) Change conditions
    - Both the commanded group No. and tool No. are registered.  
→Change the commanded tool No. data.
  - (b) Additional conditions
    - Neither the commanded group No. nor tool No. is registered.  
→Additionally register the commanded group No. and tool No. data.
    - The commanded group No. is registered, but the commanded tool No. is not registered.  
→Additionally register the commanded tool No. data to the commanded group No.
- (14) The setting range of the tool compensation No. will differ according to the specifications (number of tool compensation sets).



## Relation with other functions

- (1) During the following operations, the tool usage data will not be counted.
  - Machine lock
  - Miscellaneous function lock
  - Dry run
  - Single block
  - Skip

## 13. Program Support Functions

### 13.1 Fixed Cycles

#### 13.1.1 Standard Fixed Cycles; G80 to G89, G73, G74, G75, G76



#### Function and purpose

These standard canned cycles are used for predetermined sequences of machining operations such as positioning, hole drilling, boring, tapping, etc. which are specified in a block. The various sequences in the table below are provided for the standard canned cycles.

By editing the standard canned cycle subprogram, the canned cycle sequence can be changed by the user. The user can also register and edit an original canned cycle program. For the standard canned cycle subprogram, refer to the list of the canned cycle subprogram in the appendix of the operation manual. The list of canned cycle functions for this control unit is shown below.

G code	Hole machining start (-Z direction)	Operation at hole bottom		Return operation (+Z direction)	Retract at high-speed	Application
		Dwell	Spindle			
G80	-	-	-	-	-	Cancel
G81	Cutting feed	-	-	Rapid feed	Available	Drill, spot drilling cycle
G82	Cutting feed	Yes	-	Rapid feed	-	Drill, counter boring cycle
G83	Intermittent feed	-	-	Rapid feed	Available	Deep hole drilling cycle
G84	Cutting feed	Yes	Reverse rotation	Cutting feed	-	Tapping cycle
G85	Cutting feed	-	-	Cutting feed	-	Boring cycle
G86	Cutting feed	Yes	Stop	Rapid feed	-	Boring cycle
G87	Cutting feed	-	Forward rotation	Cutting feed	-	Back boring cycle
G88	Rapid traverse	Yes	Stop	Rapid feed	-	Boring cycle
G89	Cutting feed	Yes	-	Cutting feed	-	Boring cycle
G73	Cutting feed	Yes	-	Rapid feed	Available	Stepping cycle
G74	Intermittent feed	Yes	Forward rotation	Cutting feed	-	Reverse tapping cycle
G75	Cutting feed	-	-	Rapid feed	-	Circular cutting cycle
G76	Cutting feed	-	Oriented spindle stop	Rapid feed	-	Fine boring cycle

A canned cycle mode is canceled when the G80 or any G command in (G00, G01, G02, G03) is issued. The various data will also be cleared simultaneously to zero.



## Command format

## (1) Label L

<b>G8Δ (G7Δ)</b>	<b>X__ Y__ Z__ R__ Q__ P__(E__) F__ L__(H__) S__ ,S__ ,I__ ,J__ ;</b>
<b>G8Δ (G7Δ)</b>	<b>X__ Y__ Z__ R__ Q__ P__(E__) F__ L__(H__) S__ ,R__ ,I__ ,J__ ;</b>
G8Δ (G7Δ)	: Hole machining mode
X Y	: Hole positioning data
Z R Q P (E) F	: Hole machining data
L(H)	: Number of repetitions(When "0" is set, drilling is not executed at a command block.)
S	: Spindle rotation speed
, S	: Spindle rotation speed during retraction
, R	: Synchronization changeover
, I	: Positioning axis in-position width
, J	: Drilling axis in-position width

## (2) Label O

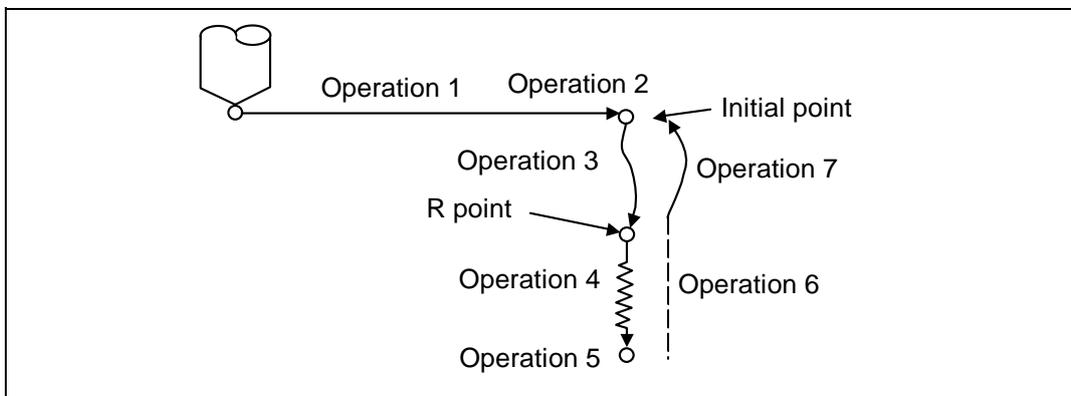
<b>G8Δ (G7Δ)</b>	<b>X__ Y__ Z__ R__ Q__ E__ F__ P__ S__ ,S__ ,I__ ,J__ ;</b>
<b>G8Δ (G7Δ)</b>	<b>X__ Y__ Z__ R__ Q__ E__ F__ P__ S__ ,R__ ,I__ ,J__ ;</b>
G8Δ (G7Δ)	: Hole machining mode
X Y	: Hole positioning data
Z R Q E F	: Hole machining data
P	: Number of repetitions(When "0" is set, drilling is not executed at a command block.)
S	: Spindle rotation speed
, S	: Spindle rotation speed during retraction
, R	: Synchronization changeover
, I	: Positioning axis in-position width
, J	: Drilling axis in-position width

As shown above, the format is divided into the hole machining mode, hole positioning data, hole machining data, No. of repetitions, spindle rotation speed, synchronization changeover (or spindle rotation speed during return), positioning axis in-position width and drilling axis in-position width.



## Detailed description

- (1) Outline of data and corresponding addresses
  - (a) Hole machining mode: Canned cycle modes such as drilling, counter boring, tapping and boring.
  - (b) Hole position data: Data used to position the X and Y axes. (Unmodal)
  - (c) Hole machining data: Actual machining data used for machining. (Modal)
  - (d) Number of repetitions: Number of times to carry out hole machining. (Unmodal)
  - (e) Synchronization changeover: Synchronous/asynchronous tapping selection command is issued during G84/G74 tapping. (Modal)
- (2) If M00 or M01 is commanded in the same block as the canned cycle or during the canned cycle mode, the canned cycle will be ignored. Instead, M00 and M01 will be output after positioning. The canned cycle will be executed if X, Y, Z or R is commanded.
- (3) There are 7 actual operations which are each described in turn below.



Operation 1 : This indicates the X and Y axes positioning, and executes positioning with G00.

Operation 2 : This is an operation done after positioning is completed (at the initial hole), and when G87 is commanded, the M10 command is output from the control unit to the machine. When this M command is executed and the finish signal (FIN) is received by the control unit, the next operation will start. If the single block stop switch is ON, the block will stop after positioning.

Operation 3 : The tool is positioned to the R point by rapid traverse.

Operation 4 : Hole machining is conducted by cutting feed.

Operation 5 : This operation takes place at the hole bottom position and it differs according to the canned cycle mode. Possible actions include spindle stop (M05) spindle reverse rotation (M04), spindle forward rotation (M03), dwell and tool shift.

Operation 6 : The tool is retracted to the R point.

Operation 7 : The tool is returned to the initial point at the rapid traverse rate.

Whether the canned cycle is to be completed at operation 6 or 7 can be selected by the following G commands.

G98 ..... Initial level return

G99 ..... R point level return

These are modal commands, and for example, if G98 is commanded once, the G98 mode will be entered until G99 is designated. The initial state when the NC is ready is the G98 mode.

The hole machining data will be ignored if X, Y, Z or R is not commanded. This function is mainly used with the special canned cycled.

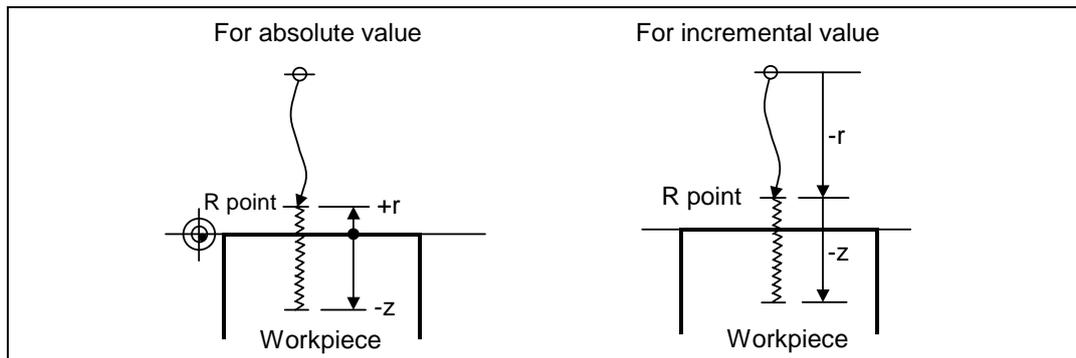
## (4) Canned cycle addresses and meanings

Address		Significance
Label L	Label O	
G	G	Selection of hole machining cycle sequence (G80 to G89, G73, G74, G76)
X	X	Designation of hole drilling position (absolute value or incremental value)
Y	Y	Designation of hole drilling position (absolute value or incremental value)
Z	Z	Designation of hole bottom position (absolute value or incremental value)
P(E)	E	Designation of dwell time at hole bottom position (decimal points will be ignored) or designation of tool radius compensation No. for G75
Q	Q	Designation of cut amount for each cutting pass for G73 or G83, designation of the shift amount for G76 or G87 (incremental value), or designation of the radius of outer circumference for G76 or G87
R	R	Designation of R point position (absolute value or incremental value)
F	F	Designation of feed rate for cutting feed (asynchronous tapping) or number of Z axis pitch (synchronous tapping)
L(H)	P	Designation of number of repetitions. (0 to 9999) When "0" is set, no execution
I, J, K	I, J, K	Designation of shift amount for G76 or G87 (incremental value) (The shift amount is designated with Q address according to the parameter setting.)
S	S	Spindle rotation speed command <b>(Note1)</b> S command that was issued with the form of "Sn = *****" will be ignored during the synchronous tapping. (n:spindle No., *****:rotation speed) <b>(Note2)</b> If S command is issued in the synchronous tapping modal, the program error (P186) will occur.
,S	,S	Spindle rotation speed designation during retraction
,R	,R	Designation of synchronous tapping/asynchronous tapping (If ",R" setting is omitted, that depends on the parameter setting.)
M	M	Designation of miscellaneous command <b>(Note)</b> On the asynchronous tapping with normal tapping, spindle forward rotation will be applied for Mm1 basically. Note that the spindle reverse rotation should be commanded with M(m1 + 1), that is, the M code of spindle forward rotation command +1, or that could lead an incorrect operation.
,I	,I	Designation of positioning axis in-position width
,J	,J	Designation of drilling axis in-position width

**(Note1)** Only for G75, P address is handled as the compensation No. but not dwell time or number of repetition.

**(Note2)** When the synchronous tapping (,R) is commanded, the E address is handled as designation of the number of ridges.

(5) Difference between absolute value command and incremental value command



(6) Feed rate for tapping cycle and tapping return

The feed rates for the tapping cycle and tapping return are as shown below.

(a) Selection of synchronous tapping cycle/asynchronous tapping cycle

Program G84xxx, Rxx	Control parameter Synchronous tapping	Synchronous/asynchronous
, R00	—	Asynchronous
, Rxx	OFF	
No designation	ON	Synchronous
, R01	—	

— is irrelevant to the setting

(b) Selection of asynchronous tapping cycle feed rate

G94/G95	Control parameter F1-digit value	F command value	Feed designation
G94	OFF	F designation	Per-minute feed
	ON	Other than F0 to F8	
G95	ON	F0 to F8 (no decimal point)	F1-digit feed
	—	F designation	Per-revolution feed

— is irrelevant to the setting

(c) Spindle rotation speed during return of synchronous tapping cycle

Address	Meaning of address	Command range (unit)	Remarks
,S	Spindle rotation speed during return	0 to 99999 (r/min)	The data is held as modal information. If the value is smaller than the speed rotation speed, the speed rotation speed value will be valid even during return. If the spindle rotation speed is not 0 during return, the tap return override value will be invalid.

- (7) M code for forward/reverse rotation command.  
The M code set with the parameter "#3028 sprcmn" is output as the M code for spindle forward/reverse rotation that is output at "hole bottom" or at "R point" during asynchronous tapping cycle.  
Note that the M code for forward rotation is output as "M3" and that for reverse rotation is as "M4" if the parameter #3208) is set to "0".
- (8) With the label L, when the number of repetition is two or more times, the axis will return to the point determined by G98/G99 modal at the last time.
- (9) With the label O, when the number of repetition is two or more times, the axis will return to the initial point regardless of G98/G99 mode at the last time.
- (10) When there is no drilling position command, the drilling will not be carried out though the drilling data is saved.
- There are x and y commands : drilling will be carried out.
  - There are only z and r commands : data will be saved only (No drilling)



### Positioning plane and hole drilling axis

The canned cycle has basic control elements for the positioning plane and hole drilling axis. The positioning plane is determined by the G17, G18 and G19 plane selection command, and the hole drilling axis is the axis perpendicular (X, Y, Z or parallel axis) to the above plane.

Plane selection	Positioning plane	Hole drilling axis
G17 (X – Y)	Xp – Yp	Zp
G18 (Z – X)	Zp – Xp	Yp
G19 (Y – Z)	Yp – Zp	Xp

Xp, Yp and Zp indicate the basic axes X, Y and Z or an axis parallel to the basic axis. A random axis other than the hole drilling axis can be commanded for positioning. The hole drilling axis is determined by the axis address of the hole drilling axis commanded in the same block as G81 to G89, G73, G74 or G76. The basic axis will be the hole drilling axis if there is no designation.

**(Example1)** When G17 (XY plane) is selected, and the axis parallel to the Z axis is set as the W axis.  
G81 ... W\_\_ ; The W axis is used as the hole drilling axis.  
G81 ... Z \_\_ ; The Z axis is used as the hole drilling axis.  
G81 ... ; (No Z or W) The Z axis is used as the hole drilling axis.

**(Note 1)** The hole drilling axis can be fixed to the Z axis with parameter #1080 Dril\_Z.

**(Note 2)** Change over the hole drilling axis in the canned cycle canceled state.

In the following explanations on the movement in each canned cycle mode, the XY plane is used for the positioning plane and the Z axis for the hole drilling axis.

Note that all command values will be incremental values, the positioning plane will be the XY plane and the hole drilling axis will be the Z axis.



**Programmable in-position width command in fixed cycle**

This command commands the in-position width for the fixed cycle from the machining program. The commanded in-position width is valid only in the G81 (drill, spot drill), G82 (drill, counter boring), G83 (deep drill cycle), G84 (tap cycle), G85 (boring), G89 (boring), G73 (step cycle) and G74 (reverse tap cycle) fixed cycles. The ",I" address is commanded in respect to the positioning axis, and the ",J" address is commanded in respect to the drilling axis.

Address	Meaning of address	Command range (unit)	Remarks
,I	In-position width (position error amount)	0.001 to 999.999 (mm)	When a value exceeding the command range is issued, a program error will occur. (P35)
,J	In-position width for drilling axis in fixed cycle (position error amount)		



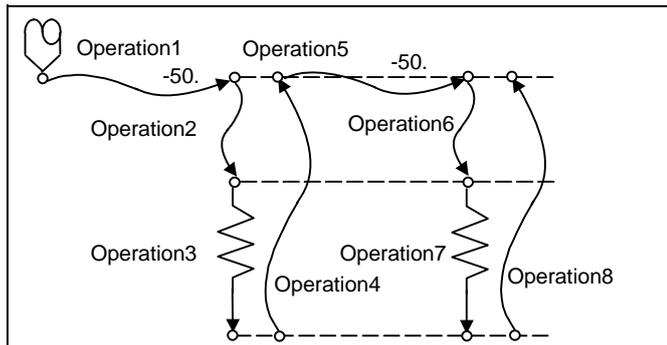
**In-position check in fixed cycle**

When the number of repetitions L is designated as two or more times in the fixed cycle, the commanded in-position width will be valid in the repetition block (operation 5 to operation 8).

```

:
G81 X-50. Z-50. R-50. L2 F2000, I0.2, J0.3 ;
:

```



Operation pattern	,I	,J
Operation 1	Valid	-
Operation 2	-	Invalid
Operation 3	-	Invalid
Operation 4	-	Valid
Operation 5	Valid	-
Operation 6	-	Invalid
Operation 7	-	Invalid
Operation 8	-	Valid

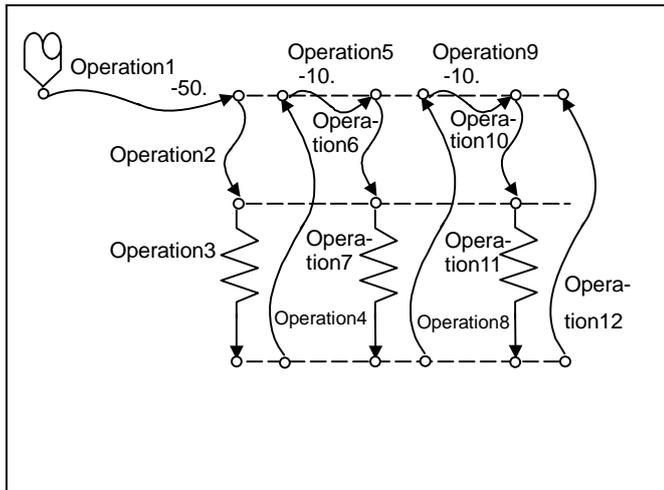
**Fig. 1 Operation when number of repetitions L is designated**

In the following machining program, the commanded in-position width is valid for the Fig. 2 block. In the (B) block, the in-position width (, I) commanded for positioning in the previous block (A) is invalid (operation 5). However, when returning from the hole bottom, the in-position width (, J) commanded in the previous block (A) is valid (operation 8). To validate the in-position width for positioning, command again as shown in block (C) (Operation 9).

```

:
G81 X-50. Z-50. R-50. F2000, I0.2, J0.3 ;.....(A)
X-10. ; .....(B)
X-10., I0.2 ; .....(C)
:

```



Operation pattern	,I	,J
Operation 1	Valid	–
Operation 2	–	Invalid
Operation 3	–	Invalid
Operation 4	–	Valid
Operation 5	Invalid	–
Operation 6	–	Invalid
Operation 7	–	Invalid
Operation 8	–	Valid
Operation 9	Valid	–
Operation 10	–	Invalid
Operation 11	–	Invalid
Operation 12	–	Valid

Fig. 2 Operation in fixed cycle modal



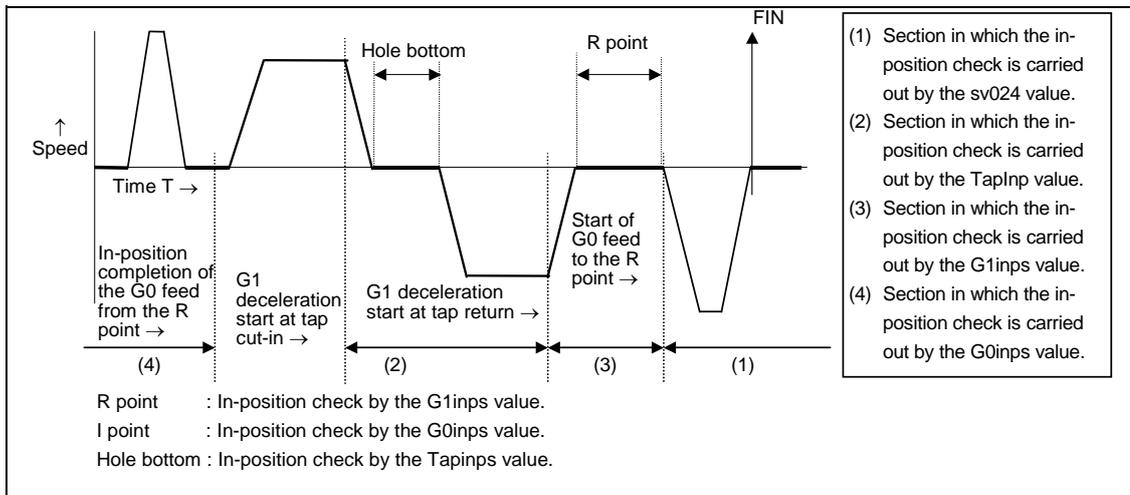
Setting values of synchronous tap in-position check parameters and tap axis movement

(1) Synchronous tap in-position check settings

#1223 aux07				"P" designation of G84/G74 command	In-position check during synchronous tapping
Bit3	Bit4	Bit5	Bit2		
Synchronous tapping in-position check	At hole bottom	At R point	I point → R point		
0	–	–	–	–	Carry out in-position check at I point → R point/R point/hole bottom.
1	–	–	–	No "P" designation Example: G84 F1. Z-5. S1000 R-5.	Hole bottom: Do not carry out in-position check. R point: Do not carry out in-position check. I point → R point: Do not carry out in-position check.
1	1	1	1	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Carry out in-position check by tap in-position width. R point: Carry out in-position check. I point → R point: Carry out in-position check.
1	1	0	1	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Carry out in-position check by tap in-position width. R point: Do not carry out in-position check. I point → R point: Carry out in-position check.
1	0	1	1	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Do not carry out in-position check. R point: Carry out in-position check. I point → R point: Carry out in-position check.
1	0	0	1	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Do not carry out in-position check. R point: Do not carry out in-position check. I point → R point: Carry out in-position check.
1	1	1	0	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Carry out in-position check by tap in-position width. R point: Carry out in-position check. I point → R point: Do not carry out in-position check.
1	1	0	0	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Carry out in-position check by tap in-position width. R point: Do not carry out in-position check. I point → R point: Do not carry out in-position check.
1	0	1	0	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Do not carry out in-position check. R point: Carry out in-position check. I point → R point: Do not carry out in-position check.
1	0	0	0	"P" designation Example: G84 F1. Z-5. S1000 <b>PO</b> R-5.	Hole bottom: Do not carry out in-position check. R point: Do not carry out in-position check. I point → R point: Do not carry out in-position check.

(Note 1) The I point refers to the initial point.

(2) Relation between the in-position width and tap axis movement for a synchronous tap in-position check

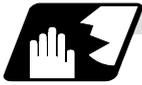


(3) Relation between the parameter setting values and tap axis movement for a synchronous tap in-position check

#1223 aux07				Hole bottom wait time	Operation at hole bottom	Operation at R point	Operation at I point → R point
Bit3	Bit4	Bit5	Bit2				
Synchronous tapping in-position check	At hole bottom	At R point	At I point → R point				
0	—	—	—	Time designated by "P". Processing time is several 10ms when no "P".	Operation determined by setting of inpos (#1193) and aux 07 (#1223): bit 1 parameters.	Operation determined by setting of inpos (#1193) and aux 07 (#1223): bit 1 parameters.	Operation determined by setting of inpos (#1193) and aux 07 (#1223): bit 1 parameters.
1	0	0	1	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in left column has elapsed.		Wait until completion of in-position check by G0inps value.
1	0	1	1	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in left column has elapsed.	Wait until completion of in-position check by G1inps value.	Wait until completion of in-position check by G0inps value.
1	1	0	1	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until dwell time in left column has elapsed after completion of in-position check.		Wait until completion of in-position check by G0inps value.
1	1	1	1	Larger value of "P" and TapDwl (#1313) is valid. Processing time is several 10ms if both values are 0.	Wait until dwell time in left column has elapsed after completion of in-position check.	Wait until completion of in-position check by G1inps value.	Wait until completion of in-position check by G0inps value.
1	0	0	0	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in left column has elapsed.		
1	0	1	0	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in left column has elapsed.	Wait until completion of in-position check by G1inps value.	
1	1	0	0	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until dwell time in left column has elapsed after completion of in-position check.		
1	1	1	0	Larger value of "P" and TapDwl (#1313) is valid. Processing time is several 10ms if both values are 0.	Wait until dwell time in left column has elapsed after completion of in-position check.	Wait until completion of in-position check by G1inps value.	

**(Note 1)** The I point refers to the initial point.

**Remarks:** The processing time is several 10ms when an in-position check is not carried out at the hole bottom.  
The processing time is between 0 to 14.2ms when an in-position check is not carried out at the R point.  
The processing time is the spindle gain changeover time when an in-position check is not carried out at I point → R point.

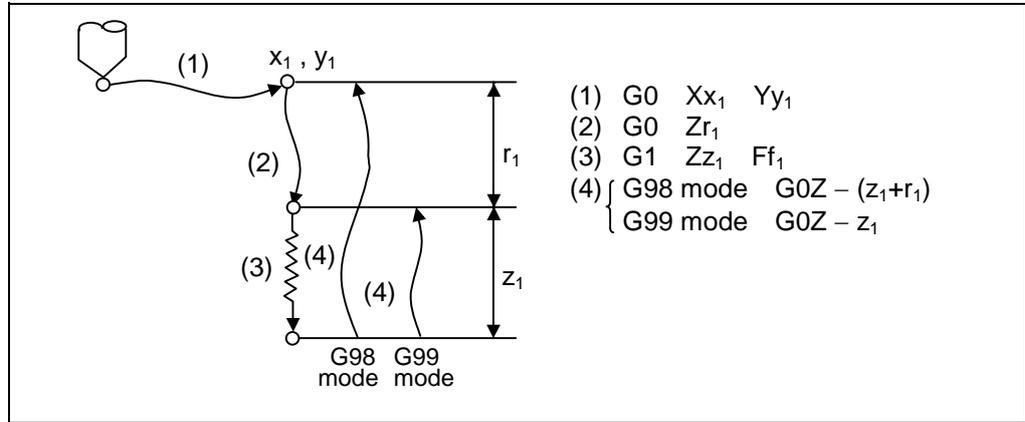


Movement when executing each fixed cycles

(a) G81 (Drilling, spot drilling)

Program

G81 Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> Rr<sub>1</sub> Ff<sub>1</sub> ,li<sub>1</sub> ,Jj<sub>1</sub>;



The operation stops at after the (1), (2) and (4) commands during single block operation.

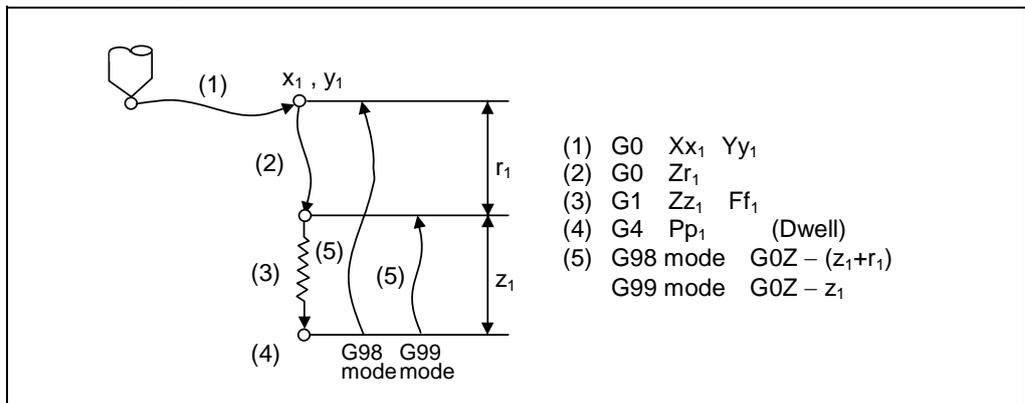
Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	Valid

(b) G82 (Drilling, counter boring)

Program

G82 Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> Rr<sub>1</sub> Ff<sub>1</sub> Pp<sub>1</sub> ,li<sub>1</sub> ,Jj<sub>1</sub>;

P : Dwell designation



Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	-
(5)	-	Valid

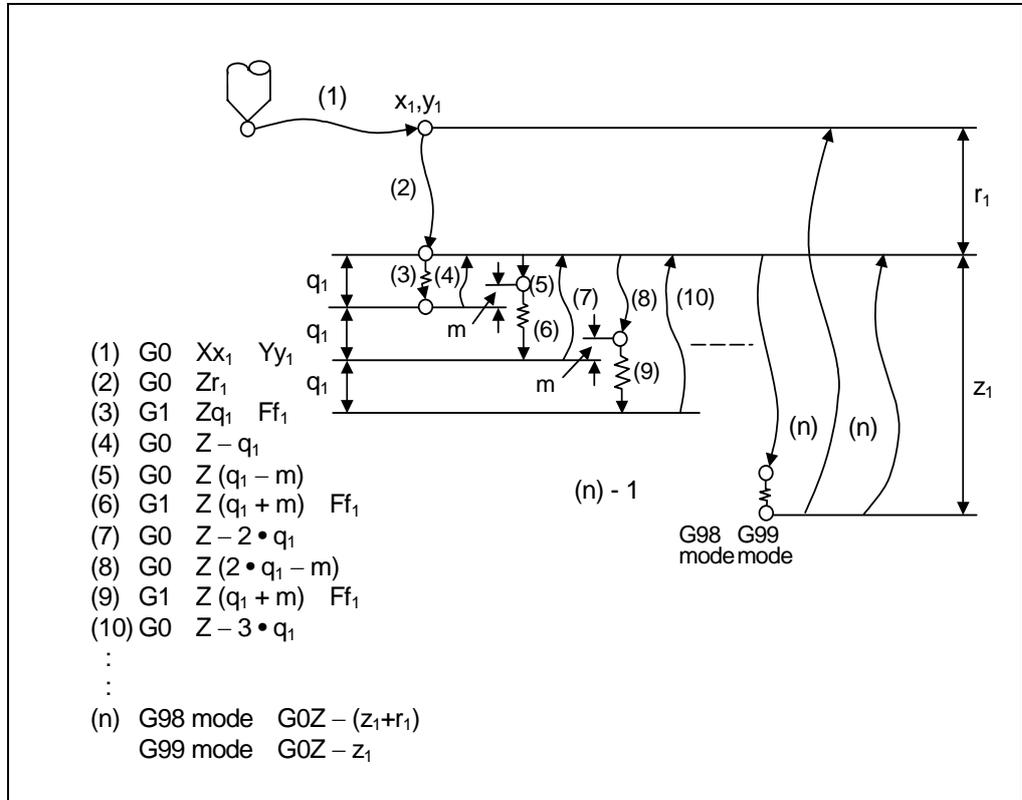
The operation stops at after the (1), (2) and (5) commands during single block operation.

(c) G83 (Deep hole drilling cycle)

Program

G83 Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> Rr<sub>1</sub> Qq<sub>1</sub> Ff<sub>1</sub> ,li<sub>1</sub> ,Jj<sub>1</sub>;

Q : This designates the cutting amount per pass, and is always designated with an incremental value.



Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	Invalid
(5)	-	Invalid
(6)	-	Invalid
(7)	-	Invalid
(8)	-	Invalid
(9)	-	Invalid
(10)	-	Invalid
⋮		
⋮		
n-1	-	Invalid
n	-	Valid

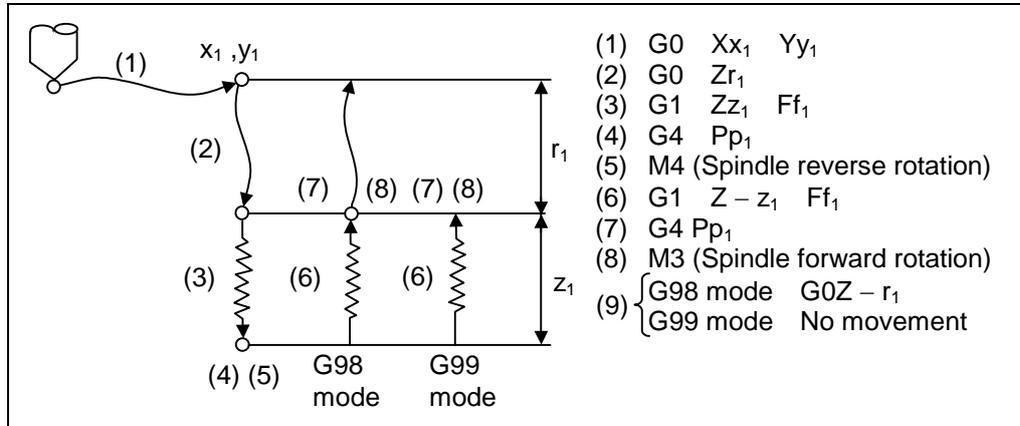
When executing a second and following cutting in the G83 as shown above, the movement will change from rapid traverse to cutting feed several mm ("m" mm in the figure above) before the position machined last. When the hole bottom is reached, the axis will return according to the G98 or G99 mode. m will differ according to the parameter "#8013 G83 return". Program so that  $q_1 > m$ . The operation stops at after the (1), (2) and n commands during single block operation.

(d) G84 (Tapping cycle)

Program

G84  $Xx_1 Yy_1 Zz_1 Rr_1 Ff_1 Pp_1 Rr_2$  (or  $S_1, S_2$ ),  $li_1, Jj_1$ ;

P : Dwell designation

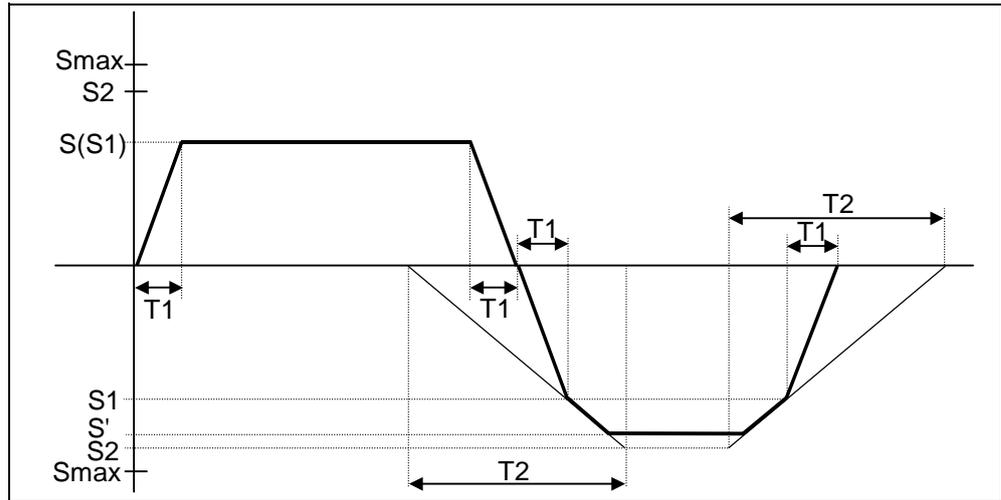


Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	-
(5)	-	-
(6)	-	Invalid
(7)	-	-
(8)	-	-
(9)	-	Valid

- When  $r_2 = 1$ , the synchronous tapping mode will be entered, and when  $r_2 = 0$ , the asynchronous tapping mode will be entered.
- When G84 is executed, the override will be canceled and the override will automatically be set to 100%.
- Dry run is valid when the control parameter "G00 DRY RUN" is on and is valid for the positioning command. If the feed hold button is pressed during G84 execution, and the sequence is at (3) to (6), the movement will not stop immediately, and instead will stop after (6). During the rapid traverse in sequence (1), (2) and (9), the movement will stop immediately.
- The operation stops at after the (1), (2) and (9) commands during single block operation.
- During the G84 modal, the "Tapping" NC output signal will be output.
- During the G84 synchronous tapping modal, the M3, M4, M5 and S code will not be output.

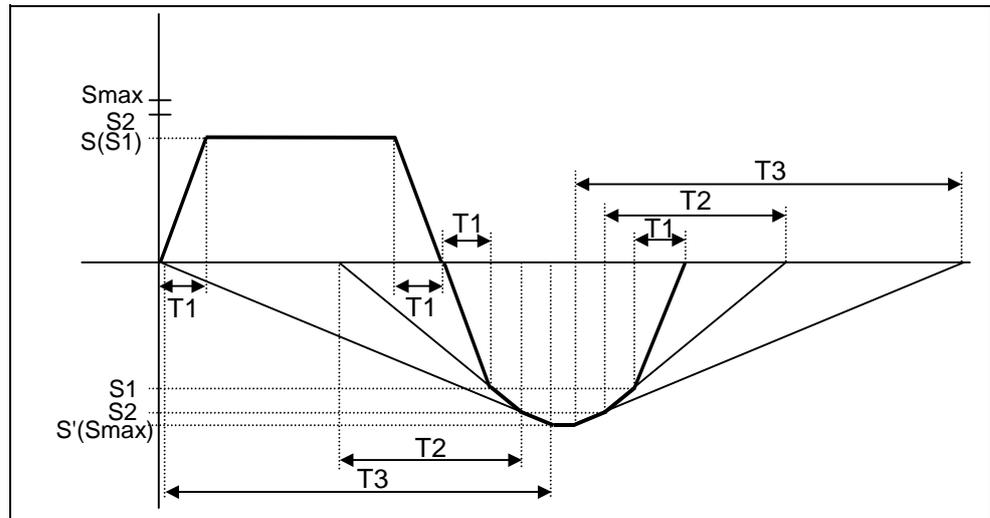
This function allows spindle acceleration/deceleration pattern to be approached to the speed loop acceleration/deceleration pattern by dividing the spindle and drilling axis acceleration/deceleration pattern into up to three stages during synchronous tapping. The acceleration/deceleration pattern can be set up to three stages for each gear. When returning from the hole bottom, rapid return is possible depending on the spindle rotation speed during return. The spindle rotation speed during return is held as modal information.

(i) When tap rotation speed < spindle rotation speed during return ≤ synchronous tap changeover spindle rotation speed 2



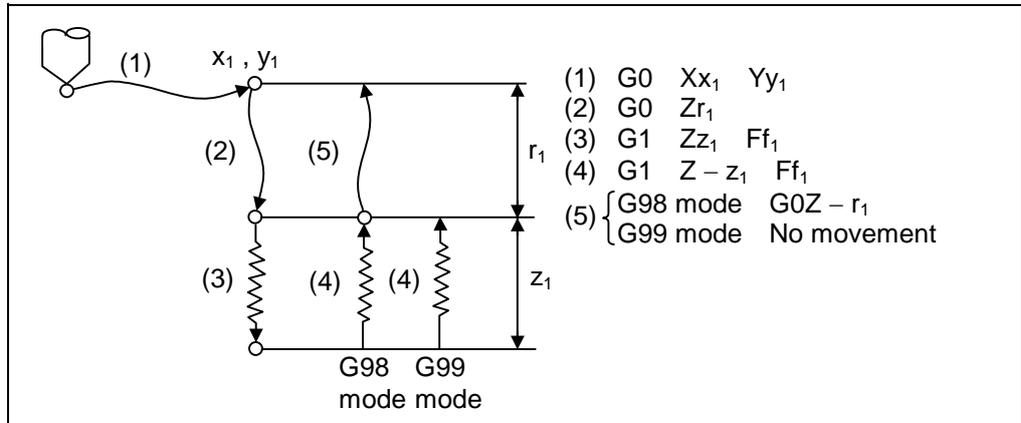
- S : Command spindle rotation speed
- S' : Spindle rotation speed during return
- S1 : Tap rotation speed (spindle base specification parameters #3013 to #3016)
- S2 : Synchronous tap changeover spindle rotation speed 2 (spindle base specification parameters #3037 to #3040)
- Smax : Maximum rotation speed (spindle base specification parameters #3005 to #3008)
- T1 : Tap time constant (spindle base specification parameters #3017 to #3020)
- T2 : Synchronous tap changeover time constant 2 (spindle base specification parameters #3041 to #3044)

(ii) When synchronous tap changeover spindle rotation speed 2 < spindle rotation speed during return



- S : Command spindle rotation speed
- S' : Spindle rotation speed during return
- S1 : Tap rotation speed (spindle base specification parameters #3013 to #3016)
- S2 : Synchronous tap changeover spindle rotation speed 2  
(spindle base specification parameters #3037 to #3040)
- Smax : Maximum rotation speed (spindle base specification parameters #3005 to #3008)
- T1 : Tap time constant (spindle base specification parameters #3017 to #3020)
- T2 : Synchronous tap changeover time constant 2  
(spindle base specification parameters #3041 to #3044)
- T3 : Synchronous tap changeover time constant 3  
(spindle base specification parameters #3045 to #3048)

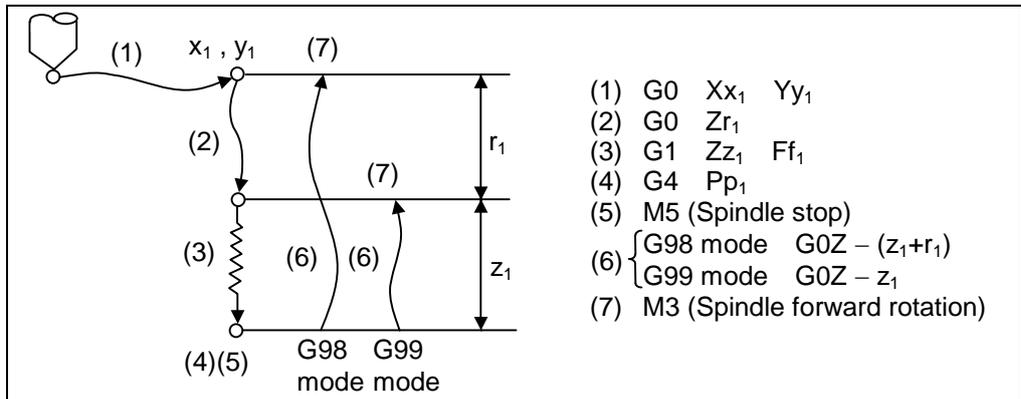
(e) G85 (Boring)  
 Program  
 G85  $Xx_1 Yy_1 Zz_1 Rr_1 Ff_1, Ii_1, Jj_1;$



Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	Invalid
(5)	-	Valid

The operation stops at after the (1), (2), and (4) or (5) commands during single block operation.

(f) G86 (Boring)  
 Program  
 G86  $Xx_1 Yy_1 Zz_1 Rr_1 Ff_1 Pp_1;$



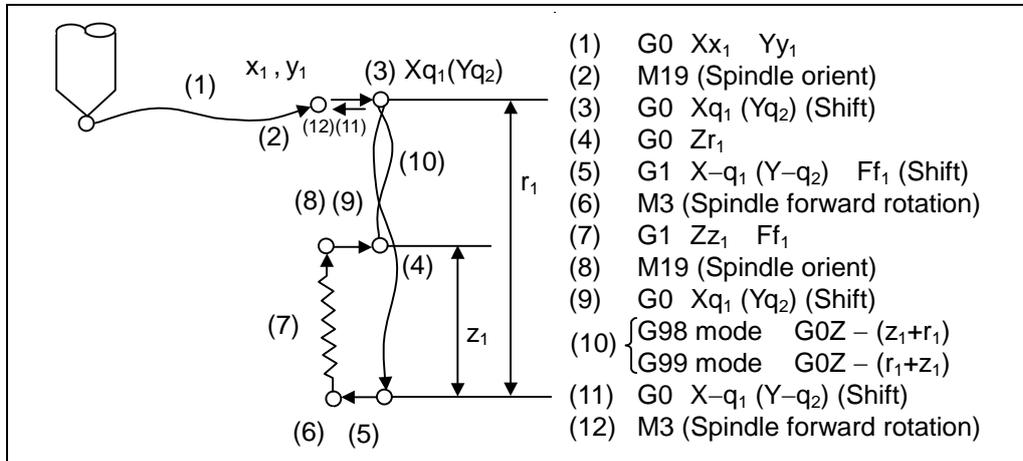
The operation stops at after the (1), (2) and (7) commands during single block operation.

(g) G87 (Back boring)

Program

G87  $Xx_1 Yy_1 Zz_1 Rr_1 Iq_1 Jq_2 Ff_1$  ;

**(Note)** Take care to the  $z_1$  and  $r_1$  designations.  
 (The  $z_1$  and  $r_1$  symbols are reversed).  
 There is no R point return.

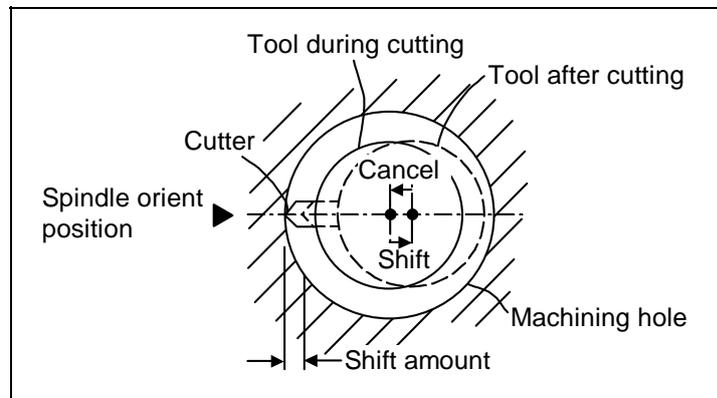


The operation stops at after the (1), (4), (6) and (11) commands during single block operation.

When this command is used, high precision drilling machining that does not scratch the machining surface can be done.

(Positioning to the hole bottom and the escape (return) after cutting is executed in the state shifted to the direction opposite of the cutter.)

The shift amount is designated as shown below with addresses I, J and K.



For G17 : I, J  
 For G18 : K, I  
 For G19 : J, K

The shift amount is executed with linear interpolation, and the feed rate follows the F command.

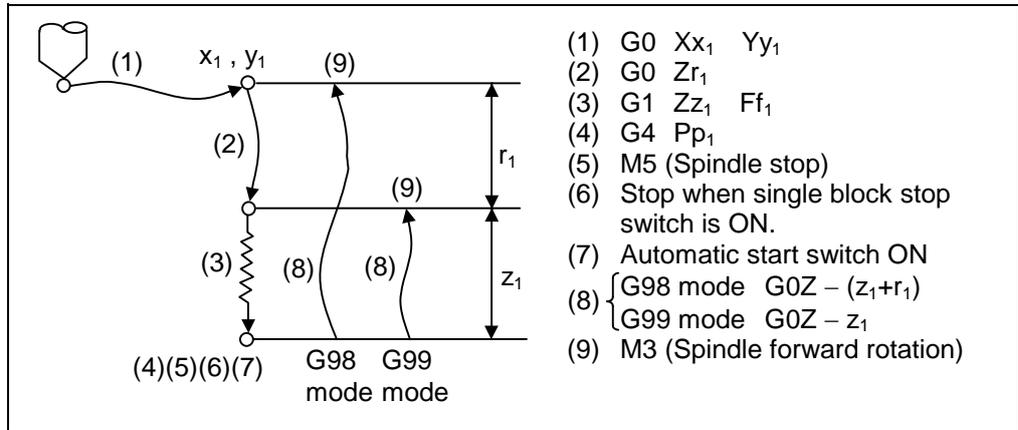
Command I, J, and K with incremental values in the same block as the hole position data. I, J and K will be handled as modals during the canned cycle.

**(Note)** If the parameter "#1080 Dril\_Z" which fixes the hole drilling axis to the Z axis is set, the shift amount can be designated with address Q instead of I and j. In this case, whether to shift or not and the shift direction are set with parameter "#8207 G76/87 No shift" and "#8208 G76/87 Shift (-)". The symbol for the Q value is ignored and the value is handled as a positive value. The Q value is a modal during the canned cycle, and will also be used as the G83, G73 and G76 cutting amount.

(h) G88 (Boring)

Program

G88  $Xx_1 Yy_1 Zz_1 Rr_1 Ff_1 Pp_1$  ;

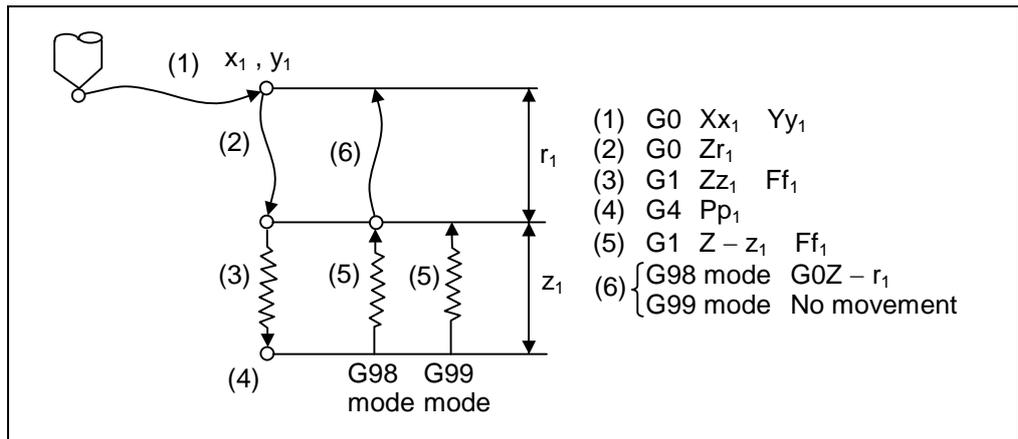


The operation stops at after the (1), (2), (6) and (9) commands during single block operation.

(i) G89 (Boring)

Program

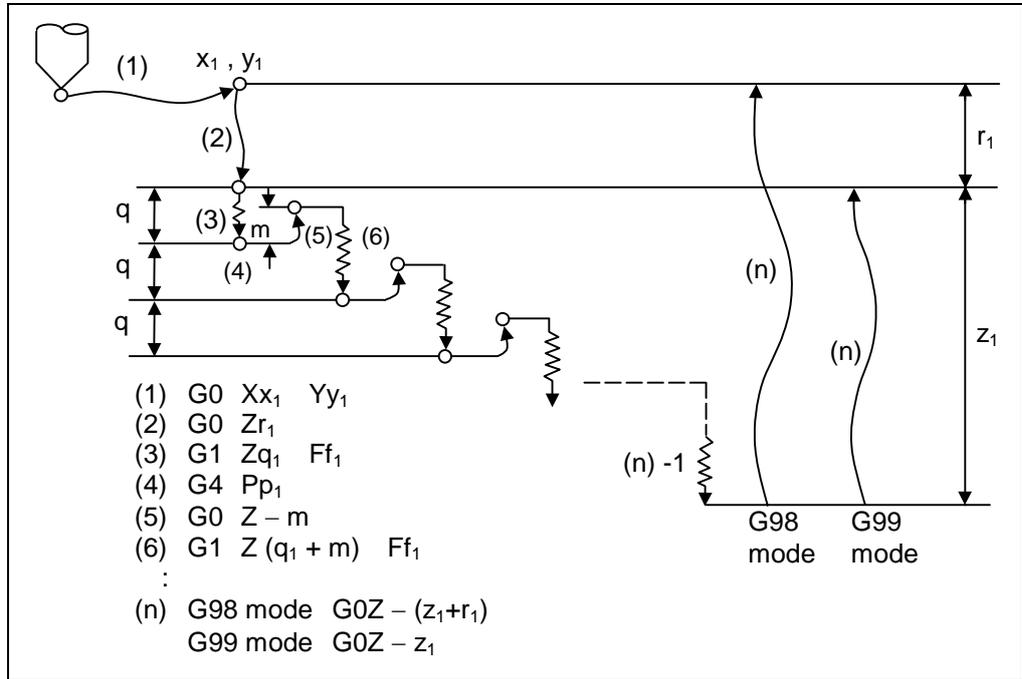
G89  $Xx_1 Yy_1 Zz_1 Rr_1 Ff_1 Pp_1, li_1, Jj_1$  ;



Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	-
(5)	-	Invalid
(6)	-	Valid

The operation stops at after the (1), (2) and (5) or (6) commands during single block operation.

- (j) G73 (Step cycle)  
 Program  
 G73 Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> Qq<sub>1</sub> Rr<sub>1</sub> Ff<sub>1</sub> Pp<sub>1</sub> ,li<sub>1</sub> ,Jj<sub>1</sub>;  
 P : Dwell designation



Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	-
(5)	-	Invalid
(6)	-	Invalid

:  
:

(n) -1	-	Invalid
(n)	-	Valid

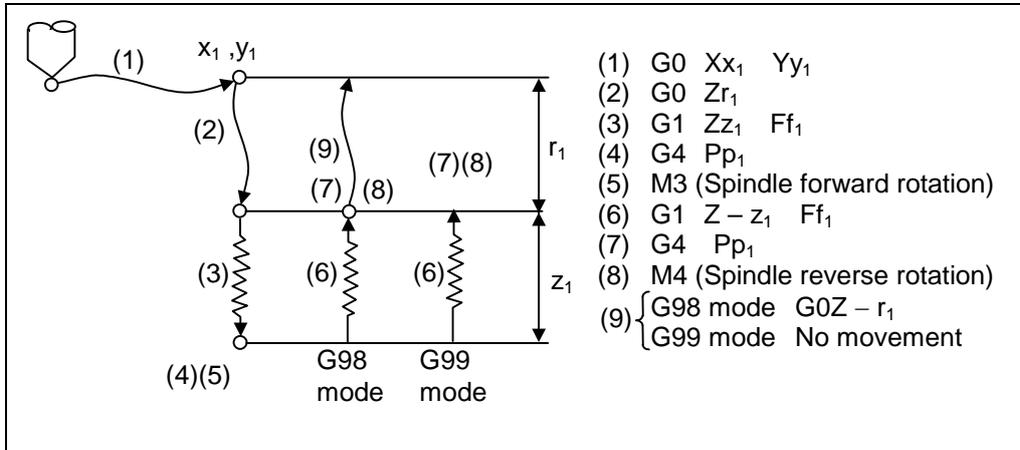
When executing a second and following cutting in the G73 as shown above, the movement will return several mm ("m" mm in the figure above) with rapid traverse and then will change to cutting feed.  
 The return amount m will differ according to the parameter "#8012 G73 return".  
 The operation stops at after the (1), (2) and (n) commands during single block operation.

(k) G74 (Reverse tapping cycle)

Program

G74  $Xx_1 Yy_1 Zz_1 Rr_1 Pp_1 Rr_2$ (or  $S_1, S_2$ ),  $li_1, Jj_1$ ;

P : Dwell designation



Operation pattern	i1	j1
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	-
(5)	-	-
(6)	-	Invalid
(7)	-	-
(8)	-	-
(9)	-	Valid

When  $r_2 = 1$ , the synchronous tapping mode will be entered, and when  $r_2 = 0$ , the asynchronous tapping mode will be entered.

When G74 is executed, the override will be canceled and the override will automatically be set to 100%. Dry run is valid when the control parameter "#1085 G00Drn" is set to "1" and is valid for the positioning command. If the feed hold button is pressed during G74 execution, and the sequence is at (3) to (6), the movement will not stop immediately, and instead will stop after (6). During the rapid traverse in sequence (1), (2) and (9), the movement will stop immediately.

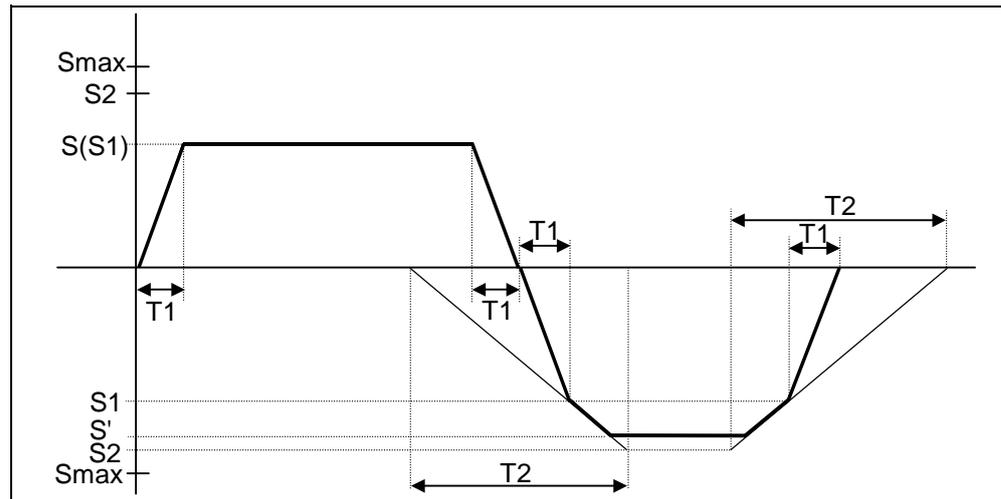
The operation stops at after the (1), (2) and (9) commands during single block operation.

During the G74 and G84 modal, the "Tapping" NC output signal will be output.

During the G74 synchronous tapping modal, the M3, M4, M5 and S code will not be output.

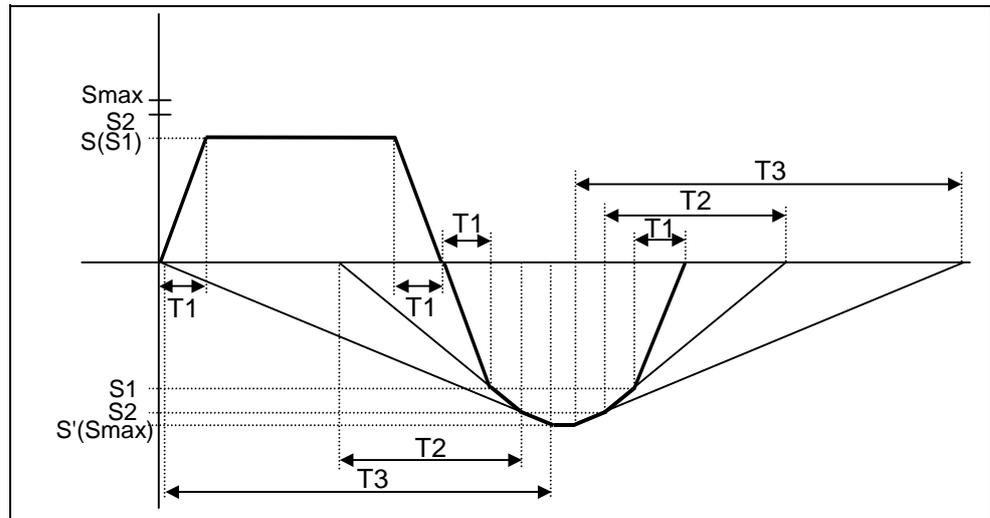
This function allows spindle acceleration/deceleration pattern to be approached to the speed loop acceleration/deceleration pattern by dividing the spindle and drilling axis acceleration/deceleration pattern into up to three stages during synchronous tapping. The acceleration/deceleration pattern can be set up to three stages for each gear. When returning from the hole bottom, rapid return is possible depending on the spindle rotation speed during return. The spindle rotation speed during return is held as modal information.

- (i) When tap rotation speed < spindle rotation speed during return  $\leq$  synchronous tap changeover spindle rotation speed 2



S	: Command spindle rotation speed
S'	: Spindle rotation speed during return
S1	: Tap rotation speed (spindle base specification parameters #3013 to #3016)
S2	: Synchronous tap changeover spindle rotation speed 2 (spindle base specification parameters #3037 to #3040)
Smax	: Maximum rotation speed (spindle base specification parameters #3005 to #3008)
T1	: Tap time constant (spindle base specification parameters #3017 to #3020)
T2	: Synchronous tap changeover time constant 2 (spindle base specification parameters #3041 to #3044)

(ii) When synchronous tap changeover spindle rotation speed 2 < spindle rotation speed during return



- S : Command spindle rotation speed
- S' : Spindle rotation speed during return
- S1 : Tap rotation speed (spindle base specification parameters #3013 to #3016)
- S2 : Synchronous tap changeover spindle rotation speed 2  
(spindle base specification parameters #3037 to #3040)
- Smax : Maximum rotation speed (spindle base specification parameters #3005 to #3008)
- T1 : Tap time constant (spindle base specification parameters #3017 to #3020)
- T2 : Synchronous tap changeover time constant 2  
(spindle base specification parameters #3041 to #3044)
- T3 : Synchronous tap changeover time constant 3  
(spindle base specification parameters #3045 to #3048)

(iii) Pecking tapping cycle

The load applied to the tool can be reduced by designating the depth of cut per pass (Q) and cutting the workpiece to the hole bottom for a multiple number of passes.

The amount retracted from the hole bottom is set to the parameter "#8018 G84/G74 return".

Select either the pecking tapping cycle or the deep-hole tapping cycle by parameter. ("#1272 ext08/bit4")

When "depth of cut per pass Q" is designated in the block containing the G84 or G74 command in the state where the pecking tapping cycle is selected, the pecking tapping cycle is executed.

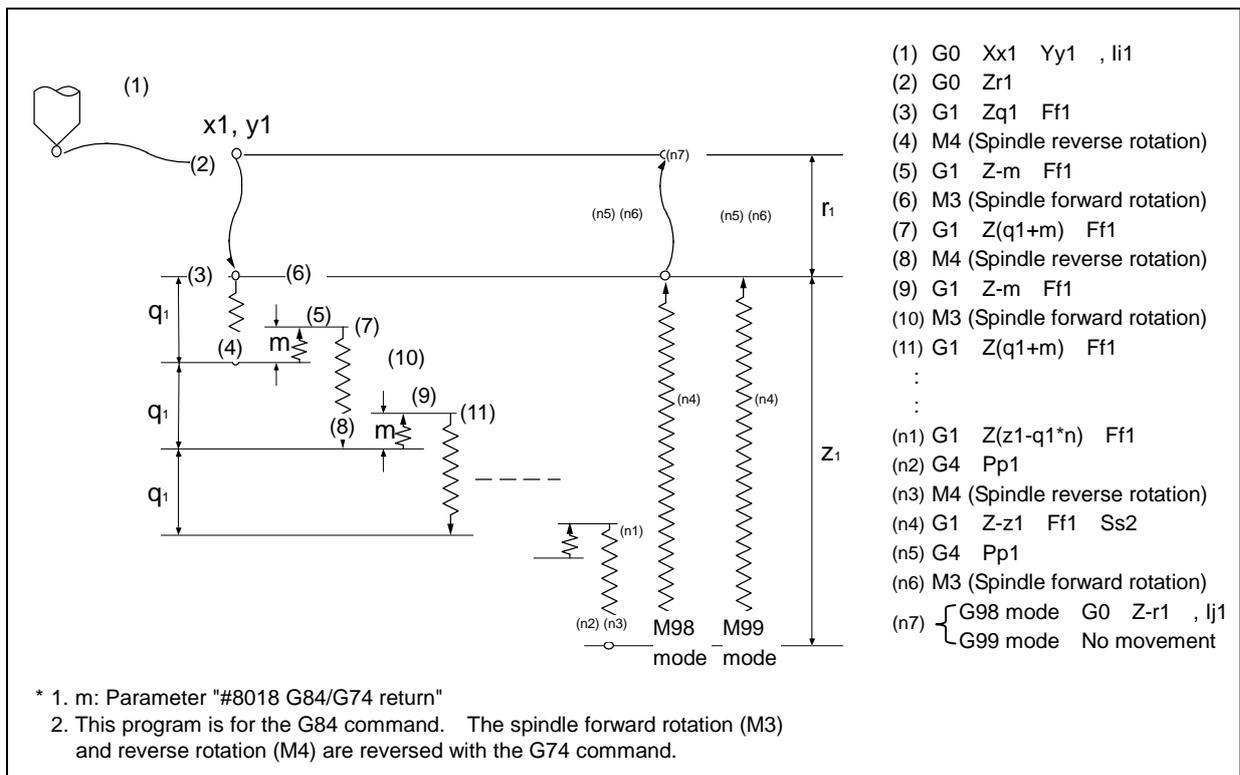
In the following cases, the normal tapping cycle is established.

- When Q is not designated
- When the command value of Q is zero

G84 Xx1 Yy1 Zz1 Rr1 Qq1 Ff1 Ee1 Pp1 Ss1 ,Ss2 ,li1 ,Jj1 ,Rr2 ;	
X, Yy	: Hole drilling position
Z	: Hole bottom position
R	: Point R position
Q	: Depth of cut per pass (designated as an incremental position)
F	: Z-axis feed amount (tapping pitch) per spindle rotation
E	: Tap thread number per 1-inch feed of Z axis
P	: Dwell time at hole bottom position
S	: Rotation speed of spindle
, S	: Rotation speed of spindle during retract
, l	: In-position width of positioning axis
, J	: In-position width of hole drilling axis
, R	: Synchronization method selection (r2=1 synchronous, r2=0 asynchronous)

(Note) When ",R0" is commanded, F address is regarded as cutting federate.

Refer to the section "Fixed cycle" for details.



(iv) Deep-hole tapping cycle

In the deep-hole tapping, the load applied to the tool can be reduced by designating the depth of cut per pass and cutting the workpiece to the hole bottom for a multiple number of passes.

Under the deep-hole tapping cycle, the tool is retracted to the R-point every time.

Select either the pecking tapping cycle or the deep-hole tapping cycle by parameter. ("#1272 ext08/bit4")

When "depth of cut per pass Q" is designated in the block containing the G84 or G74 command in the state where the deep-hole tapping cycle is selected by parameter, the deep-hole tapping cycle is executed.

In the following cases, the normal tapping cycle is established.

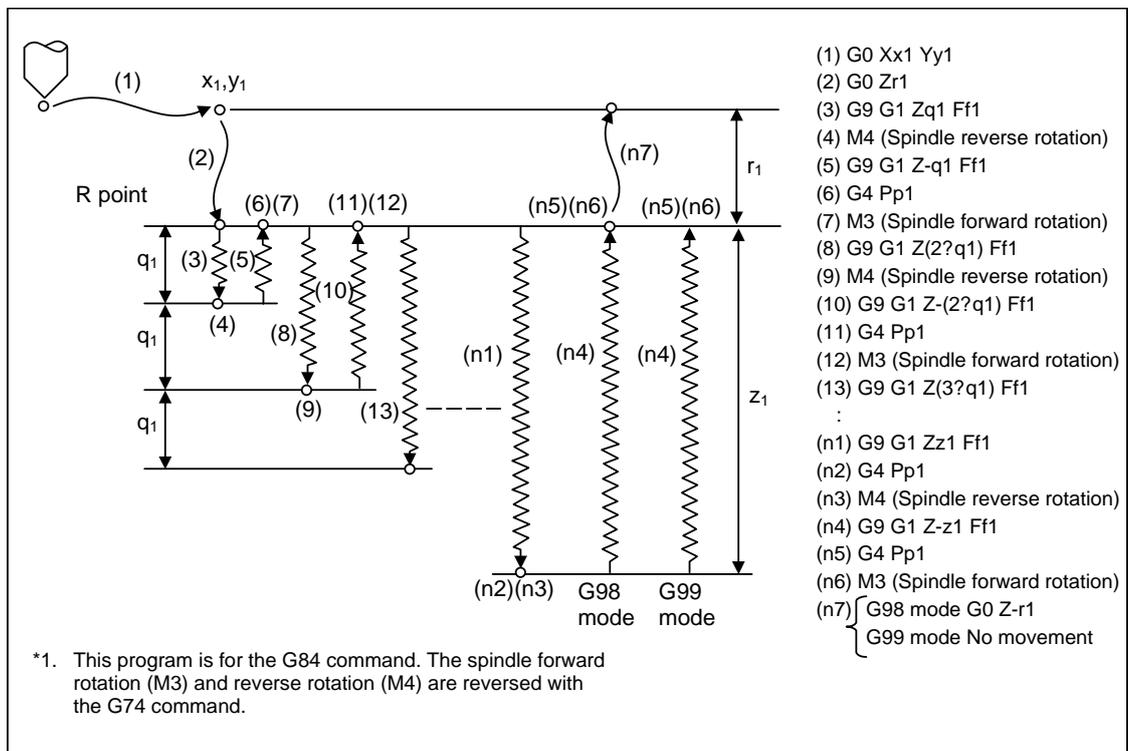
- When Q is not designated
- When the command value of Q is zero

**G84 Xx1 Yy1 Zz1 Rr1 Qq1 Ff1 Ee1 Pp1 Ss1 ,Ss2 ,li ,Jj ,Rr2 ;**

X,Y : Hole drilling position  
 Z : Hole bottom position  
 R : Point R position  
 Q : Depth of cut per pass (designated as an incremental position)  
 F : Z-axis feed amount (tapping pitch) per spindle rotation  
 E : Tap thread number per 1-inch feed of Z axis  
 P : Dwell time at hole bottom and point R return  
 S : Rotation speed of spindle  
 ,S : Rotation speed of spindle during retract  
 ,I : In-position width of positioning axis  
 ,J : In-position width of hole drilling axis  
 ,R : Synchronization method selection (r2=1 synchronous, r2=0 asynchronous)

**(Note)** When ",R0" is commanded, F address is regarded as cutting federate.

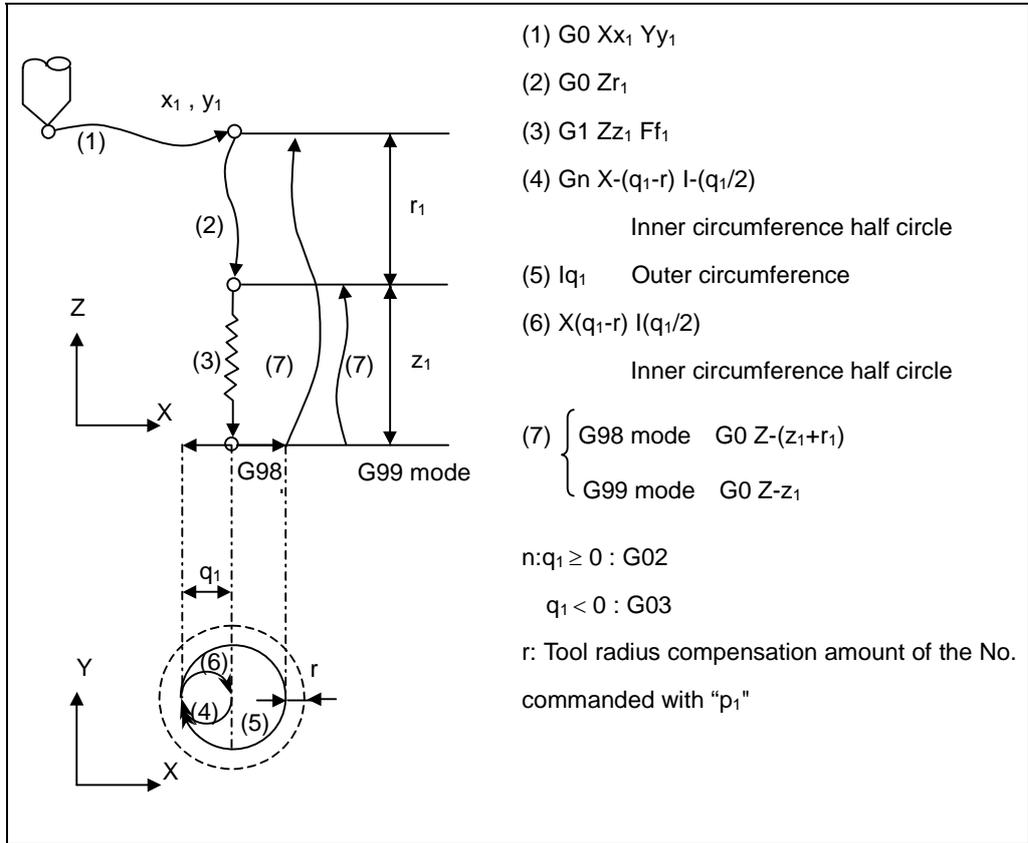
Refer to the section "Fixed cycle" for details.



- (l) G75 (Fine boring)
  - Circle cutting cycle performs a series of the cutting as follows:
  - First: positioning of X and Y axes to the circle center.
  - Next: cutting in with Z axis to the commanded position.
  - Then: moving the perfect round cutting the inside of the circle.
  - Until: returning to the circle center.

Program

G75 Xx1 Yy1 Zz1 Rr1 Qq1 Pp1 Ff1 ;



The operation stops at after the (1), (2) and (6) commands during single block operation.





## Precautions for using canned cycle

- (1) Before the canned cycle is commanded, the spindle must be rotating in a specific direction with an M command (M3 ; or M4 ;).  
Note that for the G87 (back boring) command, the spindle rotation command is included in the canned cycle so only the rotation speed command needs to be commanded beforehand.
- (2) If there is a basic axis, additional axis or R data in the block during the canned cycle mode, the hole drilling operation will be executed. If there is not data, the hole will not be drilled.  
Note that in the X axis data, if the data is a dwell (G04) time command, the hole will not be drilled.
- (3) Command the hole machining data (Q, P, I, J, K) in a block where hole drilling is executed. (Block containing a basic axis, additional axis or R data.)
- (4) The canned cycle can be canceled by the G00 to G03 or G33 command besides the G80 command. If these are designated in the same block as the canned cycle, the following will occur.

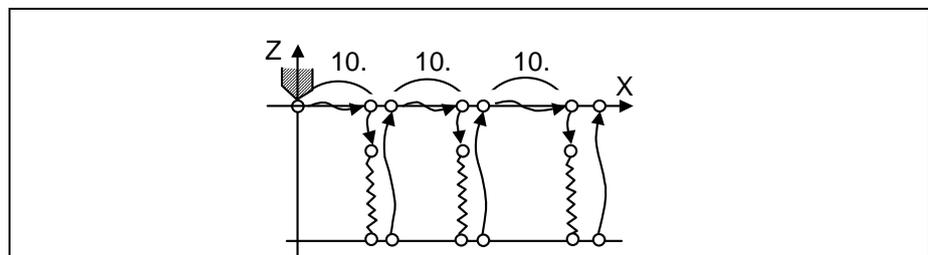
(Where, 00 to 03 and 33 are m, and the canned cycle code is n)

Gm	Gn	X__Y__Z__	R__Q__P__L__F__;
Execute	Ignore	Execute	Ignore Record
Gm	Gn	X__Y__Z__	R__Q__P__L__F__;
Ignore	Execute	Ignore	Record

Note that for the G02 and G03 commands, R will be handled as the arc radius.

- (5) If an M function is commanded in the same block as the canned cycle command, the M code and MF will be output during the initial positioning. The next operation will be moved to with FIN (finish signal).  
If there is a No. of times designated, the above control will be executed only on the first time.
- (6) If another control axis (ex., rotary axis, additional axis) is commanded in the same block as the canned cycle control axis, the canned cycle will be executed after the other control axis is moved first.
- (7) If the No. of repetitions L is not designated, L1 will be set. If L0 is designated in the same block as the canned cycle G code command, the hole machining data will be recorded, but the hole machining will not be executed.  
**(Example)** G73X\_\_Y\_\_Z\_\_R\_\_Q\_\_P\_\_F\_\_L0\_\_;
- (8) When the canned cycle is executed, only the modal command commanded in the canned cycle program will be valid in the canned cycle subprogram. The modal of the program that called out the canned cycle will not be affected.
- (9) Other subprograms cannot be called from the canned cycle subprogram.
- (10) Decimal points in the movement command will be ignored during the canned cycle subprogram.
- (11) If the No. of repetitions L is 2 or more during the incremental value mode, the positioning will also be incremented each time.

**(Example)** G91G81X10. Z-50.R-20.F100.L3 ;



- (12) If the spindle rotation speed value during return is smaller than the spindle speed, the spindle rotation speed value is valid even during return.
- (13) If the 2nd and 3rd acceleration/deceleration stage inclinations following the spindle rotation speed and time constants set in the parameters are each steeper than the previous stage's inclination, the previous stage's inclination will be valid.
- (14) If the values set in the spindle base specification parameter "stap1-4" (tap rotation speed) and "taps21-24" (synchronous tap changeover spindle rotation speed 2) exceed the maximum rotation speed, the spindle rotation speed will be clamped at the maximum rotation speed.
- (15) If the spindle rotation speed during return is not 0, the tap return override value will be invalid.
- (16) In a block where the movement direction of any axis reverses as shown below, the servo system load will greatly increase so do not command the in-position width in the machining program.  
G0 X100., I10.0 ;  
X-200. ;
- (17) If the in-position width commanded with the programmable in-position width command is increased, the positioning time and linear interpolation time can be reduced. However, the position error amount of the previous block will also increase before the next block starts, so the actual machining could be obstructed.
- (18) The in-position width and the position error amount are compared at a set time, so the position error amount at the point to be judged as in-position will be smaller than the commanded in-position width.
- (19) If the in-position width commanded with the programmable in-position command is small, the commanded deceleration check or in-position check following the parameters may be carried out first.
- (20) Synchronous and asynchronous tap can be selected with the M function.

Base specifications parameters

#	Items		Details	Setting range
1272 (PR)	ext08	bit1	M-function synchronous tap cycle valid.	0:Invalid 1:Valid

Synchronous tap cannot be selected with the M function when this parameter is OFF.

Base specifications parameters

#	Items	Details	Setting range
1513	stapM	M code for synchronous tap selection	0 to 99999999

The synchronous tap mode is selected with the miscellaneous function code set with this parameter.

The M function can be commanded in the same block before the tap command.

The synchronous and asynchronous tap will follow the combination shown below.

Program command (,R0/1)	Combination											
	0	0	0	0	1	1	1	1	No command			
#1229 (bit4) (Synchronous tap valid)	0	0	1	1	0	0	1	1	0	0	1	1
M function code (M**)	×	○	×	○	×	○	×	○	×	○	×	○
Synchronous/ asynchronous selection	A	A	A	A	S	S	S	S	A	S	S	S

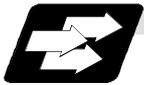
× : Does not command      A : Asynchronous tap  
○ : Commands                S : Synchronous tap

**(Note1)** Do not use M00, 01, 02, 30, 98 or 99.

**(Note2)** Depending on the model, selection with the M function may not be possible.

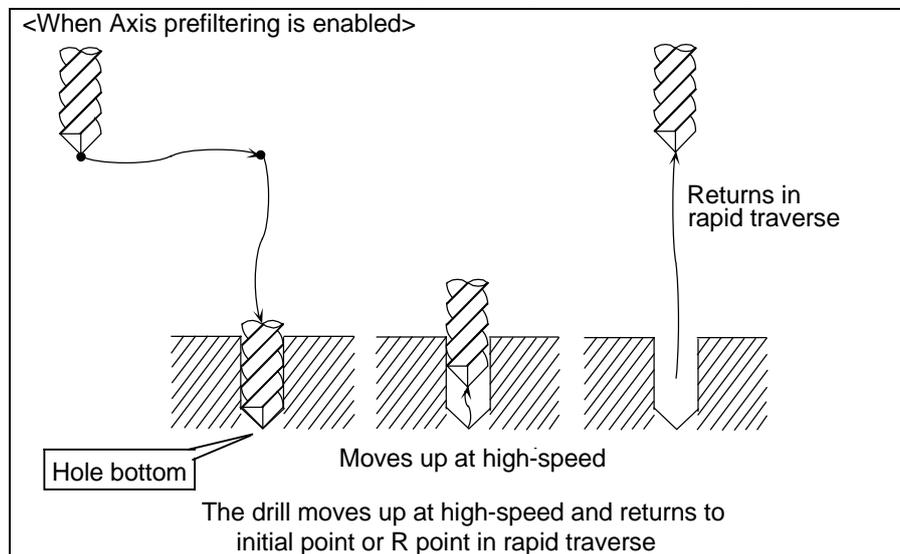
- (21) Even when the parameter "#1151 rstinit" is OFF, the fixed cycle will be canceled if NC reset 1 is carried out while executing the fixed cycle.

## 13.1.2 Drilling Cycle with High-Speed Retract



## Function And Purpose

This function retracts the drill from the hole bottom at high speed in drilling machining. This function helps extending the drill life by reducing the time of drilling in vain at hole bottom.



## Command Format

Same as the fixed cycle command format.



## Detailed Description

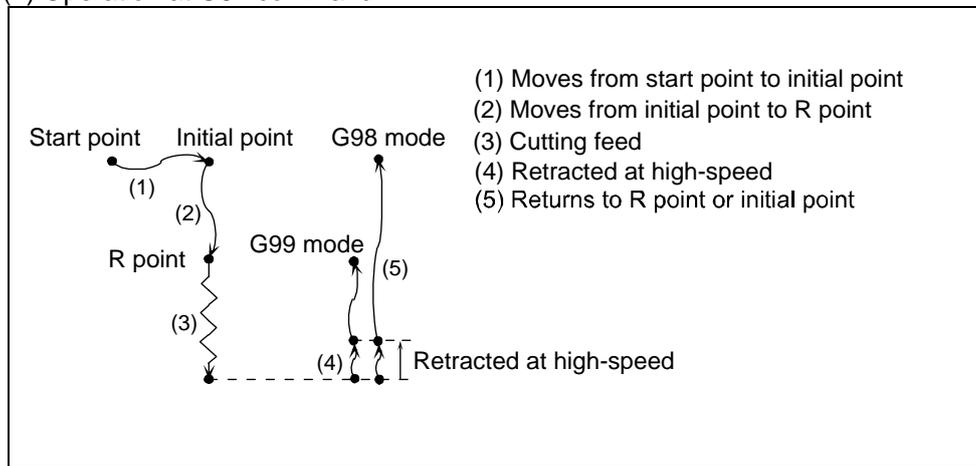
- (1) This function is available only when "#8123 H-spd retract ON" is enabled in the following fixed cycles.
  - G81 (Drill spot drilling cycle)
  - G83 (Deep whole drilling cycle)
  - G73 (Step cycle)
- (2) When "#8123 H-spd retract ON" is ON, the axis is retracted from the hole bottom at high speed with lost motion compensation.
  - (a) Set the lost motion compensation type 2 or 3 with servo parameter. Then set the following parameters to adjust the retract amount.
    - #2170 Lmc1QR (Lost motion compensation gain 1 for high-speed retract)  
(equivalent to "#2216 SV016 LMC1 Lost motion compensation 1")
    - #2171 Lmc2QR (Lost motion compensation gain 2 for high-speed retract)  
(equivalent to "#2241 SV041 LMC2 Lost motion compensation 2")
  - (b) Set the following parameters for lost motion compensation timing or lost motion compensation 3 spring constant/ viscous coefficient in addition to the ordinary lost motion compensations.
    - #2172 LmcdQR (Lost motion compensation timing for high-speed retract)  
(equivalent to "#2239 SV039 LMCD Lost motion compensation timing")
    - #2173 LmckQR (Lost motion compensation 3 spring constant for high-speed retract)  
(equivalent to "#2285 SV085 LMCK Lost motion compensation 3 spring constant")
    - #2174 LmccQR (Lost motion compensation 3 viscous coefficient for high-speed retract)  
(equivalent to "#2286 SV086 LMCc Lost motion compensation 3 viscous coefficient")

- (c) If the drilling axis is synchronously controlled, set the same value in both parameters for primary and secondary axes.
- (3) While G80 (Fixed cycle cancel) command is issued, this function is canceled by issuing any other fixed cycle of the same group (Group 9) or any Group 1 command.
- (4) This function is invalid during the following command modal.  
 In this case, the drill moves in the ordinary rapid traverse even if "#8123 H-spd retract ON" is enabled.  
 G43.1 (Tool length compensation in the tool axis direction) G43.4, G43.5 (Tool center point control) G68 (3-dimensional coordinate conversion)



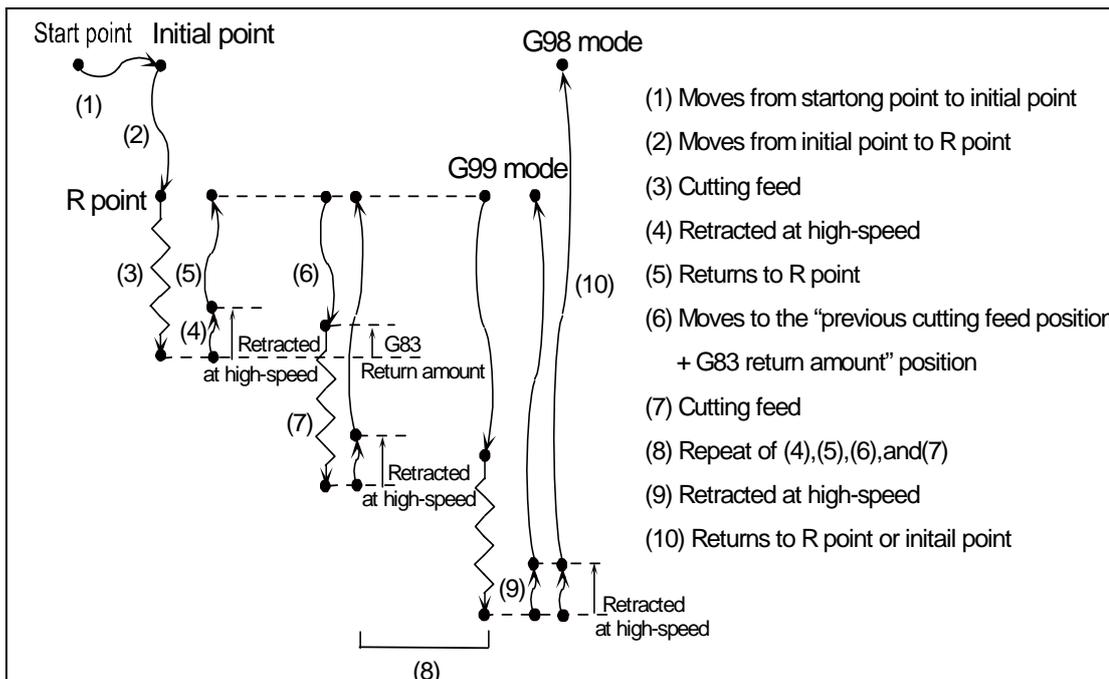
**Detailed Description**

(1) Operation at G81 command



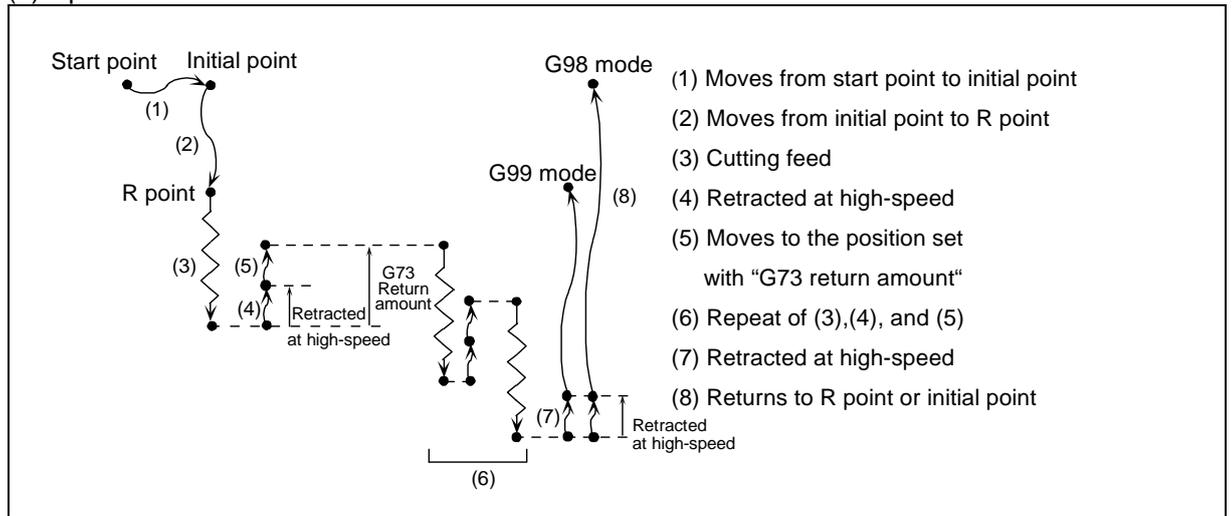
During single block operation, the axis stops after (1), (2) and (5) only.

(2) Operation at G83 command



During single block operation, the axis stops after (1), (2) and (10) only.

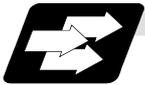
(3) Operation of G73 command



During single block operation, the axis stops after (1), (2) and (8) only.

If a dwell command is issued, the high-speed retract will be executed after the command.

13.1.3 Initial Point and R Point Level Return; G98, G99



Function and purpose

Whether to use R point or initial level for the return level in the final sequence of the canned cycle can be selected.



Command format

G98 ;  
 G99 ;  
 G98 ; Initial level return  
 G99 ; R point level return



Detailed description

The relation of the G98/G99 mode and No. of repetition designation is as shown below.

No. of hole drilling	Program example	G98 At power ON, at cancel with M02, M30, and reset button	G99
Only one execution	G81X100. Y100. Z-50. R25. F1000;	<p>Initial level return is executed.</p>	<p>R point level return is executed.</p>
Second and following executions	G81X100. Y100. Z-50. R25. L5F1000;	<p>Initial level return is executed for all times.</p>	<p>Final time (Note)</p>

**(Note)** The final return position will be different between label O and L when drilling is executed more than twice with G99 command.  
 Label O... R point return  
 Label L... Initial point return  
 With label L, R point return can be selected by setting "#1246 set18/bit3 Change final return position at M2L"



Example of program

(Example 1)

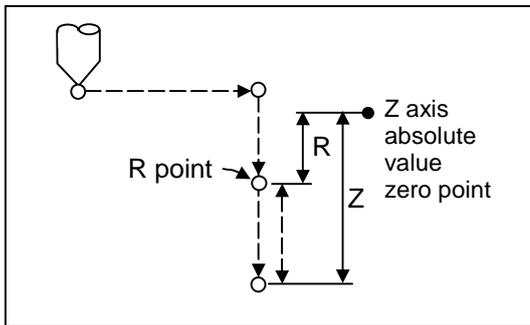
G82 Zz <sub>1</sub> Rr <sub>1</sub> Pp <sub>1</sub> Ff <sub>1</sub> L0 ;	Record only the hold machining data (Do not execute)
Xx <sub>1</sub> Yy <sub>1</sub> ;	Execute hole drilling operation with G82 mode

The No. of canned cycle repetitions is designated with L. If L1 is designated or L not designated, the canned cycle will be executed once. The setting range is 1 to 9999.

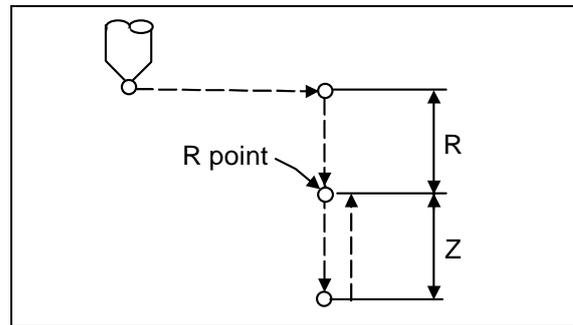
If L0 is commanded, only the hole machining data will be recorded.

G8Δ (7Δ) Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> Rr<sub>1</sub> Pp<sub>1</sub> Qq<sub>1</sub> Ff<sub>1</sub> Ll<sub>1</sub> ;

The ideology of the data differs between the absolute value mode (G90) and incremental value mode (G91) as shown below.



Absolute value mode (G90)



Incremental value mode (G91)

Designate a command value with a symbol for X, Y and Z. R indicates the coordinate value from the zero point in the absolute value mode, so a symbol must always be added. However, in the incremental value the symbol will be ignored and will be viewed as the same symbol as for Z. Note that the symbols will be viewed in reverse for G87.

The hole machining data is held as shown below in the canned cycle. The hole machining data is canceled when the G80 command or G commands (G00, G01, G02, G03, G2.1, G3.1, G33) in the 01 group are reached.

(Example 2)

N001 G81 Xx <sub>1</sub> Yy <sub>1</sub> Zz <sub>1</sub> Rr <sub>1</sub> Ff <sub>1</sub> ;	
N002 G81 ;	Only selection of canned cycle sequence
N003 Xx <sub>2</sub> Yy <sub>2</sub> ;	Change of positioning point, and execution of canned cycle
N004 M22 ;	Execution of only M22
N005 G04 Xx <sub>3</sub> ;	Execution of only dwell
N006 G92 Xx <sub>4</sub> Yy <sub>4</sub> ;	Execution of only coordinate system setting
N007 G28 (G30) Z0 ;	Execution of only reference position (zero point) return
N008 ;	No work
N009 G99 Zz <sub>2</sub> Rr <sub>2</sub> Ff <sub>2</sub> L0 ;	Execution of only hole machining data recording
N010 Xx <sub>5</sub> Yy <sub>5</sub> Ll <sub>5</sub> ;	Change of positioning point, and execution of R point return canned cycle for l <sub>5</sub> times
N011 G98 Xx <sub>6</sub> Yy <sub>6</sub> Zz <sub>6</sub> Rr <sub>6</sub> ;	Change of positioning point, and execution of canned cycle
N012 Ww <sub>1</sub> ;	Execute W axis according to 01 group modal before N001, and then execute canned cycle

### 13.1.4 Setting of Workpiece Coordinates in Fixed Cycle Mode

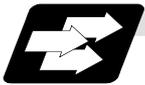
The designated axis moves with the workpiece coordinate system set for the axis.  
The Z axis is valid after the R point positioning after positioning or from Z axis movement.

**(Note)** When the workpiece coordinates are changed over for address Z and R, re-program even if the values are the same.

**(Example)**

G54 Xx <sub>1</sub> Yy <sub>1</sub> Zz <sub>1</sub> ;	
G81 Xx <sub>2</sub> Yy <sub>2</sub> Zz <sub>2</sub> Rr <sub>2</sub> ;	
G55 Xx <sub>3</sub> Yy <sub>3</sub> Zz <sub>2</sub> Rr <sub>2</sub> ;	Re-command even if Z and R are the same as the previous value.
Xx <sub>4</sub> Yy <sub>4</sub> ;	
Xx <sub>5</sub> Yy <sub>5</sub> ;	

### 13.2 Special Fixed Cycle; G34, G35, G36, G37



#### Function and purpose

The special fixed cycle is used with the standard fixed cycle. Before using the special fixed cycle, program the fixed cycle sequence selection G code and hole machining data to record the hole machining data. (If there is no positioning data, the fixed cycle will not be executed, and only the data will be recorded.)

The axis is positioned to the hole machining position when the special fixed cycle is executed. The hole machining operation is executed with the canned cycle for hole machining.

Even after the special fixed cycle is executed, the recorded standard fixed cycle will be held until canceled.

If the special fixed cycle is designated when not in the fixed cycle mode, only positioning will be executed, and the hole drilling operation will not be done.

If the special fixed cycle is commanded without commanding the fixed cycle for hole machining, positioning will be executed following the current O1 group modal G code.



## Bolt hole circle (G34)

**G34 X x<sub>1</sub> Y y<sub>1</sub> I r J θ K n ;**

X, Y :Positioning of bolt hole cycle center. This will be affected by G90/G91.

I :Radius r of the circle. The unit follows the input setting unit, and is given with a positive number.

J :Angle θ of the point to be drilled first. The CCW direction is positive.  
(The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)

K :No. of holes n to be drilled. 1 to 9999 can be designated, but 0 cannot be designated. When the value is positive, positioning will take place in the CCW direction, and when negative, will take place in the CW direction.  
If "0" is designated, the program error (P221) will occur.

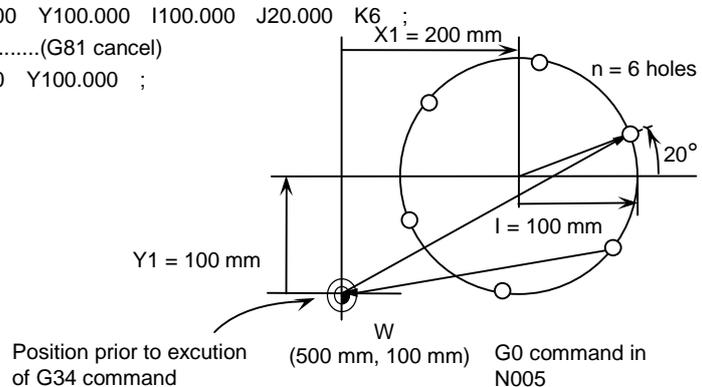
Drilling of n obtained by dividing the circumference by n will start at point created by the Z axis and angle θ. The circumference is that of the radius R centering on the coordinates designated with XX and Y. The hole drilling operation at each hole will hold the drilling data for the standard canned cycle such as G81.

The movement between hole positions will all be done in the G00 mode. G34 will not hold the data even when the command is completed.

**(Example)****(Example)**

With 0.001mm least command increment

```
N001 G91 ;
N002 G81 Z -10.000 R5.000 L0 F200 ;
N003 G90 G34 X200.000 Y100.000 I100.000 J20.000 K6 ;
N004 G80 ; .....(G81 cancel)
N005 G90 G0 X500.000 Y100.000 ;
```



As shown in the example, the tool position after the G34 command is completed is over the final hole. When moving to the next position, the coordinate value must be calculated to issue the command with an incremental value. Thus, use of the absolute value mode is handy.

**(Note 1)** If an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G34 command, a program error (P32) will occur.



## Line at angle (G35)

**G35 X x1 Y y1 I d J  $\theta$  K n ;**

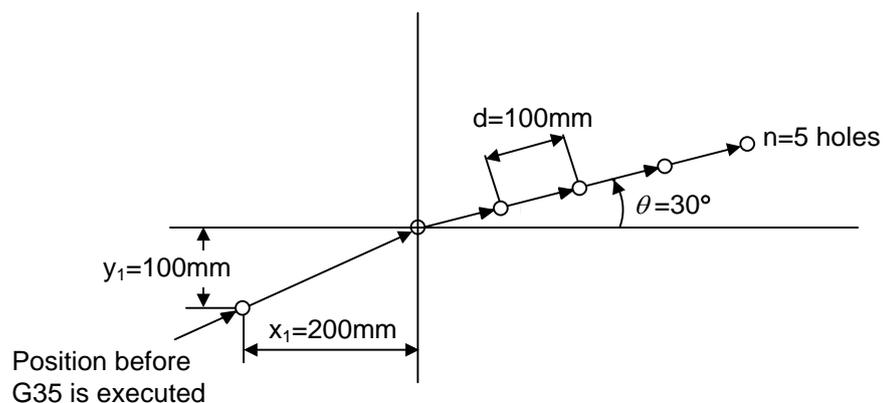
- X, Y :Designation of start point coordinates. This will be affected by G90/G91.  
 I :Interval d. The unit follows the input setting unit. If d is negative, the drilling will take place in the direction symmetrical to the point that is the center of the start point.  
 J :Angle  $\theta$ . The CCW direction is positive.  
 (The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)  
 K :No. of holes n to be drilled. 1 to 9999 can be designated, and the start point is included.

Using the position designated by X and Y as the start point, the Zn holes will be drilled with interval d in the direction created by X axis and angle  $\theta$ . The hole drilling operation at each hole position will be determined by the standard canned cycle, so the hole drilling data (hole machining mode and hole machining data) must be held beforehand. The movement between hole positions will all be done in the G00 mode. G35 will not hold the data even when the command is completed.

**(Example)**

When input setting unit is 0.001mm

```
G91 ;
G81 Z - 10000 R5000 L0 F100 ;
G35 X200000 Y100000 I100000 J30000 K5 ;
```



**(Note 1)** If the K command is K0 or if there is no K command, the program error (P221) will occur.

**(Note 2)** If the K value is more than four digits, the last four digits will be valid.

**(Note 3)** If an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G35 command, a program error (P32) will occur.

**(Note 4)** If a group 0 G command is issued in the same block as the G35 command, the command issued later is the priority.

**(Example)** G35 G28 Xx<sub>1</sub> Yy<sub>1</sub> Ii<sub>1</sub> Jj<sub>1</sub> Kk<sub>1</sub> ;

G35 is ignored G 28 is executed as Xx<sub>1</sub> Yy<sub>1</sub>

**(Note 5)** If there is a G72 to G89 command in the same block as the G35 command, the canned cycle will be ignored, and the G35 command will be executed.



## Arc (G36)

**G36 X x1 Y y1 I r J  $\theta$  P  $\Delta\theta$  K n ;**

X, Y :Center coordinates of arc. This will be affected by G90/G91.

I :Radius r of arc. The unit follows the input setting unit, and is given with a positive No.

J :Angle  $\theta$  of the point to be drilled first. The CCW direction is positive. (The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)

P :Angle interval  $\Delta\theta$ . When the value is positive, the drilling will take place in the CCW direction, and in the CW direction when negative. (The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)

K :No. of holes n to be drilled. 1 to 9999 can be designated.

The n holes aligned at the angle interval  $\Delta\theta$  will be drilled starting at point created by the X axis and angle  $\theta$ . The circumference is that of the radius R centering on the coordinates designated with XX and Y. As with the bolt hole circle, the hole drilling operation at each hole will depend on the standard canned cycle.

The movement between hole positions will all be done in the G00 mode. G36 will not hold the data even when the command is completed.

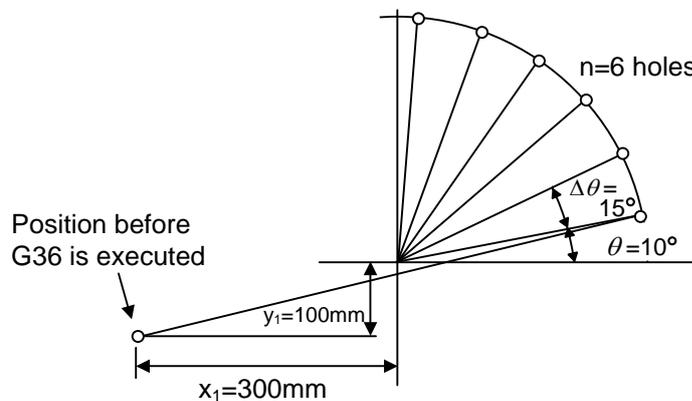
**(Example)**

When input setting unit is 0.001mm

N001 G91 ;

N002 G81 Z - 10000 R5000 F100 ;

N003 G36 X300000 Y100000 I300000 J10000 P15000 K6 ;



**(Note 1)** If an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G36 command, a program error (P32) will occur.



## Grid (G37)

**G37 X x1 Y y1 I Dx P nx J Dy K ny ;**

X, Y :Designation of start point coordinates. This will be affected by G90/G91.  
 I :Interval Dx of the X axis. The unit will follow the input setting unit. If Dx is positive, the interval will be in the forward direction looking from the start point, and when negative, will be in the reverse direction looking from the start point.  
 P :No. of holes nx in the X axis direction. The setting range is 1 to 9999.  
 J :Interval Dy of the Y axis. The unit will follow the input setting unit. If Dy is positive, the interval will be in the forward direction looking from the start point, and when negative, will be in the reverse direction looking from the start point.  
 K :No. of holes ny in the Y axis direction. The setting range is 1 to 9999.

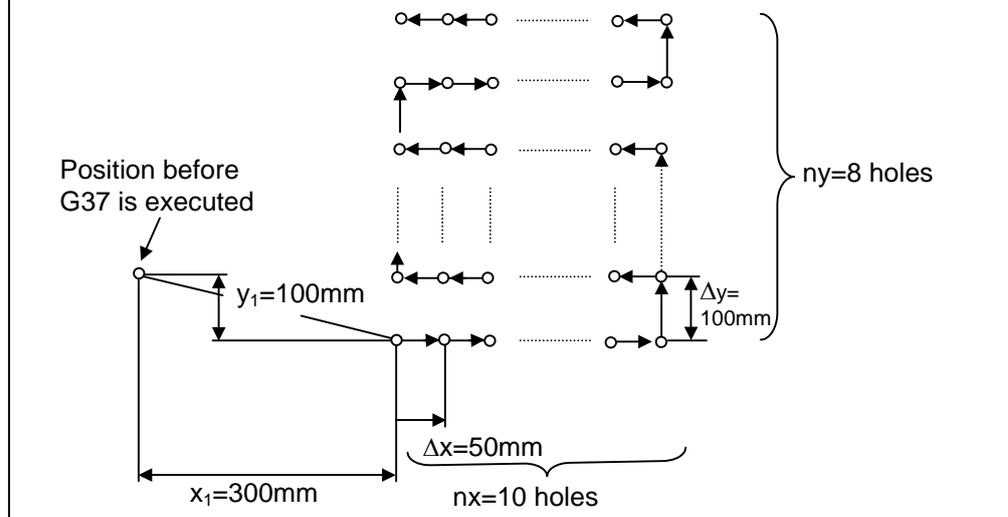
The nx points on a grid are drilled with an interval  $\Delta x$  parallel to the X axis, starting at the position designated with X, Y. The drilling operation at each hole position will depend on the standard canned cycle, so the hole machining data (hole machining mode and hole machining data) must be held beforehand.

The movement between hole positions will all be done in the G00 mode. G37 will not hold the data even when the command is completed.

**(Example)**

When input setting unit is 0.01mm

```
G91 ;
G81 Z-10000 R5000 F20 ;
G37 X300000 Y-100000 I50000 P10 J100000 K8 ;
```



**(Note 1)** If the P and K commands are P0 or K0, or if there is no P or K command, the program error (P221) will occur.

If the P or K value is more than four digits, the last four digits will be valid.

**(Note 2)** If an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G37 command, a program error (P32) will occur.

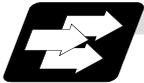
**(Note 3)** If a group 0 G command is issued in the same block as the G37 command, the command issued later is the priority.

**(Note 4)** If there is a G72 to G89 command in the same block as the G37 command, the canned cycle will be ignored, and the G37 command will be executed.

**(Note 5)** If the G22.1/G23.1 command is programmed in the same block as the G37 command, the G22.1/G23.1 command will be ignored, and the G37 command will be executed.

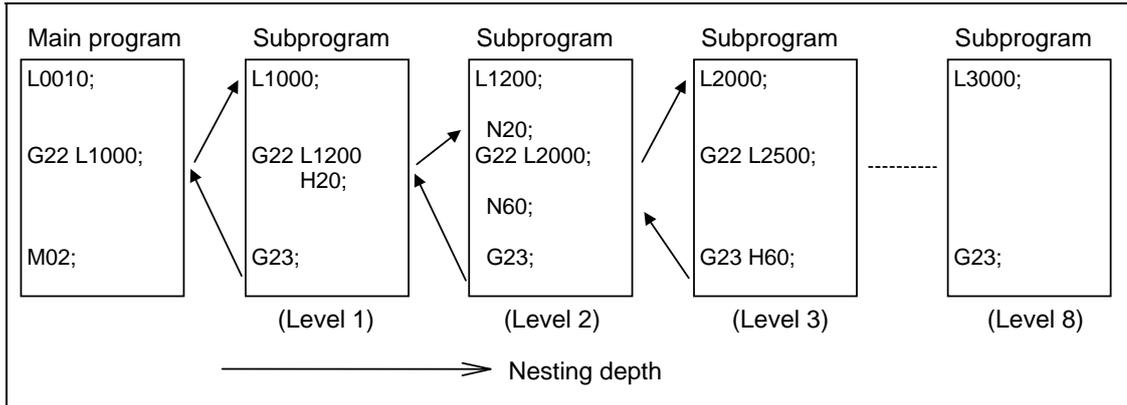
13.3 Subprogram Control; G22, G23

13.3.1 Calling Subprogram with G22 and G23 Commands



Function and purpose

Fixed sequences or repeatedly used parameters can be stored in the memory as subprograms which can then be called from the main program when required. G22 serves to call subprograms and G23 serves to return operation from the subprogram to the main program. Furthermore, it is possible to call other subprograms from particular subprograms and the nesting depth can include as many as 8 levels.



The table below shows the functions which can be executed by adding and combining the tape storing and editing functions, subprogram control functions and fixed cycle functions.

	Case 1	Case 2	Case 3	Case 4
1. Tape storing and editing	Yes	Yes	Yes	Yes
2. Subprogram control	No	Yes	Yes	No
3. Fixed cycles	No	No	Yes	Yes
<b>Function</b>				
1. Memory mode	○	○	○	○
2. Tape editing (main memory)	○	○	○	○
3. Subprogram call	×	○	○	×
4. Subprogram variable designation (Note 2)	×	○	○	×
5. Subprogram nesting level call (Note 3)	×	○	○	×
6. Fixed cycles	×	×	○	○
7. Subprogram editing for fixed cycle	×	×	○	○

(Note 1) "○" denotes a function which can be used and "×" a function which cannot be used.

(Note 2) Variables cannot be transferred with the G22 command but variable commands in subprograms can be used provided that the variable command option is available.

(Note 3) A maximum of 8 nesting levels form the nesting depth.



**Command format**

**Subprogram call**

**G22 L\_\_ H\_\_ P\_\_ ,D\_\_ ;**  
**or G22 <File name> H\_\_ P\_\_ ,D\_\_ ;**

G22 Subprogram call command

L Program No. of subprogram to be called (own program if omitted)

A (label O) L(A) address can be omitted only during memory mode and MDI mode.  
 (Max. 8 digits)

<File name> File name  
 A file name can be specified instead of a program No.  
 In this case, enclose the file name with brackets <>.  
 (The file name can have up to 32 characters including the extension.)  
 (Example) G22 <parts-12. RAF>

H Sequence No. in subprogram to be called (head block if omitted)  
 (Max. 5 digits)

L Number of subprogram repetitions  
 (When omitted, this is interpreted at P1, and is not executed when P0.)  
 (Between 1 and 9999 times according to 4-digit value.)  
 For instance, G22 L1 P3 ; is equivalent to the following:  
 G22 L1 ;  
 G22 L1 ;  
 G22 L1 ;

,D Subprogram device No. (0 to 4).  
 The subprogram in the memory can be used when ,D is omitted.  
 The device No. is set with the machining parameters.

**Return to main program from subprogram**

**G22 H\_\_ ;**

G22 Subprogram return command

H Sequence No. of return destination (returned to block that follows the calling block)



**Creating and entering subprograms**

Subprograms have the same format as machining programs for normal memory mode except that the subprogram completion instruction G22 (L\_ P\_) ; is entered as an independent block at the last block.

<pre> L##### ; ..... ; ..... ;       :       :       : ..... ; G23 ; % (EOR)                 </pre>	<p>) Program No. as subprogram No.</p> <p>Main body of subprogram</p> <p>) Subprogram return command Entry completion code</p>
---	--

(1) The above program is entered by editing operations at the setting and display unit. For further details, refer to the section on program editing in the Instruction Manual.

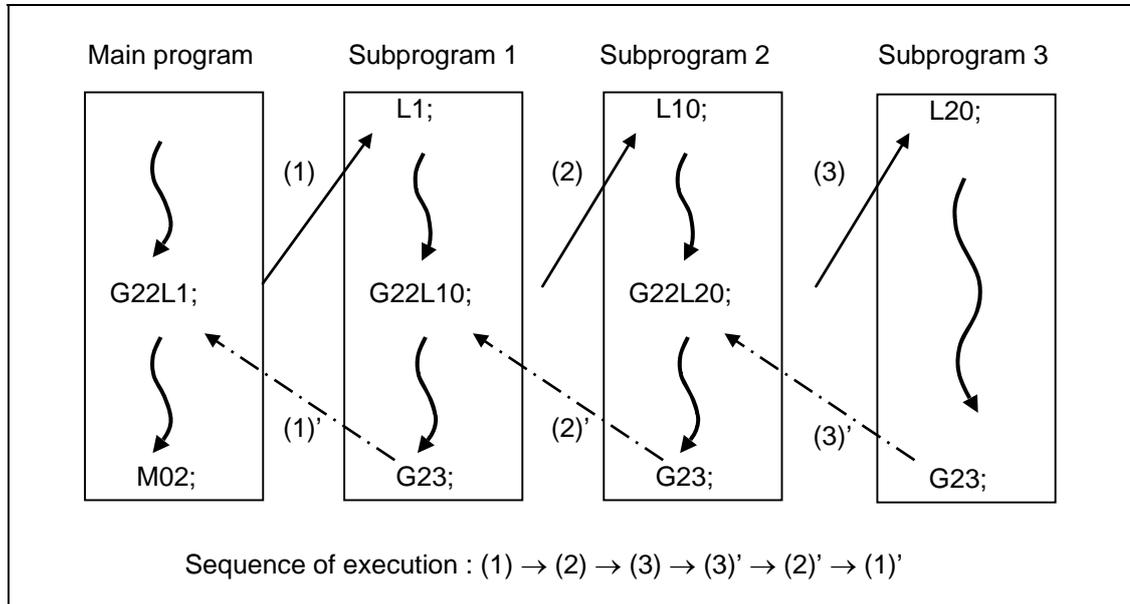
- (2) Only those subprograms Nos. ranging from 1 to 99999999 designated by the optional specifications can be used. When there are no program Nos. on the tape, they are entered as the setting No. for "program input."
- (3) Up to 8 nesting levels can be used for calling programs from subprograms, and program error (P230) results if this number is exceeded.
- (4) No distinction between main programs and subprograms is made since they are entered in the sequence in which they were read. This means that main programs and subprograms should not be given the same Nos. (If they are, error "E11" appears during entry.)
- (5) Besides the G22 command, subprogram nesting is subject to the following commands:
- G65 : Macro call
  - G66 : Modal call
  - G66.1 : Modal call
  - G code call
  - Miscellaneous function call
  - MDI interruption
  - Automatic tool length measurement
  - Macro interruption
  - Multiple-step skip function
- (6) Subprogram nesting is not subject to the following commands which can be called even beyond the 8th nesting level.
- Fixed cycles
  - Pattern cycles
- (7) To repeatedly use the subprogram, it can be repeated p1 times by programming G22 Ll<sub>1</sub> Pp<sub>1</sub>;
- (8) When using the multi-part system, if the subprogram attributed to the part system with the call command is empty, the subprogram call operation will change according to the parameters.

#1050 MemPrg	#1285 ext21/bit1	Details
0, 2, 4, 6	-	The subprogram registered in the memory common for the part systems is called out.
1, 3, 5, 7	OFF	The subprogram registered in the memory for the selected part system is called out.
	ON	The subprogram registered in the memory for the selected part system is called out. If the subprogram in the selected part system is empty, the subprogram with the same No. in the 1st part system is called out.



## Example of program 1

When there are 3 subprogram calls (known as 3 nesting levels)

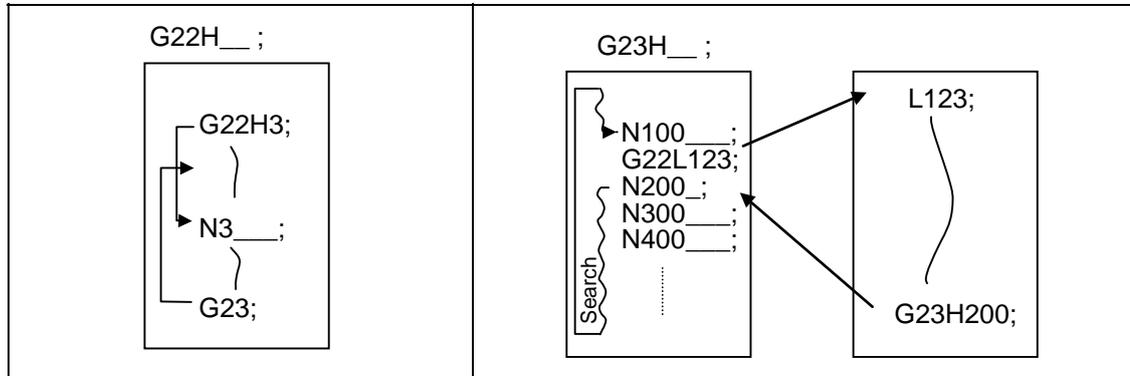


- (1) For nesting, the G22 and G23 commands should always be paired off on a 1:1 basis (1)' for (1), (2)' for (2), etc.
- (2) Modal information can be rewritten according to the execution sequence without distinction between main programs and subprograms. This means that after calling a subprogram, attention must be paid to the modal data status when programming.



### Example of program 2

The G22 H\_ ; G23 H\_ ; commands designate the sequence Nos. in a program with a call instruction.



### Precautions

- (1) Program error (P232) results when the designated L (program No.) is not located.
- (2) Single block stop does not occur with the G22 L \_ ; G23 ; block. If any address except L(O), N, P, L or H is used, single block stop can be executed. (With X100. G22 L100 ;, operation branches to L100 after X100. is executed.)
- (3) When G23 is commanded by the main program, operation returns to the head. (This is same for MDI.)
- (4) Branching from tape and BTR operation to the subprogram with G22 L \_ is possible, but the return destination sequence No. cannot be designated with G23H\_ ;. (H\_ is ignored).
- (5) Bear in mind that the search operation will take time when the sequence No. is designated by G23H\_ ;.
- (6) When using a file name for the subprogram, specify the file name with 32 characters or less, including the extension. If a file name exceeding 32 characters is specified, a program error (P232) will occur.
- (7) All the programs are registered as files. For example, when calling the file "0100" as a subprogram, "0100" cannot be searched with G22L100 or G22L0100. When a value is specified following L, reading 0 is omitted; therefore, it is assumed that the program No. (file) "100" was specified in this case. When wishing the program like "0100" to be called, specify the file name using the G22<0100> format.

### 13.3.2 Figure Rotation; G22 I\_ J\_ K\_



#### Function and purpose

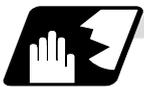
If the same pattern is used repeatedly on a concentric circle, one of the rotating machining patterns can be registered as a subprogram. When the subprogram is called from the main program, if the rotation center is designated, a path similar to the rotary phase can be easily created on the concentric circle. This simplifies creation of the program.



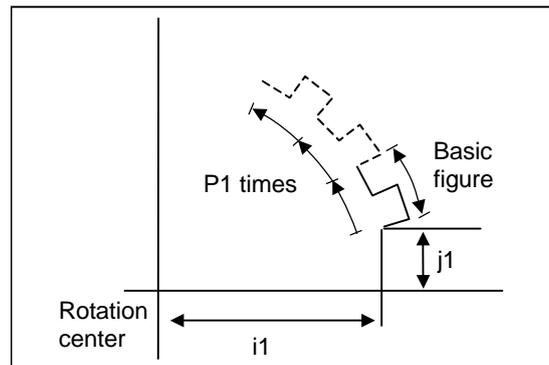
## Command format

**G22 I\_\_ J\_\_ K\_\_ L\_\_ H\_\_ P\_\_ ,D\_\_ ;**  
**or, G22 I\_\_ J\_\_ K\_\_ <File name> H\_\_ P\_\_ ,D\_\_ ;**

**G22** : Subprogram call command  
**I, J, K** : Rotation center  
**L** : Program No. in subprogram to be called. (Own program if omitted.)  
 Note that L can be omitted only during memory operation and MDI operation.  
 (Max. 8-digit value)  
**<File name>** : A file name can be designated instead of the program No. In this case, enclosed the file name with brackets <>.  
 (The file name can have up to 32 characters, including the extension.)  
 (Example) G22 <PARTS-12.RAF>;  
**H** : Sequence No. in subprogram to be called (Head block if omitted)  
 (Max. 5-digit value)  
**P** : Number of subprogram repetitions  
 (If omitted, this is handled as P1. When P0 is set, the subprogram is not called.)  
 (1 to 9999 times set with 4-digit value.)  
**,D** : Subprogram device No. (0 to 4).  
 The subprogram in the memory can be used when ,D is omitted.  
 The device No. is set with the machining parameters.



## Detailed description



- (1) The first subprogram called out with subprogram call is executed at a 0° rotation angle. The path is created as commanded.
- (2) If the number of repetitions is set to two or more, the rotation angle is obtained from the called subprogram's start point, end point and rotation center coordinate. The path of the first subprogram is used as a basic figure and is rotated and arranged for the designated number of call repetitions, using the rotation center coordinates as a reference.
- (3) All blocks in the subprogram are rotated.

- (4) If the subprogram start point and end point are not on the same circle having the commanded figure rotation center coordinates as the center, the axis will interpolate using the subprogram's end point as the start point, and the end point in the first movement command block in the rotated subprogram as the end point.
- (5) Both absolute values and incremental values can be used in the figure rotation subprogram. Even if commanded with an absolute value command, the rotation will be the same as when commanded with an incremental value.
- (6) I, J, and K are commanded with the incremental amount from the start point.
- (7) A subprogram of which figure is rotating cannot be branched to the other subprogram.
- (8) The figure is rotated on the workpiece coordinate system, and can be shifted with the G92, G52, G54 to G59 (workpiece coordinate system shift) command.
- (9) Functions (reference position return, unidirectional positioning, etc.) on the machine coordinate system for the rotary plane axis cannot be used while the figure is rotated. However, the machine coordinate system functions can be used for axes other than the rotation plane.
- (10) Refer to "13.3.1 Calling subprogram with G22 and G23 commands" for <file name>.



### Precautions

- (1) A program error will occur if figure rotation is commanded during figure rotation.
- (2) Figure rotation and program coordinate rotation cannot be commanded simultaneously.



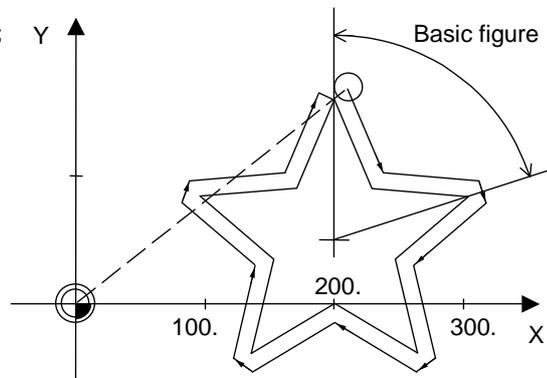
### Example of program

```

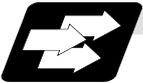
Main program (L1000)
N01 G90 G54 G00 X0 Y0 ;
N02 G01 G41 X200.Y150.D01 F500 ;
N03 G01 Z-50.F300 ;
N04 G22 L2200 P5 J-100. ;
N05 G90 G01 Z50.F500 ;
N06 G40 ;
N07 G00 X0 Y0 ;

Subprogram (L2200)
N01 G91 G01 X29.389 Y-59.549 ;
N02 X65.717 Y-9.549 ;
N03 G23 ;

```



13.4 Variable Commands



Function and purpose

Programming can be endowed with flexibility and general-purpose capabilities by designating variables, instead of giving direct numerical values to particular addresses in a program, and by assigning the values of those variables as required when executing a program.



Command format

# $\Delta\Delta\Delta$  =  $\circ\circ\circ\circ\circ\circ\circ\circ$  or # $\Delta\Delta\Delta$  = [formula]



Detailed description

(1) Variable expressions

		Example
(a) #m	m = value consisting of 0 to 9	#100
(b) # [f]	f = one of the following in the formula	# [-#120]
	Numerical value m	123
	Variable	#543
	Formula operator formula	#110+#119
	- (minus) formula	-#120
	[Formula]	[#119]
	function [formula]	SIN [#110]

**(Note 1)** The 4 standard operators are +, -, \* and /.

**(Note 2)** Functions cannot be used unless the user macro specifications are available.

**(Note 3)** Error "P241" results when a variable number is negative.

**(Note 4)** Examples of incorrect variable expressions are given below.

Incorrect	→	Correct
#6/2	→	#[6/2] (Note that expression such as "#6/2" is regarded as "[#6]/2")
#- -5	→	#[- [-5]]
#- [#1]	→	#[-#1]

**(2) Type of variables**

The following table gives the types of variables.

Type of variable		No.		Function
Common variables		Common variables 1	Common variables 2	<ul style="list-style-type: none"> <li>• Can be used in common throughout main, sub and macro programs.</li> <li>• When using common variables in the multi-part system, the number of common variable shared between the part systems can be specified by the parameter "#1052 MemVal".</li> </ul> <b>(Note 5)</b>
1st part system	100 sets	500 to 549	100 to 149	
	200 sets	500 to 599	100 to 199	
	300 sets	500 to 699	100 to 199	
	600 sets	500 to 999 100100 to 800199 <b>(Note 7)</b>	100 to 199	
	700 sets	400 to 999 <b>(Note 4)</b> 100100 to 800199 <b>(Note 7)</b>	100 to 199	
Multi-part system (n = number of part systems)	50 + 50 * n sets	500 to 549	100 to 149 * n	
	100 + 100 * n sets	500 to 599	100 to 199 * n	
	200 + 100 * n sets	500 to 699	100 to 199 * n	
	500 + 100 * n sets	500 to 999 100100 to 800199 <b>(Note 7)</b>	100 to 199 * n	
	600 + 100 * n sets	400 to 999 <b>(Note 4)</b> 100100 to 800199 <b>(Note 7)</b>	100 to 199 * n	
Local variables		1 to 33		Can be used for local variables in macro programs.
System variables		From 1000		Application is fixed by system.
Fixed cycle variables		1 to 32		Local variables in fixed cycle programs.

**(Note 1)** All common variables are retained even when the power is turned OFF.

**(Note 2)** When the power is turned OFF or reset, the common variables can be set to <null> by setting the parameter (#1128 RstVC1, #1129 PwrVC1).

**(Note 3)** The common variables are divided into the following two types.

Common variables 1 : Used in common through all part systems

Common variables 2 : Used in common in the programs of the part system

**(Note 4)** Address #400s common variable can be used only when the sets of common variable is "700 sets" and the parameter "#1336 #400\_Valtype" is "1". If address #400s common variable is used when the set of common variable is other than 700 sets or the parameter "#1336 #400\_Valtype" is "0", a program error (P241) will occur.

When common variable address #400s can be used, these can be displayed and set on the common variable screen.

It is possible to input/output data of common variable address #400s.

**(Note 5)** When the parameter "#1052 MemVal" is set to "1" in multi-part system, a part or all of common variable "#100 to #199" and "#500 to #999" can be shared and used between part systems. The number of the shared common variable is set by the parameter "#1303 V1comN" (#100 to set value) and "#1304 V0comN" (#500 to set value).

(Example) "10" is set to "#1303 V1comN" "5" is set to "#1304 V0comN"  
 #100 to #109 : Common for the #500 to #504 : Common for the  
 part systems part systems  
 #110 to #199 : Each part system #505 to #999 : Each part system

When the parameter "#1052 MemVal" is set to "0", the common variables "#100 to #199" are used for each part system, and variables "#500 to #999" are common for the part systems. Address #400s, which can be used as common variable with 700 sets of variable, is common for the part systems regardless of the setting of parameter "#1052 MemVal".

**(Note 6)** In the common variable data input, when the following illegal variable No. data exist in input file, the illegal variable No. data is ignored and only the correct common variable data is input.

- Variable data which is not common variables of local variable (#1 to #33) and system variable (#1000 to ), etc.
- Variable data to which conditions of number of common variable sets are not corresponding

(Example) If the #100 to #199, #500 to #599 exist in the input file when the common variable 100 sets (#100 to #149, #500 to #549), the #150 to #199, #550 to #599 are ignored, and #100 to #149, #500 to #549 are input.

**(Note 7)** When the parameter "#1316 CrossCom" is set to "1", the common variables #100100 to #800199 can be used for common variable shared between the part systems. The common variable shared between part systems which can be used is shown in the table below.

Variable sets		Common variables 1 (When "#1316 CrossCom" = "1")
Variable sets specification	600 sets (500 + 100)	#100100 to #100199 (Equivalent to # 100 to #199 in 1st part system) #200100 to #200199 (Equivalent to # 100 to #199 in 2nd part system)
	700 sets (600 + 100)	#300100 to #300199 (Equivalent to # 100 to #199 in 3rd part system) #400100 to #400199 (Equivalent to # 100 to #199 in 4th part system) #500100 to #500199 #600100 to #600199 #700100 to #700199 #800100 to #800199

**<Example>**

When 1-part system

```
#100100=200 ;      Equivalent to #100 = 200 ;
#200105=#100 ;    "200" is set to #200105
#300110=#100100  "200" is set to #300110
;                  The variable value of #500120 is set to #800199
#800199=#500120
;
```



**(3) Variable quotations**

Variables can be used for all addresses except L(O), N and / (slash).

- (a) When the variable value is used directly:  
X#1 ..... Value of #1 is used as the X value.
- (b) When the complement of the variable value is used:  
X-#2 ..... Value with the #2 sign changed is used as the X value.
- (c) When defining variables:  
#3 = #5 ..... Variable #3 uses the equivalent value of variable #5.  
#1 = 1000 ..... Variable #1 uses the equivalent value 1000 (which is treated as 1000.).
- (d) When defining the variable arithmetic formula:  
#1 = #3 + #2 - 100 ..... The value of the operation result of #3 + #2 - 100. is used as the #1 value.  
X [#1 + #3 + 1000] ..... The value of the operation result of #1 + #3 + 1000 is used as the X value.

**(Note 1)** A variable cannot be defined in the same block as an address. It must be defined in a separate block.

Incorrect	→	Correct
X#1 = #3 + 100;		#1 = #3 + 100; X#1;

**(Note 2)** Up to five sets of square parentheses [ ] may be used.

#543 = - [ [ [ [ [ #120 ] / 2 + 15. ] \* 3 - #100 ] / #520 + #125 + #128 ] \* #130 + #132 ]

**(Note 3)** There are no restrictions on the number of characters and number of variables for variable definition.

**(Note 4)** The variable values should be within a range form 0 to ±999999999.  
If this range is exceeded, the arithmetic operations may not be conducted properly.

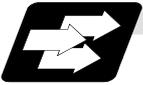
**(Note 5)** The variable definitions are valid from the moment that the variables are actually defined.

#1 = 100 ; ..... #1 = 100 (Valid from the next command)  
#1 = 200 #2 = #1 + 200 ; ..... #1 = 200, #2 = 400 (Valid from the next command)  
#3 = #1 + 300 ; ..... #3 = 500 (Valid from the next command)

**(Note 6)** Variable quotations are always regarded as having a decimal point at the end.  
When #100 = 10, then X#100 ; is treated as X10.

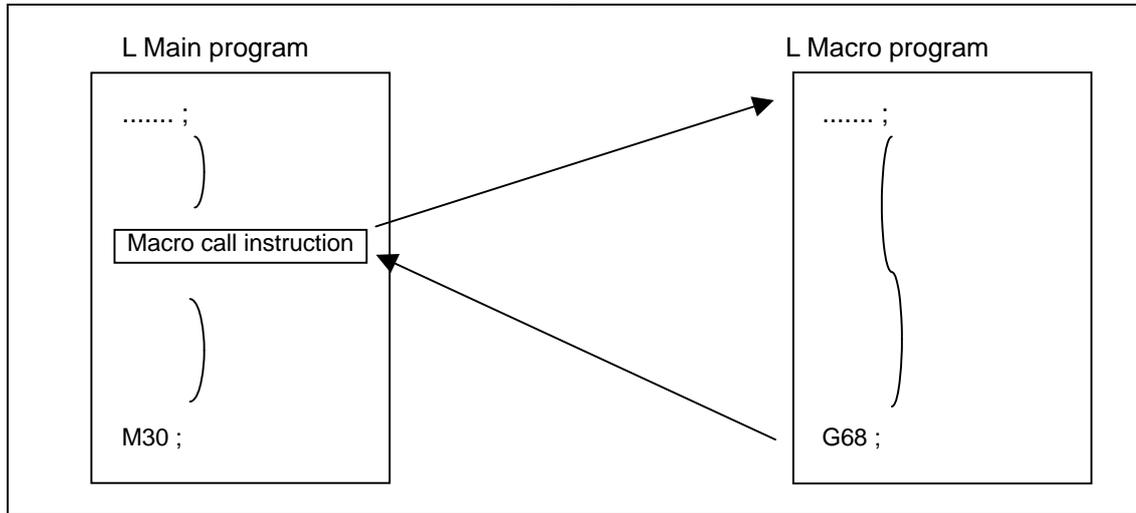
13.5 User Macro Specifications

13.5.1 User Macro Commands; G65, G66, G66.1, G67, G68(G23)



Function and purpose

By combining the user macros with variable commands, it is possible to use macro program call, arithmetic operation, data input/output with PLC, control, decision, branch and many other instructions for measurement and other such applications.



Macro programs use variables, arithmetic instructions and control instructions to create subprograms which function to provide special-purpose control. These special-purpose control functions (macro programs) are called by the macro call instructions exactly when required from the main program. The following G codes are available for the macro call commands.

G code	Function
G65	User macro Simple call
G66	User macro Modal call A (Movement command call)
G66.1	User macro Modal call B (Per-block call)
G67	User macro Modal call cancel
G68	User macro program termination (same as G23)



Detailed description

- (1) When the G66 (or 66.1) command is entered, the specified user macro subprogram will be called after each block has been executed (or after the movement command in the block) with the movement commands has been executed until the G67 (cancel) command is entered.
- (2) The G66 (or G66.1) and G67 commands must be paired in the same program.

## 13.5.2 Macro Call Command

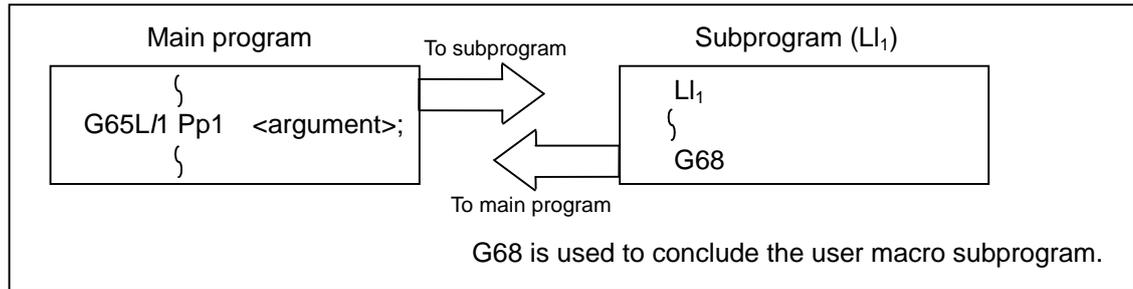


## Function and purpose

Included among the macro call commands are the simple calls which apply only to the instructed block and also modal calls (types A and B) which apply to each block in the call modal.



## Simple macro calls



## Format

```

G65 L_(A_) P_ argument ;
or G65 <File name> P_ argument ;
G65          Call instruction
L(A)        Program No.
<File name> File name
              A file name can be specified instead of a program No.
              In this case, enclose the file name with brackets <>.
              (The file name can have up to 32 characters including the extension.)
P           Number of repetitions
Argument    Specify variable data
  
```

When the argument must be transferred as a local variable to a user macro subprogram, the actual value should be designated after the address.

Regardless of the address, a sign and decimal point can be used in the argument. There are 2 ways in which arguments are designated.

**(1) Argument designation I**

Format : A\_ B\_ C\_ ... X\_ Y\_ Z\_

**Detailed description**

- (a) Arguments can be designated using any address except G, L(A), N, O and P.
- (b) Except for I, J and K, there is no need for designation in alphabetical order.
- (c) I, J and K must be designated in alphabetical order.

I\_ J\_ K\_ ..... Correct  
J\_ I\_ K\_ ..... Incorrect

- (d) Addresses which do not need to be designated can be omitted.
- (e) The following table shows the correspondence between the addresses which can be designated by argument designation I and the variable Nos. in the user macro main body.

Address and variable number correspondence		Call instructions and usable address	
Argument designation I address	Variable in macro	G65, G66	G66.1
A	#1	×	×
B	#2	○	○
C	#3	○	○
D	#7	○	○
E	#8	○	○
F	#9	○	○
G	#10	×	×*
H	#11	○	○
I	#4	○	○
J	#5	○	○
K	#6	○	○
L	#12	×	×*
M	#13	○	○
N	#14	×	×*
O	#15	×	×
P	#16	×	×*
Q	#17	○	○
R	#18	○	○
S	#19	○	○
T	#20	○	○
U	#21	○	○
V	#22	○	○
W	#23	○	○
X	#24	○	○
Y	#25	○	○
Z	#26	○	○

○ : Can be used.

× : Cannot be used.

\* : Can be used while G66.1 command is modal.

**(Note)** With the M2/M0 format-label O type, the address "A" is used as the address for subprogram No., thus, "A" can not be used as the local variable No.

**(2) Argument designation II**

Format : A\_\_ B\_\_ C\_\_ I\_\_ J\_\_ K\_\_ I\_\_ J\_\_ K\_\_ ••••

Detailed description

- (a) In addition to address A, B and C, up to 10 groups of arguments with I, J, K serving as 1 group can be designated.
- (b) When the same address is duplicated, designate the addresses in the specified order.
- (c) Addresses which do not need to be designated can be omitted.
- (d) The following table shows the correspondence between the addresses which can be designated by argument designation II and the variable numbers in the user macro main body.

Argument designation II address	Variable within macro
A	# 1
B	# 2
C	# 3
I1	# 4
J1	# 5
K1	# 6
I2	# 7
J2	# 8
K2	# 9
I3	#10
J3	#11
K3	#12
I4	#13
J4	#14
K4	#15
I5	#16

Argument designation II address	Variable within macro
J5	#17
K5	#18
I6	#19
J6	#20
K6	#21
I7	#22
J7	#23
K7	#24
I8	#25
J8	#26
K8	#27
I9	#28
J9	#29
K9	#30
I10	#31
J10	#32
K10	#33

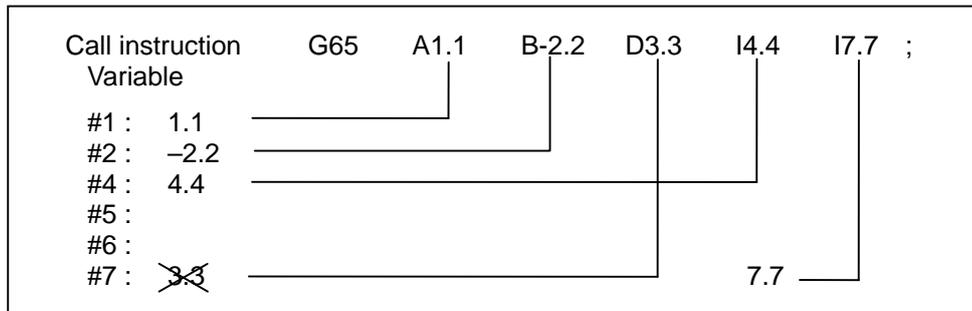
**(Note 1)** The numbers 1 through 10 accompanying I, J and K denote the sequence of the commanded groups and they are not required for the actual instructions.

**(Note 2)** With the M2/M0 format-label O type, the address "A" is used as the address for subprogram No., thus, "A" can not be used as the local variable No.

**(3) Using arguments designations I and II together**

If addresses corresponding to the same variable are commanded when both types I and II are used to designate arguments, the latter address is valid.

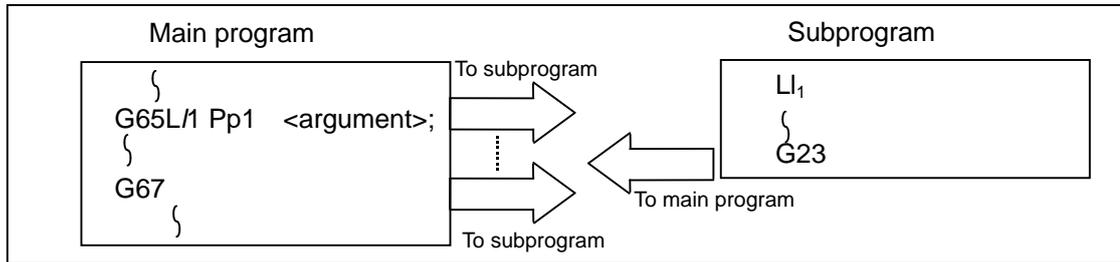
**(Example 1)**



In the above example, the last I7.7 argument is valid when both arguments D3.3 and I7.7 are commanded for the #7 variable.



## Modal call A (movement command call)



When the block with a movement command is commanded between G66 and G67, the movement command is first executed and then the designated user macro subprogram is executed. The number of times the subprogram is executed is  $\ell 1$  times with each call. The argument is the same as for a simple call.

## Format

```

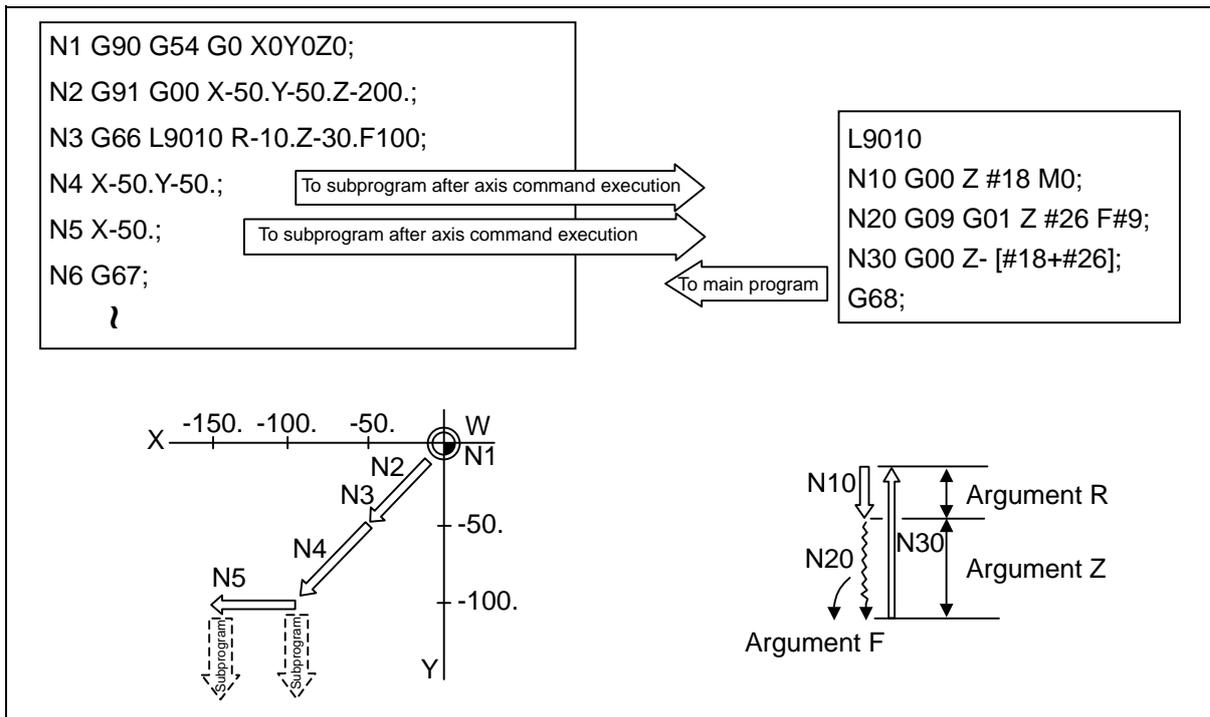
G66 L_ P_ argument ;
or G66 <File name> P_ argument ;
G66          Call instruction
L_           Program No.
<File name>  File name
              A file name can be specified instead of a program No.
              In this case, enclose the file name with brackets <>.
              (The file name can have up to 32 characters including the extension.)
P_           Number of repetitions
Argument     Specify variable data

```

## Detailed description

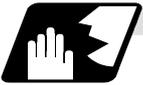
- (1) When the G66 command is entered, the specified user macro subprogram will be called after the movement command in the block with the movement commands has been executed until the G67 (cancel) command is entered.
- (2) The G66 and G67 commands must be paired in the same program. A program error will result when G67 is issued without the G66 command.

**(Example)** Drill cycle



**(Note 1)** After the axis command is executed in the main program, the subprogram is executed.

**(Note 2)** The subprogram is not executed in the blocks following G67.

**Modal call B (for each block)**

The specified user macro subprogram is called unconditionally for each command block which is assigned between G66.1 and G67 and the subprogram is executed the specified number of times.

Format

<b>G66.1 L__ P__ <i>argument</i> ;</b>	
<b>or G66.1 &lt;File name&gt; P__ <i>argument</i> ;</b>	
G66.1	Call instruction
L__	Program No.
<File name>	File name
	A file name can be specified instead of a program No. In this case, enclose the file name with brackets <>. (The file name can have up to 32 characters including the extension.)
P__	Number of repetitions
Argument	Specify variable data

Detailed description

- (1) In the G66.1 mode, everything except the L(O), N and G codes in the various command blocks which are read are handled as the argument without being executed. Any G code designated last or any N code commanded after anything except L(O) and N will function as the argument.
- (2) The same applies as when G65L\_\_ is assigned at the head of a block for all significant blocks in the G66.1 mode.

**(Example 1)**

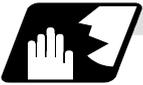
N100 G01 G90 X100. Y200. F400 R1000; in the G66.1 L1000; mode is the same as: N100 G65 L1000 G01 G90 X100. Y200. F400 R1000;

**(Note 1)**

The Call is performed even in the G66.1 command block in the G66.1 mode and the correspondence between the argument address and the variable number is the same as for G65 (simple call).

- (3) The range of the G and N command values which can be used anew as variables in the G66.1 mode is subject to the restrictions applying to values as normal NC command values.
- (4) Program number L(O), sequence numbers N and modal G codes are updated as modal information.





**Miscellaneous command macro call (for M, S, T, B code macro call)**

The user macro subprogram of the specified program number can be called merely by issuing an M (or S, T, B) code. (Only entered codes apply for M but all S, T and B codes apply.)

**Format**

<b>M** ; (or S** ;, T** ;, B** ;)</b>
M** M code for macro call (or S, T, B code)

**Detailed description**

(1) The above instruction functions in the same way as the instructions below, and parameters are set for each M code to determine the correspondence with the instructions. (Same for S, T and B codes)

- a : G22 LΔΔΔΔ ;
  - b : G65 LΔΔΔΔ M\*\* ;
  - c : G66 L ΔΔΔΔ M\*\* ;
  - d : G66.1 LΔΔΔΔ M\*\* ;
- } M\*\* are not output

When the parameters corresponding to c and d above are set, issue the cancel command (G67) either in the user macro or after the call code has been commanded so as to cancel the modal call.

- (2) The correspondence between the "M\*\*" which conducts the macro call and the macro program number LΔΔΔΔ to be called is set by parameter. Up to 10 M codes from M00 to M95 can be entered. Note that the codes to be registered are the codes basically required for the machine, and codes excluding M0, M1, M2, M30.
- (3) As with G22, it is displayed on the screen display of the setting and display unit but the M codes and MF are not output.
- (4) Even if the miscellaneous command entered above is issued during a user macro subprogram called by the M code, macro call will not result and it will be handled as an ordinary miscellaneous command.
- (5) All S, T and B codes call the subprograms in the prescribed program numbers of the corresponding S, T and B functions.
- (6) A maximum of 10 M codes can be set. However when not setting all 10. Set the parameters as shown below.

[ MACRO ]			
	<Code>	<Type>	<Program No.>
M [01]	20	0	8000
M [02]	21	0	8001
M [03]	9999	0	19999999
M [04]	9999	0	19999999
M [05]	9999	0	19999999
	:	:	:
	:	:	:
M [10]	9999	0	19999999

Setting to call L8000 with type 0 (G22 type) during M20 command

Setting to call L8001 with type 0 (G22 type) during M21 command

Set parameters not being used as shown on left.



**Differences between G22 and G65 commands**

- (1) The argument can be designated for G65 but not for G22.
- (2) The sequence number can be designated for G22 but no for G65, G66 and G66.1.
- (3) G22 executes a subprogram after all the commands except M, P, H and L(O) in the G22 block have been executed, but G65 branches to the subprogram without any further operation.
- (4) When any address except L(O), N, P, H or L is included in the G22 block, single block stop results. This is not the case with G65.
- (5) The level of the G22 local variables is fixed but it can be varied in accordance with the nesting depth for G65. (#1, for instance, has the same significance either before or after G22 but a different significance in each case with G65.)
- (6) The G22 nesting depth extends up to 8 levels in combination with G65, G66 and G66.1. The G65 nesting depth extends up to only 4 levels in combination with G66 and G66.1.



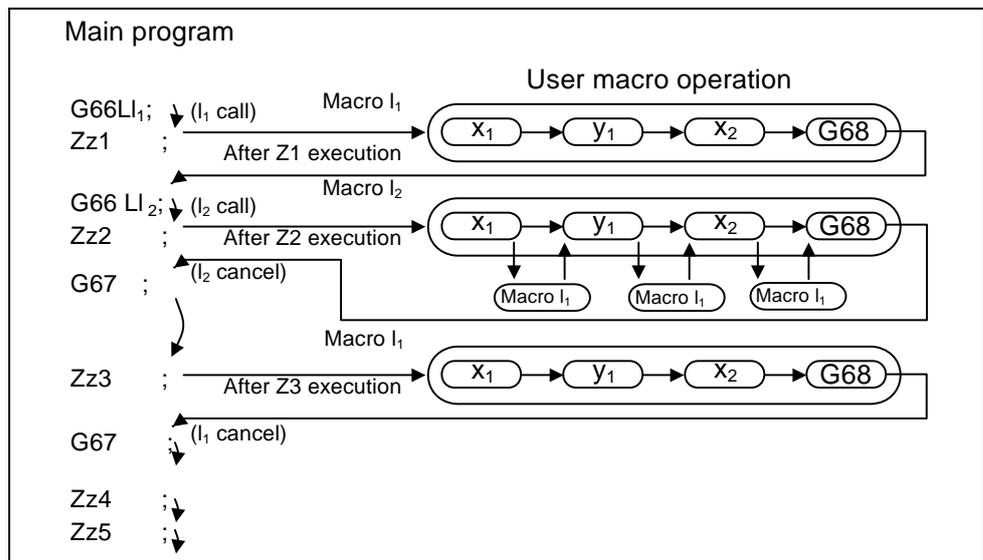
**Macro call command nesting depth**

Up to 4 nesting levels are available for macro subprogram calls based on simple call or modal call. The argument with a macro call instruction is valid only on the called macro level. Since the nesting depth for macro calls extends up to 4 levels, the argument can be used as a local variable for the program with each respective macro call.

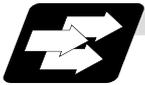
**(Note 1)** When a G65, G66, G66.1 G code macro call or miscellaneous command macro call is conducted, this is regarded as nesting level 1 and the level of the local variables is also incremented by one.

**(Note 2)** The designated user macro subprogram is called every time the movement command is executed with modal call A. However, when the G66 command has been duplicated, the next user macro subprogram is called every time an axis is moved even with movement commands in the macro. User macro subprograms are called in sequence from the subprogram commanded last.

**(Example 1)**



13.5.3 ASCII Code Macro



Function and purpose

A macro program can be called out by setting the correspondence of a subprogram (macro program) pre-registered with the parameters to codes, and then commanding the ASCII code in the machining program.

This function can be used in addition to the G, M, S, T and B miscellaneous command macro call function.

(Execution example 1) G22 type

Main program

```
L0002 ;
:
D2000 ;
:
M30 ;
```

Subprogram

```
L200
:
:
:
G23 ;
```

After outputting 2000 to common variable #146, the program No. 200 subprogram is called with the G22 subprogram call type.

Parameter	
#7401 (ASCII [01] Valid/Invalid)	1 (Valid)
#7402 (ASCII [01] Code)	D
#7403 (ASCII [01] Type)	0 (G22 type)
#7404 (ASCII [01] Program No.)	200
#7405 (ASCII [01] Variable)	146

(Execution example 2) G65 type

Main program

```
L0003 ;
:
A500 ;
:
M30 ;
```

Subprogram

```
L3000
:
:
:
G23 ;
```

After outputting 500 to local variable #1, the program No. 3000 subprogram is called out with the G65 macro call type.

Parameter	
#7411 (ASCII [02] Valid/Invalid)	1 (Valid)
#7412 (ASCII [02] Code)	A
#7413 (ASCII [02] Type)	0 (G65 type)
#7414 (ASCII [02] Program No.)	3000
#7415 (ASCII [02] Variable)	100 (Not used)



## Command format

<input type="checkbox"/> ****;	<b>Designates the address and code</b>
<input type="checkbox"/>	: ASCII code for calling out macro (one character)
****	: Value or expression output to variable (Setting range: ±999999.9999)



## Detailed description

- (1) The command above functions in the same way as that below. The correspondence of commands is set for each ASCII code with the parameters.
  - 0: G22 LΔΔΔΔ ;
  - 1: G65 LΔΔΔΔ <Argument> ;
  - 2: G66 LΔΔΔΔ <Argument> ;
  - 3: G66.1 LΔΔΔΔ <Argument> ;
 In order to cancel the modal call while parameters are set for 2 and 3 above, issue the cancel command (G67) after commanding the call code or the during the user macro.
- (2) The ASCII code for calling the macro and the program No. LΔΔΔΔ to be called are set with the parameters.  
Up to two ASCII codes can be registered.
- (3) The code section is output to the variables, but the output destination differs according to the call type and address.
  - (a) For G22 type  
The code section is output to the common variable, and the variable No. is set with the parameters.  
When corresponding to the first address (parameter #7401), the section is output to the common variable where the first variable No. (parameter #7404) is indicating.
  - (b) For G65/G66/G66.1 type  
The code section is output to the local variable. The variable No. differs according to the address, and corresponds to the following table.

Address	#	Address	#	Address	#
A	1	K	6	U	21
B	2	L	12	V	22
C	3	M	13	W	23
D	7	N	14	X	24
E	8	O	15	Y	25
F	9	P	16	Z	26
G	10	Q	17		
H	11	R	18		
I	4	S	19		
J	5	T	20		

**(Note)** The following addresses can be used.

A, B, D, F, H, I, J, K, M, Q, R, S, T
---------------------------------------



### Restrictions

- (1) Calling a macro with an ASCII code from a program macro-called with an ASCII code

A macro cannot be called with an ASCII code from a program macro-called with an ASCII code.

The other patterns are shown below.

If it is determined that the macro cannot be called, the command will be handled as a normal command.

		Called side			
		ASCII	GMSTB macro	G65/66/66.1	G22
Calling side	ASCII	x	x	○	○
	GMSTB macro	x	x	○	○
	G65/66/66.1	○	○	○	○
	G22	○	○	○	○

- (2) Nest level of macro call command

The macro subprogram can be called in up to four levels using simple call (G65) and modal call (G66/G66.1).

The macro call command's argument is valid only in the called macro level.

Since the macro call nest level is four levels, the argument for each macro call can be used in the program as a local variable.

- (3) Nest level of subprogram call

Counting the main program as 0, up to eight levels of subprograms can be called (G22) from a subprogram.

The following commands are used for subprogram nesting.

- (a) G22
- (b) G65 G66 G66.1
- (c) G code call Miscellaneous function call (M/S/T/B)
- (d) MDI interruption
- (e) Automatic tool length measurement
- (f) Multiple-step skip function

The following commands can be commanded regardless of nesting.

- (g) Fixed cycle
- (h) Macro interruption

## (4) Order of command priority

If "M" is designated for the ASCII code address, the codes basically necessary for that machine will be overlapped. In this case, commands will be identified with the following priority using code values.

- (a) M00 (program stop command), M01 (optional stop command)  
M02, M30 (end command)
- (b) When corresponding to miscellaneous code (M) call macro command
- (c) When corresponding to ASCII code macro command
- (d) Used as normal command

If "S", "T" and "B" are designated for the ASCII code address, commands will be identified with the following priority using code values.

- (a) When corresponding to miscellaneous code (S, T, B) call macro command
- (b) When corresponding to ASCII code macro command
- (c) Used as normal command

If the other addresses do not correspond to the ASCII code macro command, they will be identified as normal commands. If the command to be used overlaps with an ASCII code macro command, it must be commanded in the program macro-called with the ASCII code. Note that there are cases where the command will be unconditionally handled as a normal command, as explained in (5) below.

## (5) Conditions where the address set in ASCII code macro command is handled as a normal command

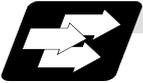
- (a) When there is a data setting command (G10) in the same block.
- (b) When ASCII code macro call is executed after the G code macro call command in the same block (also applies for M, S, T, B and ASCII)

**(Example)** When address "D" (G65 type) is set in the ASCII code macro, and M50 is set in the macro call (G65 type).

M50 D200 ;     Execute M code macro with argument (Set 200 in #7)

- (c) When inputting parameters
- (d) When there is a comma (,) before the address. **(Example)** ",D", ",R", etc.
- (e) When commanded in fixed cycle
- (f) When commanded in macro subprogram called with G code macro call  
(Also applies when macro is called with M, S, T, B or ASCII)

## 13.5.4 Variables

**Function and purpose**

Both the variable specifications and user macro specifications are required for the variables which are used with the user macros.

The offset amounts of the local, common and system variables among the variables for this MELDAS NC system except #33 are retained even when the unit's power is switched off. (Common variables can also be cleared by parameter "#1129 PwrVC1".)

**Variable multiplexing**

When the user macro specifications applied, variable numbers can be turned into variables (multiple use of variables) or replaced by <formula>. Only one of the four basic arithmetic rule (+, -, ×, ÷) operations can be conducted with <formula>.

**(Example 1)** Multiple use of variables

#1 = 10 #10 = 20 #20 = 30 ; #5 = #[#[#1]] ;	} #[#[#1]] = #[#10] from #1 = 10. } #[#10] = #20 from #10 = 20. Therefore, #5 = #20 or #5 = 30.
--	---

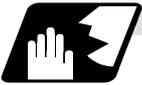
#1 = 20 #10 = 20 #20 = 30 #5 = 1000 ; #[#[#1]] = #5 ;	} #[#[#1]] = #[#10] from #1 = 10. } #[#10] = #20 from #10 = 20. Therefore, #20 = #5 or #20 = 1000.
--	--

**(Example 2)** Example of multiple designation of variables

#10 = 5 In which case ##10 = 100 ; #5 = 100	<Formula>##10 = 100; is handled in the same manner as # [#10] = 100.
--	--

**(Example 3)** Replacing variable numbers with <formula>

#10 = 5 ; #[#10 + 1] = 1000 ; #[#10 - 1] = -1000 ; #[#10*3] = 100 ; #[#10/2] = -100 ;	In which case, #6 = 1000. In which case, #4 = -1000. In which case, #15 = 100. In which case, #3 = -100. (Rounded off.)
---	--



**Undefined variables**

Variables applying with the user macro specifications such as variables which have not been used even once after the power was switched on or local variables not quoted by the G65, G66 or G66.1 commands can be used as <vacant>. Also, variables can forcibly be set to <vacant>. Variable #0 is always used as the <vacant> variable and cannot be defined in the left-side member.

**(1) Arithmetic expressions**

- #1 = #0 ; ..... #1 = <vacant>
- #2 = #0 + 1 ; ..... #2 = 1
- #3 = 1 + #0 ; ..... #3 = 1
- #4 = #0\*10 ; ..... #4 = 0
- #5 = #0 + #0 ; ..... #5 = 0

It should be borne in mind that <vacant> in an arithmetic expression is handled in the same way as 0.

<Vacant> + <Vacant> = 0  
 <Vacant> + <Constant> = Constant  
 <Constant> + <Vacant> = Constant

**(2) Variable quotations**

When undefined variables only are quoted, they are ignored up to the address.  
 When #1 = <Vacant>  
 G0 X#1 Y1000 ; ..... Equivalent to G0 Y1000 ;  
 G0 X#1 + 10 Y1000 ; ..... Equivalent to G0 X10 Y1000 ;

**(3) Conditional expressions**

<Vacant> and 0 are not equivalent for EQ and NE only. (#0 means <vacant>.)

When #101 = <Vacant>	When #101 = 0
#101 EQ #0 <Vacant> = <Vacant> established	#101 EQ #0 0 = <Vacant> not established
#101 NE 0 <Vacant> ≠ 0 established	#101 NE 0 0 ≠ 0 not established
#101 GE #0 <Vacant> ≥ <Vacant> established	#101 GE #0 0 ≥ <Vacant> established
#101 GT 0 <Vacant> > 0 not established	#101 GT 0 0 > 0 not established
#101 LE #0 <Vacant> ≤ <Vacant> established	#101 LE #0 0 ≤ <Vacant> established
#101 LT 0 <Vacant> < 0 not established	#101 LT 0 0 < 0 not established

**(Note 1)** EQ and NE should be used only for integers. For comparison of numeric values with decimals, GE, GT, LE, and LT should be used.

13.5.5 Types of Variables



**Common variables**

Common variables can be used commonly from any position. Number of the common variables sets depends on the specifications. Refer to "13.4 Variable commands" for details.



**Local variables (#1 to #33)**

These can be defined as an <argument> when a macro subprogram is called or used locally within main programs and subprograms. They can be duplicated regardless of the relationship existing between macros (up to 4 levels).

```
G65 L__ P__ <argument> ;
L           : Program number
P           : Number of repetitions
```

The <argument> is assumed to be Aa1 Bb1 Cc1 ..... Zz1.

The following table shows the correspondences between the addresses designated by <argument> and the local variable numbers used in the user macro main bodies.

[Argument specification I]

Call command		Argument address	Local variable number
G65 G66	G66.1		
×	×	A	#1
○	○	B	#2
○	○	C	#3
○	○	D	#7
○	○	E	#8
○	○	F	#9
×	×*	G	#10
○	○	H	#11
○	○	I	#4
○	○	J	#5
○	○	K	#6
×	×*	L	#12
○	○	M	#13
×	×*	N	#14
×	×	O	#15
×	×*	P	#16

Call command		Argument address	Local variable number
G65 G66	G66.1		
○	○	Q	#17
○	○	R	#18
○	○	S	#19
○	○	T	#20
○	○	U	#21
○	○	V	#22
○	○	W	#23
○	○	X	#24
○	○	Y	#25
○	○	Z	#26
		-	#27
		-	#28
		-	#29
		-	#30
		-	#31
		-	#32
		-	#33

"×" in the above table denotes an argument address which cannot be used. However, provided that the G66.1 mode has been established, an argument address denoted by the asterisk can be added for use.

"-" denotes that a corresponding address is not available.

**(Note)** With the M2/M0 format-label O type, the address "A" is used as the address for subprogram No., thus, "A" can not be used as the local variable No.

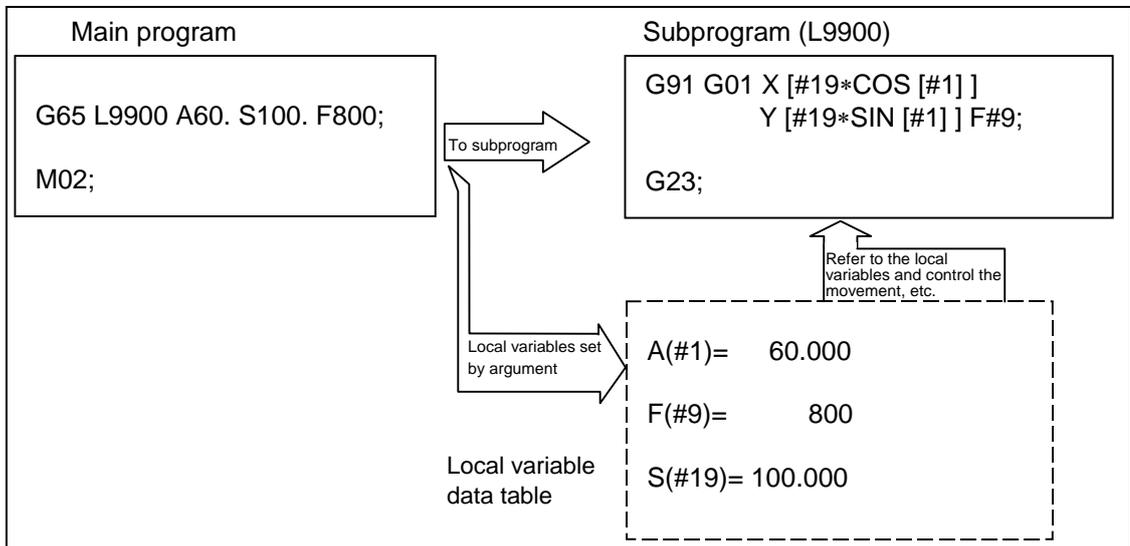
[Argument specification II]

Argument specification II address	Variable in macro
A	# 1
B	# 2
C	# 3
I1	# 4
J1	# 5
K1	# 6
I2	# 7
J2	# 8
K2	# 9
I3	#10
J3	#11
K3	#12
I4	#13
J4	#14
K4	#15
I5	#16
J5	#17
K5	#18

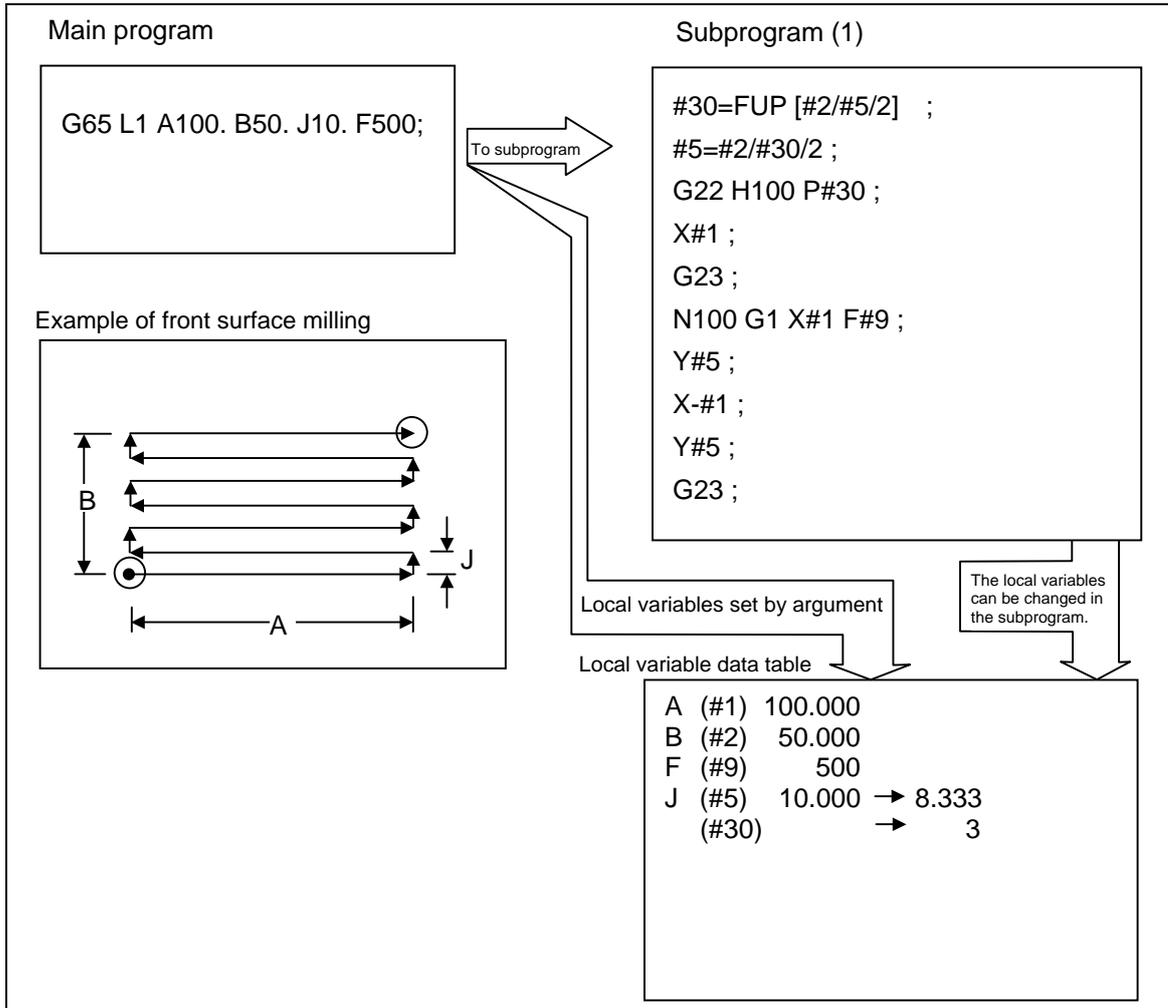
Argument specification II address	Variable in macro
I6	#19
J6	#20
K6	#21
I7	#22
J7	#23
K7	#24
I8	#25
J8	#26
K8	#27
I9	#28
J9	#29
K9	#30
I10	#31
J10	#32
K10	#33

**(Note 1)** Subscripts 1 to 10 for I, J, and K indicate the order of the specified command sets. They are not required to specify instructions.

(1) Local variables in subprograms can be defined by means of the <argument> designation during macro call.

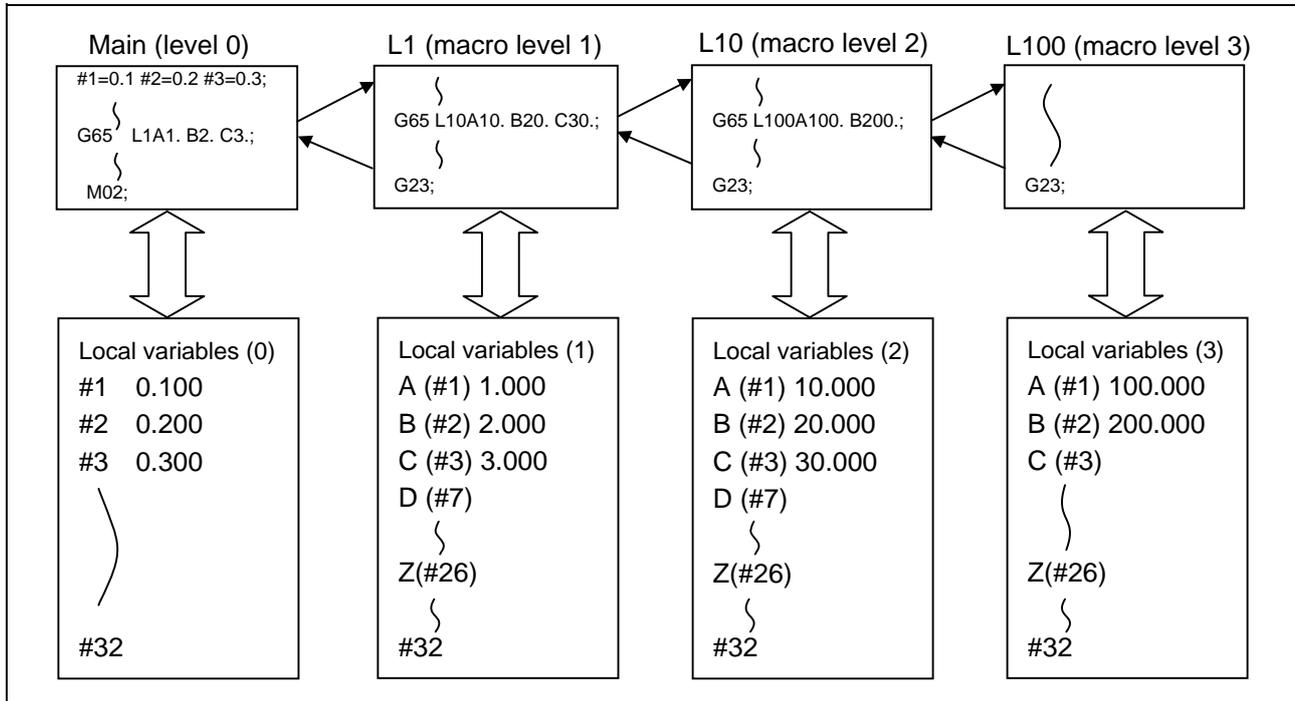


(2) The local variables can be used freely in that subprogram.



In the front surface milling example, argument J is programmed as the milling pitch 10.mm. However, this is changed to 8.333mm to create an equal interval pitch. The results of the No. of reciprocation data calculation is set in local variable #30.

- (3) Local variables can be used independently on each of the macro call levels (4 levels). Local variables are also provided independently for the main program (macro level 0). Arguments cannot be used for the level 0 local variables.



The status of the local variables appear on the setting and display unit. Refer to the Operation Manual for details.


**Macro interface inputs (#1000 to #1035, #1200 to #1295) : PLC → NC**

The status of the interface input signals can be ascertained by reading out the values of variable numbers #1000 to #1035, #1200 to #1295. A variable value which has been read out can be only one of 2 values: 1 or 0 (1: contact closed, 0: contact open). All the input signals from #1000 to #1031 can be read at once by reading out the value of variable number #1032.

Similarly, the input signals #1200 to #1231, #1232 to #1263, and #1264 to #1295 can be read by reading the values of the variable numbers #1033 to #1035.

Variable numbers #1000 to #1035, #1200 to #1295 are for readout only, and cannot be placed in the left side member of their arithmetic formula. Input here refers to input to the control unit.

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1000	1	Register R6436 bit 0	#1016	1	Register R6437 bit 0
#1001	1	Register R6436 bit 1	#1017	1	Register R6437 bit 1
#1002	1	Register R6436 bit 2	#1018	1	Register R6437 bit 2
#1003	1	Register R6436 bit 3	#1019	1	Register R6437 bit 3
#1004	1	Register R6436 bit 4	#1020	1	Register R6437 bit 4
#1005	1	Register R6436 bit 5	#1021	1	Register R6437 bit 5
#1006	1	Register R6436 bit 6	#1022	1	Register R6437 bit 6
#1007	1	Register R6436 bit 7	#1023	1	Register R6437 bit 7
#1008	1	Register R6436 bit 8	#1024	1	Register R6437 bit 8
#1009	1	Register R6436 bit 9	#1025	1	Register R6437 bit 9
#1010	1	Register R6436 bit 10	#1026	1	Register R6437 bit 10
#1011	1	Register R6436 bit 11	#1027	1	Register R6437 bit 11
#1012	1	Register R6436 bit 12	#1028	1	Register R6437 bit 12
#1013	1	Register R6436 bit 13	#1029	1	Register R6437 bit 13
#1014	1	Register R6436 bit 14	#1030	1	Register R6437 bit 14
#1015	1	Register R6436 bit 15	#1031	1	Register R6437 bit 15

System variable	No. of points	Interface input signal
#1032	32	Register R6436, R6437
#1033	32	Register R6438, R6439
#1034	32	Register R6440, R6441
#1035	32	Register R6442, R6443

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1200	1	Register R6438 bit 0	#1216	1	Register R6439 bit 0
#1201	1	Register R6438 bit 1	#1217	1	Register R6439 bit 1
#1202	1	Register R6438 bit 2	#1218	1	Register R6439 bit 2
#1203	1	Register R6438 bit 3	#1219	1	Register R6439 bit 3
#1204	1	Register R6438 bit 4	#1220	1	Register R6439 bit 4
#1205	1	Register R6438 bit 5	#1221	1	Register R6439 bit 5
#1206	1	Register R6438 bit 6	#1222	1	Register R6439 bit 6
#1207	1	Register R6438 bit 7	#1223	1	Register R6439 bit 7
#1208	1	Register R6438 bit 8	#1224	1	Register R6439 bit 8
#1209	1	Register R6438 bit 9	#1225	1	Register R6439 bit 9
#1210	1	Register R6438 bit 10	#1226	1	Register R6439 bit 10
#1211	1	Register R6438 bit 11	#1227	1	Register R6439 bit 11
#1212	1	Register R6438 bit 12	#1228	1	Register R6439 bit 12
#1213	1	Register R6438 bit 13	#1229	1	Register R6439 bit 13
#1214	1	Register R6438 bit 14	#1230	1	Register R6439 bit 14
#1215	1	Register R6438 bit 15	#1231	1	Register R6439 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1232	1	Register R6440 bit 0	#1248	1	Register R6441 bit 0
#1233	1	Register R6440 bit 1	#1249	1	Register R6441 bit 1
#1234	1	Register R6440 bit 2	#1250	1	Register R6441 bit 2
#1235	1	Register R6440 bit 3	#1251	1	Register R6441 bit 3
#1236	1	Register R6440 bit 4	#1252	1	Register R6441 bit 4
#1237	1	Register R6440 bit 5	#1253	1	Register R6441 bit 5
#1238	1	Register R6440 bit 6	#1254	1	Register R6441 bit 6
#1239	1	Register R6440 bit 7	#1255	1	Register R6441 bit 7
#1240	1	Register R6440 bit 8	#1256	1	Register R6441 bit 8
#1241	1	Register R6440 bit 9	#1257	1	Register R6441 bit 9
#1242	1	Register R6440 bit 10	#1258	1	Register R6441 bit 10
#1243	1	Register R6440 bit 11	#1259	1	Register R6441 bit 11
#1244	1	Register R6440 bit 12	#1260	1	Register R6441 bit 12
#1245	1	Register R6440 bit 13	#1261	1	Register R6441 bit 13
#1246	1	Register R6440 bit 14	#1262	1	Register R6441 bit 14
#1247	1	Register R6440 bit 15	#1263	1	Register R6441 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1264	1	Register R6442 bit 0	#1280	1	Register R6443 bit 0
#1265	1	Register R6442 bit 1	#1281	1	Register R6443 bit 1
#1266	1	Register R6442 bit 2	#1282	1	Register R6443 bit 2
#1267	1	Register R6442 bit 3	#1283	1	Register R6443 bit 3
#1268	1	Register R6442 bit 4	#1284	1	Register R6443 bit 4
#1269	1	Register R6442 bit 5	#1285	1	Register R6443 bit 5
#1270	1	Register R6442 bit 6	#1286	1	Register R6443 bit 6
#1271	1	Register R6442 bit 7	#1287	1	Register R6443 bit 7
#1272	1	Register R6442 bit 8	#1288	1	Register R6443 bit 8
#1273	1	Register R6442 bit 9	#1289	1	Register R6443 bit 9
#1274	1	Register R6442 bit 10	#1290	1	Register R6443 bit 10
#1275	1	Register R6442 bit 11	#1291	1	Register R6443 bit 11
#1276	1	Register R6442 bit 12	#1292	1	Register R6443 bit 12
#1277	1	Register R6442 bit 13	#1293	1	Register R6443 bit 13
#1278	1	Register R6442 bit 14	#1294	1	Register R6443 bit 14
#1279	1	Register R6442 bit 15	#1295	1	Register R6443 bit 15



#### Macro interface outputs (#1100 to #1135, #1300 to #1395) NC → PLC

The interface output signals can be sent by substituting values in variable numbers #1100 to #1135, #1300 to #1395. An output signal can be only 0 or 1.

All the output signals from #1100 to #1131 can be sent at once by substituting a value in variable number #1132.

Similarly, the output signals #1300 to #1311, #1332 to #1363, and #1364 to #1395 can be sent by assigning values to the variable numbers #1133 to #1135. ( $2^0$  to  $2^{31}$ )

The status of the writing and output signals can be read in order to offset the #1100 to #1135, #1300 to #1395 output signals. Output here refers to the output from the NC side.

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1100	1	Register R6372 bit 0	#1116	1	Register R6373 bit 0
#1101	1	Register R6372 bit 1	#1117	1	Register R6373 bit 1
#1102	1	Register R6372 bit 2	#1118	1	Register R6373 bit 2
#1103	1	Register R6372 bit 3	#1119	1	Register R6373 bit 3
#1104	1	Register R6372 bit 4	#1120	1	Register R6373 bit 4
#1105	1	Register R6372 bit 5	#1121	1	Register R6373 bit 5
#1106	1	Register R6372 bit 6	#1122	1	Register R6373 bit 6
#1107	1	Register R6372 bit 7	#1123	1	Register R6373 bit 7
#1108	1	Register R6372 bit 8	#1124	1	Register R6373 bit 8
#1109	1	Register R6372 bit 9	#1125	1	Register R6373 bit 9
#1110	1	Register R6372 bit 10	#1126	1	Register R6373 bit 10
#1111	1	Register R6372 bit 11	#1127	1	Register R6373 bit 11
#1112	1	Register R6372 bit 12	#1128	1	Register R6373 bit 12
#1113	1	Register R6372 bit 13	#1129	1	Register R6373 bit 13
#1114	1	Register R6372 bit 14	#1130	1	Register R6373 bit 14
#1115	1	Register R6372 bit 15	#1131	1	Register R6373 bit 15

System variable	No. of points	Interface output signal
#1132	32	Register R6372, R6373
#1133	32	Register R6374, R6375
#1134	32	Register R6376, R6377
#1135	32	Register R6378, R6379

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1300	1	Register R6374 bit 0	#1316	1	Register R6375 bit 0
#1301	1	Register R6374 bit 1	#1317	1	Register R6375 bit 1
#1302	1	Register R6374 bit 2	#1318	1	Register R6375 bit 2
#1303	1	Register R6374 bit 3	#1319	1	Register R6375 bit 3
#1304	1	Register R6374 bit 4	#1320	1	Register R6375 bit 4
#1305	1	Register R6374 bit 5	#1321	1	Register R6375 bit 5
#1306	1	Register R6374 bit 6	#1322	1	Register R6375 bit 6
#1307	1	Register R6374 bit 7	#1323	1	Register R6375 bit 7
#1308	1	Register R6374 bit 8	#1324	1	Register R6375 bit 8
#1309	1	Register R6374 bit 9	#1325	1	Register R6375 bit 9
#1310	1	Register R6374 bit 10	#1326	1	Register R6375 bit 10
#1311	1	Register R6374 bit 11	#1327	1	Register R6375 bit 11
#1312	1	Register R6374 bit 12	#1328	1	Register R6375 bit 12
#1313	1	Register R6374 bit 13	#1329	1	Register R6375 bit 13
#1314	1	Register R6374 bit 14	#1330	1	Register R6375 bit 14
#1315	1	Register R6374 bit 15	#1331	1	Register R6375 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1332	1	Register R6376 bit 0	#1348	1	Register R6377 bit 0
#1333	1	Register R6376 bit 1	#1349	1	Register R6377 bit 1
#1334	1	Register R6376 bit 2	#1350	1	Register R6377 bit 2
#1335	1	Register R6376 bit 3	#1351	1	Register R6377 bit 3
#1336	1	Register R6376 bit 4	#1352	1	Register R6377 bit 4
#1337	1	Register R6376 bit 5	#1353	1	Register R6377 bit 5
#1338	1	Register R6376 bit 6	#1354	1	Register R6377 bit 6
#1339	1	Register R6376 bit 7	#1355	1	Register R6377 bit 7
#1340	1	Register R6376 bit 8	#1356	1	Register R6377 bit 8
#1341	1	Register R6376 bit 9	#1357	1	Register R6377 bit 9
#1342	1	Register R6376 bit 10	#1358	1	Register R6377 bit 10
#1343	1	Register R6376 bit 11	#1359	1	Register R6377 bit 11
#1344	1	Register R6376 bit 12	#1360	1	Register R6377 bit 12
#1345	1	Register R6376 bit 13	#1361	1	Register R6377 bit 13
#1346	1	Register R6376 bit 14	#1362	1	Register R6377 bit 14
#1347	1	Register R6376 bit 15	#1363	1	Register R6377 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1364	1	Register R6378 bit 0	#1380	1	Register R6379 bit 0
#1365	1	Register R6378 bit 1	#1381	1	Register R6379 bit 1
#1366	1	Register R6378 bit 2	#1382	1	Register R6379 bit 2
#1367	1	Register R6378 bit 3	#1383	1	Register R6379 bit 3
#1368	1	Register R6378 bit 4	#1384	1	Register R6379 bit 4
#1369	1	Register R6378 bit 5	#1385	1	Register R6379 bit 5
#1370	1	Register R6378 bit 6	#1386	1	Register R6379 bit 6
#1371	1	Register R6378 bit 7	#1387	1	Register R6379 bit 7
#1372	1	Register R6378 bit 8	#1388	1	Register R6379 bit 8
#1373	1	Register R6378 bit 9	#1389	1	Register R6379 bit 9
#1374	1	Register R6378 bit 10	#1390	1	Register R6379 bit 10
#1375	1	Register R6378 bit 11	#1391	1	Register R6379 bit 11
#1376	1	Register R6378 bit 12	#1392	1	Register R6379 bit 12
#1377	1	Register R6378 bit 13	#1393	1	Register R6379 bit 13
#1378	1	Register R6378 bit 14	#1394	1	Register R6379 bit 14
#1379	1	Register R6378 bit 15	#1395	1	Register R6379 bit 15

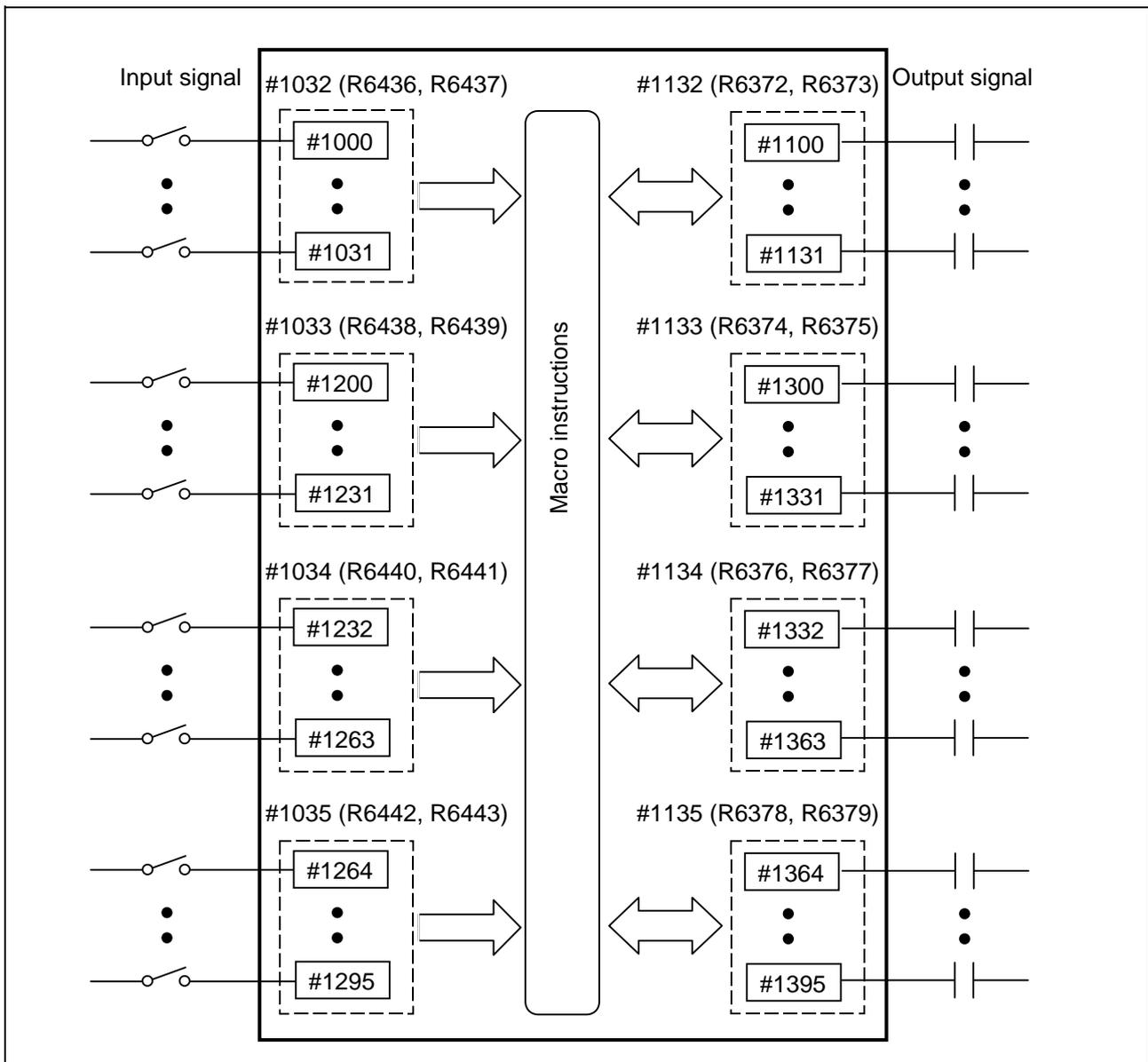
**(Note 1)** The last values of the system variables #1100 to #1135, #1300 to #1395 sent are retained as 1 or 0. (They are not cleared even with resetting.)

**(Note 2)** The following applies when any number except 1 or 0 is substituted into #1100 to #1131, #1300 to #1395.

<Vacant> is treated as 0.

Any number except 0 and <vacant> is treated as 1.

Any value less than 0.00000001 is indefinite.





**Tool compensation**

Tool data can be read and set using the variable numbers.

Variable number range		Type 1	Type 2
#10001 to #10000 + n	#2001 to #2000 + n	○	○ (Length dimension)
#11001 to #11000 + n	#2201 to #2200 + n	×	○ (Length wear)
#16001 to #16000 + n	#2401 to #2400 + n	×	○ (Radius dimension)
#17001 to #17000 + n	#2601 to #2600 + n	×	○ (Radius wear)

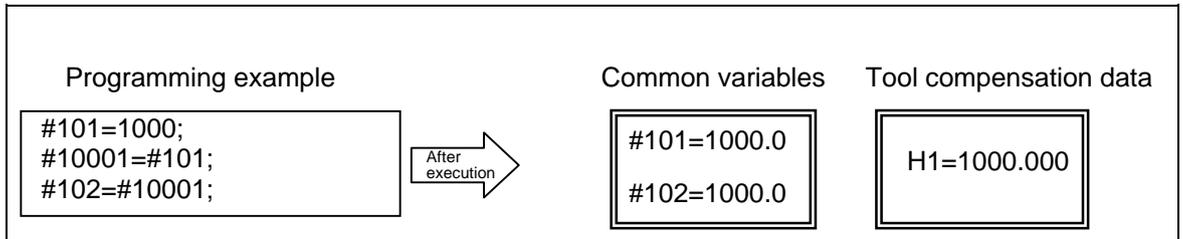
"n" in the table indicates the tool No.

The maximum value of "n" matches the number of tool offset sets.

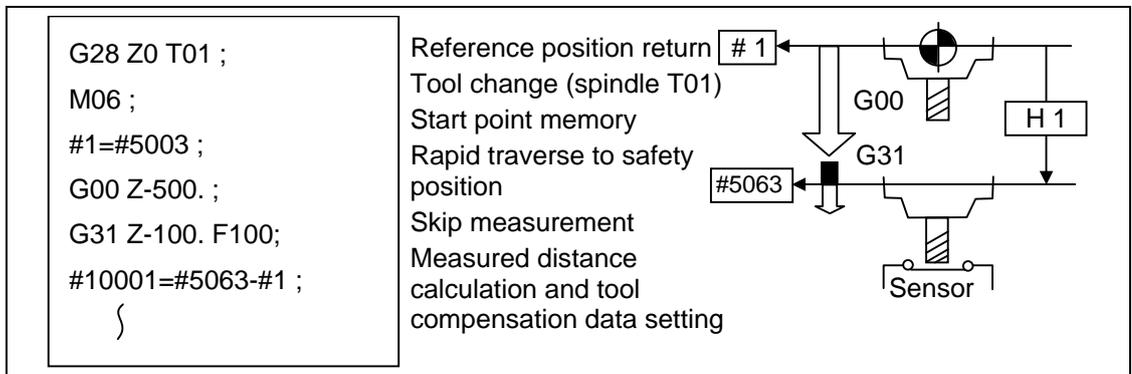
Numbers in the #10000 order have the equivalent function to that in the #2000 order, however, the maximum value of "n" for #2000 order is "200".

When the number of tool offset sets is larger than "200", use the variables of #10000 order.

The tool compensation data are configured as data with a decimal point in the same way as for other variables. Consequently, programming as "#10001 = 1000;" results the setting "1000.000" in the tool compensation data.



**(Example 1) Calculation and tool offset data setting**



**(Note)** In this example, no consideration is given to the delay in the skip sensor signal. #5003 is the Z axis start point position and #5063 is the Z axis skip coordinates, and indicated is the position at which the skip signal is input while G31 is being executed.



**Workpiece coordinate system offset**

By using variable numbers #5201 to #532n, it is possible to read out the workpiece coordinate system offset data or to substitute values.

**(Note)** The number of axes which can be controlled differs according to the specifications.  
The last digit of the variable No. corresponds to the control axis No.

Axis No. Axis name	Axis 1	Axis 2	Axis 3	Axis 4	..	Axis n	Remarks
External workpiece offset	#5201	#5202	#5203	#5204	..	#520n	External workpiece offset specifications are required.
G54	#5221	#5222	#5223	#5224	..	#522n	
G55	#5241	#5242	#5243	#5244	..	#524n	
G56	#5261	#5262	#5263	#5264	..	#526n	
G57	#5281	#5282	#5283	#5284	..	#528n	
G58	#5301	#5302	#5303	#5304	..	#530n	
G59	#5321	#5322	#5323	#5324	..	#532n	

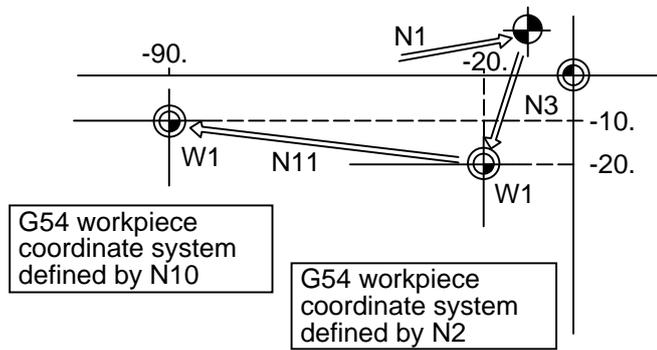
**(Example 1)**

```

N1 G28 X0 Y0 Z0 ;
N2 #5221=-20. #5222=-20. ;
N3 G90 G00 G54 X0 Y0 ;

N10 #5221=-90. #5222=-10. ;
N11 G90 G00 G54 X0 Y0 ;

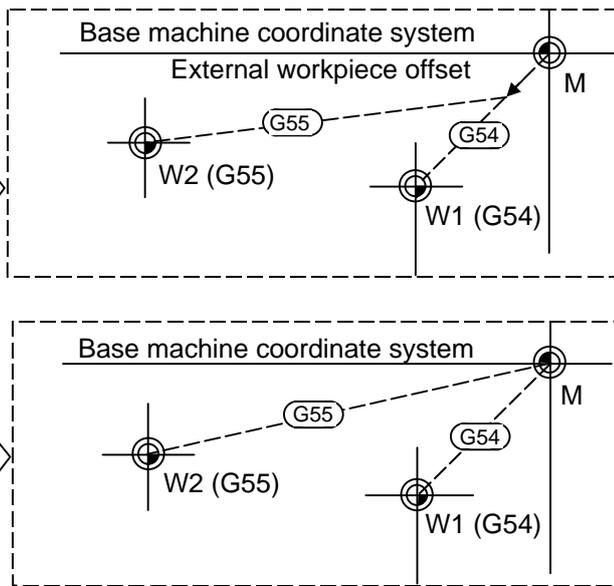
M02 ;
    
```



**(Example 2)**

```

N100 #5221=#5221+#5201 ;
      #5222=#5222+#5202 ;
      #5241=#5241+#5201 ;
      #5242=#5242+#5202 ;
      #5201=0 #5202=0;
    
```



This is an example where the external workpiece offset values are added to the workpiece coordinate (G54, G55) system offset values without changing the position of the workpiece coordinate systems.



## Alarm (#3000)

The NC system can be forcibly set to the alarm state by using variable number #3000.

Format

**#3000 = 70 (CALL#PROGRAMMER#TEL#530) :**  
 70 : Alarm number  
 CALL#PROGRAMMER#TEL#530 : Alarm message

Any alarm number from 1 to 9999 can be specified.

The alarm message must be less than 31 characters long.

"NC alarm 3 (program error)" signal will be output.

The "P277" user macro alarm message appears in the <alarm> column on diagnosis screen 1 while the alarm number and alarm message CALL #PROGRAMMER #TEL#530 is indicated in the <operator message>.

Example of program (alarm when #1 = 0)

```

§
IF [#1 NE 0] GOTO 100 ;
#3000=70
(CALL#PROGRAMMER#TEL#530) ;
N100
§
  
```

Stops with  
NC alarm

```

<Alarm>
P277 : Macro alarm message
<Operator message>
CALL#PROGRAMMER#TEL#530 70
  
```

**(Note 1)** Alarm number 0 is not displayed and any number exceeding 9999 cannot be indicated.

**(Note 2)** The characters following the first alphabet letter in the right member is treated as the alarm message. Therefore, a number cannot be designated as the first character of an alarm message. It is recommended that the alarm messages be enclosed in round parentheses.

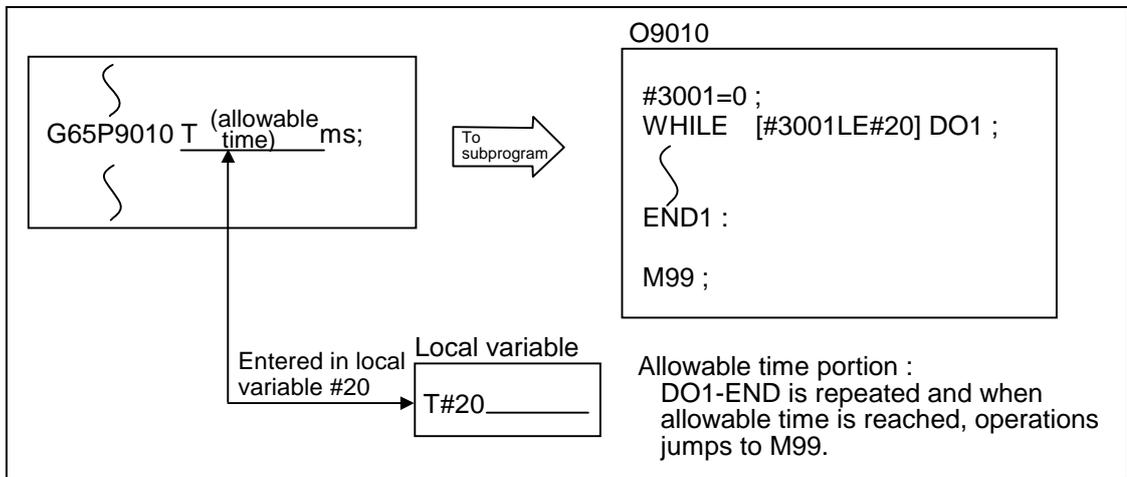


**Integrating (run-out) time (#3001, #3002)**

The integrating (run-out) time can be read during automatic operation or automatic start or values can be substituted by using variable numbers #3001 and #3002.

Type	Variable number	Unit	Contents when power is switched on	Initialization of contents	Count condition
Integrating (run-out) time 1	3001	ms	Same as when power is switched off	Value substituted for variable	At all times while power is ON
Integrating (run-out) time 2	3002				In-automatic start

The integrating run time returns to zero in about  $2.44 \times 10^{11}$  ms (approximately 7.7 years).

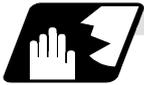


**Suppression of single block stop and miscellaneous function finish signal waiting**

By substituting the values below in variable number #3003, it is possible to suppress single block stop in the subsequent blocks or to advance to the next block without waiting for the miscellaneous function (M, S, T, B) finish (FIN) signal.

#3003	Single block stop	Miscellaneous function finish signal
0	Not suppressed	Awaited
1	Suppressed	Awaited
2	Not suppressed	Not awaited
3	Suppressed	Not awaited

**(Note 1)** #3003 is cleared to zero by NC reset.



### Feed hold, feedrate override, G09 valid/invalid

By substituting the values below in variable number #3004, it is possible to make the feed hold, feedrate override and G09 functions either valid or invalid in the subsequent blocks.

#3004 Contents (value)	Bit 0	Bit 1	Bit 2
	Feed hold	Feedrate override	G09 check
0	Valid	Valid	Valid
1	Invalid	Valid	Valid
2	Valid	Invalid	Valid
3	Invalid	Invalid	Valid
4	Valid	Valid	Invalid
5	Invalid	Valid	Invalid
6	Valid	Invalid	Invalid
7	Invalid	Invalid	Invalid

**(Note 1)** Variable number #3004 is set to zero by NC reset.

**(Note 2)** The functions are valid when the above bits are 0 and invalid when they are 1.

**(Note 3)** When the feed hold is set to invalid with #3004, the following will occur when the feed hold switch is pressed.

- During thread cutting, block stop will be carried out at the end of the next block of the block where thread cutting is completed.
- During tapping with tap cycle, block stop will be carried out after the operation of R point return.
- In the case other than above, block stop will be carried out after termination of the currently executing block.



### Message display and stop

By using variable number #3006, the execution is stopped after the previous block has been executed and, if message display data have been commanded, then the corresponding message will be indicated on the operator message area.

Format

#3006 = 1 ( TAKE FIVE ) :	
1	Fixed value (Nothing is displayed if a value other than 1 is set.)
TAKE FIVE	Message

The message should not be longer than 31 characters and it should be enclosed within round ( ) parentheses.



### Mirror image

By reading variable number #3007, it is possible to ascertain the status of mirror image at a particular point in time for each axis.

The axes correspond to the bits of #3007.

When the bits are 0, it means that the mirror image function is not valid; when they are 1, it means that it is valid.

#3007

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
nth axis											6	5	4	3	2	1



## G command modals

Using variable numbers #4001 to #4021, it is possible to read the G modal commands which have been issued up to the block immediately before.

Similarly, it is possible to read the modals in the block being executed with variable numbers #4201 to #4221.

Variable number		Function
Pre-read block	Execution block	
#4001	#4201	Interpolation mode : G00:0, G01:1, G02:2, G03:3, G33:33
#4002	#4202	Plane selection : G17:17, G18:18, G19:19
#4003	#4203	Absolute/incremental : G90:90, G91:91
#4004	#4204	No variable No.
#4005	#4205	Feed designation : G94:94, G95:95
#4006	#4206	Inch/metric : G20:20, G21:21
#4007	#4207	Tool nose R compensation : : G40:40, G41:41, G42:42
#4008	#4208	Tool length compensation : G43:43, G44:44, G49:49
#4009	#4209	Canned cycle : G80:80, G73 to 74, G76:76, G81 to G89:81 to 89
#4010	#4210	Return level : G98:98, G99:99
#4011	#4211	
#4012	#4212	Workpiece coordinate system : G54 to G59:54 to 59
#4013	#4213	Acceleration/deceleration : G61 to G64:61 to 64, G61.1:61.1
#4014	#4214	Macro modal call : G66:66, G66.1:66.1, G67:67
#4015	#4215	
#4016	#4216	
#4017	#4217	Constant surface speed control : G96:96, G97:97
#4018	#4218	No variable No.
#4019	#4219	Mirror image : G50.1:50.1, G51.1:51.1
#4020	#4220	
#4021	#4221	No variable No.

**(Example)**

```

G28 X0 Y0 Z0 ;
G90 G1 X100. F1000;
G91 G65 L300 X100. Y100.;
M02;
L300;
#1 = #4003;      → Group 3G modal (pre-read) #1 = 91.0
#2 = #4203;      → Group 3G modal (now being executed) #2 = 90.0
G#1 X#24 Y#25;
G68;
%
```



## Other modals

Using variable numbers #4101 to #4120, it is possible to read the modal commands assigned up to the block immediately before.  
Similarly, it is possible to read the modals in the block being executed with variable numbers #4301 to #4320.

Variable number		Modal information	Variable number		Modal information
Pre-read	Execution		Pre-read	Execution	
#4101	#4301		#4111	#4311	Tool length compensation No. H
#4102	#4302		#4112	#4312	
#4103	#4303		#4113	#4313	Miscellaneous function M
#4104	#4304		#4114	#4314	Sequence number N
#4105	#4305		#4115	#4315	Program number L(O) (Note 1)
#4106	#4306		#4116	#4316	
#4107	#4307	Tool radius compensation No. D	#4117	#4317	
#4108	#4308		#4118	#4318	
#4109	#4309	Feedrate F	#4119	#4319	Spindle function S
#4110	#4310		#4120	#4320	Tool function T

(Note 1) The programs are registered as files. When the program No. (file name) is read with #4115, #4315, the character string will be converted to a value.

(Example 1) The file name "123" is the character string 0x31, 0x32, 0x33, so the value will be  $(0x31-0x30)*100 + (0x32-0x30)*10 + (0x33-0x30) = 123.0$ . Note that if the file name contains characters other than numbers, it will be "blank".

(Example 2) If the file name is "123ABC", it contains characters other than numbers; so, the result will be "blank".



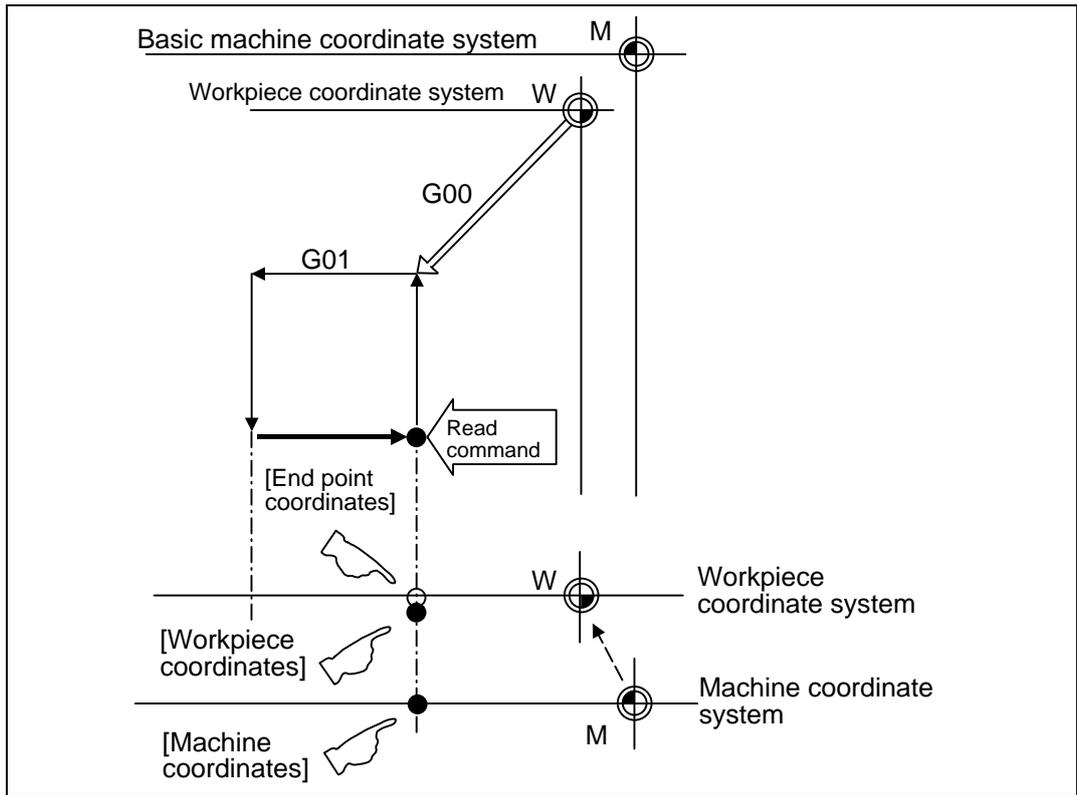
## Position information

Using variable numbers #5001 to #5104, it is possible to read the servo deviation amounts, tool position compensation amount, skip coordinates, workpiece coordinates, machine coordinates and end point coordinates in the block immediately before.

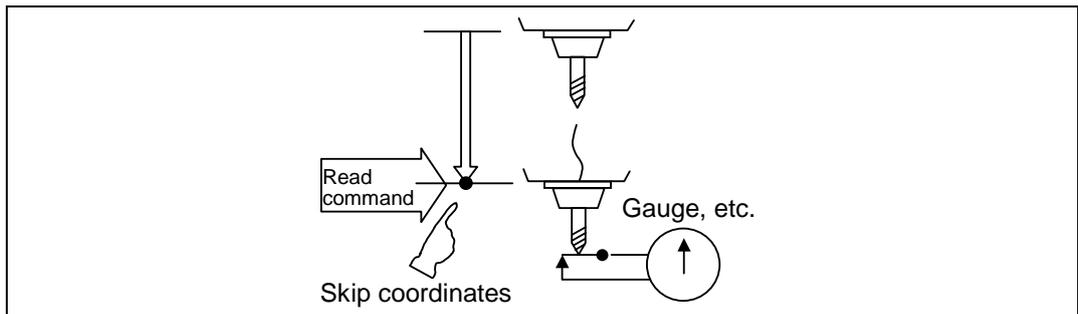
Axis No.	1	2	3	4	...	n	Remarks (reading during movement)
<b>Position information</b>							
<b>End point coordinate of block immediately before</b>	#5001	#5002	#5003	#5004	...	#5000+n	Yes
<b>Machine coordinate</b>	#5021	#5022	#5023	#5024	...	#5020+n	No
<b>Workpiece coordinate</b>	#5041	#5042	#5043	#5044	...	#5040+n	No
<b>Skip coordinate</b>	#5061	#5062	#5063	#5064	...	#5060+n	Yes
<b>Tool position compensation amount</b>	#5081	#5082	#5083	#5084	...	#5080+n	No
<b>Servo deviation amount</b>	#5101	#5102	#5103	#5104	...	#5100+n	Yes
<b>Start point coordinate of block with a macro interrupt</b>	#5121	#5122	#5123	#5124	...	#5120+n	Yes
<b>End point coordinate of block with a macro interrupt</b>	#5141	#5142	#5143	#5144	...	#5140+n	Yes

**(Note 1)** The number of axes which can be controlled differs according to the specifications.

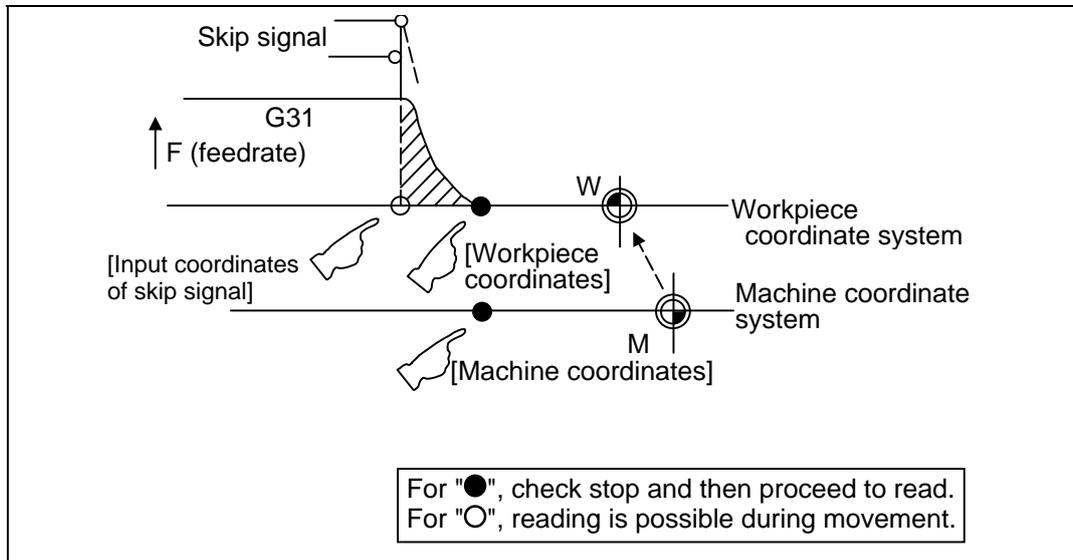
**(Note 2)** The last digit of the variable No. corresponds to the control axis No.



- (1) The positions of the end point coordinates and skip coordinates are positions in the workpiece coordinate system.
- (2) The end point coordinates, skip coordinates and servo deviation amounts can be read even during movement. However, it must first be checked that movement has stopped before reading the machine coordinates and the workpiece coordinates.
- (3) The position where the skip signal is turned ON in the G31 block is indicated for the skip coordinates. The end point position is indicated when the skip signal has not been turned ON. (For further details, refer to the section on tool length measurement.)



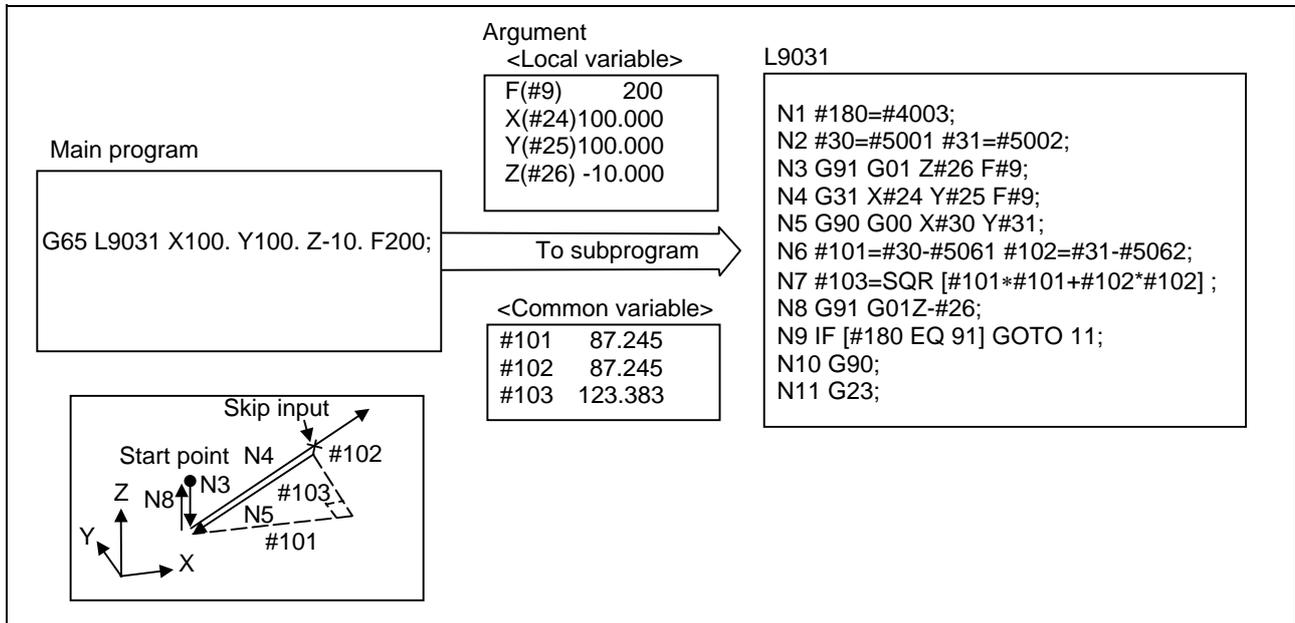
- (4) The tool nose position where the tool compensation and other such factors are not considered is indicated as the end point position. The tool reference position with consideration given to tool compensation is indicated for the machine coordinates, workpiece coordinates and skip coordinates.



The position of the skip signal input coordinates is the position in the workpiece coordinate system. The coordinates in variable numbers #5061 to #5064 memorize the moments when the skip input signal during movement was input and so they can be read at any subsequent time. For further details, reference should be made to the section on the skip function.

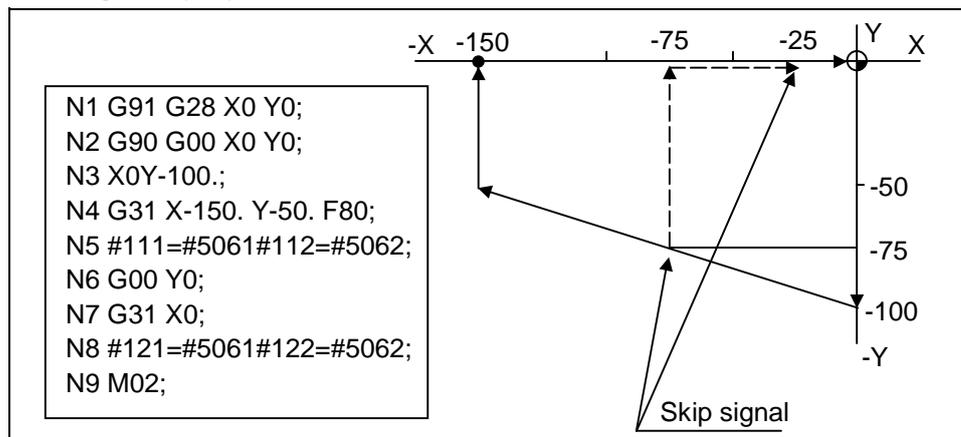
**(Example 1)** Example of workpiece position measurement

An example to measure the distance from the measured reference position to the workpiece edge is shown below.



- |       |                                   |         |  |
|-------|-----------------------------------|---------|--|
| #101  | X axis measurement amount         | N1      | G90/G91 modal recording                        |
| #102  | X axis measurement amount         | N2      | X, Y start point recording                     |
| #103  | Measurement linear segment amount | N3      | Z axis entry amount                            |
| #5001 | X axis measurement start point    | N4      | X, Y measurement (Stop at skip input)          |
| #5002 | Y axis measurement start point    | N5      | Return to X, Y start point                     |
|       |                                   | N6      | X, Y measurement incremental value calculation |
|       |                                   | N7      | Measurement linear segment calculation         |
| #5061 | X axis skip input point           | N8      | Z axis escape                                  |
| #5062 | Y axis skip input point           | N9, N10 | G90/G91 modal return                           |
|       |                                   | N11     | Subprogram return                              |

**(Example 2)** Reading of skip input coordinates



$$\begin{aligned} \#111 &= -75. + \epsilon & \#112 &= -75. + \epsilon \\ \#121 &= -25. + \epsilon & \#122 &= -75. + \epsilon \end{aligned}$$

$\epsilon$  is the error caused by response delay.

(Refer to the section on the skip function for details.)

#122 is the N4 skip signal input coordinates as there is no Y command at N7.



### Variable name setting and quotation

Any name (variable name) can be given to common variables #500 to #519. It must be composed of not more than 7 alphanumeric characters and it must begin with a letter. Do not use "#" in variable names. It causes an alarm when the program is executed.

Format

<b>SETVN n [ NAME1, NAME2, ..... ] :</b>	
n	: Head number of variable to be named
NAME1	: #n name (variable name)
NAME2	: #n + 1 name (variable name)

Variable names are separated by a comma (,).

Detailed description

- (1) Once variable names have been set, they will not be cleared even when the power is switched off.
- (2) Variables in programs can be quoted by their variable names. In cases like this, the variables should be enclosed in square parentheses.  
**(Example 1)** G01X [#POINT1];  
                   [#NUMBER] = 25 ;
- (3) The variable numbers, data and variable names appear on the screen of the setting and display unit.

#### (Example 2)

Program ..... SETVN500 [A234567, DIST, TOOL25] ;

[Common variables]		
#500	-12345.678	A234567
#501	5670.000	DIST
#502	-156.500	TOOL25
~~~~~		
#518	10.000	NUMBER
Common variable	#(502) Data (-156.5)	Name (TOOL25)

**(Note)** At the head of the variable name, do not use the characters determined by the NC for use in arithmetic commands, etc. (e.g. SIN, COS).



### Number of workpiece machining times

The number of workpiece machining times can be read using variables #3901 and #3902. By substituting a value in these variables, the number of workpiece machining times can be changed.

Type	Variable No.	Data setting range
Number of workpiece machining times	#3901	0 to 999999
Maximum workpiece value	#3902	

**(Note)** Always substitute a positive value for the number of workpiece machining times.



### Coordinate rotation parameter

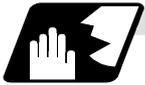
The following variables can be read by the system variables of the variable command. Note that writing is not possible onto these variables.

Variable No.	Description
#30060	Control axis No. on the coordinate rotation plane (horizontal axis)
#30061	Control axis No. on the coordinate rotation plane (vertical axis)
#30062	Coordinate rotation center (horizontal axis)
#30063	Coordinate rotation center (vertical axis)
#30064	Coordinate rotation angle
#30065	SIN data for the coordinate rotation angle [SIN(Coordinate rotation angle)]
#30066	COS data for the coordinate rotation angle [COS(Coordinate rotation angle)]
#30067	Coordinate rotation vector (horizontal axis)
#30068	Coordinate rotation vector (vertical axis)



### Reverse run information

Variable number	Usage	Description	Range
#31100	Number of available blocks for reverse run	+1 added number of the blocks that retained the reverse run information while the "Reverse run control mode" signal was ON	0 to 201
#31101	Counter of available blocks for reverse run	Number of available blocks for reverse run (value of #31100) when the "Reverse run" signal turned ON to start. Turns "0" when the forward run has been executed for all the blocks. Shows "0" in the normal operation.	0 to 201



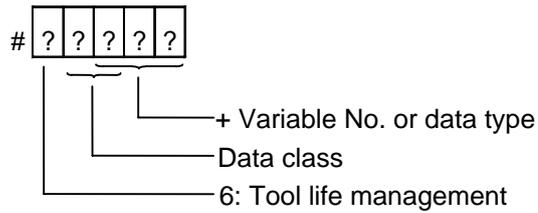
**Tool life management**

**(1) Definition of variable numbers**

- (a) Designation of group No.  
#60000

The tool life management data group No. to be read with #60001 to #64700 is designated by substituting a value in this variable. If a group No. is not designated, the data of the group registered first is read. This is valid until reset.

- (b) Tool life management system variable No. (Read)  
#60001 to #64700



- (c) Details of data classification

Data class	M System	L System	Remarks
00	For control	For control	Refer to following types
05	Group No.	Group No.	Refer to registration No.
10	Tool No.	Tool No.	Refer to registration No.
15	Tool data flag	Method	Refer to registration No.
20	Tool status	Status	Refer to registration No.
25	Life data	Life time/No. of times	Refer to registration No.
30	Usage data	Usage time/No. of times	Refer to registration No.
35	Tool length compensation data	-	Refer to registration No.
40	Tool radius compensation data	-	Refer to registration No.
45	Auxiliary data	-	Refer to registration No.

The group No., L System method, and life data are common for the group.

- (d) Registration No.

M system	1 to 200
L system	1 to 16

(e) Data type

Type	M System	L System	Remarks
1	Number of registered tools	Number of registered tools	
2	Life current value	Life current value	
3	Tool selected No.	Tool selected No.	
4	Number of remaining registered tools	Number of remaining registered tools	
5	Signal being executed	Signal being executed	
6	Cutting time cumulative value (minute)	Cutting time cumulative value (minute)	
7	Life end signal	Life end signal	
8	Life prediction signal	Life prediction signal	

Variable No.	Item	Type	Details	Data range
60001	Number of registered tools	Common to system	Total number of tools registered in each group.	0 to 200
60002	Life current value	For each group (Designate group No. #60000)	Usage time/No. of uses of tool being used. Spindle tool usage data or usage data for tool in use (#60003).	0 to 4000 minutes 0 to 9999 times
60003	Tool selected No.		Registration No. of tool being used. Spindle tool registration No. (If spindle tool is not data of the designated group, ST:1 first tool, or if ST:1 is not used, the first tool of ST:0. When all tools have reached their lives, the last tool.)	0 to 200
60004	Number of remaining registered tools		No. of first registered tool that has not reached its life.	0 to 200
60005	Signal being executed		"1" when this group is used in program being executed. "1" when spindle tool data group No. and designated group No. match.	0/1
60006	Cutting time cumulative value (minute)		Indicates the time that this group is used in the program being executed.	(Not used)
60007	Life end signal		"1" when lives of all tools in this group have been reached. "1" when all tools registered in designated group reach lives.	0/1
60008	Life prediction signal		"1" when new tool is selected with next command in this group. "1" when there are no tools in use (ST: 1) while there is an unused tool (ST: 0) in the specified group.	0/1

Variable No.	Item	Type	Details	Data range
60500 +***	Group No.	Each group/ registration No.	This group's No.	1 to 99999999
61000 +***	Tool No.	(Designate the group No. #60000 and registration No. ***.)  Note the group No., method and life are common for the groups.	Tool No.	1 to 99999999
61500 +***	Tool data flag		Usage data count method, length compensation method, radius compensation method, etc., parameters.  bit 0, 1 : Tool length compensation data format bit 2, 3 : Tool radius compensation data format 0: Compensation No. method 1: Incremental value compensation amount method 2: Absolute value compensation amount method bit 4, 5 : Tool life management method 0: Usage time 1: No. of mounts 2: No. of usages	0 to FF (H)
62000 +***	Tool status		Tool usage state 0: Not used tool 1: Tool being used 2: Normal life tool 3: Tool error 1 4: Tool error 2	0 to 4
62500 +***	Life data		Life time or No. of lives for each tool	0 to 4000 minutes 0 to 9999 times
63000 +***	Usage data		Usage time or No. of uses for each tool	0 to 4000 minutes 0 to 9999 times
63500 +***	Tool length compensation data		Length compensation data set as compensation No., absolute value compensation amount or increment value compensation amount method.	Compensation No.: 0 to No. of tool compensation sets  Absolute value compensation amount ±99999.999  Increment value compensation amount ±99999.999
64000 +***	Tool radius compensation data		Radius compensation data set as compensation No., absolute value compensation amount or increment value compensation amount method.	Compensation No.: 0 to No. of tool compensation sets  Absolute value compensation amount ±99999.999  Increment value compensation amount ±99999.999
64500 +***	Auxiliary data		Spare data	0 to 65535



## Example of program for tool life management

**(1) Normal commands**

```
#101 = #60001 ; ..... Reads the number of registered tools.
#102 = #60002 ; ..... Reads the life current value.
#103 = #60003 ; ..... Reads the tool selection No.
#60000 = 10 ; ..... Designates the group No. of the life data
                    to be read.
#104 = #60004 ; ..... Reads the remaining number of registered
                    tools in group 10.
#105 = #60005 ; ..... Reads the signal being executed in group 10.
#111 = #61001 ; ..... Reads the group 10, #1 tool No.
#112 = #62001 ; ..... Reads the group 10, #1 status.
#113 = #61002 ; ..... Reads the group 10, #2 status.
%
```

} Designated program No. is valid until reset.

**(2) When group No. is not designated.**

```
#104 = #60004 ; ..... Reads the remaining number of registered tools in the group
                    registered first.
#111 = #61001 ; ..... Reads the #1 tool No. in the group registered first.
%
```

**(3) When non-registered group No. is designated. (Group 9999 does not exist.)**

```
#60000 = 9999 ; ..... Designates the group No.
#104 = #60004 ; ..... #104 = -1.
```

**(4) When registration No. not used is designated. (Group 10 has 15 tools)**

```
#60000 = 10 ; ..... Designates the group No.
#111 = #61016 ; ..... #101 = -1.
```

**(5) When registration No. out of the specifications is designated.**

```
#60000 = 10 ;
#111 = #61017 ; ..... Program error (P241)
```

**(6) When tool life management data is registered with G10 command after group No. is designated.**

```
#60000 = 10 ; ..... Designates the group No.
G10 L3 ; ..... Starts the life management data registration.
P10 LLn NNn ; ..... 10 is the group No., Ln is the life per tool,
                    Nn is the method.
TTn ; ..... Tn is the tool No.
:
G11 ; ..... Registers the group 10 data with the G10
                    command.
#111 = #61001 ; ..... Reads the group 10, #1 tool No.
G10 L3 ; ..... Starts the life management data registration.
P1 LLn NNn ; ..... 1 is the group No., Ln is the life per tool,
                    Nn is the method.
TTn ; ..... Tn is the tool No.
:
G11 ; ..... Registers the life data with the G10
                    command.
                    (The registered data is deleted.)
#111 = 61001 ; ..... Group 10 does not exist. #201 = -1.
```

} The group 10 life data is registered.

} The life data other than group 10 is registered.

**Precautions for tool life management**

- (1) If the tool life management system variable is commanded without designating a group No., the data of the group registered at the head of the registered data will be read.
- (2) If a non-registered group No. is designated and the tool life management system variable is commanded, "-1" will be read as the data.
- (3) If an unused registration No. tool life management system variable is commanded, "-1" will be read as the data.
- (4) Once commanded, the group No. is valid until NC reset.



### Reading the parameters

System data can be read in with the system variables.

**(Note)** These can be used only with some models.

Variable No.	Application
#100000	Parameter # designation
#100001	Part system No. designation
#100002	Axis No./spindle No. designation
#100010	Parameter value read

The parameter values are read in with the following four blocks using these four system variables.

#100000 = 1001 ; ..... Designates the parameter #.

#100001 = 1 ; ..... Designates the part system No.

#100002 = 1 ; ..... Designates the axis No./spindle No.

#100 = #100010 ; ..... Reads the parameter value.

#### (1) Parameter # designation (#100000)

The parameter to be read in is designated by substituting the parameter # in this system variable.

If the parameters are read without designating this number, the parameters will be read in the same manner as if the minimum parameter # (#1) is designated. Once designated, the setting is held until the parameter # is designated again or until it is reset.

A program error (P39) will occur if a parameter # that does not exist is set.

#### (2) Part system No. designation (#100001)

Always set "0".

When using the PLC axis, set "10".

#### (3) Axis/spindle No. designation (#100002)

(a) System variable for axis/spindle No. designation

The axis No./spindle No. of the parameter to be read in is designated by substituting an index value in this system variable. This designation will be ignored when reading in parameters that are not for a specific axis or spindle.

If the parameters are read without designating this number, the parameters will be read in the same manner as if the index value 1 (1st axis/1st spindle) is designated. Once designated, the setting is held until the index value is designated again or until it is reset. A program error (P39) will occur if an axis/spindle No. that does not exist is set.

(b) Index values

Index values	Axis parameter	Spindle parameter
1	1st axis	1st spindle
2	2nd axis	2nd spindle
3	3rd axis	3rd spindle
4	4th axis	4th spindle
5	5th axis	-
6	6th axis	-

**(4) Parameter read (#100010)**

The designated parameter data is read with this system variable.  
The following data is read according to the parameter type.

Type	Read data
Value	The values displayed on the Parameter screen are output.
Text	ASCII codes are converted into decimal values.

**Example of programs for reading parameters****(1) To read the parameter [#1002 axisno (Number of axes)]**

```
#100000 = 1002 ; .....Designates [#1002].
#100001 = 0 ;
#101 = #100010 ; .....Reads the number of axes.
#100001 = 5 ; .....Designates [5th part system]. (Program error (P39) occurs.)
#100001 = 10 ; .....Designates [PLC axis].
#110 = #100010 ; .....Reads the number of PLC axes.
```

**(2) To read the parameter [#2037 G53ofs (#1 reference position)]**

[Conditions]

```
#2037 G53ofs      <1st axis>  <2nd axis>
                  1000.000  200.000
```

```
#100002 = 1 ; .....Designates [1st axis].
#100000 = 2037 ; .....Designates [#2037].
#101 = #100010 ; .....Reads the [#1 reference position] for the 1st axis.
                      (#101 = 100.000.)
#100002 = 2 ; .....Designates [2nd axis].
#102 = #100010 ; .....Reads the [#1 reference position] for the 2nd axis.
                      (#102 = 200.000.)
```

**(3) To read each parameter for each axis and spindle**

```
#100002 = 1 ; .....Designates [1st spindle].
#100000 = 3001 ; .....Designates [#3001].
#101 = #100010 ; .....Reads the [#3001 slimt1 (Number of limit rotation gears 00)]
                      for 1st spindle.
#100000 = 3002 ; .....Designates [#3002].
#102 = #100010 ; .....Reads the [#3002 slimt2 (Number of limit rotation gears 01)]
                      for 1st spindle.
#100002 = 2 ; .....Designates [2nd spindle].
#100000 = 3001 ; .....Designates [#3001].
#201 = #100010 ; .....Reads the [#3001 slimt1 (Number of limit rotation gears 00)]
                      for 2nd spindle.
#100000 = 3002 ; .....Designates [#3002].
#202 = #100010 ; .....Reads the [#3002 slimt2 (Number of limit rotation gears 01)]
                      for 2nd spindle
```



### Example of parameter read macro program

#### <Macro specifications>

Q341 A_ Q_ ;		
A_.....	Storage common variable	Designates the common variable No. for storing the data read in.
Q_.....	Parameter # designation	For an axis/spindle parameter, designates the axis/spindle No. with one digit after the decimal point.

#### <Macro contents>

#100000 = FIX [#17] ;	.....	Designates parameter #.
#100002 = FIX [#17*10] MOD 10 ;	.....	Designates axis/spindle No.
#[#1] = #100010 ;	.....	Reads parameter data.
M99 ;		



### Precautions for reading parameters

- (1) The number of axes and spindles is the maximum number in the specifications determined by the model.
- (2) The inch/metric changeover function for the setting and display is valid even for the data read in.



**Reading PLC data**

PLC data can be read in with the system variables.

**(Note 1)** These can be used only with some models.

**(Note 2)** The read devices are limited.

Variable No.	Application
#100100	Device type designation
#100101	Device No. designation
#100102	Number of read bytes designation
#100103	Read bit designation
#100110	PLC data read

The PLC data is read in with the following five blocks using these five system variables.

#100100 = 1 ; .....Designates the device type.

#100101 = 0 ; .....Designates the device No.

#100102 = 1 ; .....Designates the number of bytes.

#100103 = 2 ; .....Designates the bit. (Valid only when reading word device bits.)

#100 = #100110 ; .....Reads in the PLC data.

**(1) Device designation (#100100)**

(a) System variable for device designation

The type of device to be read in can be designated by substituting the device designation value in this system variable.

If the data is read without designating this number, the data will be read in the same manner as if the minimum device designation value (0: M device) is designated. Once designated, the setting is held until the device is designated again or until it is reset.

A program error (P39) will occur if a device that does not exist is set.

(b) Device designation values

Device designation value	Device			Device designation value	Device		
	Device	Unit	Device No.		Device	Unit	Device No.
0	M	Bit	M0 to M10239	10	F	Bit	F0 to F1023
1	D	Word	D0 to D2047	13	L	Bit	L0 to L511
2	C	Bit	C0 to C255	18	V	Bit	V0 to V255
4	X *	Bit	X0 to X1FFF	19	ST	Bit	ST0 to ST63
5	Y *	Bit	Y0 to Y1FFF	20	SD	Word	SD0 to SD127
6	R	Word	R0 to R13311	21	SB*	Bit	SB0 to SB1FF
7	T	Bit	T0 to T703	22	SW*	Word	SW0 to SW1FF
9	SM	Bit	SM0 to SM127	23	B*	Bit	B0 to B1FFF
				24	W*	Word	W0 to W1FFF

The unit is the amount of data per device No. "Word" has 16 bits, and "Bit" has one bit.

\* is a device that expresses the device No. as a hexadecimal.

**(2) Device No. designation (#100101)**

The device to be read in is designated by substituting the device No. in this system variable. Convert a device expressed as a hexadecimal into a decimal when designating.

If the data is read without designating this number, the data will be read in the same manner as if the minimum device No. (0) is designated. Once designated, the setting is held until the device No. is designated again or until it is reset.

A program error (P39) will occur if a device No. that does not exist is set.

**(3) Number of bytes designation (#100102)****(a) System variable for number of bytes designation**

The reading size is designated by substituting the number of bytes designation value in this system variable.

If the data is read without designating this number, the data will be read in the same manner as if the minimum number of bytes designation value (0: bit designation) is designated. Once designated, the setting is held until the number of bytes is designated again or until it is reset.

A program error (P39) will occur if a number of bytes that does not exist in the specifications is set.

**(b) Number of bytes designation value**

Number of bytes designation value	Read in data			Operation	
	Size	Sign	Range	Word device	Bit device
0	1 bit	-	0 to 1	The number of bits designated is read in.	The bits for the designated device No. are read in.
1	1 byte	No	0 to 255	The low-order byte is read in.	8 bits are read in from the designated device No.
101		Yes	-128 to 127		
2	2 bytes	No	0 to 65535	Two bytes are read in.	16 bits are read in from the designated device No.
102		Yes	-32768 to 32767		
4	4 bytes	No	0 to 4294967295	The designated device (L) and next device (H) are read in.	32 bits are read in from the designated device No.
104		Yes	-2147483648 to 2147483647		

0 to 4 are designated without a sign, and 101 to 104 are designated with a sign.

**(4) Bit designation (#100103)**

## (a) System variable for bit designation

The bit to be read in is designated by substituting the bit designation value in this system variable.

This designation is valid only when reading the bits for a 16-bit device, and is invalid in all other cases.

If the data is read without designating this number, the data will be read in the same manner as if the minimum bit designation value (0: bit 0) is designated. Once designated, the setting is held until the bit is designated again or until it is reset.

A program error (P39) will occur if a bit that does not exist is set.

## (b) Bit designation values

Bit designation value	Read in bit
0	Bit 0
1	Bit 1
:	:
15	Bit 15

**(5) PLC data read (#100110)**

The data for the designated device is read in with this system variable.

Refer to the table for number of bytes designation for details on the range of data read in.



## Examples of programs for reading PLC data

**(1) To read a bit device**

#100100 = 0 ; ..... Designates [M device].  
 #100101 = 0 ; ..... Designates [Device No. 0].  
 #100102 = 0 ; ..... Designates [Bit].  
 #100 = #100110 ; ..... Reads M0 (one bit).  
 #100102 = 1 ; ..... Designates [1 byte].  
 #101 = #100110 ; ..... Reads M0 to M7 (eight bits).  
 (If M7 to M0 is 0001 0010, this will be #102 = 18 (0x12).)  
 #100102 = 102 ; ..... Designates [Signed two bytes].  
 #102 = #100110 ; ..... Reads M0 to M15 (16 bits).  
 (If M15 to M0 is 1111 1110 1101 1100, this will be  
 #102 = -292 (0xFEDC).)  
 #100102 = 4 ; ..... Designates [4 bytes].  
 #104 = #100110 ; ..... Reads M0 to M31 (32 bits).  
 (If M31 to M0 is 0001 0010 0011 0100 0101 0110 0111 1000,  
 this will be #104 = 305419896 (0x12345678).)

**(2) To read a word device**

#100100 = 1 ; ..... Designates [D device].  
 #100101 = 0 ; ..... Designates [Device No. 0].  
 #100102 = 0 ; ..... Designates [Bit].  
 #100103 = 1 ; ..... Designates [Bit 1].  
 #100 = #100110 ; ..... Reads the D0 bit 1.  
 (If D0 = 0x0102, this will be #101 =1.)  
 #100102 = 1 ; ..... Designates [1 byte].  
 #101 = #100110 ; ..... Reads the low-order byte of D0.  
 (If D0 = 0x0102, this will be #101 =2.)  
 #100102 = 2 ; ..... Designates [2 bytes].  
 #102 = #100110 ; ..... Reads D0. (If D0 = 0x0102, this will be #102 =258.)  
 #100102 = 104 ; ..... Designates [Signed four bytes].  
 #104 = #100110 ; ..... Reads D0 and D1.  
 (If D0 = 0xFFFFE and D1 = 0xFFFF, this will be #104 =-2.)



Examples of using macro program for reading PLC data

<Macro specifications>

G340 F\_ A\_ Q\_ H\_ ;

F\_ .....Number of bytes designation

A\_ .....Device designation

Q\_ .....Device No. designation

H\_ ..... Storage common variable

F0..... Designates bit.

F1..... Designates one byte.

F2..... Designates two bytes.

A0 ..... Designates M.

A1 ..... Designates D.

A2 ..... Designates C.

A3 ..... Designates G.

A4 ..... Designates X.

A5 ..... Designates Y.

A6 ..... Designates R.

A7 ..... Designates T.

For a bit, designates the bit with two digits after the decimal point.

For a byte, a decimal value is not designated.

Designates the common variable No. for storing the read data.

<Macro contents>

#100100 = #1 ;

#100101 = FIX [#17] ;

#100102 = #9 ;

#100103 = FIX [#17\*100] MOD 100 ;

#[#11] = #100110 ;

M99 ;

..... Designates device.

..... Designates device No.

..... Designates number of bytes.

..... Designates bit.

..... Reads PLC data.



Precautions for reading PLC data

- (1) As the PLC data is read asynchronously from the ladder execution, the data is not necessarily from the running program. Take care when reading devices that change.
- (2) If reading of a device that does not exist is attempted by designating the device No. and number of bytes, the 0 value will be read in only for the sections that do not exist.



## Time reading variables

The following operations can be carried out using the system variable extension for the user macro time.

- (1) By adding time information system variable #3011 and #3012, the current date (#3011) and current time (#3012) can be read and written.
- (2) By adding parameter #1273/bit1, the unit (millisecond unit/hour unit) of the cumulative time during system variable #3002 automatic start can be changed.

Variable No.	Details
#3001	The cumulative time during power ON can be read and the value can be substituted. The unit is a millisecond unit.
#3002	The cumulative time during automatic start can be read and the value can be substituted. The unit can be changed between millisecond and hour with parameter #1273/bit1.
#3011	The current date can be read and written. YYYY/MM/DD is read as a YYYYMMDD value. When a YYYYMMDD value is written in, it will be set as YY/MM/DD (the last two digits of the year are displayed). Command range for year/month/date setting Year (YYYY) : 2000 to 2099 Month (MM) : 1 to 12 Date (DD) : 1 to maximum number of days in month
#3012	The current time can be read and written. HH/MM/SS is read as a HHMMSS value. When an HHMMSS value is written in, it will be set as HH/MM/DD. Command range for hour/minute/second setting Hour (HH) : 0 to 23 (24-hour system) Minute (MM) : 0 to 59 Second (SS) : 0 to 59

- (3) The cumulative time returns to 0 at approx.  $2.44 \times 10^{11}$  milliseconds (approx. 7.7 years).
- (4) If a negative value or a value exceeding 244335917226 milliseconds (67871.08811851 hours for #3002 time designation) is set for the cumulative time, a program error (P35) will occur.
- (5) If a value exceeding the command range is set for the date or time, a program error (P35) will occur.
- (6) Always set the month/date/hour/minute/second as a two-digit value when setting the date and time.  
If the value only has one digit, always add 0. (February 14, 2001  $\Rightarrow$  #3001= 20010214 ;, etc.)



### Examples of using time reading variable

- (Example 1)** To read the current date (February 14, 2001) in common variable #100  
 #100 = #3011 ; (20010214 is inserted in #100)
- (Example 2)** To write current time (18 hours, 13 minutes, 6 seconds) into system variable #3012  
 #3012 = 181306 ; (The command value cumulative time #2: time is set to 18:13:06.)
- (Example 3)** By setting the following type of program, the machining start/end time (year/month/date/hour/minute/second) can be viewed.

```
#100=#3011 ; ⇒ Machining start year/month/date
#101=#3012 ; ⇒ Machining start hour/minute/second
G28 X0 Y0 Z0 ;
G92 ;
G0 X50. ;
.
.
.
#102=#3011 ; ⇒ Machining end year/month/date
#103=#3012 ; ⇒ Machining end hour/minute/second
M30 ;
```



### Restrictions and precautions for using time reading variable

- (1) #3011 reads the date as an eight-digit value, so the difference of the two dates read in will not be the difference of days.
- (2) #3012 reads the time as a six-digit value, so the difference of the two times read in will not be the difference of hours.

13.5.6 Arithmetic Commands

A variety of arithmetic operations can be performed between variables.



Command format

**#i = <formula>**

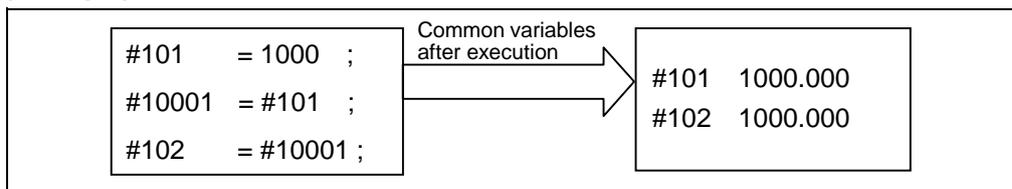
<Formula> is a combination of constants, variables, functions and operators. Constants can be used instead of #j and #k below.

(1) Definition and substitution of variables	#i = #j	Definition, substitution
(2) Addition arithmetic	#i = #j + #k	Addition
	#i = #j - #k	Subtraction
	#i = #j OR #k	Logical sum (at every bit of 32 bits)
	#i = #j XOR #k	Exclusive OR (at every bit of 32 bits)
(3) Multiplication arithmetic	#i = #j * #k	Multiplication
	#i = #j / #k	Division
	#i = #j MOD #k	Remainder
	#i = #j AND #k	Logical product (at every bit of 32 bits)
(4) Functions	#i = SIN [#k]	Sine
	#i = COS [#k]	Cosine
	#i = TAN [#k]	Tangent (sin/cos used for tan)
	#i = ASIN [#k]	Arcsine
	#i = ATAN [#j]	Arctangent (ATAN or ATN may be used)
	#i = ACOS [#j]	Arc-cosine
	#i = SQRT [#k]	Square root (SQRT or SQR may be used)
	#i = ABS [#k]	Absolute value
	#i = BIN [#k]	Conversion from BCD to BIN
	#i = BCD [#k]	Conversion from BIN to BCD
	#i = ROUND [#k]	Rounding off (ROUND or RND may be used)
	#i = FIX [#k]	Discard fractions less than 1
#i = FUP [#k]	Add for fractions less than 1	
#i = LN [#k]	Natural logarithm	
#i = EXP [#k]	Exponent with e (=2.718 ..... ) as bottom	

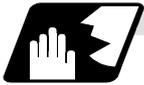
**(Note 1)** A value without a decimal point is basically treated as a value with a decimal point at the end (1 = 1.000).

**(Note 2)** Compensation amounts from #10001 and workpiece coordinate system offset values from #5201 are handled as data with a decimal point. Consequently, data with a decimal point will be produced even when data without a decimal point have been defined in the variable numbers.

**(Example)**



**(Note 3)** The <formula> after a function must be enclosed in the square parentheses.



**Sequence of arithmetic operations**

- (1) The sequence of the arithmetic operations (1) through (3) is, respectively, the functions followed by the multiplication arithmetic followed in turn by the addition arithmetic.

$$\#101 = \#111 + \#112 * \text{SIN}[\#113]$$

(1) Function  
(2) Multiplication arithmetic  
(3) Addition arithmetic

- (2) The part to be given priority in the operation sequence should be enclosed in square parentheses. Up to 5 pairs of such parentheses including those for the functions may be used.

$$\#101 = \text{SQRT} [ [ [ \#111 = \#112 ] * \text{SIN}[\#113] + \#114 ] * \#15 ] ;$$

First pair of parentheses  
Second pair of parentheses  
Third pair of parentheses



**Examples of arithmetic commands**

(1) Main program and argument designation	G65 P100 A10 B20. ; #101=100.000 #102=200.000 ;	#1 10.000 #2 20.000 #101 100.000 #102 200.000	
(2) Definition and substitution (=)	#1=1000 #2=1000. #3=#101 #4=#102 #5=#10001 (#10001=-10.)	#1 1000.000 #2 1000.000 #3 100.000 #4 200.000 #5 -10.000 ...	} From common variables ... From compensation amount
(3) Addition and subtraction (+, -)	#11=#1+1000 #12=#2-50. #13=#101+#1 #14=#10001-3. (#10001 = -10.) #15=#10001+#102	#11 2000.000 #12 950.000 #13 1100.000 #14 -13.000 #15 190.000	
(4) Multiplication and division (*, /)	#21=100*100 #22=100.*100 #23=100*100 #24=100.*100. #25=100/100 #26=100./100 #27=100/100. #28=100./100. #29=#10001*#101 (#10001 = -10.) #30=#10001/#102	#21 10000.000 #22 10000.000 #23 10000.000 #24 10000.000 #25 1.000 #26 1.000 #27 1.000 #28 1.000 #29 -1000.000 #30 -0.050	
(5) Remainder (MOD)	#19=48 #20=9 #31=#19 MOD #20	#19/#20 = 48/9 = 5 with 3 over  #31 = 3	

(6)	Logical sum (OR)	#3=100 #4=#3 OR 14	#3 = 01100100 (binary) 14 = 00001110 (binary) #4 = 01101110 = 110
(7)	Exclusive OR (XOR)	#3=100 #4=#3 XOR 14	#3 = 01100100 (binary) 14 = 00001110 (binary) #4 = 01101010 = 106
(8)	Logical product (AND)	#9=100 #10=#9 AND 15	#9 = 01100100 (binary) 15 = 00001111 (binary) #10 = 00000100 = 4
(9)	Sin (SIN)	#501 = SIN [60] #502 = SIN [60.] #503 = 1000*SIN [60] #504 = 1000*SIN [60.] #505 = 1000.*SIN [60] #506 = 1000.*SIN [60.] <b>(Note)</b> SIN [60] is equivalent to SIN [60.]	#501 0.866 #502 0.866 #503 866.025 #504 866.025 #505 866.025 #506 866.025
(10)	Cosine (COS)	#541 = COS [45] #542 = COS [45.] #543 = 1000*COS [45] #544 = 1000*COS [45.] #545 = 1000.*COS [45] #546 = 1000.*COS [45.] <b>(Note)</b> COS [45] is equivalent to COS [45.]	#541 0.707 #542 0.707 #543 707.107 #544 707.107 #545 707.107 #546 707.107
(11)	Tangent (TAN)	#551 = TAN [60] #552 = TAN [60.] #553 = 1000*TAN [60] #554 = 1000*TAN [60.] #555 = 1000.*TAN [60] #556 = 1000.*TAN [60.] <b>(Note)</b> TAN [60] is equivalent to TAN [60.]	#551 1.732 #552 1.732 #553 1732.051 #554 1732.051 #555 1732.051 #556 1732.051
(12)	Arcsine ASIN	#531 = ASIN [100.500/201.] #532 = ASIN [100.500/201] #533 = ASIN [0.500] #534 = ASIN [-0.500]	#531 30.000 #432 30.000 #533 30.000 #534 -30.000 <b>(Note)</b> When #1273/bit 0 is set to 1, #534 will be 330°.
(13)	Arctangent (ATAN or ATN)	#561 = ATAN [173205/100000] #562 = ATAN [173205/100000.] #563 = ATAN [173.205/100] #564 = ATAN [173.205/100.] #565 = ATAN [1.73205]	#561 60.000 #562 60.000 #563 60.000 #564 60.000 #565 60.000

(14)	Arccosine (ACOS)	#521 = ACOS [100./141.421] #522 = ACOS [100./141.421]	#521 45.000 #522 45.000
(15)	Square root (SQR or SQRT)	#571 = SQRT [1000] #572 = SQRT [1000.] #573 = SQRT [10. *10. +20. *20.] <b>(Note)</b> In order to increase the accuracy, proceed with the operation inside parentheses.	#571 31.623 #572 31.623 #573 22.360
(16)	Absolute value (ABS)	#576 = -1000 #577 = ABS [#576] #3 = 70. #4 = -50. #580 = ABS [#4 - #3]	#576 -1000.000 #577 1000.000 #580 120.000
(17)	BIN, BCD	#1 = 100 #11 = BIN [#1] #12 = BCD [#1]	#11 64 #12 256
(18)	Rounding off (ROUND or RND)	#21 = ROUND [14/3] #22 = ROUND [14./3] #23 = ROUND [14/3.] #24 = ROUND [14./3.] #25 = ROUND [-14/3] #26 = ROUND [-14./3] #27 = ROUND [-14/3.] #28 = ROUND [-14./3.]	#21 5 #22 5 #23 5 #24 5 #25 -5 #26 -5 #27 -5 #28 -5
(19)	Discarding fractions below decimal point (FIX)	#21 = FIX [14/3] #22 = FIX [14./3] #23 = FIX [14/3.] #24 = FIX [14./3.] #25 = FIX [-14/3] #26 = FIX [-14./3] #27 = FIX [-14/3.] #28 = FIX [-14./3.]	#21 4.000 #22 4.000 #23 4.000 #24 4.000 #25 -4.000 #26 -4.000 #27 -4.000 #28 -4.000
(20)	Adding fractions less than 1 (FUP)	#21 = FUP [14/3] #22 = FUP [14./3] #23 = FUP [14/3.] #24 = FUP [14./3.] #25 = FUP [-14/3] #26 = FUP [-14./3] #27 = FUP [-14/3.] #28 = FUP [-14./3.]	#21 5.000 #22 5.000 #23 5.000 #24 5.000 #25 -5.000 #26 -5.000 #27 -5.000 #28 -5.000
(21)	Natural logarithms (LN)	#101 = LN [5] #102 = LN [0.5] #103 = LN [-5]	#101 1.609 #102 -0.693 Error "P282"
(22)	Exponents (EXP)	#104 = EXP [2] #105 = EXP [1] #106 = EXP [-2]	#104 7.389 #105 2.718 #106 0.135



### Arithmetic accuracy

As shown in the following table, errors will be generated when performing arithmetic operations once and these errors will accumulate by repeating the operations.

Arithmetic format	Average error	Maximum error	Type of error
a = b + c a = b - c	$2.33 \times 10^{-10}$	$5.32 \times 10^{-10}$	Min. $ \varepsilon/b $ , $ \varepsilon/c $
a = b*c	$1.55 \times 10^{-10}$	$4.66 \times 10^{-10}$	Relative error $ \varepsilon/a $
a = b/c	$4.66 \times 10^{-10}$	$1.86 \times 10^{-9}$	
a = $\sqrt{b}$	$1.24 \times 10^{-9}$	$3.73 \times 10^{-9}$	
a = SIN [b] a = COS [b]	$5.0 \times 10^{-9}$	$1.0 \times 10^{-8}$	Absolute error $ \varepsilon ^\circ$
a = ATAN [b/c]	$1.8 \times 10^{-6}$	$3.6 \times 10^{-6}$	

(Note) SIN/COS is calculated for the function TAN.



### Notes on reduced accuracy

#### (1) Addition and subtraction

It should be noted that when absolute values are used subtractively in addition or subtraction, the relative error cannot be kept below  $10^{-8}$ .

For instance, it is assumed that the real values produced as the arithmetic calculation result of #10 and #20 are as follows (these values cannot be substituted directly) :

#10 = 2345678988888.888

#20 = 2345678901234.567

Performing #10 - #20 will not produced #10 - 320 = 87654.321. There are 8 significant digits in the variables and so the values of #10 and #20 will be as follows (strictly speaking, the internal values will differ somewhat from the values below because they are binary numbers) :

#10 = 2345679000000.000

#20 = 2345678900000.000

Consequently, #10 - #20 = 100000.000 will generate a large error.

#### (2) Logical operations

EQ, NE, GT, LT, GE and LE are basically the same as addition and subtraction and so care should be taken with errors. For instance, to determine whether or not #10 and #20 are equal in the above example :

IF [#10EQ#20]

It is not always possible to provide proper evaluation because of the above mentioned error. Therefore, when the error is evaluated as in the following expression :

IF [ABS [#10 - #20] LT200000]

and the difference between #10 and #20 falls within the designated range error, both values should be considered equal.

#### (3) Trigonometric functions

Absolute errors are guaranteed with trigonometric functions but since the relative error is not under  $10^{-8}$ , care should be taken when dividing or multiplying after having used a trigonometric function.

13.5.7 Control Commands

The flow of programs can be controlled by IF-GOTO- and WHILE-DO-.



**Branching**

Format

**IF [conditional expression] GOTO n; (n = sequence number in the program)**

When the condition is satisfied, control branches to "n" and when it is not satisfied, the next block is executed.

IF [conditional expression] can be omitted and, when it is, control passes to "n" unconditionally. The following types of [conditional expressions] are available.

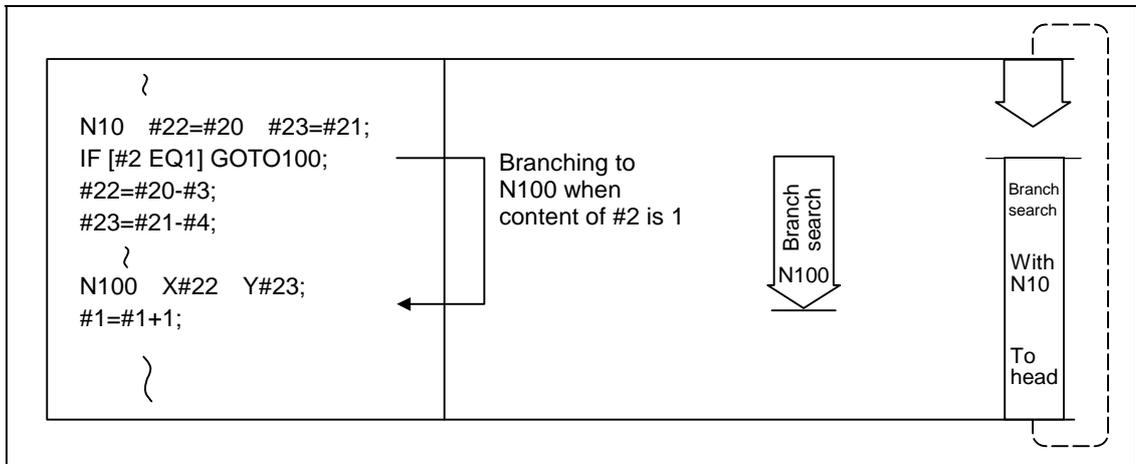
#i EQ #j	=	When #i and #j are equal
#i NE #j	≠	When #i and #j are not equal
#i GT #j	>	When #i is greater than #j
#i LT #j	<	When #i is less than #j
#i GE #j	≥	When #i is #j or more
#i LE #j	≤	When #i is #j or less

"n" of GOTO n must always be in the same program. Program error (P231) will result if it is not. A formula or variable can be used instead of #i, #j and "n".

In the block with sequence number "n" which will be executed after a GOTO n command, the sequence number must always be at the head of the block.

Otherwise, program error (P231) will result.

If "/" is at the head of the block and Nn follows, control can be branched to the sequence number.



**(Note 1)** When the sequence number of the branch destination is searched, the search is conducted up to the end of the program (% code) from the block following IF.....; and if it is not found, it is then conducted from the top of the program to the block before IF.....;. Therefore, branch searches in the opposite direction to the program flow will take longer to execute compared with branch searches in the forward direction.

**(Note 2)** EQ and NE should be used only for integers. For comparison of numeric values with decimals, GE, GT, LE, and LT should be used.



Iteration

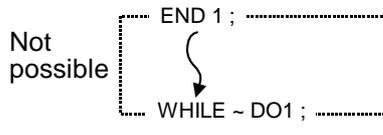
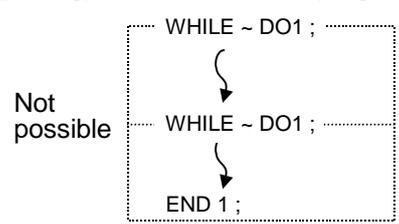
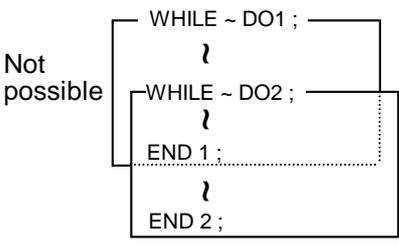
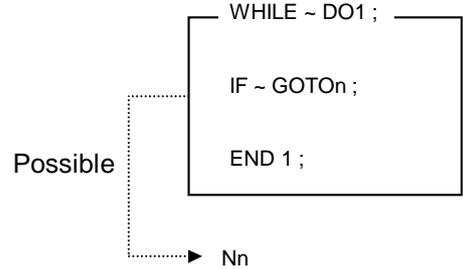
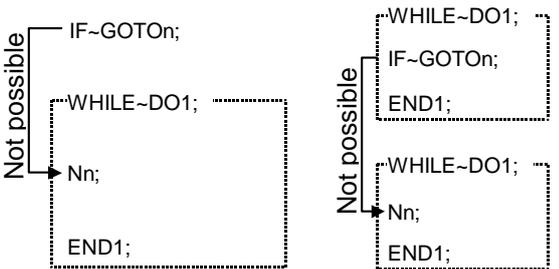
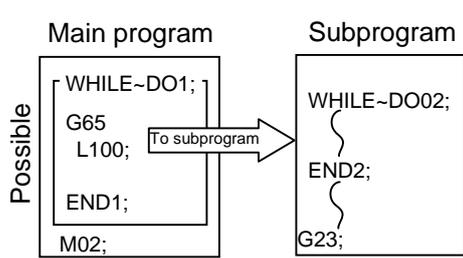
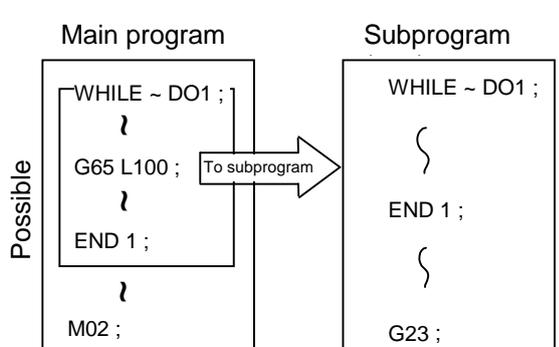
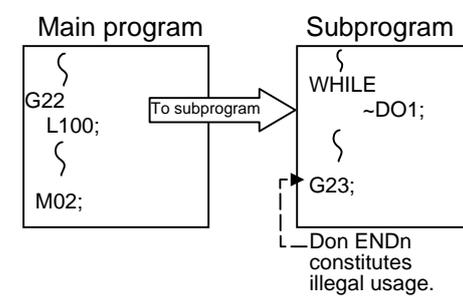
Format

```

WHILE [conditional expression] DOm ; (m = 1, 2, 3 ..... 127)
    {
END m ;
    
```

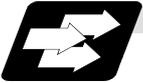
While the conditional expression is established, the blocks from the following block to ENDM are repeatedly executed; when it is not established, execution moves to the block after ENDM. DOM may come before WHILE, WHILE [conditional expression] DOM and ENDM must be used as a pair. IF WHILE [conditional expression] is omitted, these blocks will be repeatedly ad infinitum. The repeating identification numbers range from 1 through 127 (DO1, DO2, DO3, ..... DO127). Up to 27 nesting levels can be used.

<p>(1) Same identifier number can be used any number of times.</p> <p>Possible</p> <pre>         WHILE ~ DO1 ;         END1 ;     </pre> <p>Possible</p> <pre>         WHILE ~ DO1 ;         END1 ;     </pre>	<p>(2) Any number may be used for the WHILE –DOM identifier number.</p> <pre>         WHILE ~ DO1 ;         {         END1 ;         }         WHILE ~ DO3 ;         {         END3 ;         }     </pre> <p>Possible</p> <pre>         {         WHILE ~ DO2 ;         {         END2 ;         }         }         WHILE ~ DO1 ;         {         END1 ;         }     </pre>
<p>(3) Up to 27 nesting levels for WHILE– DOM. "m" is any number from 1 to 127 for the nesting depth.</p> <p>Possible</p> <p>(Note) :With nesting, "m" which has been used once cannot be used.</p>	<p>(4) The number of WHILE – DOM nesting levels cannot exceed 27.</p> <p>Not possible</p>

<p>(5) WHILE - DOm must be designated first and ENDm last.</p> <p>Not possible</p> 	<p>(6) WHILE - DOm and ENDm must correspond on a 1:1 (pairing) basis in the same program.</p> <p>Not possible</p> 
<p>(7) Two WHILE - DOm's must not overlap.</p> <p>Not possible</p> 	<p>(8) Branching externally is possible from the WHILE - DOm range.</p> <p>Possible</p> 
<p>(9) No branching is possible inside WHILE - DOm.</p> <p>Not possible</p> 	<p>(10) Subprograms can be called by G22, G65 or G66 between WHILE - DOm's.</p> <p>Possible</p> 
<p>(11) Calls can be initiated by G65 or G66 between WHILE - DOm's and commands can be issued again from 1. Up to 27 nesting levels are possible for the main program and subprograms.</p> <p>Possible</p> 	<p>(12) A program error will occur at G23 if WHILE and END are not paired in the subprogram (including macro subprogram).</p> <p>Don ENDn constitutes illegal usage.</p> 

**(Note)** As the canned cycles G73 and G83 and the special canned cycle G34 use WHILE, these will be added multiple times.

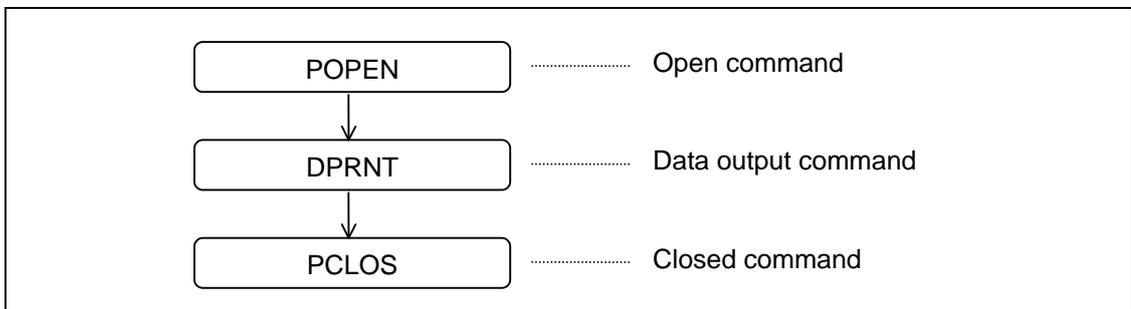
## 13.5.8 External Output Commands

**Function and purpose**

Besides the standard user macro commands, the following macro instructions are also available as external output commands. They are designed to output the variable values or characters via the RS-232C interface.

**Command format**

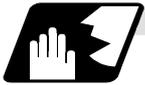
**POPEN** For preparing the processing of data outputs  
**PCLOS** For terminating the processing of data outputs  
**BPRNT** For character output and variable value binary output  
**DPRNT** For character output and digit-by-digit variable numerical output

**Command sequence****Open command : POPEN**

- (1) The command is issued before the series of data output commands.
- (2) The DC2 control code and % code are output from the NC system to the external output device.
- (3) Once POPEN; has been issued, it will remain valid until PCLOS; is issued.

**Close command : PCLOS**

- (1) This command is issued when all the data outputs are completed.
- (2) The DC4 control code and % code are output from the NC unit to the external output device.
- (3) This command is used together with the open command and it should not be issued unless the open mode has been established.
- (4) Issue the close command at the end of the program even when operation has been suspended by resetting or some other operation during data output.



## Data output command : DPRNT

**DPRNT [ /1 # v1 [ d1 c1 ] /2 # v2 [ d2 c2 ] ..... ]**

l1	:	Character string	
v1	:	Variable number	
d1	:	Significant digits above decimal point	} c + d ≤ 8
c1	:	Significant digits below decimal point	

- (1) The character output and decimal output of the variable values are done with ISO codes.
- (2) The commanded character string is output as is by the ISO code.  
Alphanumerics (A to Z, 0 to 9) and special characters (+, -, \*, /) can be used.  
Note that asterisk (\*) is output as a space code.
- (3) The required significant digits above and below the decimal point in the variable values are commanded within square parentheses. As a result, the variable values equivalent to the commanded number of digits including the decimal point are output in ISO code in decimal notation from the high-order digits. Trailing zeroes are not omitted.
- (4) Leading zeroes are suppressed.  
The leading zeroes can also be replaced by blank if so specified with a parameter. This can justify printed data on the last column.

**(Note)** A data output command can be issued even in two-part system mode. In this case, however, note that the output channel is shared for both part systems. So, take care not to execute data output in both part systems simultaneously.

13.5.9 Precautions

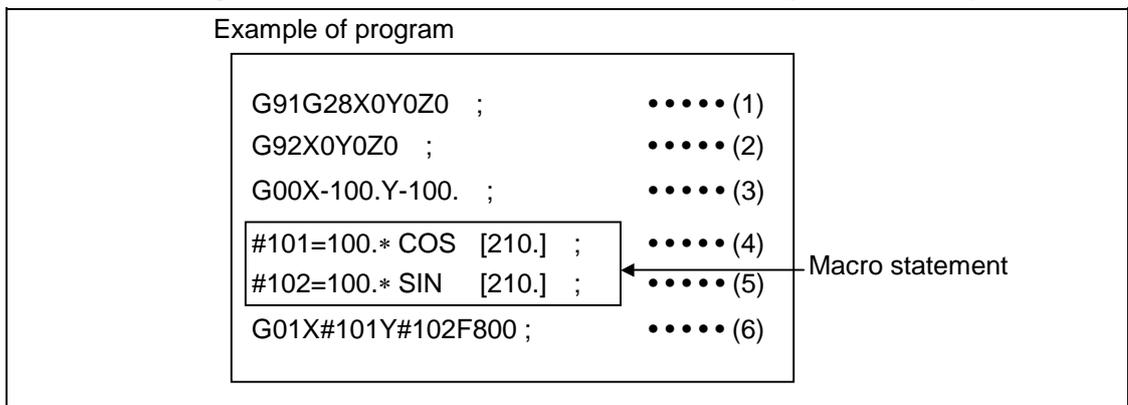


Precautions

When the user macro commands are employed, it is possible to use the M, S, T and other NC control commands together with the arithmetic, decision, branching and other macro commands for preparing the machining programs. When the former commands are made into executable statements and the latter commands into macro statements, the macro statement processing should be accomplished as quickly as possible in order to minimize the machining time, because such processing is not directly related to machine control.

As a result, the parameter "#8101 macro single" can be set and the macro statements can be processed in parallel with the execution of the executable statement.

(The parameter can be set OFF during normal machining to process all the macro statements together or set ON during a program check to execute the macro statements block by block. This enables the setting to be made in accordance with the intended objective in mind.)

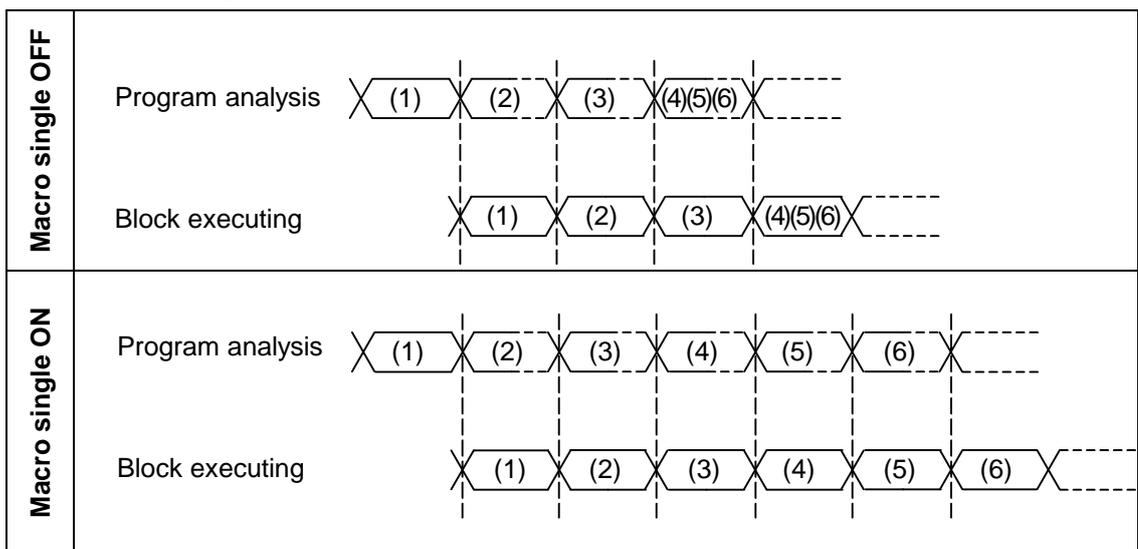


Macro statements are:

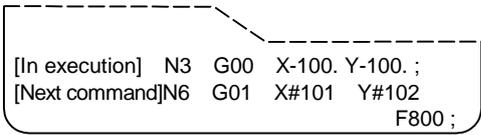
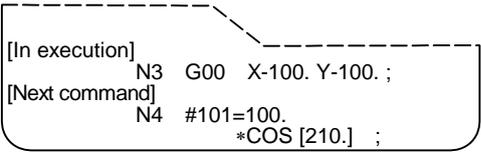
- (1) Arithmetic commands (block including =)
- (2) Control commands (block including GOTO, DO-END, etc.)
- (3) Macro call commands (including macro calls based on G codes and cancel commands (G65, G66, G66.1, G67))

Executable statements indicate statements other than macro statements.

Flow of processing



Machining program display

<b>Macro single ON</b>	 <pre> [In execution] N3 G00 X-100. Y-100. ; [Next command]N6 G01 X#101 Y#102                 F800 ;             </pre>	<p>N4, N5 and N6 are processed in parallel with the control of the executable statement of N3, N6 is an executable statement and so it is displayed as the next command. If the N4, N5 and N6 analysis is in time during N3 control, the machine movement will be continuously controlled.</p>
<b>Macro single OFF</b>	 <pre> [In execution] N3 G00 X-100. Y-100. ; [Next command] N4 #101=100.                 *COS [210.] ;             </pre>	<p>N4 is processed in parallel with the control of the NC executable statement of N3, and it is displayed as the next command. N5 and N6 is executed after N3 has finished, and so the machine control is held on standby during the N5 and N6 analysis time.</p>

13.5.10 Actual Examples of Using User Macros

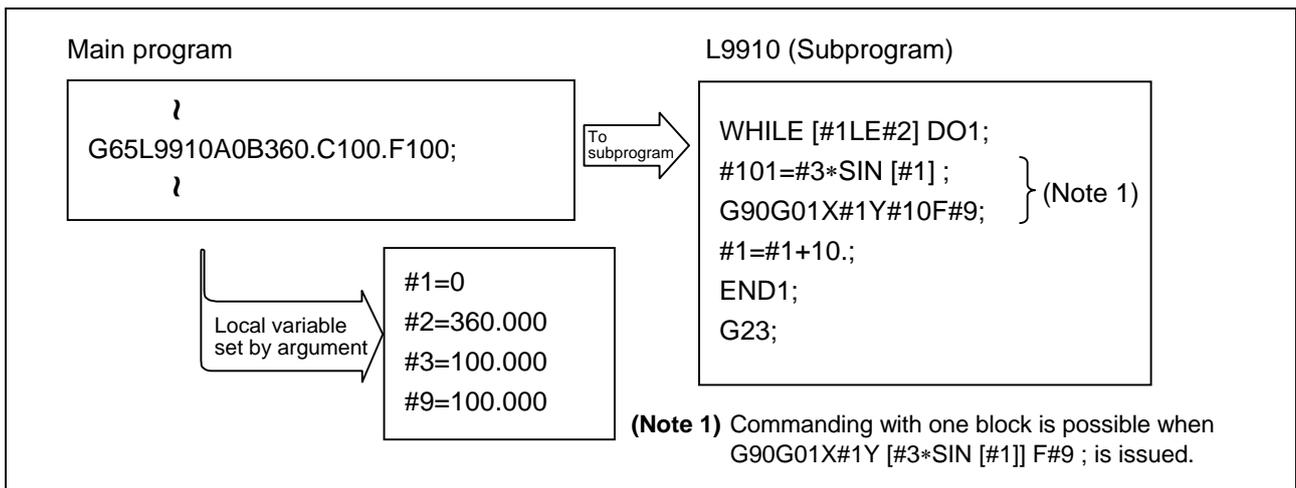
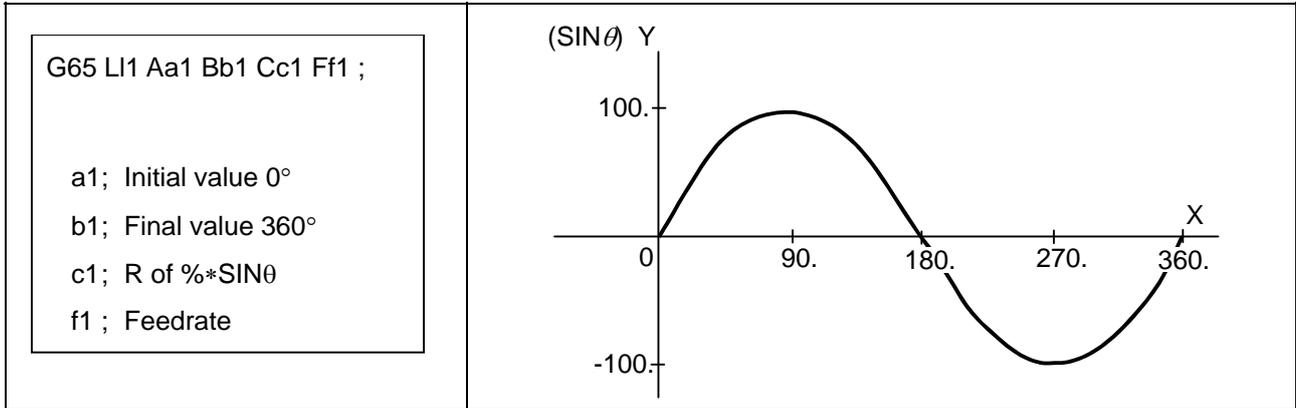
The following three examples will be described.

**(Example 1)** SIN curve

**(Example 2)** Bolt hole circle

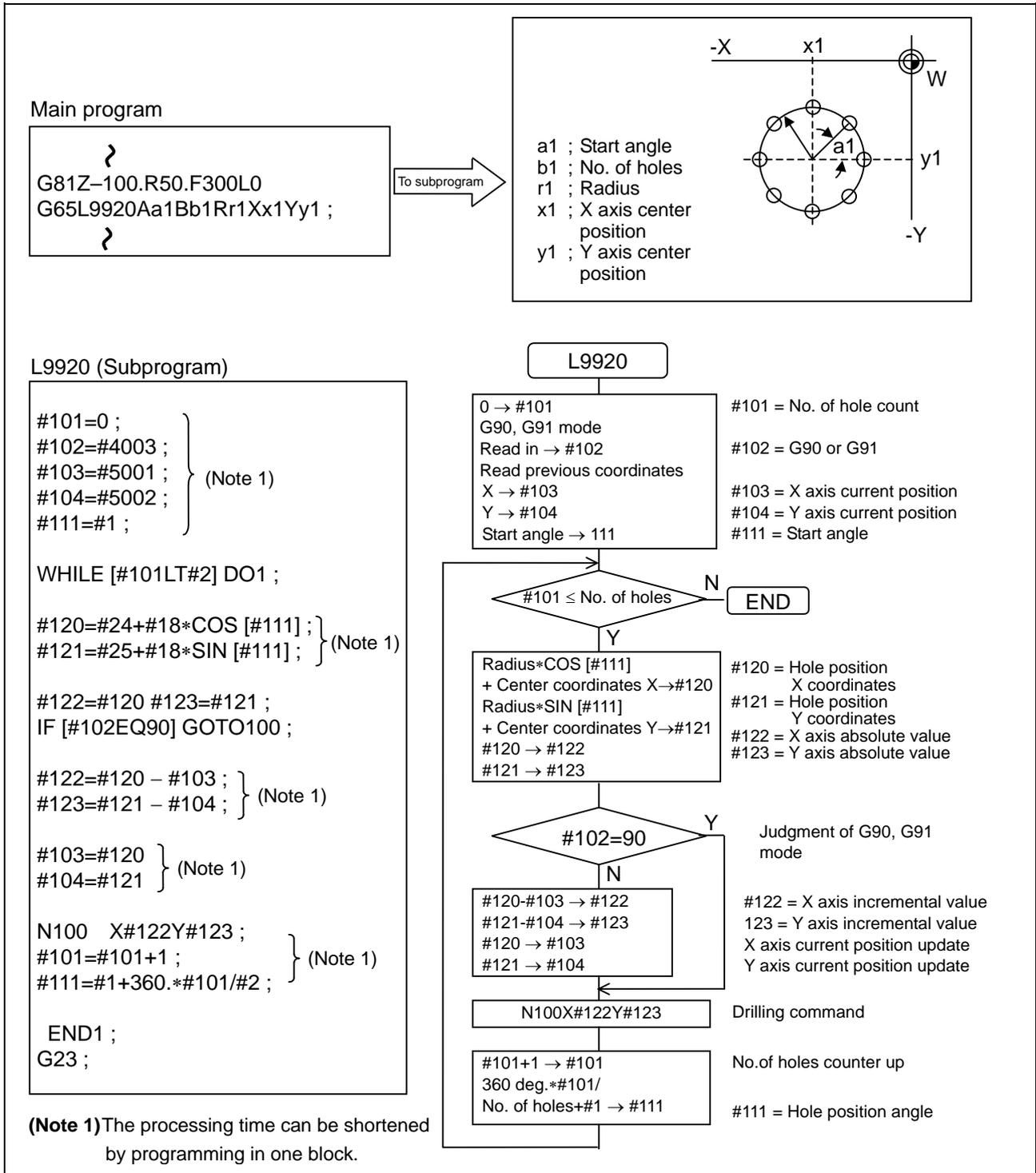
**(Example 3)** Grid

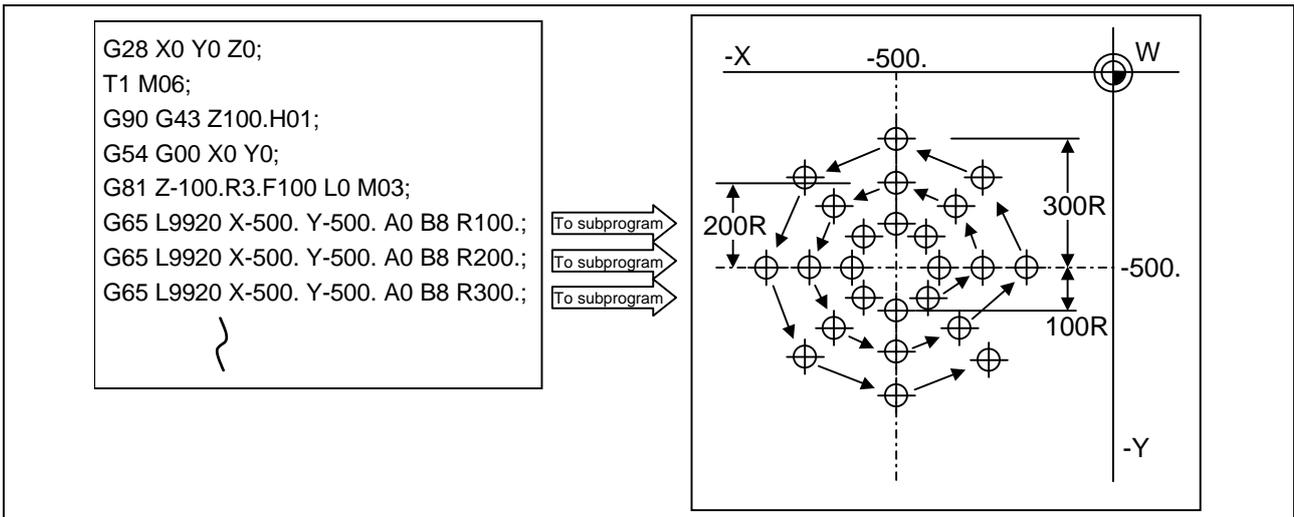
**(Example 1) SIN curve**



**(Example 2) Bolt hole circle**

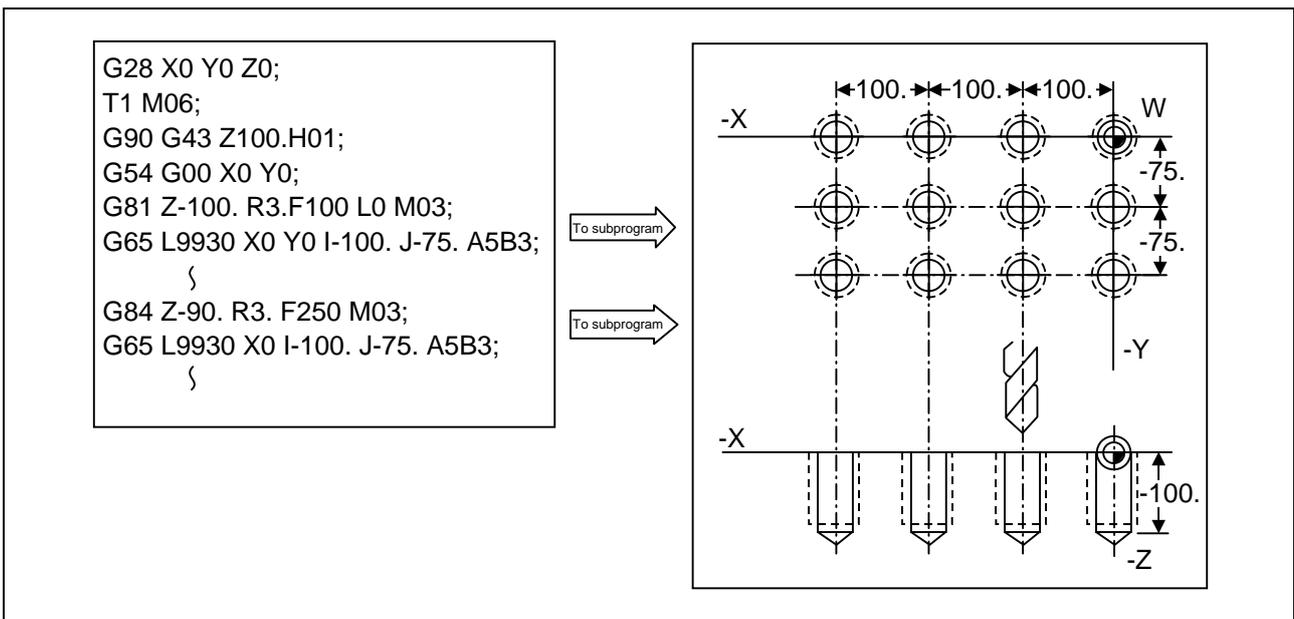
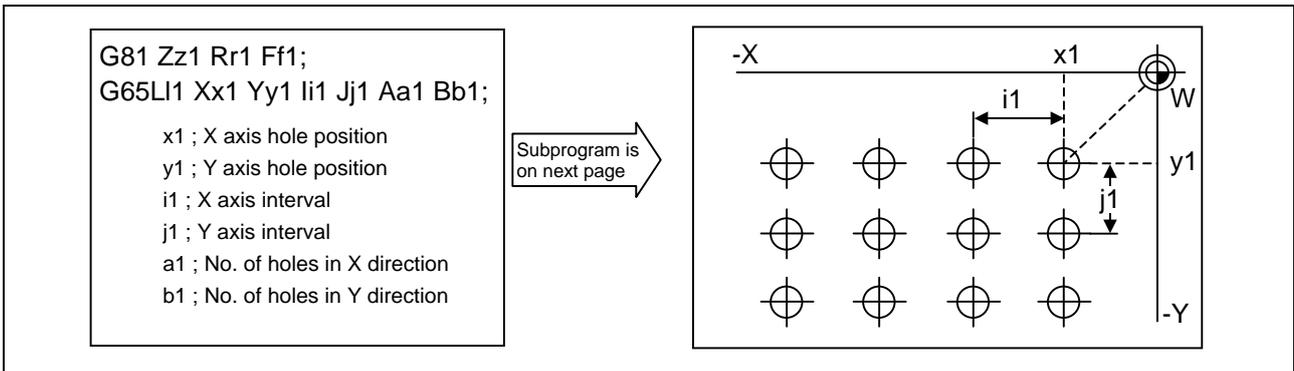
After defining the hole data with canned cycle (G72 to G89), the macro command is issued as the hole position command.





**(Example 3) Grid**

After defining the hole data with the canned cycle (G72 to G89), macro call is commanded as a hole position command.



L9930 (Subprogram)

L9930

```

#101=#24 ;
#102=#25 ;

#103=#4 ;
#104=#5 ;

#106=#2 ;

WHILE [#106GT0] DO1 ;

#105=#1 ;

WHILE [#105GT0] DO2 ;

G90 X#101 Y#102 ;

#101=#101+#103 ;
#105=#105-1 ;

END2 ;

#101=#101-#103 ;
#102=#102+#104 ;

#103=-#103 ;
#106=#106-1 ;

END1 ;

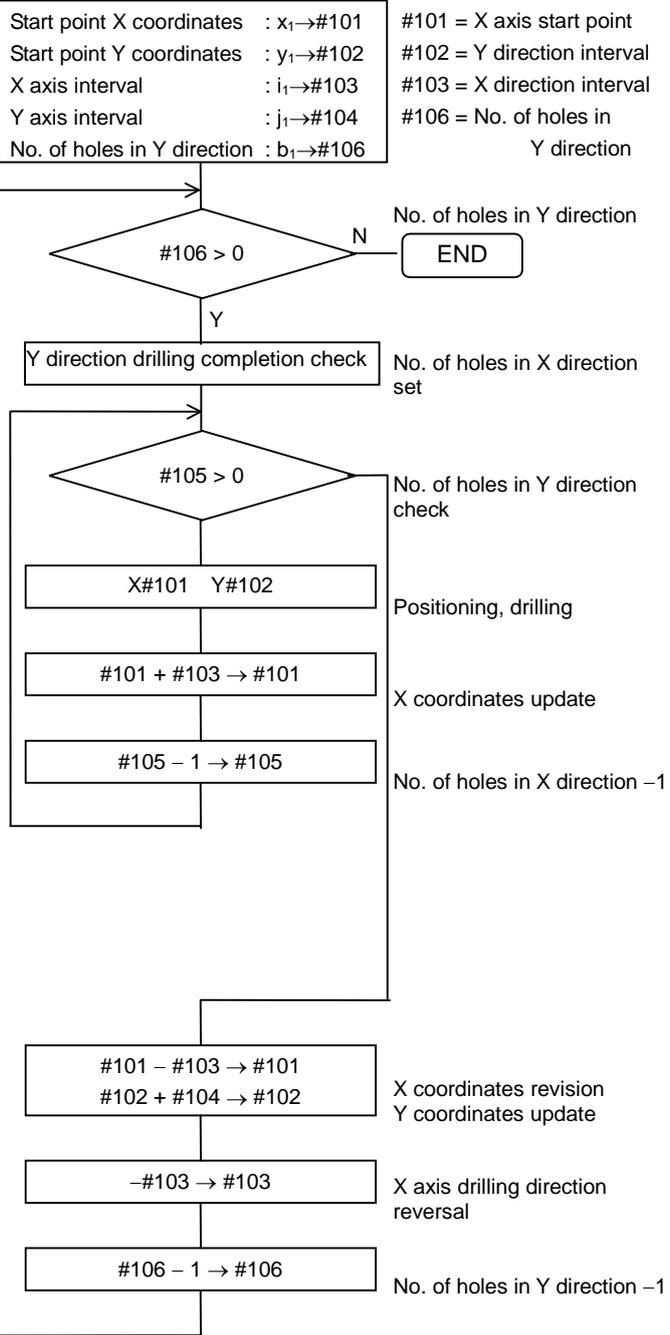
G23 ;
    
```

(Note 1)

(Note 1)

(Note 1)

**(Note 1)** The processing time can be shortened by programming in one block.



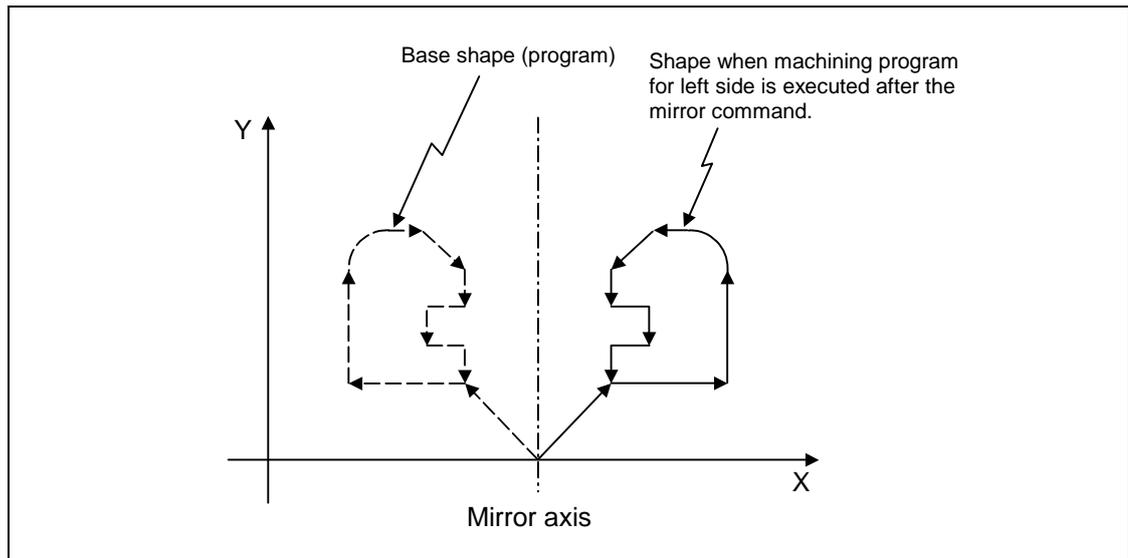
## 13.6 G Command Mirror Image; G50.1, G51.1 / G62



## Function and purpose

When cutting a shape that is symmetrical on the left and right, programming time can be shortened by machining the one side and then using the same program to machine the other side. The mirror image function is effective for this.

For example, when using a program as shown below to machine the shape on the left side, a symmetrical shape can be machined on the right side by applying mirror image and executing the program.



## Command format

## (1) For G50.1, G51.1

<b>G51.1</b>	<b>Xx1</b>	<b>Yy1</b>	<b>Zz1 ;</b>	<b>Mirror image ON</b>
<b>G50.1</b>	<b>Xx2</b>	<b>Yy2</b>	<b>Zz2 ;</b>	<b>Mirror image OFF</b>
x1, y1, z1	: Mirror image center coordinates			(Mirror image will be applied regarding this position as a center)
x2, y2, z2	: Mirror image cancel axis			(The values of x2, y2, z2 will be ignored.)

Command these items with the absolute or incremental position.

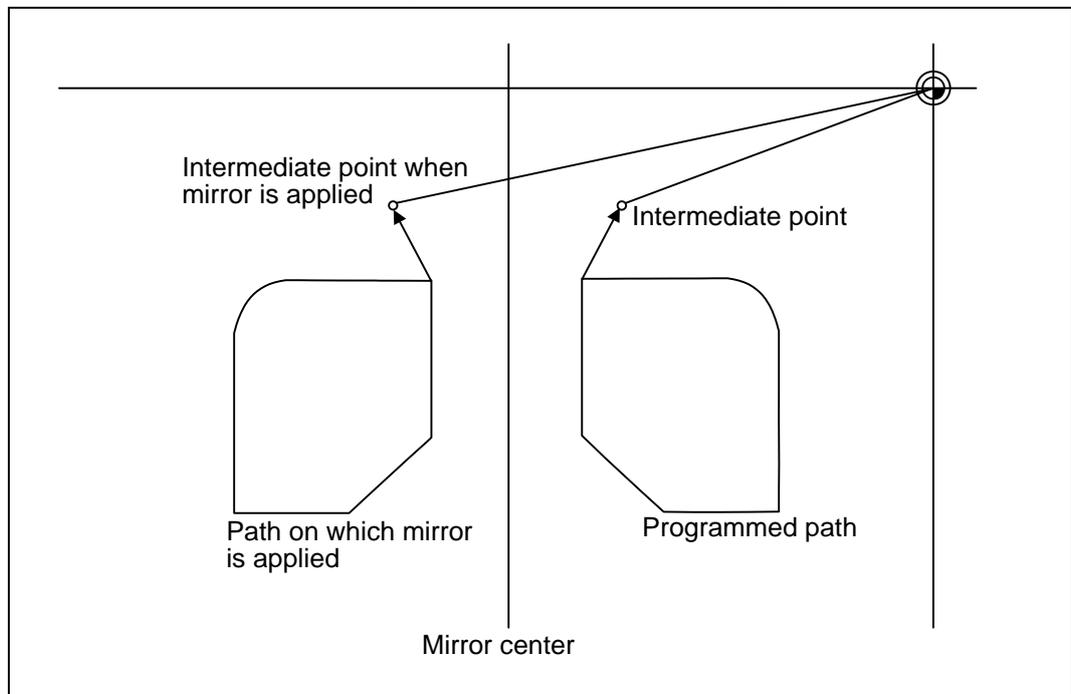
## (2) For G62

<b>G62</b>	<b>X1</b>	<b>Y1</b>	<b>Z1 ;</b>	<b>Mirror image ON</b>
<b>G62</b>	<b>X0</b>	<b>Y0</b>	<b>Z0 ;</b>	<b>Mirror image OFF</b>
G62	: Mirror image ON/OFF command			
X1, Y1, Z1	: Mirror image applied with the position where the command is issued as a center.			
X0, Y0, Z0	: Set "0" to the axis that mirror image is to be canceled.			



## Detailed description

- (1) At G51.1, command the mirror image axis and the coordinate to be a center of mirror image with the absolute command or incremental command.
- (2) At G50.1, command the axis for which mirror image is to be turned OFF. The values of x2, y2, and z2 will be ignored.
- (3) If mirror image is applied on only one axis in the designated plane, the rotation direction and compensation direction will be reversed for the arc or tool radius compensation and coordinate rotation, etc.
- (4) This function is processed on the local coordinate system, so the center of the mirror image will change when the counter is preset or when the workpiece coordinates are changed.
- (5) Reference position return during mirror image  
If the reference position return command (G28, G30) is executed during the mirror image, the mirror image will be valid during the movement to the intermediate point, but will not be applied on the movement to the reference position after the intermediate point.



- (6) Return from zero point during mirror image  
If the return command (G29) from the zero point is commanded during the mirror image, the mirror will be applied on the intermediate point.
- (7) The mirror image will not be applied on the G53 command.
- (8) For G62, mirror image ON/OFF will be applied on all the axes commanded in the same block. The position where G62 is executed is regarded as the center of the mirror image.
- (9) For G62, command "0" or "1" following to the axis name basically. The value other than "0" will be regarded as "1"  
In addition, mirror image axis ON/OFF can be commanded together.



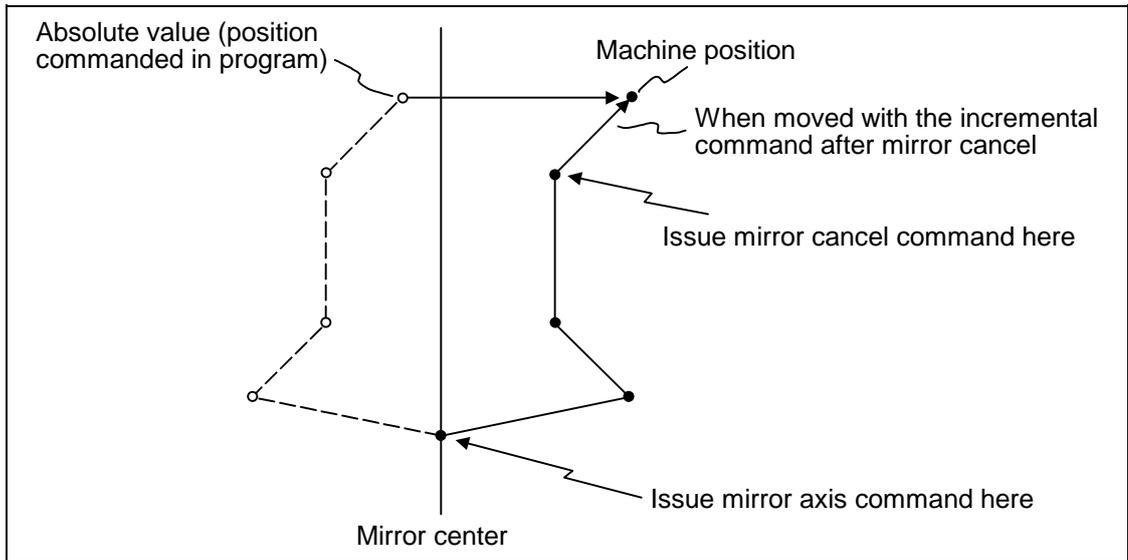
**Precautions**

**CAUTION**

⚠ Turn the mirror image ON and OFF at the mirror image center.

If mirror image is canceled at a point other than the mirror center, the absolute value and machine position will deviate as shown below. (In this state, execute the absolute value command (positioning with G90 mode), or execute reference position return with G28 or G30 to continue the operation.) The mirror center is set with an absolute value, so if the mirror center is commanded again in this state, the center may be set to an unpredictable position.

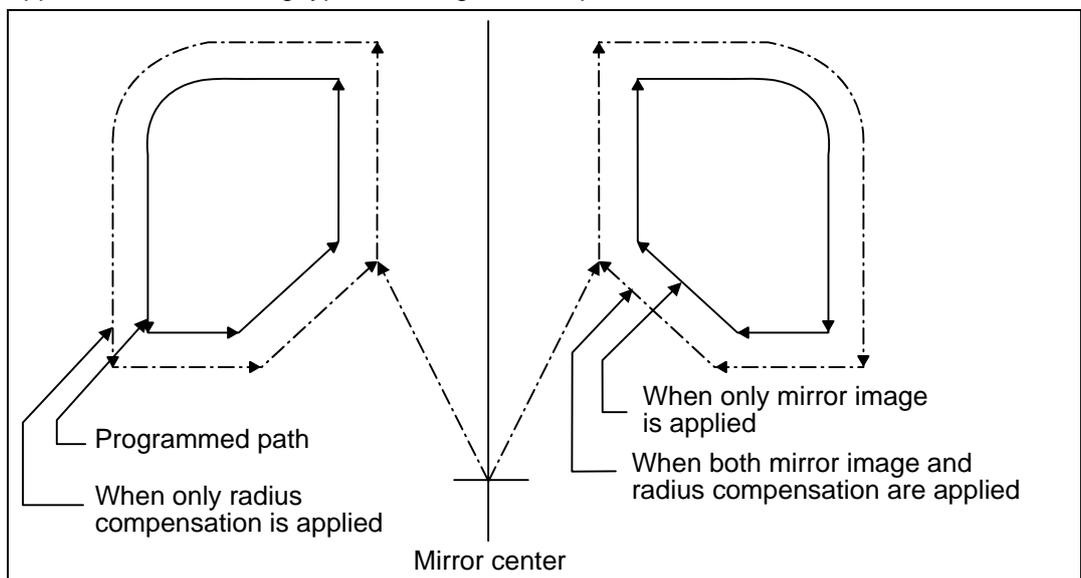
Cancel the mirror at the mirror center or position with the absolute value command after canceling.



**Combination with other functions**

**(1) Combination with radius compensation**

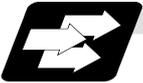
The mirror image (G51.1) will be processed after the radius compensation (G41, G42) is applied, so the following type of cutting will take place.



### 13.7 Corner Chamfering/Corner Rounding I

Chamfering at any angle or corner rounding is performed automatically by adding ",C\_" or ",R\_" to the end of the block to be commanded first among those command blocks which shape the corner with lines only.

#### 13.7.1 Corner Chamfering ",C\_"



##### Function and purpose

The corner is chamfered in such a way that the positions produced by subtracting the lengths commanded by ",C\_" from the imaginary starting and final corners which would apply if no chamfering were to be performed, are connected.



##### Command format

```
N100 G01 X__ Y__ ,C__ ;
```

```
N200 G01 X__ Y__ ;
```

,C : Length up to chamfering starting point or end point from imaginary corner

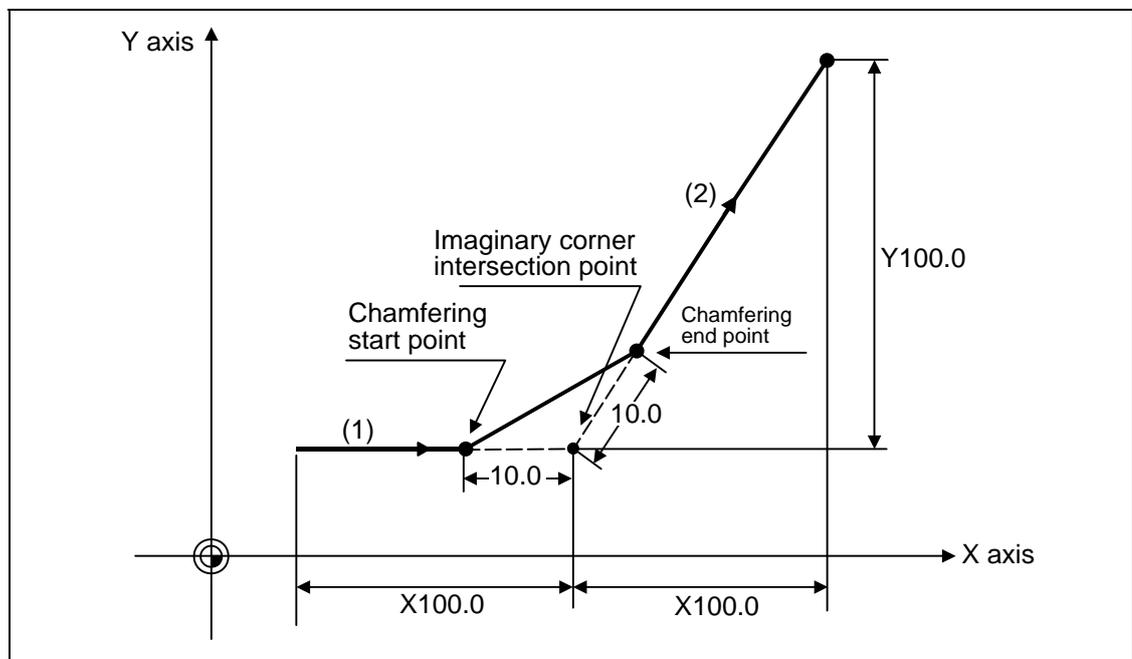
Chamfering is performed at the point where N100 and N200 intersect.



##### Example of program

```
(1) G91 G01 X100., C10. ;
```

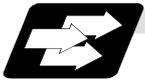
```
(2) X100. Y100. ;
```



**Detailed description**

- (1) The start point of the block following the corner chamfering serves as the imaginary corner intersection point.
- (2) When the comma in ",C" is not present, it is handled as a C command.
- (3) When both the corner chamfer and corner rounding commands exist in the same block, the latter command is valid.
- (4) Tool compensation is calculated for the shape which has already been subjected to corner chamfering.
- (5) When scaling is commanded, scaling will also be applied to the commanded corner chamfer amount.
- (6) Program error (P381) results when there is an arc command in the block following the corner chamfering block.
- (7) Program error (P382) results when the block following the corner chamfering block does not have a linear command.
- (8) Program error (P383) results when the movement amount in the corner chamfering block is less than the chamfering amount.
- (9) Program error (P384) results when the movement amount in the block following the corner chamfering block is less than the chamfering amount.

## 13.7.2 Corner Rounding ",R\_"



## Function and purpose

The imaginary corner, which would exist if the corner were not to be rounded, is rounded with the arc having the radius which is commanded by ",R\_" only when configured of linear lines.



## Command format

```
N100 G01 X__ Y__, R__ ;
N200 G02 X__ Y__ ;
```

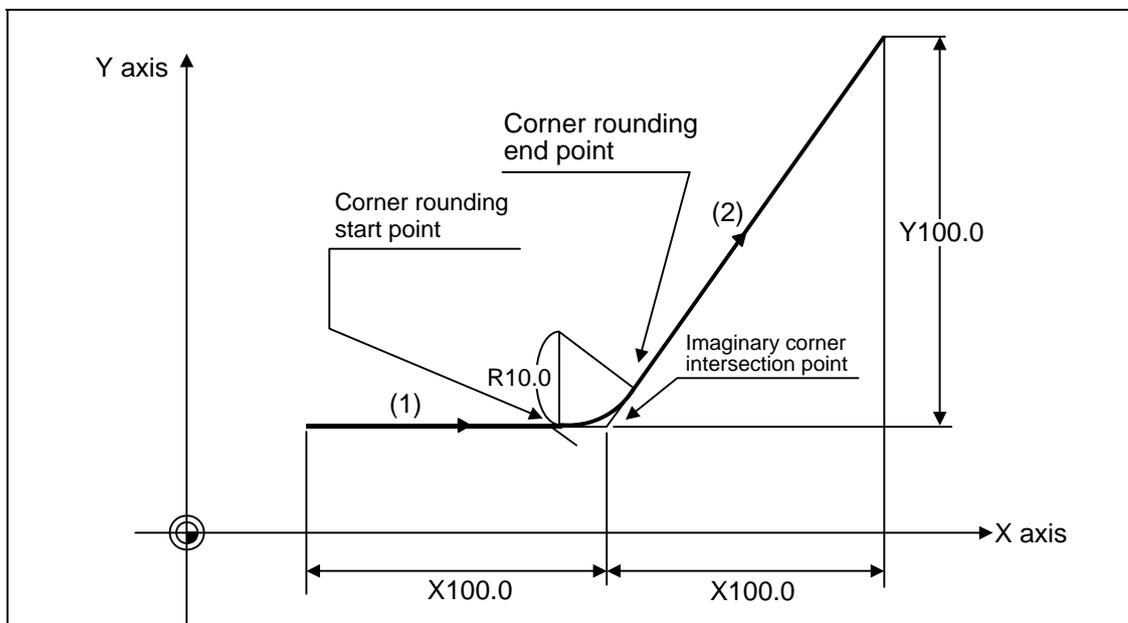
,R : Arc radius of corner rounding

Corner rounding is performed at the point where N100 and N200 intersect.



## Example of program

```
(1) G91 G01 X100., R10. ;
(2) X100. Y100. ;
```



## Detailed description

- (1) The start point of the block following the corner R serves as the imaginary corner intersection point.
- (2) When the comma in ",R" is not present, it is handled as an R command.
- (3) When both the corner chamfer and corner rounding commands exist in the same block, the latter command is valid.
- (4) Tool compensation is calculated for the shape which has already been subjected to corner rounding.
- (5) Program error (P381) results when there is an arc command in the block following the corner rounding block.
- (6) Program error (P382) results when the block following the corner rounding block does not have a linear command.
- (7) Program error (P383) results when the movement amount in the corner rounding block is less than the R value.
- (8) Program error (P384) results when the movement amount in the block following the corner rounding block is less than the R value.

## 13.8 Linear Angle Command



## Function and purpose

The end point coordinates are calculated automatically by commanding the linear angle and one of the end point coordinate axes.

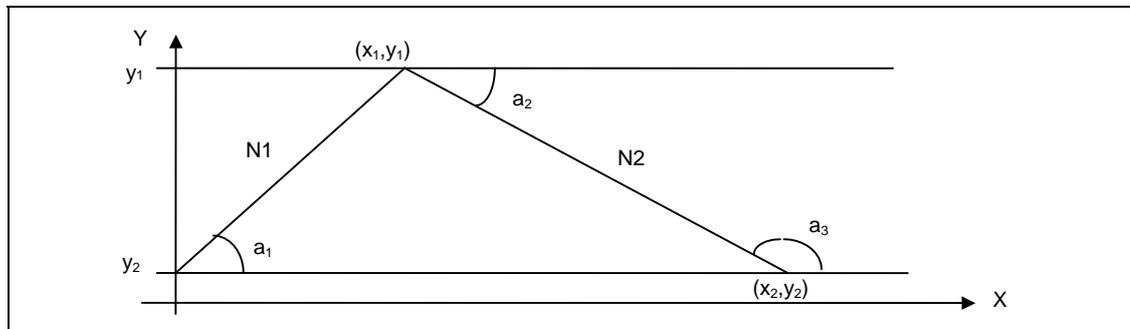


## Command format

```
N1 G01 Xx1 (Yy1) Aa1;  
N1 G01 Xx2 (Yy2) A-a2; (A-a2 can also be set as As3.)  
or  
N1 G01 Xx1 (Yy1) ,Aa1;  
N1 G01 Xx2 (Yy2) ,A-a2;
```

This designates the angle and the X or Y axis coordinates.

Select the command plane with G17 to G19.

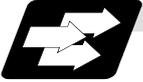


## Detailed description

- (1) The angle is from the + direction of the horizontal axis on the selected plane. The counter-clockwise (CCW) direction is considered to be + and the clockwise direction (CW) -.
- (2) Either of the axes on the selected plane is commanded for the end point.
- (3) The angle is ignored when the angle and the coordinates of both axes are commanded.
- (4) When only the angle has been commanded, this is treated as a geometric command.
- (5) The angle of either the start point ( $a_1$ ) or end point ( $a_2$ ) may be used.
- (6) This function is valid only for the G01 command; it is not valid for other interpolation or positioning commands.
- (7) The range of slope "a" is  $-360.000 \leq a \leq 360.000$ .  
When a value outside this range is commanded, it will be divided by 360 (degrees) and the remainder will be commanded.  
**(Example)** If 400 is commanded,  $40^\circ$  (remainder of  $400/360$ ) will become the command angle.
- (8) If address A is used for the axis name or 2nd miscellaneous function, use ",A" as the angle.
- (9) If "A" and ",A" are commanded in the same block, ",A" will be interpreted as the angle.

**(Note)** A program error (P33) will occur if this function is commanded during the high-speed machining mode or high-speed high-accuracy mode.

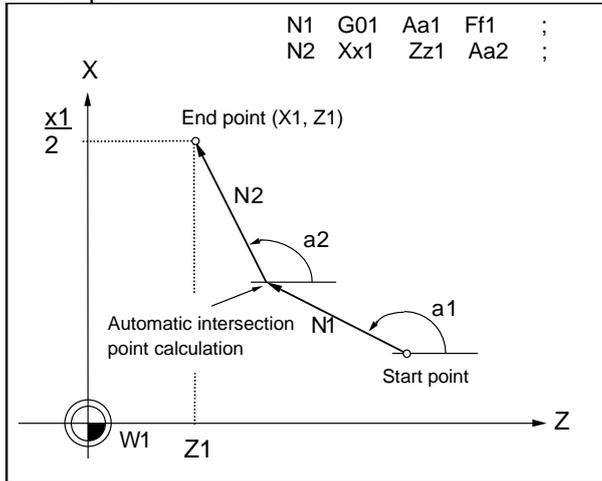
## 13.9 Geometric Command



## Function and purpose

When it is difficult to find the intersection point of two straight lines with a continuous linear interpolation command, this point can be calculated automatically by programming the command for the angle of the straight lines.

Example



a: Angle ( $^{\circ}$ ) formed between straight line and horizontal axis on plane.  
The plane is the selected plane at this time.

**(Note 1)** This function cannot be used when using the A axis or 2nd miscellaneous function A.



## Command format

**N1 G01 Xx<sub>1</sub> (Yy<sub>1</sub>) Aa<sub>1</sub> ;**  
**N1 G01 Xx<sub>2</sub> (Yy<sub>2</sub>) A-a<sub>2</sub> ; (A-a<sub>2</sub> can also be set as As<sub>3</sub>.)**

This designates the angle and the X or Y axis coordinates.

Select the command plane with G17 to G19.



## Detailed description

## (1) Automatic calculation of two-arc contact

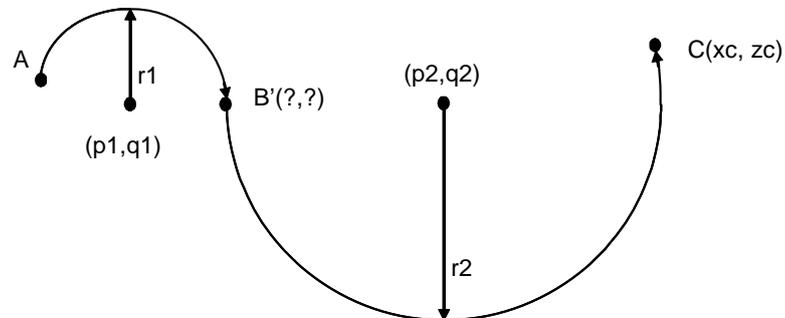
When two continuous circular arcs contact with each other and it is difficult to find the contact, the contact is automatically calculated by specifying the center coordinates position or radius of the first circular arc and the end point (absolute position) and center position or radius of the second circular arc.

## Example

```

G18 G02 Ii1 Kk1 Ff1 ;
      G03 Xxc Zzc Ii2 Kk2 Ff2 ;
OR
G18 G02 Ii1 Kk1 Ff1 ;
      G03 Xxc Zzc Rr2 Ff2 ;
OR
G18 G02 Rr1 Ff1 ;
      G03 Xxc Zzc Ii2 Kk2 Ff2 ;

```



I and K : Incremental position from arc end point  
P and Q : Arc center position (absolute position)

I and K are the arc center position (incremental position); distances from the start point in the first block or distances from the end point in the second block. P and Q (X, Z arc center position (absolute position)) can be commanded instead of I and K commands.

(2) Automatic calculation of linear-arc intersection

When it is difficult to find the intersections of a given line and circular arc, the intersections are automatically calculated by programming the following blocks.

**Example**

```
G18 G01 Aa1 Ff1 ;
      G02 Xxc Zzc Ii2 Kk2 Hh2 Ff2 ;
```

I and K : Incremental position from arc end point  
 P and Q : Arc center position (absolute position)  
 H = 0 : Intersection with shorter line (B point)  
 H = 1 : Intersection with longer line (B' point)

(3) Automatic calculation of arc-linear intersection

When it is difficult to find the intersections of a given circular arc and line, the intersections are automatically calculated by programming the following blocks.

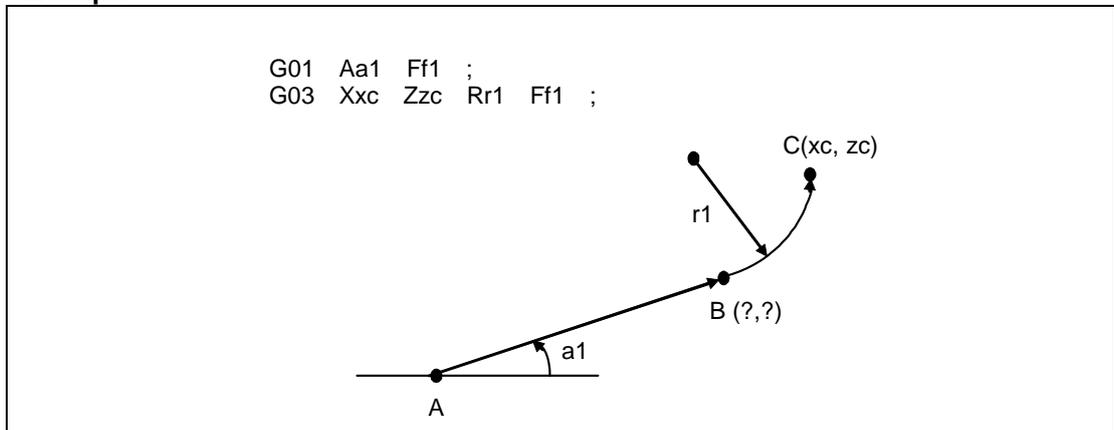
**Example**

```
G18 G03 Ii1 Kk1 Hh1 Ff1 ;
      G01 Xxc Zzc Aa1 Ff1 ;
```

I and K : Incremental position from arc end point  
 P and Q : Arc center position (absolute position) (L3 only)  
 H = 0 : Intersection with shorter line (B point)  
 H = 1 : Intersection with longer line (B' point)

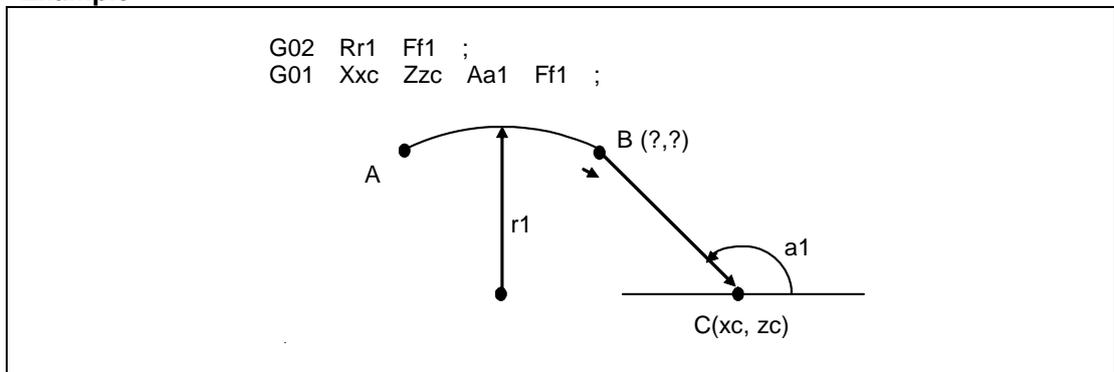
## (4) Automatic calculation of linear-arc contact

When it is difficult to find the contact of a given line and circular arc, the contact is automatically calculated by programming the following blocks.

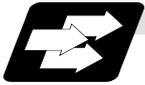
**Example**

## (5) Automatic calculation of arc-linear contact

When it is difficult to find the contact of a given circular arc and line, the contact is automatically calculated by programming the following blocks.

**Example**

13.10 Circle Cutting; G12, G13



Function and purpose

Circle cutting starts the tool from the center of the circle, and cuts the inner circumference of the circle. The tool continues cutting while drawing a circle and returns to the center position.



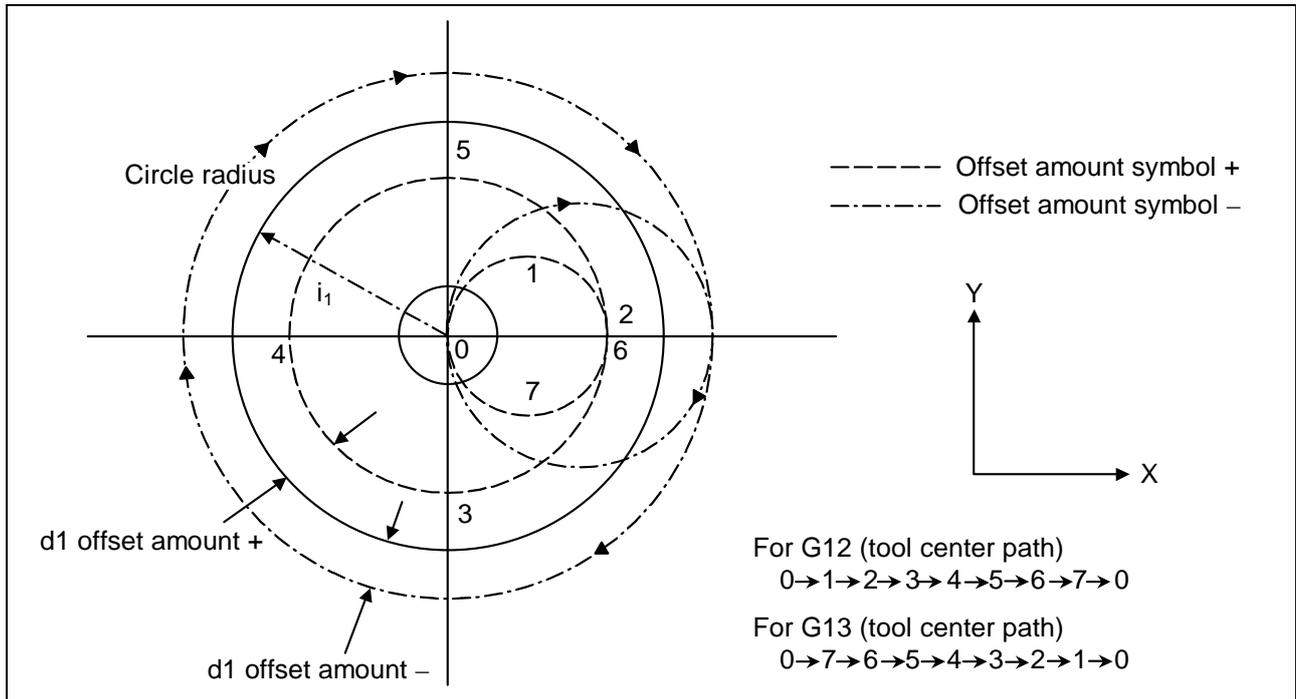
Command format

<b>G12 (G13)</b>	<b>I__ D__ F__ ;</b>
G12	: Clockwise (CW)
G13	: Counterclockwise (CCW)
I	: Radius of circle (incremental value), the symbol is ignored
D	: Offset No. (The offset No. and offset data are not displayed on the setting and display unit.)
F	: Feedrate



Detailed description

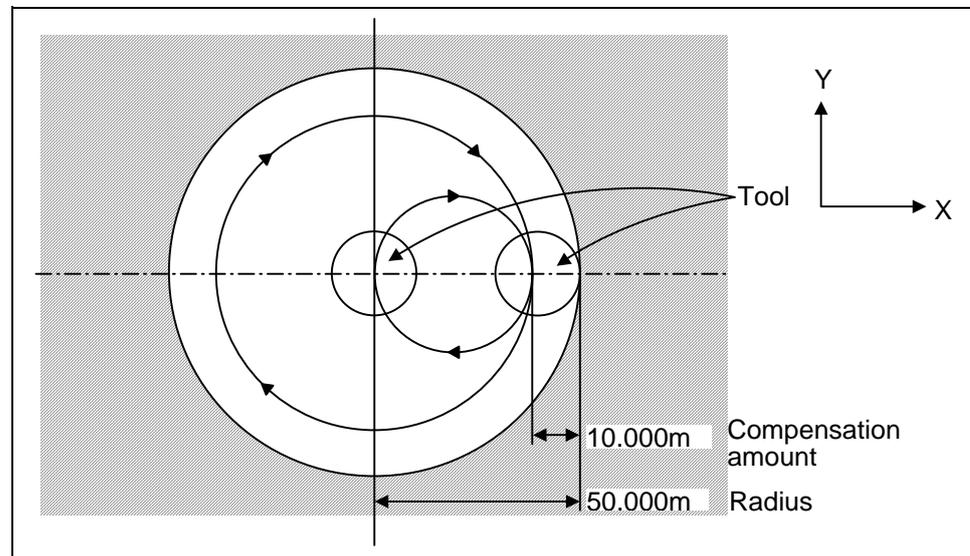
- (1) The symbol + for the offset amount indicates reduction, and – indicates enlargement.
- (2) The circle cutting is executed on the plane G17, G18 or G19 currently selected.





## Example of program

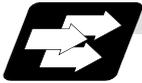
(Example 1) G12 I5000 D01 F100 ; (Input setting unit 0.01)  
When compensation amount is +10.00mm



## Precautions

- (1) If the offset No. "D" is not issued or if the offset No. is illegal, the program error (P170) will occur.
- (2) If [Radius (I) = offset amount] is 0 or negative, the program error (P233) will occur.
- (3) If G12 or G13 is commanded during radius compensation (G41, G42), the radius compensation will be validated on the path after compensating with the D commanded with G12 or G13.
- (4) If an address, not included in the format, is commanded in the same block as G12 and G13, a program error (P32) will occur.

13.11 Parameter Input by Program; G10, G11.1



Function and purpose

The parameters set from the setting and display unit can be changed in the machining programs. The data format used for the data setting is as follows.



Command format

<b>G10 L70 ; Data setting start command</b>			
P parameter No.	S part system No.	A axis No.	H□data ; ..... Bit parameter
P parameter No.	S part system No.	A axis No.	D data ; ..... Numerical value parameter
P parameter No.	S part system No.	A axis No.	<character string> ... Character string parameter
<b>G11 Data setting end command</b>			

- (Note 1)** The sequence of addresses in a block must be as shown above.  
When an address is commanded two or more times, the last command will be valid.
- (Note 2)** The part system No. is set in the following manner: "1" for 1st part system, "2" for 2nd part system, and so forth.  
If the address S is omitted, the part system of the executing program will be applied  
As for the parameters common to part systems, the command of part system No. will be ignored.
- (Note 3)** The axis No. is set in the following manner: "1" for 1st axis, "2" for 2nd axis, and so forth.  
If the address A is omitted, the 1st axis will be applied.  
As for the parameters common to axes, the command of axis No. will be ignored.
- (Note 4)** Address H is commanded with the combination of setting data (0 or 1) and the bit designation (□) (0 to 7).
- (Note 5)** Only the decimal number can be commanded with the address D.  
The value that is smaller than the input setting increment (#1003 iunit) will be round off to the nearest increment.
- (Note 6)** The character string must be put in angled brackets "<" and ">".  
If these brackets are not provided, the program error (P33) will occur.  
Up to 63 characters can be set.
- (Note 7)** Command G10L70, G11.1 in independent blocks. A program error (P33, P421) will occur if not commanded in independent blocks.

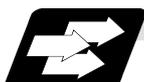


Example of program

(Example)

G10 L70;	
P6401 H71 ;	Sets "1" to "#6401 bit7".
P8204 S1 A2 D1.234 ;	Sets "1.234" to #8204 of the 1st part system 2nd axis.
P8621 <X> ;	Sets "x" to "#8621".
G11.1	

## 13.12 Macro Interrupt; ION, IOF



## Function and purpose

A user macro interrupt signal (UIT) is input from the machine to interrupt the program being currently executed and instead call another program and execute it. This is called the user macro interrupt function.

Use of this function allows the program to operate flexibly enough to meet varying conditions. For setting the parameters of the function, refer to the Setup manual.



## Command format

<b>ION <i>program number</i> H__ ;    Macro interruption enable</b>	
<b>or ION &lt;File name&gt; H__ ;</b>	
ION	:User macro interruption command
Program number	:Interrupt program No.
<File name>	:File name
	A file name can be specified instead of a program No.
	In this case, enclose the file name with brackets <>.
	(The file name can have up to 32 characters including the extension.)
H	:Interrupt sequence No.

<b>IOF ;</b>	<b>User macro interruption disable</b>
IOF	:User macro interruption end command

The user macro interrupt function is enabled and disabled by the ION and IOF commands programmed to make the user macro interrupt signal (UIT) valid or invalid. That is, if an interrupt signal (UIT) is input from the machine side in a user macro interrupt enable period from when ION is issued to when IOF is issued or the NC is reset, a user macro interrupt is caused to execute the program specified following to IOF instead of the one being executed currently.

Another interrupt signal (UIT) is ignored while one user macro interrupt is being in service. It is also ignored in a user macro interrupt disable state such as after an IOF command is issued or the system is reset.



## Interrupt enable conditions

A user macro interrupt is enabled only during execution of a program. The requirements for the user macro interrupt are as follows:

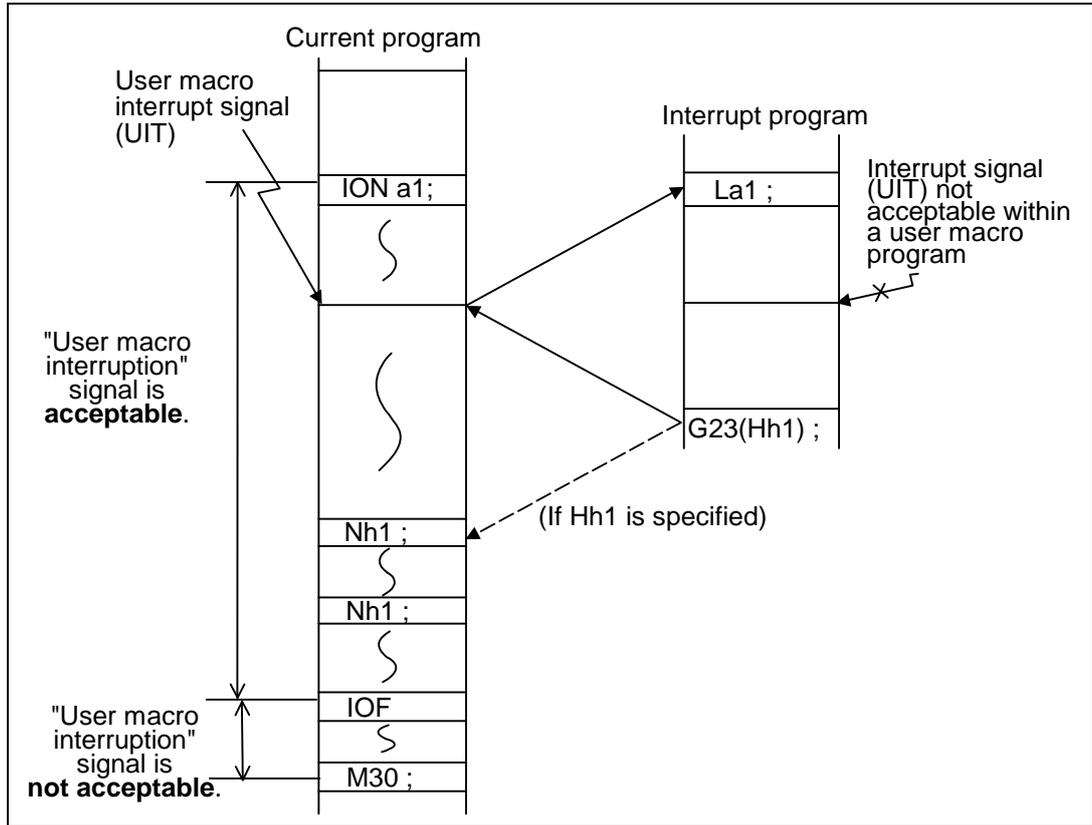
- (1) An automatic operation mode or MDI has been selected.
- (2) The system is running in automatic mode.
- (3) No other macro interrupt is being processed.

**(Note 1)** A macro interrupt is disabled in manual operation mode (JOG, STEP, HANDLE, etc.)



Outline of operation

- (1) When a user macro interrupt signal (UIT) is input after an ION a1 ; command is issued by the current program, interrupt program La1 is executed. When an G23 ; command is issued by the interrupt program, control returns to the main program.
- (2) If G23 Hh1 ; is specified, the blocks from the one next to the interrupted block to the last one are searched for the block with sequence number Nh1 ;. Control thus returns to the block with sequence number Nh1 that is found first in the above search.



**Interrupt type**

Interrupt types 1 and 2 can be selected by the parameter "#1113 INT\_2".

**[Type 1]**

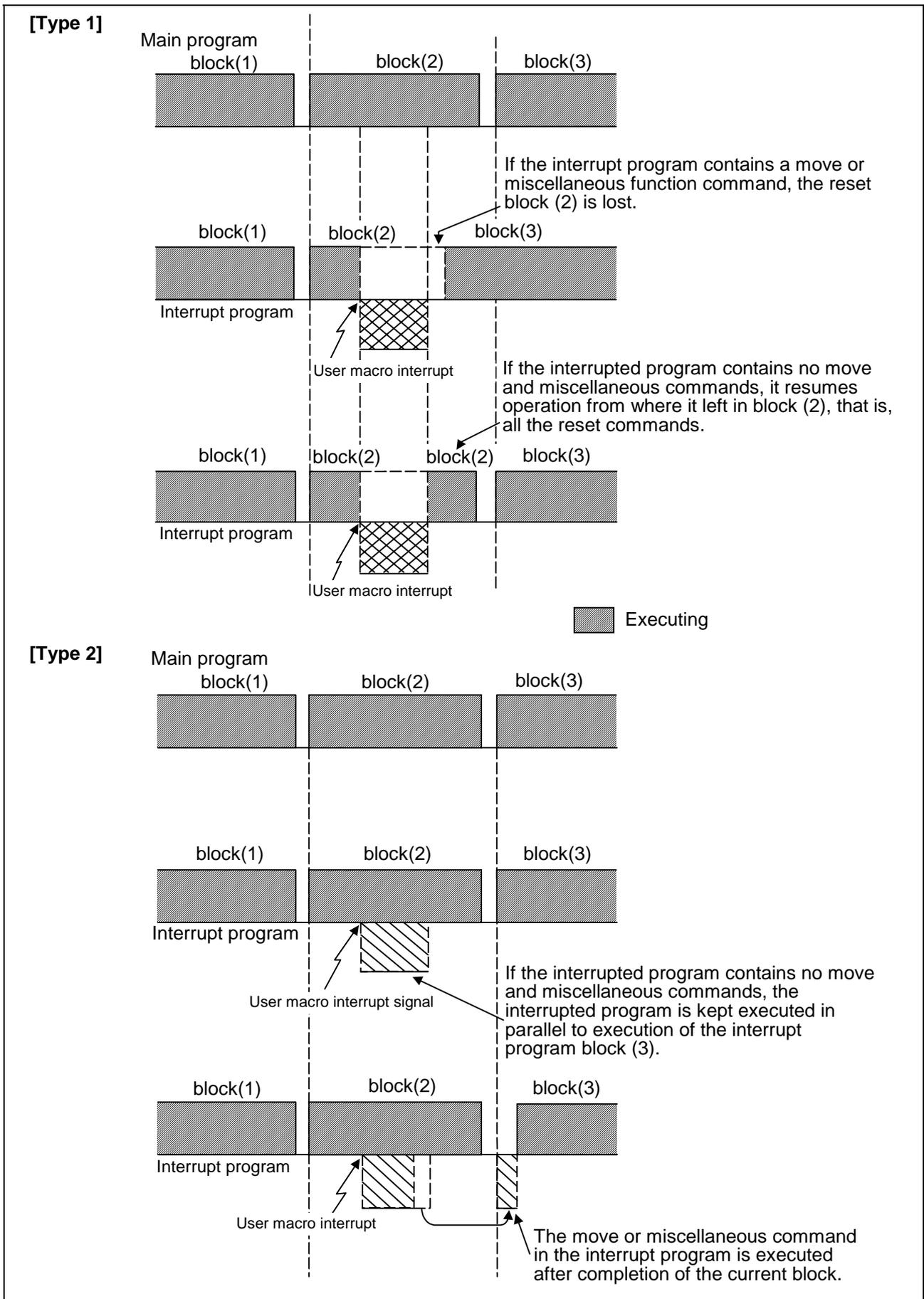
- When an interrupt signal (UIT) is input, the system immediately stops moving the tool and interrupts dwell, then permits the interrupt program to run.
- If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are lost. After the interrupt program completes, the main program resumes operation from the block next to the interrupted one.
- If the interrupt program contains no move and miscellaneous (MSTB) commands, it resumes operation, after completion of the interrupt program, from the point in the block where the interrupt was caused.

If an interrupt signal (UIT) is input during execution of a miscellaneous function (MSTB) command, the NC system waits for a completion signal (FIN). The system thus executes a move or miscellaneous function command (MSTB) in the interrupt program only after input of FIN.

**[Type 2]**

- When an interrupt signal (UIT) is input, the interrupt program will be executed in parallel with the executing block.
- If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are completed, then, these commands will be executed.
- If the interrupt program contains no move and miscellaneous function (MSTB) commands, the interrupt program is executed without interrupting execution of the current block.

However, if the interrupt program has not ended even after the execution of the original block is completed, the system may stop machining temporarily.





**Calling method**

User macro interrupt is classified into the following two types depending on the way an interrupt program is called. These two types of interrupt are selected by parameter "#1229 set01/bit0". Both types of interrupt are included in calculation of the nest level. The subprograms and user macros called in the interrupt program are also included in calculation of the nest level.

- a. Subprogram type interrupt
- b. Macro type interrupt

Subprogram type interrupt	The user macro interrupt program is called as a subprogram. As with calling by G22, the local variable level remains unchanged before and after an interrupt.
Macro type interrupt	The user macro interrupt program is called as a user macro. As with calling by G65, the local variable level changes before and after an interrupt. No arguments in the main program can be passed to the interrupt program.

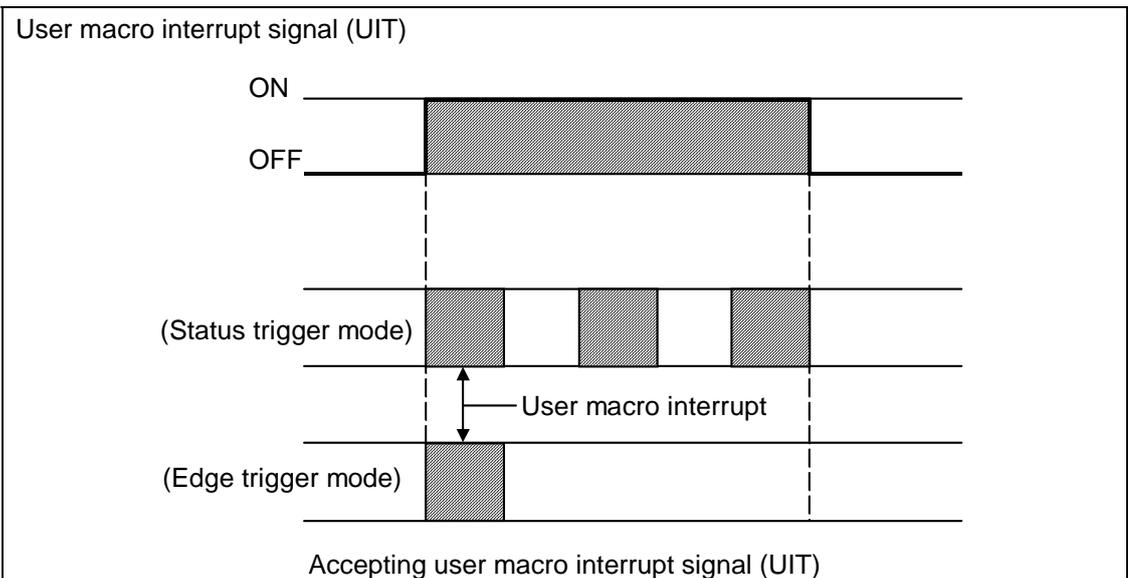


**Acceptance of user macro interrupt signal (UIT)**

A user macro interrupt signal (UIT) is accepted in the following two modes: These two modes are selected by a parameter "#1112 S\_TRG".

- a. Status trigger mode
- b. Edge trigger mode

Status trigger mode	The user macro interrupt signal (UIT) is accepted as valid when it is on. If the interrupt signal (UIT) is ON when the user macro interrupt function is enabled by ION, the interrupt program is activated. By keeping the interrupt signal (UIT) ON, the interrupt program can be executed repeatedly.
Edge trigger mode	The user macro interrupt signal (UIT) is accepted as valid at its rising edge, that is, at the instance it turns on. This mode is useful to execute an interrupt program once.





**Returning from user macro interrupt**

G23 (H\_\_);

An G23 command is issued in the interrupt program to return to the main program. Address H is used to specify the sequence number of the return destination in the main program. The blocks from the one next to the interrupted block to the last one in the main program are first searched for the block with designated sequence number. If it is not found, all the blocks before the interrupted one are then searched. Control thus returns to the block with sequence number that is found first in the above search.

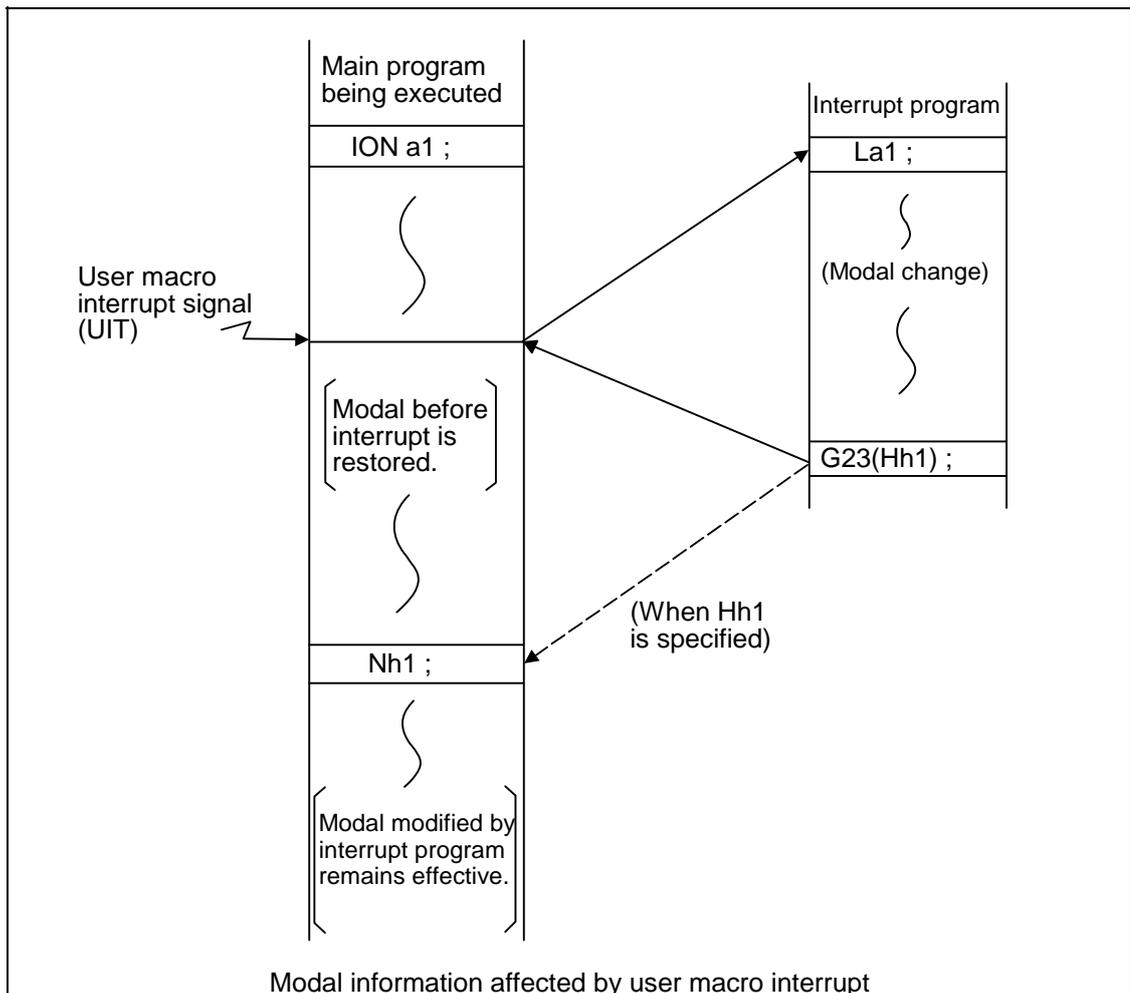
(This is equivalent to G23H\_\_ used after G22 calling.)



**Modal information affected by user macro interrupt**

If modal information is changed by the interrupt program, it is handled as follows after control returns from the interrupt program to the main program.

Returning with G23;	The change of modal information by the interrupt program is invalidated and the original modal information is not restored. With interrupt type 1, however, if the interrupt program contains a move or miscellaneous function (MSTB) command, the original modal information is not restored.
Returning with G23H__;	The original modal information is updated by the change in the interrupt program even after returning to the main program. This is the same as in returning with G23H__; from a program called by G22.





### Modal information variables (#4401 to #4520)

Modal information when control passes to the user macro interrupt program can be known by reading system variables #4401 to #4520.

The unit specified with a command applies.

System variable	Modal information
#4401 to #4421	G code (group 01 to group 21)
#4507	D code
#4509	F code
#4511	H code
#4513	M code
#4514	Sequence number
#4515	Program number ( <b>Note 1</b> )
#4519	S code
#4520	T code

} Some groups are not used.

The above system variables are available only in the user macro interrupt program. If they are used in other programs, program error (P241) results.

**(Note 1)** The programs are registered as files. When the program No. (file name) is read with #4515, the character string will be converted to a value.

**(Example 1)** The file name "123" is the character string 0x31, 0x32, 0x33, so the value will be  $(0x31-0x30)*100 + (0x32-0x30)*10 + (0x33-0x30) = 123.0$ .

Note that if the file name contains characters other than numbers, it will be "blank".

**(Example 2)** If the file name is "123ABC", characters other than numbers are included in the file name; so, the result will be "blank".



### Parameters

Refer to the Setup Manual for details on the setting methods.

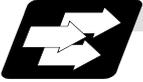
- (1) Subprogram call validity "#1229 set 01/bit 0"
  - 1 : Subprogram type user macro interrupt
  - 0 : Macro type user macro interrupt
- (2) Status trigger mode validity "#1112 S\_TRG"
  - 1 : Status trigger mode
  - 0 : Edge trigger mode
- (3) Interrupt type 2 validity "#1113 INT\_2"
  - 1 : The executable statements in the interrupt program are executed after completion of execution of the current block. (Type 2)
  - 0 : The executable statements in the interrupt program are executed before completion of execution of the current block. (Type 1)
- (4) Validity of alternate M code for user macro interrupt control "#1109 subs\_M"
  - 1 : Valid
  - 0 : Invalid



### Restrictions

- (1) If the user macro interrupt program uses system variables #5001 and after (position information) to read coordinates, the coordinates pre-read in the buffer are used.
- (2) If an interrupt is caused during execution of the tool radius compensation, a sequence number (G23 H\_\_;) must be specified with a command to return from the user macro interrupt program. If no sequence number is specified, control cannot return to the main program normally.

## 13.13 Tool Change Position Return; G30.1 to G30.6



## Function and purpose

By specifying the tool changing position in a parameter "#8206 TOOL CHG. P" and also specifying a tool changing position return command in a machining program, the tool can be changed at the most appropriate position.

The axes that are going to return to the tool changing position and the order in which the axes begin to return can be changed by commands.



## Command format

- (1) The format of tool changing position return commands is as follows.

**G30. n;**

n = 1 to 6 : Specify the axes that return to the tool changing position and the order in which they return.

For the commands and return order, see next table.

Command	Return order
G30.1	Z axis → X axis • Y axis ( → added axis)
G30.2	Z axis → X axis → Y axis ( → added axis)
G30.3	Z axis → Y axis → X axis ( → added axis)
G30.4	X axis → Y axis • Z axis ( → added axis)
G30.5	Y axis → X axis • Z axis ( → added axis)
G30.6	X axis • Y axis • Z axis ( → added axis)

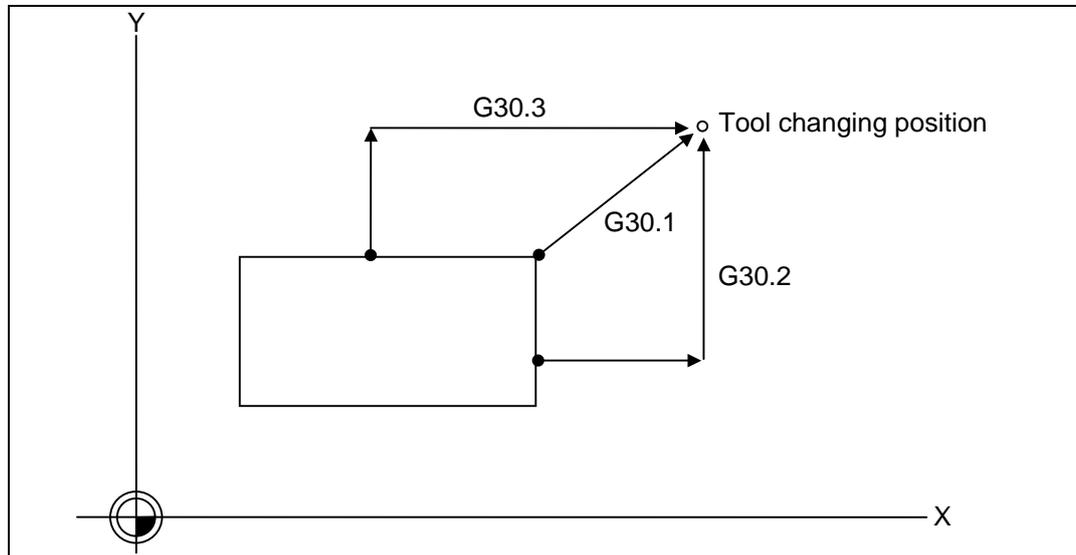
**(Note 1)** An arrow ( → ) indicates the order of axes that begin to return. An period ( • ) indicates that the axes begin to return simultaneously. (Example : "Z axis → X axis, Y axis" indicate that the Z axis returns to the tool changing position, then the X and Y axes does.)

- (2) The tool changing position return on/off for the added axis can be set with parameter "#1092 Tchg\_A" for the added axis.  
Note, however, that the added axis always return to the tool changing position only after the standard axes complete returning (see the above table).  
The added axis alone cannot return to the tool changing position.
- (3) When the axis address is commanded in the same block as the tool change position return command, the program error (P33) will occur.



## Example of operates

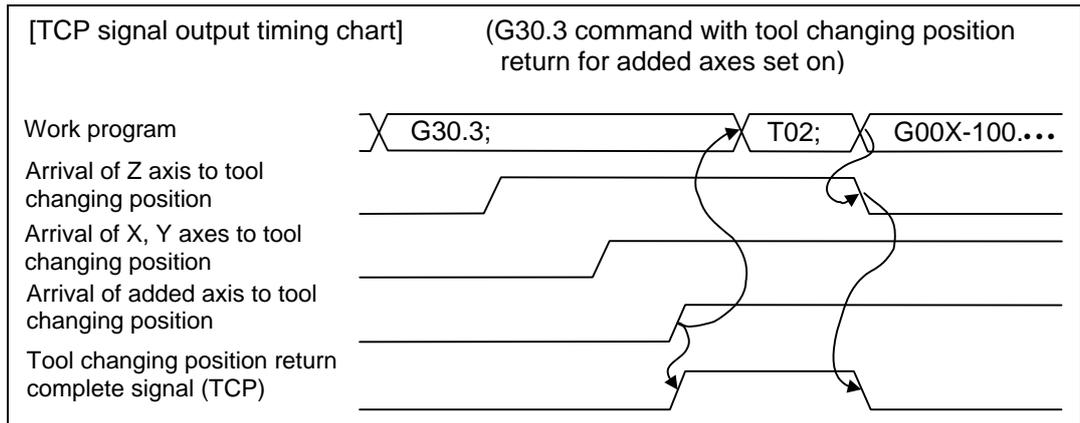
- (1) The figure below shows an example of how the tool operates during the tool change position return command. (Only operations of X and Y axes in G30.1 to G30.3 are figured.)



- (a) G30.1 command: The Z axis returns to the tool changing position, then the X and Y axes simultaneously do the same thing. (If tool changing position return is on for an added axis, the added axis also returns to the tool changing position after the X, Y and Z axes reach the tool changing position.)
- (b) G30.2 command: The Z axis returns to the tool changing position, then the X axis does the same thing. After that, the Y axis returns to the tool changing position. (If tool changing position return is on for an added axis, the added axis also returns to the tool changing position after the X, Y and Z axes reach the tool changing position.)
- (c) G30.3 command: The Z axis returns to the tool changing position, then the X axis does the same thing. After that, the X axis returns to the tool changing position. (If tool changing position return is on for an added axis, the added axis also returns to the tool changing position after the X and Z axes reach the tool changing position.)
- (d) G30.4 command: The X axis returns to the tool changing position, then the Y axis and Z axis simultaneously do the same thing. (If tool changing position return is on for an added axis, the added axis also return to the tool changing position after the X, Y and X axes reach the tool changing position.)
- (e) G30.5 command: The Y axis returns to the tool changing position, then the X and Z axes return to the tool changing position simultaneously. (If tool changing position return is on for an added axis, the added axis also returns to the tool changing position after the X, Y and Z axes reach the tool changing position.)
- (f) G30.6 command: The X, Y and Z axes return to the tool changing position simultaneously. (If tool changing position return is on for an added axis, the added axis also returns to the tool changing position after the X, Y and Z axes reach the tool changing position.)

- (2) After all necessary tool changing position return is completed by a G30.n command, tool changing position return complete signal TCP (XC93) is turned on. When an axis out of those having returned to the tool changing position by a G30.n command leaves the tool changing position, the TCP signal is turned off.

With a G30.1 command, for example, the TCP signal is turned on when the Z axis has reached the tool changing position after the X and Y axes did (after the additional axis did if additional axis tool change position return is valid). The TCP signal is then turned off when the X or Y axis leaves the position. If tool changing position return for added axes is on with parameter "#1092 Tchg\_A", the TCP signal is turned on when the added axis or axes have reached the tool changing position after the standard axes did. It is then turned off when one of the X, Y, Z, and added axes leaves the position.



- (3) When a tool changing position return command is issued, tool offset data such as for tool length offset and tool radius compensation for the axis that moved is canceled.
- (4) This command is executed by dividing blocks for every axis. If this command is issued during single-block operation, therefore, a block stop occurs each time one axis returns to the tool changing position. To make the next axis return to the tool changing position, therefore, a cycle start needs to be specified.

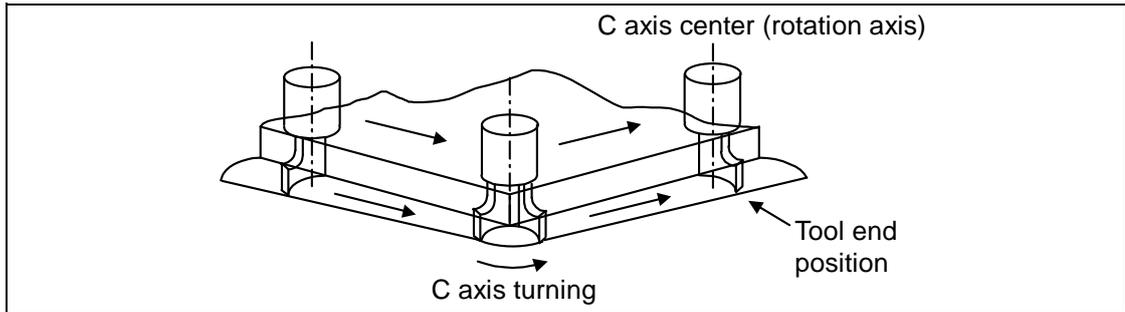
13.14 Normal Line Control ; G40.1/G41.1/G42.1



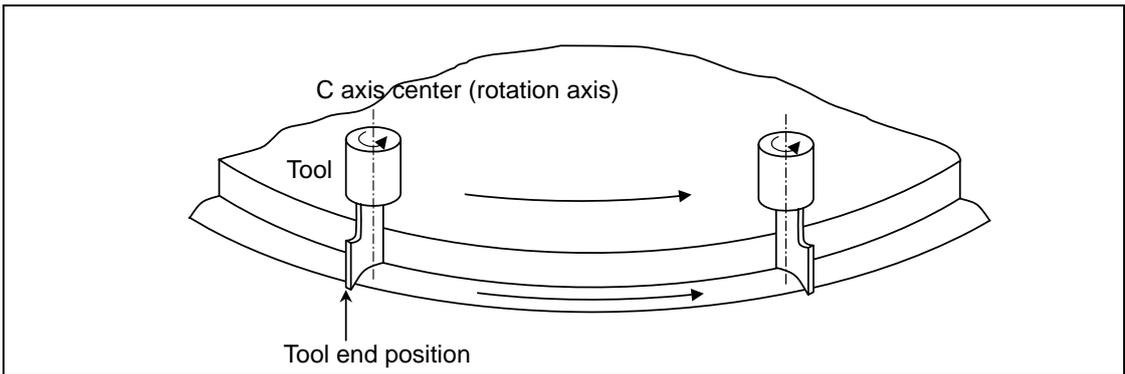
Function and purpose

If the C axis is set as the normal line control axis, the C axis (rotation axis) turning will be controlled so that the tool constantly faces the normal line direction control in respect to the XY axis movement command during program operation.

At the block seams, the C axis turning is controlled so that the tool faces the normal line direction at the next block's start point.



During arc interpolation, the rotation axis turning is controlled in synchronization with the operation of arc interpolation.



The normal line control I and II can be used according to the C axis turn direction during normal line control. This is set by the parameters.

Normal line control type	Turning direction	Turning speed	Turning speed in arc interpolation
Type I (#1524 C_type=0)	Direction that is 180° or less (shortcut direction)	Parameter speed (#1523 C_feed)	Speed that the program path follows the F command
Type II (#1524 C_type=1)	As a principle, the commanded direction	Feedrate	Speed that the tool nose follows the F command



## Command format

```

G40.1 X__ Y__ F__ ;
G41.1 X__ Y__ F__ ;
G42.1 X__ Y__1 F__ ;
G40.1      :Normal line control cancel
G41.1      :Normal line control left ON
G42.1      :Normal line control right ON
X          : X axis end point coordinates
Y          : Y axis end point coordinates
F          : Feedrate

```

The axis No. for the normal line control axis is designated with the parameter (#1522 C\_axis).  
 Normal line control is carried out in respect to the axis movement direction of the selected plane.  
 G17 plane X-Y axes  
 G18 plane Z-X axes  
 G19 plane Y-Z axes



## Detailed description

**(1) Definition of C axis turning direction**

The C axis angle is 0° (degree) when the tool is facing the +X direction.

The counterclockwise direction turning is + (plus), and the clockwise direction turning is – (minus).

**(2) C axis turning operation in respect to movement command****(a) Start up**

After the normal line control axis turns to a right angle of the advance direction at the start point of the normal line control command block, the axis of the selected plane is moved. Note that the normal line control axis at the start up turns in the direction that is 180° or less (shortcut direction) in both the normal line control type I and II.

**(b) During normal line control mode**

## 1. Block seam

No tool radius compensation	After the C axis is turned to be at a right angle of the movement of selected plane in the next block, that block is moved.
With tool radius compensation	If tool radius compensation is applied, normal line control is carried out along the path to which the tool radius compensation is applied.

## 2. During block movement

The C axis angle is kept constant during the linear command, and the C axis does not turn.

During the arc command, the C axis turns in synchronization with the operation of the arc interpolation.

**(c) Cancel**

The C axis does not turn, and the axis movement of the selected plane by the program command is carried out.

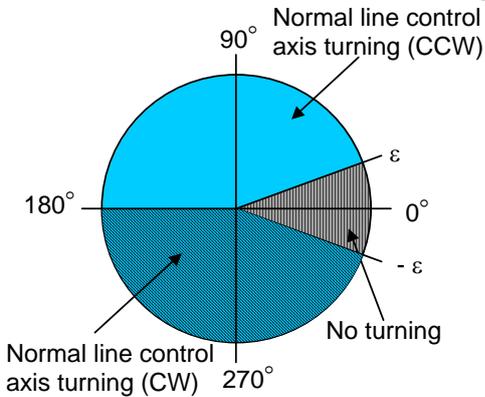
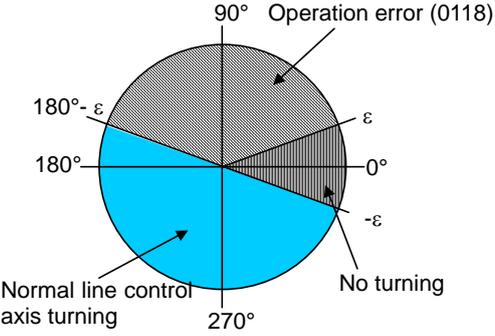
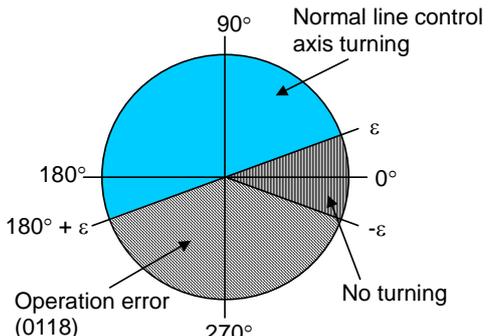
**(3) Normal line control temporally cancel**

During normal line control, the turning operation for the normal line control axis is not carried out at the seam of the block that the movement amount is smaller than that set with the parameter (#1535 C\_leng) and its previous block.

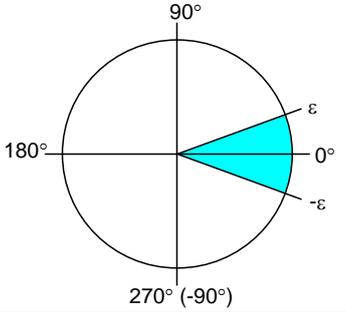
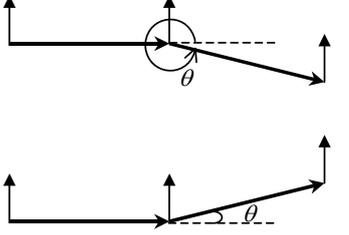
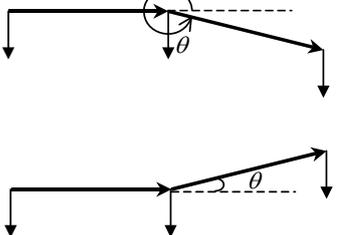
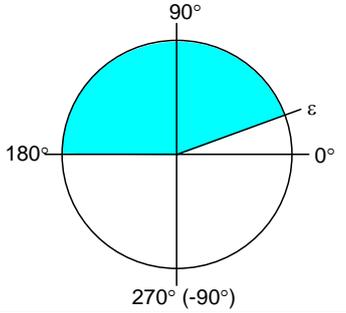
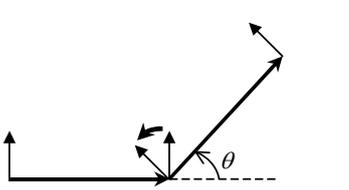
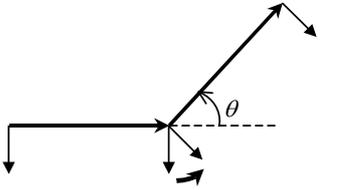
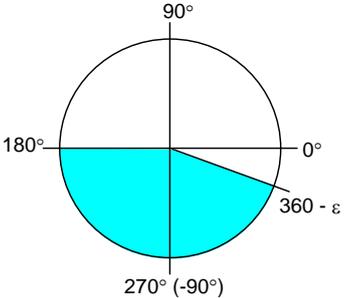
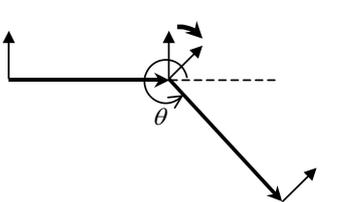
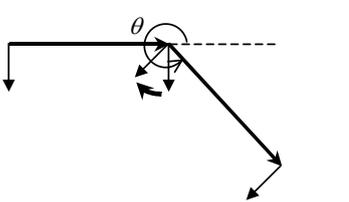
**(Note)** Since operation fractions are created by calculating the intersection point of two segments, the turning operation may be carried out or not when the parameter (#1535 C\_leng) and the segment length are equal.

**(4) Normal line control axis turning direction at block seam**

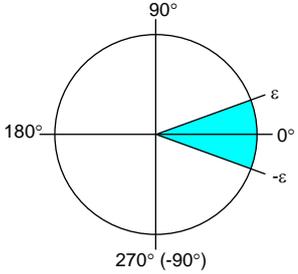
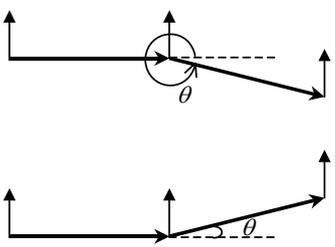
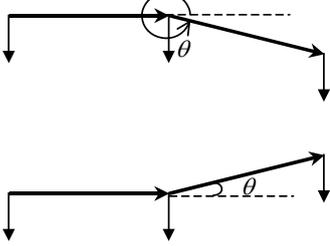
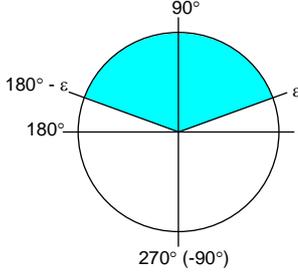
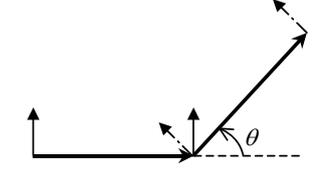
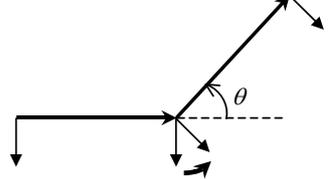
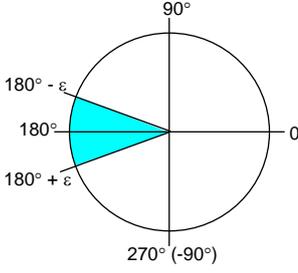
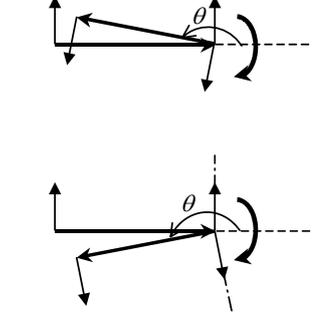
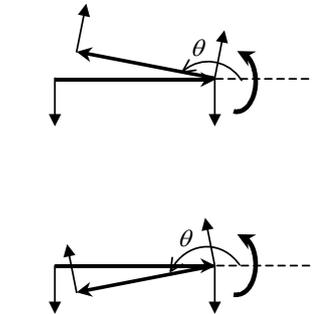
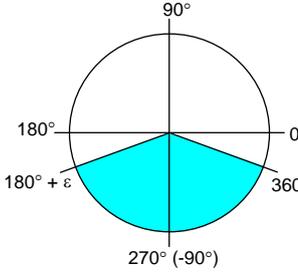
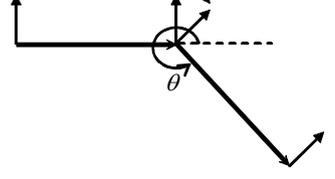
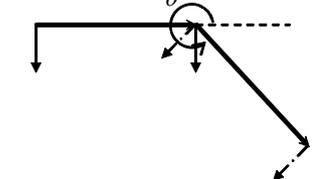
The normal line control axis turning direction at block seam differs according to the normal line control type I or II. The turning angle is limited by the angle  $\epsilon$  set with the parameter (#1521 C\_min).

Item	Type I	Type II
Normal line control axis turning direction at block seam	Direction that is 180° or less. (shortcut direction)	G41.1 : - direction (CW) G42.1 : + direction (CCW)
Normal line control axis turning angle at block seam	<ul style="list-style-type: none"> <li>when <math> \theta  &lt; \epsilon</math> is applied, turning is not performed. <math>\theta</math> : Turning angle <math>\epsilon</math> : Parameter (#1521 C_min)</li> <li>When the turning angle is 180°, the turning direction is indefinite regardless of the command mode.</li> </ul> <p>[G41.1/G42.1 When the normal line control axis is in 0°]</p> 	<ul style="list-style-type: none"> <li>when <math> \theta  &lt; \epsilon</math> is applied, turning is not performed. <math>\theta</math> : Turning angle <math>\epsilon</math> : Parameter (#1521 C_min)</li> <li>In the following cases, an operation error (0118) will occur. &lt;For G41.1&gt; <math>\epsilon \leq \theta &lt; 180^\circ - \epsilon</math> &lt;For G42.1&gt; <math>180^\circ + \epsilon &lt; \theta \leq 360^\circ - \epsilon</math></li> </ul> <p>[G41.1 When the normal line control axis is in 0°]</p>  <p>[G42.1 When the normal line control axis is in 0°]</p> 

(a) Normal line control type I

Normal line control axis turning angle at block seam: $\theta$	G41.1	G42.1
<p>1. <math>-\varepsilon &lt; \theta &lt; \varepsilon</math></p> 	 <p>No turning</p>	 <p>No turning</p>
<p>2. <math>\varepsilon \leq \theta &lt; 180^\circ</math></p> 		
<p>3. <math>180^\circ \leq \theta \leq 360^\circ - \varepsilon</math></p> 	 <p>Shortcut direction</p>	 <p>Shortcut direction</p>

(b) Normal line control type II

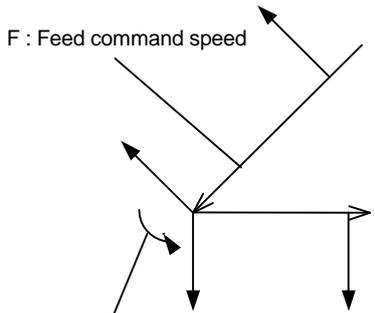
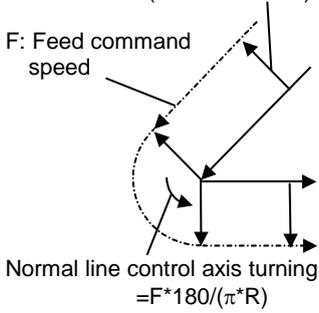
Normal line control axis turning angle at block seam: $\theta$	G41.1	G42.1
<p>1. <math>-\varepsilon &lt; \theta &lt; \varepsilon</math></p> 	 <p>No turning</p>	 <p>No turning</p>
<p>2. <math>\varepsilon \leq \theta &lt; 180^\circ - \varepsilon</math></p> 	 <p>Operation error (0118) (Note)</p>	
<p>3. <math>180^\circ - \varepsilon \leq \theta \leq 180^\circ + \varepsilon</math></p> 		
<p>3. <math>180^\circ + \varepsilon &lt; \theta \leq 360^\circ - \varepsilon</math></p> 		 <p>Operation error (0118) (Note)</p>

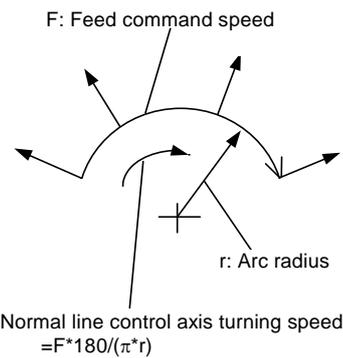
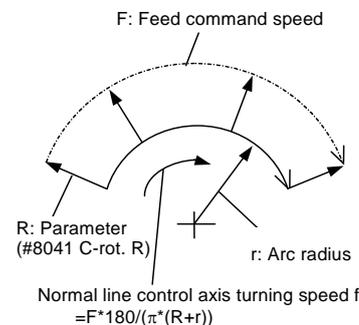
**(Note)** Turning operation in the command direction is performed in the inside of the workpiece. Therefore, an operation error will occur.

**(5) C axis turning speed**

Turning speed at block seam (select from type 1 or type 2)

Item	Type 1	Type 2
<p>Normal line control axis turning speed at block seam</p>	<p>(a) Rapid traverse</p> <ul style="list-style-type: none"> <li>• Dry run OFF                             <p>The rapid traverse rate (#2001 rapid) is applied.</p> <p>Normal line control axis turning speed <math>f</math> = Rapid traverse rate * (Rapid traverse override) (°/min)</p> </li> <li>• Dry run ON                             <p>The manual feedrate is applied.</p> <p>Normal line control axis turning speed <math>f</math> = Manual feedrate * (Cutting feed override) (°/min)</p> </li> </ul> <p>(Note 1) When the manual override valid is ON, the cutting feed override is valid.</p> <p>(Note 2) If the normal line control axis turning speed exceeds the cutting feed clamp speed (#2002 clamp), the speed is applied to the cutting feed clamp speed.</p> <p>(Note 3) When the rapid traverse is ON, the dry run is invalid.</p> <p style="text-align: right;">(Continued to the next page)</p>	<p>(a) Rapid traverse</p> <ul style="list-style-type: none"> <li>• Dry run OFF                             <p>Normal line control axis turning speed <math>f</math> = <math>F * 180 / (\pi * R) * (\text{Rapid traverse override})</math> (°/min)</p> <p>For <math>R=0</math>, the following expression is applied.</p> <p>Normal line control axis turning speed <math>f</math> = <math>F * (\text{Rapid traverse override})</math> (°/min)</p> <p>F: Rapid traverse rate (#2001 rapid) (mm/min)</p> <p>R: Parameter (#8041 C-rot.R) (mm) (Length from normal line control axis center to tool nose)</p> </li> <li>(Note 1) If the normal line control axis turning speed exceeds the rapid traverse rate (#2001 rapid), the speed is clamped to the rapid traverse rate.</li> <li>• Dry run ON                             <p>Normal line control axis turning speed <math>f</math> = <math>F * 180 / (\pi * R) * (\text{Cutting feed override})</math> (°/min)</p> <p>For <math>R=0</math>, the following expression is applied.</p> <p>Normal line control axis turning speed <math>f</math> = <math>F * (\text{Cutting feed override})</math> (°/min)</p> <p>F: Manual feedrate (mm/min)</p> <p>R: Parameter (#8041 C-rot.R) (mm) (Length from normal line control axis center to tool nose)</p> </li> <li>(Note 1) When the manual override valid is ON, the cutting feed override is valid.</li> <li>(Note 2) If the normal line control axis turning speed exceeds the rapid traverse rate (#2001 rapid), the speed is applied to the rapid traverse rate.</li> <li>(Note 3) When the rapid traverse is ON, the dry run is invalid.</li> </ul> <p style="text-align: right;">(Continued to the next page)</p>

Item	Type 1	Type 2
<p>Normal line control axis turning speed at block seam</p>	<p>(b) Cutting feed</p> <ul style="list-style-type: none"> <li>• Dry run OFF The normal line control axis turning speed set with the parameter (#1523 C_feed) is applied. Normal line control axis turning speed <math>f</math> = Parameter (#1523 C_feed) * (Cutting feed override) (°/min)</li> <li>• Dry run ON (Rapid traverse ON) The cutting feed clamp speed (#2002 clamp) is applied. Normal line control axis turning speed <math>f</math> = Cutting feed clamp speed (°/min)</li> <li>• Dry run ON (Rapid traverse OFF) The manual feedrate is applied. Normal line control axis turning speed <math>f</math> = Manual feedrate * (Cutting feed override) (°/min)</li> </ul> <p>(Note 1) When the manual override valid is ON, the cutting feed override is valid.</p> <p>(Note2) If the normal line control axis turning speed exceeds the cutting feed clamp speed (#2002 clamp), the speed is clamped to the cutting feed clamp speed.</p>  <p><math>f</math> : Normal line control axis turning speed = parameter (#1523 C_feed)</p> <p>(Continued to the next page)</p>	<p>(b) Cutting feed</p> <p>The feedrate at the tool nose is the F command. The normal line control axis turning speed is the normal line control axis speed that follows this F command.</p> <p>Normal line control axis turning speed <math>f</math> = <math>F * 180 / (\pi * R) * (\text{Cutting feed override})</math> (°/min)</p> <p>For <math>R=0</math>, the following expression is applied. Normal line control axis turning speed <math>f</math> = <math>F</math> (°/min) F: Feedrate command (mm/min) R: Parameter (#8041 C-rot.R) (mm) (Length from normal line control axis center to tool nose)</p> <p>(Note 1) If the normal line control axis turning speed exceeds the cutting feed clamp speed (#2002 clamp), the speed is applied to the cutting feed clamp speed.</p> <p>(Note2) When the dry run is ON, the normal line control axis turning speed is obtained with the same expression as the rapid traverse.</p> <p>R: Parameter (#8041 C-rot. R) F: Feed command speed</p>  <p>Normal line control axis turning speed <math>f</math> = <math>F * 180 / (\pi * R)</math></p> <p>(Continued to the next page)</p>

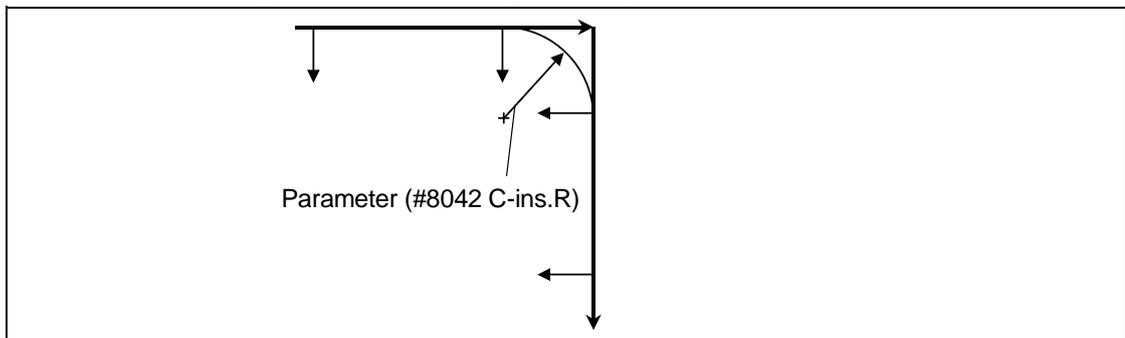
Item	Type 1	Type 2
<p>Normal line control axis turning speed during circular interpolation</p>	<p>The normal line control axis turning speed is the rotation speed obtained by feedrate F.</p> <p>Normal line control axis turning speed <math>f</math>  <math>= F \cdot 180 / (\pi \cdot r)</math> (degree/min)                      F : Feed command speed (mm/min)                      r : Arc radius (mm)</p> <p style="text-align: right;"><b>(Note 1)</b></p>  <p>Normal line control axis turning speed <math>f</math>  <math>= F \cdot 180 / (\pi \cdot r)</math></p>	<p>The feedrate at the tool nose is the F command. The normal line control axis turning speed is the rotation speed that follows this F command.</p> <p>Normal line control axis turning speed <math>f</math>  <math>= F \cdot 180 / (\pi \cdot (R+r))</math> (degree/min)                      F : Feed command speed (mm/min)                      R : Parameter (#8041 C-rot. R) (mm)                      Length from normal line control axis center to tool nose                      r : Arc radius (mm)</p>  <p>Normal line control axis turning speed <math>f</math>  <math>= F \cdot 180 / (\pi \cdot (R+r))</math></p>

- (Note 1)** If the normal line control axis turning speed exceeds the cutting feed clamp speed (#2002 clamp), the speed will be as follows;
- Normal line control axis turning speed = Cutting feed clamp speed.
  - Moving speed during arc interpolation = The speed according to the normal line control axis turning speed



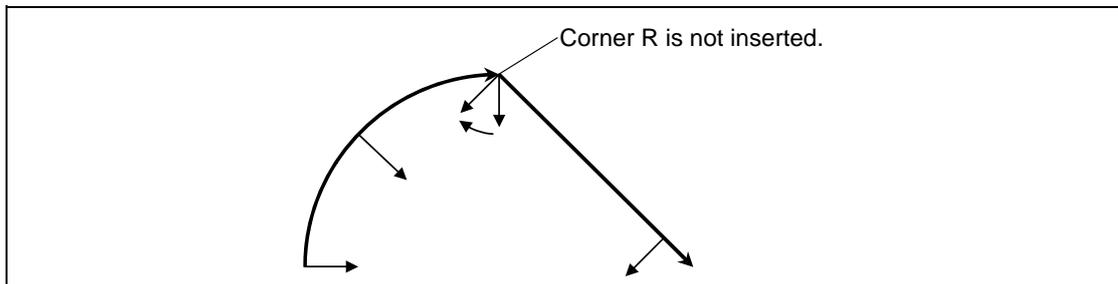
**Automatic corner arc insertion function**

During normal line control, an arc is automatically inserted into the corner in the axis movement of the plane selection. This function is for the normal line control type I.  
 The radius of the arc to be inserted is set with the parameter (#8042 C-ins.R).  
 This parameter can be read and write using the macro variable #1901.  
 Normal line control is performed also during the interpolation for the arc to be inserted.

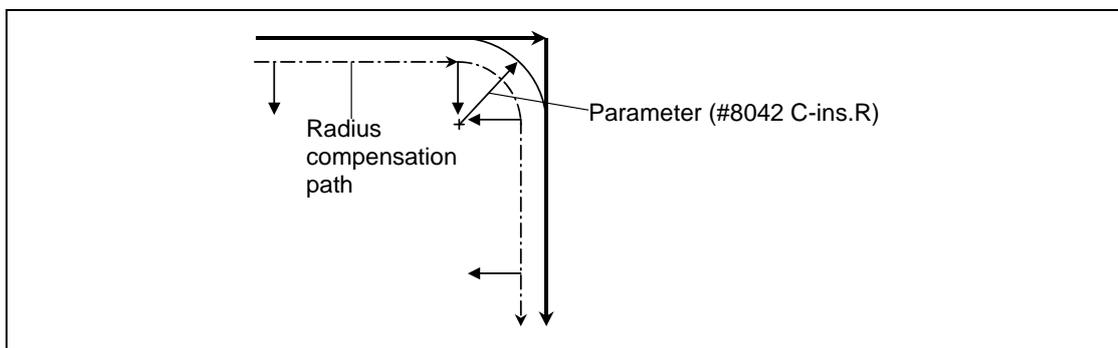


<Supplements>

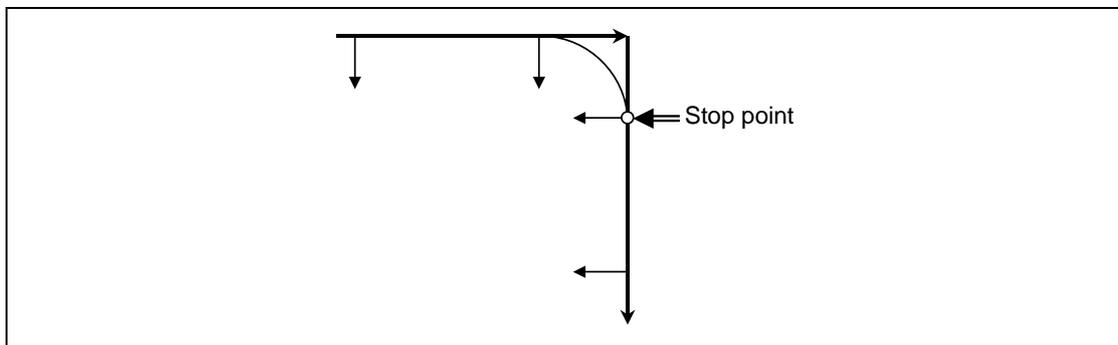
The corner arc is not inserted into the straight line that is smaller than a linear-arc, arc-arc, linear-block with no movement, block with no movement-linear or radius of the arc to be inserted.



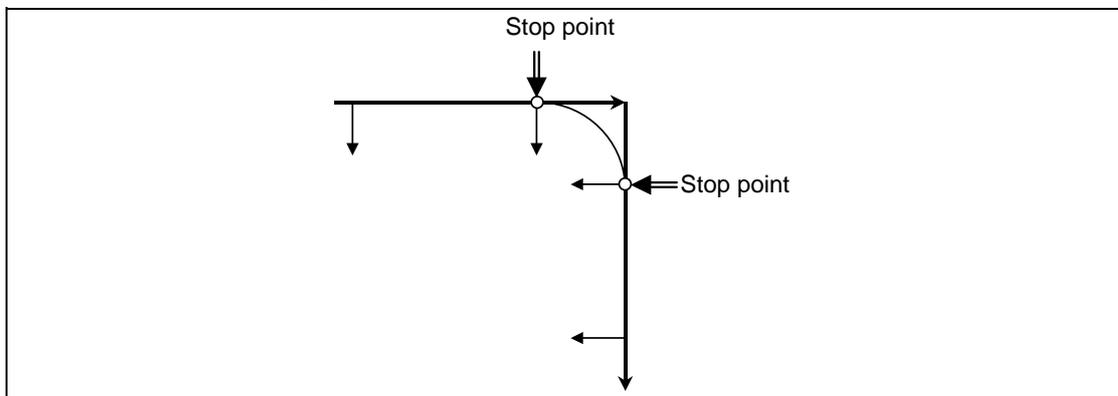
During the radius compensation, the radius compensation is applied to the path that the corner arc is inserted into.



The stop point of the single block and block start interlock is as follows.



The stop point of the cutting start interlock is as follows.





## Precautions

- (1) During normal line control, the program coordinates are updated following the normal line control axis movement.  
Thus, program the normal line control with the program coordinate system.
- (2) The normal line control axis will stop at the turning start position at the single block, cutting block start interlock and block start interlock.
- (3) The C axis movement command is ignored during normal line control.
- (4) During C axis normal line control (during the G41.1 and G42.1 modal), the C axis workpiece offset rewrite command (G92C\_;) cannot be issued. The program error (P901) will occur if commanded.
- (5) If mirror image is applied to either the 1-axis or 2-axis, the normal line control direction will be reversed.
- (6) Designate the rotation axis for the normal line control axis (parameter (#1522 C\_axis)). Designate so that the axis is not duplicated with the axis on the plane where normal line control is to be carried out.  
If an illegal axis is designated, the program error (P902) will occur when the program (G40.1, G41.1, G42.1) is commanded.  
The program error (P902) will also occur if the parameter (#1522 C\_axis) is "0" when these codes are commanded.
- (7) Depending on the model, this function cannot be used.
- (8) The movement of the normal line control axis is counted as one axis of number of simultaneous contouring control axes.  
If the number of simultaneous contouring control axes exceeds the specification range by movement of the normal line control axis, the program error (P10) will occur.



## Relation with other functions

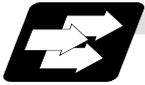
Function name	Notes
Uni-directional positioning	Normal line control is not applied.
Helical cutting	Normal line control is applied normally.
Spiral interpolation	The start point and end point are not on the same arc, so normal line control is not applied correctly.
Exact stop check	The operation will not decelerate and stop for the turning movement of the normal line control axis.
Error detect	Error detect is not applied on the turning movement of the normal line control axis.
Override	Override is applied on the turning movement by normal line control axis.
Coordinate rotation by program	Normal line control is applied to the shape after coordinate rotation.
Scaling	Normal line control is applied to the shape after scaling.
Mirror image	Normal line control is applied to the shape after mirror image.
Thread cutting	Normal line control is not applied.
Automatic reference position return	Normal line control is not applied.
Start position return	Normal line control is not applied on the movement to the middle point position. If the base specification parameter "#1086 G0Intp" is 0, normal line control is applied to the movement from the middle point to a position designated in the program.

(Continued to the next page)

(Continued from the previous page)

Function name	Notes
High-accuracy control	This cannot be commanded during normal line control. A program error (P29) will occur. The normal line control command during high-accuracy control cannot also be issued. A program error (P29) will occur.
Spline	This cannot be commanded during normal line control. A program error (P29) will occur. The normal line control command during spline cannot also be issued. A program error (P29) will occur.
High-speed High-accuracy control I/II	This cannot be commanded during normal line control. A program error (P29) will occur. The normal line control command during high-speed High-accuracy control I/II cannot also be issued. A program error (P29) will occur.
Cylindrical interpolation	This cannot be commanded during normal line control. A program error (P486) will occur. The normal line control command during cylindrical interpolation cannot also be issued. A program error (P481) will occur.
Workpiece coordinate system offset	The workpiece coordinate system cannot be changed during normal line control. A program error (P29) will occur. The program parameter input (G10L2) cannot also be commanded. A program error (P29) will occur.
Local coordinate system offset	The local coordinate system cannot be changed during normal line control. A program error (P29) will occur.
Program restart	The program including the normal line control command cannot be restarted. "E98 CAN'T RESEARCH" will occur.
Dry run	The feedrate is changed by the dry run signal even in respect to the turning movement of the normal line control axis.
Chopping	The axis cannot be used as the normal line control axis during the chopping command.
Graphic check	The section turned by normal line control is not drawn. The path is drawn for the axes subjected to the graphic check.
G00 non-interpolation	Normal line control is not applied.
Pole coordinate interpolation	This cannot be commanded during normal line control. A program error (P486) will occur. The normal line control command during pole coordinate interpolation cannot also be issued. A program error (P481) will occur.
Exponential interpolation	If the normal line control axis is the same as the rotation axis of exponential interpolation, a program error (P612) will occur. If the normal line control axis is different from the rotation axis of exponential interpolation, an error will not occur, however normal line control is not applied.
Plane selection	This cannot be commanded during normal line control. A program error (P903) will occur.
System variable	The block end coordinate (#5001~) for the normal line control axis during normal line control cannot obtain a correct axis position.

## 13.15 High-accuracy Control ; G61.1, G08

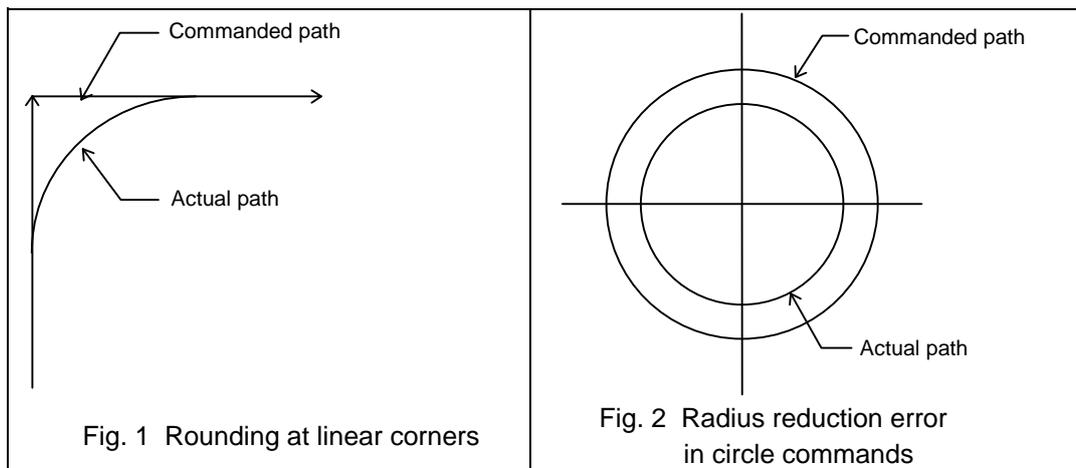


## Function and purpose

This function aims to improve the error caused by the accuracy of the control system during machine machining. The parameter method and G code command method, which turn initial high-accuracy ON, are used to enter the high-accuracy control mode.

Trouble such as the following occurred when using normal control:

- (1) Corner rounding occurred at linear and linear-connected corners because the following command movement started before the previous command finished. (Refer to Fig. 1)
- (2) When cutting circle commands, an error occurred further inside the commanded path, and the resulting cutting path was smaller than the commanded path. (Refer to Fig. 2)



This function uses the following fix functions to minimize the increase in machining time while reducing the shape error.

- (1) Pre-interpolation acceleration/deceleration (linear acceleration/deceleration)
- (2) Optimum speed control
- (3) Vector accuracy interpolation
- (4) Feed forward
- (5) Arc entrance/exit speed control
- (6) S-pattern filter control



## Command format

**G61.1 F\_\_ ;**

G61.1 : High-accuracy control mode ON  
 F : Feedrate command

The high-accuracy control mode is validated from the block containing the G61.1 command.

The "G61.1" high-accuracy control mode is canceled with one of the G code group 13's functions.

- G61 (Exact stop check mode)
- G62 (Automatic corner override)
- G63 (Tapping mode)
- G64 (Cutting mode)
- G08 P1 (High-accuracy control mode)

**G08 P1(P0) ;**

G08 : High-accuracy control mode  
 P1 : High-accuracy control mode start  
 P0 : High-accuracy control mode cancel

The "G08 P1" high-accuracy control mode is canceled with P0.

Command G08P\_ in an independent block.

The decimal places below the decimal point are ignored for P address.

**(Note)** G code group for in G08 is "0"; the priority is given to the function of the G code group 0 over the function of the G code group 13.

After "G08 P1" is commanded, G code group 13 is changed automatically to G64 (cutting) mode. Other command of "13" results in error.

Even if high-accuracy control mode is canceled by "G08 P0" command, G64 (cutting) mode will not be changed.

If you want to return to the function of G code group "13" when "G08 P1" has been commanded, command again after high-accuracy control mode is canceled.



Detailed description

- (1) Feedrate command F is clamped with "#2110 Clamp(H-precision)" (Cutting feedrate during high-accuracy control mode for clamp function) set by the parameter.
- (2) Rapid traverse rate enables "#2109 Rapid(H-precision)" (Rapid traverse rate during high-accuracy control mode) set by the parameter.
- (3) When the "#2109 Rapid(H-precision)" is set to "0", the movement follows "2001 rapid" (rapid traverse rate) set by the parameter. Also, when the setting value for #2110 Clamp(H-precision) is "0", the feedrate is clamped with #2002 clamp (cutting clamp speed) set by the parameter.
- (4) The modal holding state of the high-accuracy control mode differs according to the combination of the base specification parameter "#1151 rstint" (reset initial) and "#1148 I\_G611" (initial high-accuracy).

Parameter		Default state	Reset			Emergency stop	Emergency stop cancel	Block interruption	Block stop	NC alarm	OT
Reset initial (#1151)	Initial high accuracy (#1148)	Power ON	Reset 1	Reset 2	Reset & rewind	Emergency stop switch External emergency stop	Emergency stop switch External emergency stop	Mode changeover (automatic/manual/Feed hold)	Single block	Servo alarm	H/W OT
OFF	OFF	OFF	H	OFF		H	H	H			
ON			OFF				OFF				
OFF	ON	ON	H	ON		H	H				
ON			ON				ON				

H (hold): Modal hold

ON: Switches to high-accuracy mode

As for G61.1, the mode is switched to the high-accuracy mode, even if the other modes (G61 to G64) are valid.

OFF: The status of the high-accuracy mode is OFF.



Pre-interpolation acceleration/deceleration

Acceleration/deceleration control is carried out for the movement commands to suppress the impact when the machine starts or stops moving. However, with conventional post-interpolation acceleration/deceleration, the corners at the block seams are rounded, and path errors occur regarding the command shape.

In the high-accuracy control function mode, acceleration/deceleration is carried out before interpolation to solve the above problems. This pre-interpolation acceleration/deceleration enables machining on a machining path that more closely follows the command.

The acceleration/deceleration time can be reduced because constant inclination acceleration/deceleration is carried out.

(1) Basic patterns of acceleration/deceleration control in linear interpolation commands

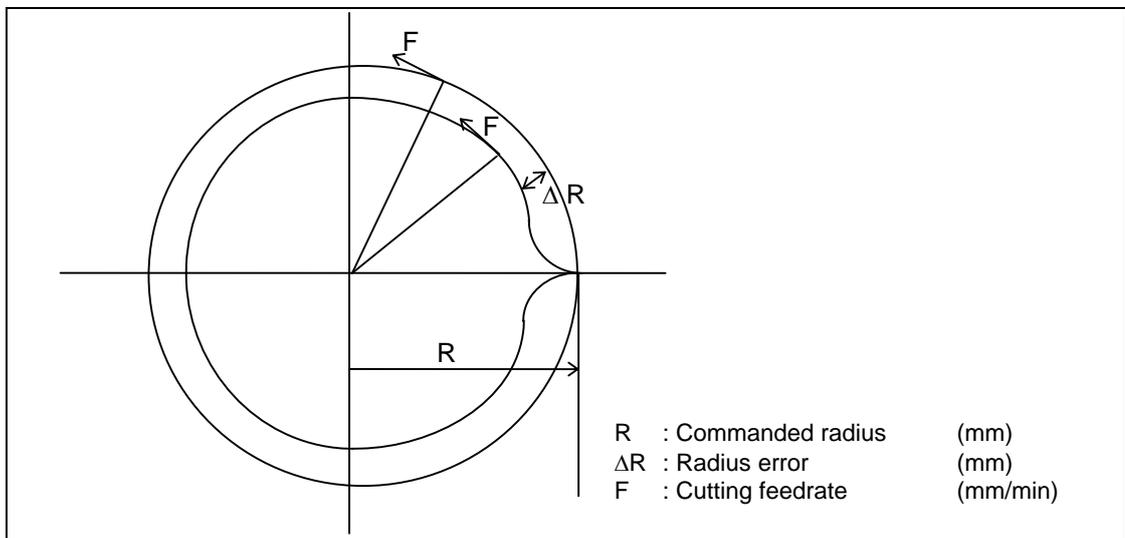
	Acceleration/deceleration waveform pattern	
Normal mode		<p>(a) Because of the constant time constant acceleration/deceleration, the rising edge/falling edge of the waveform becomes more gentle as the command speed becomes slower.</p> <p>(b) The acceleration/deceleration time constant can be independently set for each axis. Linear type, exponential function type, or both can be selected. Note that if the time constant of each axis is not set to the same value, an error will occur in the path course.</p> <p>#2002 clamp : G01 clamp speed                      #2007 G1tL : Linear type acceleration/ deceleration time constant                      #2008 G1t1 : Exponential type acceleration/ deceleration time constant</p>
High-accuracy control mode		<p>(a) Because of the constant inclination type linear acceleration/deceleration, the acceleration/deceleration time is reduced as the command speed becomes slower.</p> <p>(b) The acceleration/deceleration time constant becomes one value (common for each axis) in the system.</p> <p>#2002 clamp : G01 clamp speed                      #1206 G1bF : Target speed                      #1207 G1btL : Acceleration/deceleration time to target speed</p> <p><b>(Note)</b> G1bF and G1btL are values for specifying the inclination of the acceleration/ deceleration time; the actual cutting feed maximum speed is clamped by the "#2002 clamp" value.</p>

(2) Path control in circular interpolation commands

When commanding circular interpolation with the conventional post-interpolation acceleration/deceleration control method, the path itself that is output from the CNC to the servo runs further inside the commanded path, and the circle radius becomes smaller than that of the commanded circle. This is due to the influence of the smoothing course droop amount for CNC internal acceleration/deceleration.

With the pre-interpolation acceleration/deceleration control method, the path error is eliminated and a circular path faithful to the command results, because interpolation is carried out after the acceleration/deceleration control. Note that the tracking lag due to the position loop control in the servo system is not the target here.

The following shows a comparison of the circle radius reduction error amounts for the conventional post-interpolation acceleration/deceleration control and pre-interpolation acceleration/deceleration control in the high-accuracy control mode.



The compensation amount of the circle radius reduction error (ΔR) is theoretically calculated as shown in the following table.

Post-interpolation acceleration/deceleration control (normal mode)	Pre-interpolation acceleration/deceleration control (high-accuracy control mode)
Linear acceleration/deceleration $\Delta R = \frac{1}{2R} \left( \frac{1}{12} T_s^2 + T_p^2 \right) \left( \frac{F}{60} \right)^2$	Linear acceleration/deceleration $\Delta R = \frac{1}{2R} \left\{ T_p^2 (1 - K_f^2) \right\} \left( \frac{F}{60} \right)^2$
Exponential function acceleration/deceleration $\Delta R = \frac{1}{2R} (T_s^2 + T_p^2) \left( \frac{F}{60} \right)^2$	(a) Because the item $T_s$ can be ignored by using the pre-interpolation acceleration/deceleration control method, the radius reduction error amount can be reduced. (b) Item $T_p$ can be negated by making $K_f = 1$ .

$T_s$  : Acceleration/deceleration time constant in the CNC (s)  
 $T_p$  : Servo system position loop time constant (s)  
 $K_f$  : Feed forward coefficient

**(Note)** When a speed is set to "#2110 Clamp (H-precision)", which is the cutting clamp speed parameter for the high-accuracy control mode, clamping will be carried out at that speed.



Optimum speed control

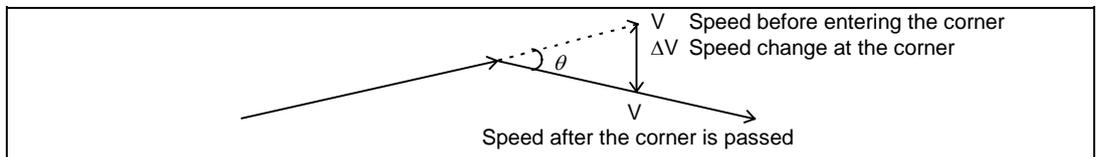
(1) Optimum corner deceleration

By calculating the angle of the seam between blocks, and carrying out acceleration/ deceleration control in which the corner is passed at the optimum speed, highly accurate edge machining can be realized.

When the corner is entered, that corner's optimum speed (optimum corner speed) is calculated from the angle with the next block. The machine decelerates to that speed in advance, and then accelerates back to the command speed after the corner is passed.

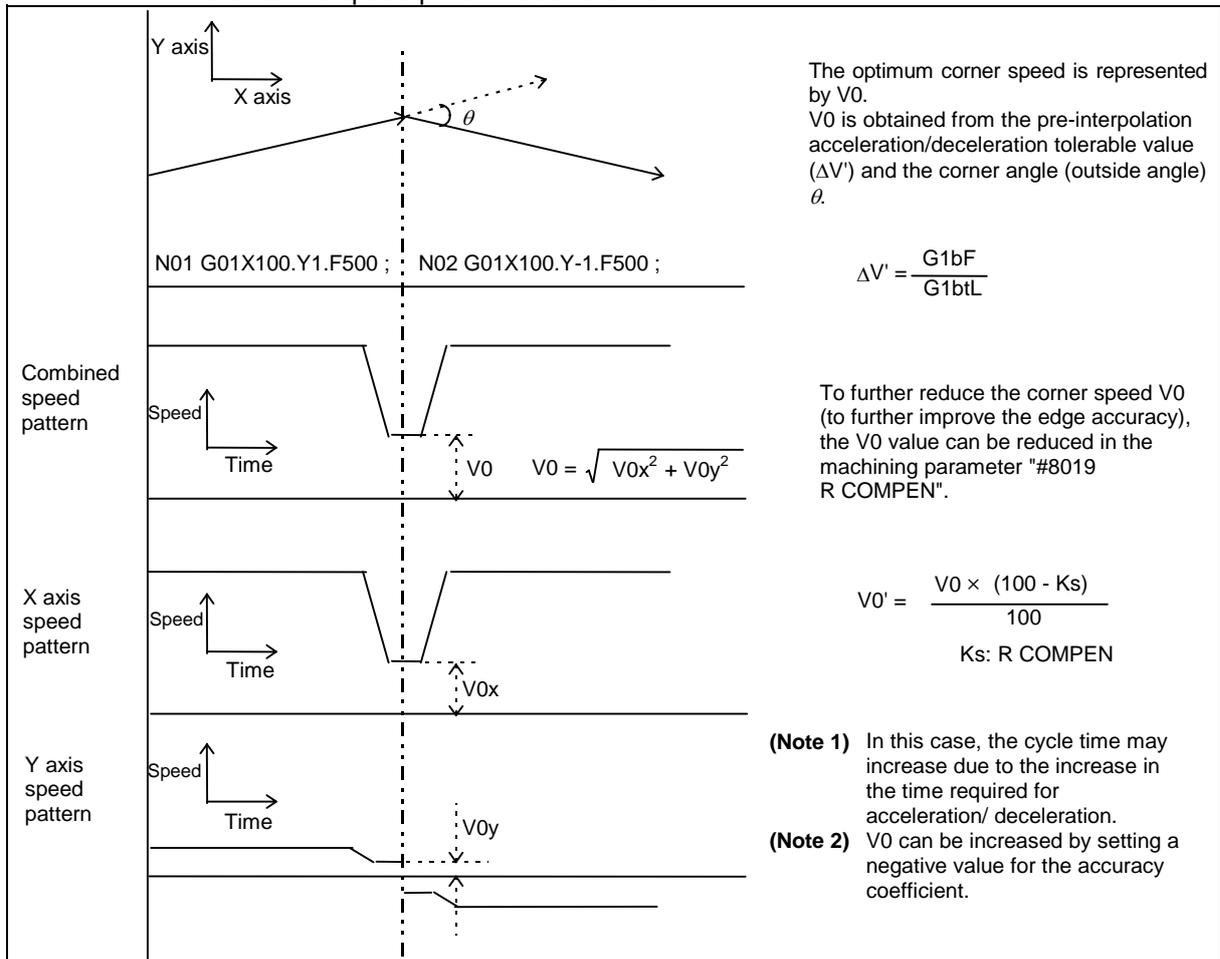
Corner deceleration is not carried out when blocks are smoothly connected. In this case, the criteria for whether the connection is smooth or not can be designated by the machining parameter "#8020 DCC ANGLE".

When the corner angle is larger than the parameter "DCC ANGLE" for a linear-linear connection, or for a circle, etc., the acceleration  $\Delta V$  occurs due to the change in the direction of progress after the corner is passed at a speed  $V$ .



The corner angle  $V$  is controlled so that this  $\Delta V$  value becomes less than the pre-interpolation acceleration/ deceleration tolerable value set in the parameters ("#1206 G1bF", "#1207 G1btL").

In this case the speed pattern is as follows.

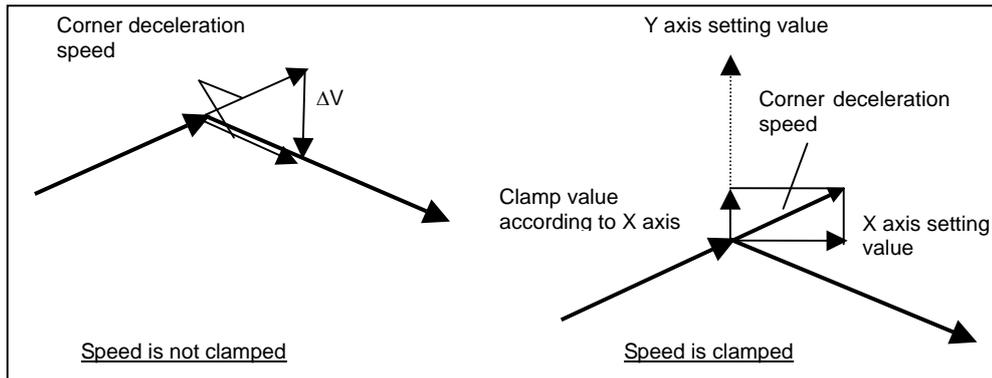


The accuracy coefficient differs according to parameter "#8201 COMP CHANGE".

#8201 COMP CHANGE	Accuracy coefficient used
0	#8019 R COMPEN
1	#8022 CORNER COMP

The corner speed V0 can be maintained at a set speed or more so that the corner speed does not drop too far.

Set "#2096 crncsp (corner deceleration minimum speed)" for each axis, and make a composite speed so that the moving axis does not exceed this setting.



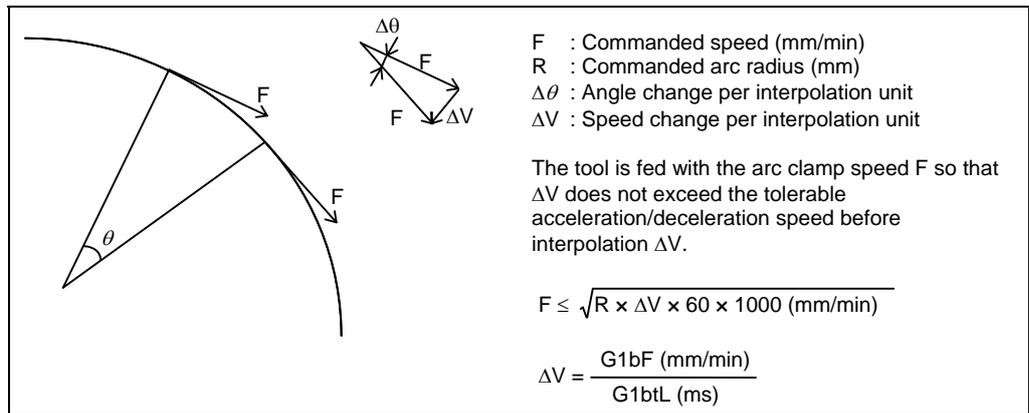
Note that the speed is controlled with the optimum corner deceleration speed in the following cases.

- When the composite corner deceleration speed is less than the optimum corner deceleration speed.
- When the corner deceleration minimum speed parameter setting for the moving axes is set to "0" for even one axis.

(2) Arc speed clamp

During circular interpolation, even when moving at a constant speed, acceleration is generated as the advance direction constantly changes. When the arc radius is large compared to the commanded speed, control is carried out at the commanded speed. However, when the arc radius is relatively small, the speed is clamped so that the generated acceleration does not exceed the tolerable acceleration/deceleration speed before interpolation, calculated with the parameters.

This allows arc cutting to be carried out at an optimum speed for the arc radius.



When the above  $F'$  expression is substituted in the expression expressing the maximum logical arc radius reduction error amount  $\Delta R$  explained in the section "a) Pre-interpolation acceleration/deceleration", the commanded radius  $R$  is eliminated, and  $\Delta R$  does not rely on  $R$ .

$$\Delta R \leq \frac{1}{2R} \{Tp^2 (1 - Kf^2)\} \left(\frac{F}{60}\right)^2$$

$$\leq \frac{1}{2} \{Tp^2 (1 - Kf^2)\} \left(\frac{\Delta V' \times 1000}{60}\right)$$

$\Delta R$ : Arc radius reduction error amount  
 $Tp$ : Position loop gain time constant of servo system  
 $Kf$ : Feed forward coefficient  
 $F$ : Cutting feedrate

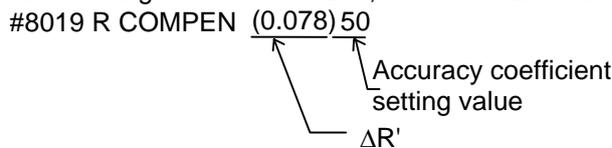
In other words, with the arc command in the high-accuracy control mode, in logical terms regardless of the commanded speed  $F$  or commanded radius  $R$ , machining can be carried out with a radius reduction error amount within a constant value.

To further lower the arc clamp speed (to further improve the roundness), the arc clamp speed can be lowered with the machining parameter "#8019 R COMPEN". In this case, speed control is carried out to improve the maximum arc radius reduction error amount  $\Delta R$  by the set percentage.

$$\Delta R' \leq \frac{\Delta R \times (100 - Ks)}{100} \text{ (mm)}$$

$\Delta R'$ : Maximum arc radius reduction error amount  
 $Ks$ : R COMPEN (%)

After setting the "R COMPEN", the above  $\Delta R'$  will appear on the parameter screen.



**(Note 1)** The maximum arc radius reduction error amount  $\Delta R$  can be increased by setting a negative value for the "accuracy coefficient".

**(Note 2)** When the "R COMPEN" is set with positive value, the arc clamp speed will drop, so in a machining program with many arc commands, the machining time will take longer.

**(Note 3)** The "R COMPEN" is valid only when the arc speed clamp is applied. To reduce the radius reduction error when not using the arc speed clamp, the commanded speed F must be lowered.

**(Note 4)** If the "accuracy coefficient" is not set (0), arc speed clamping will not be applied.

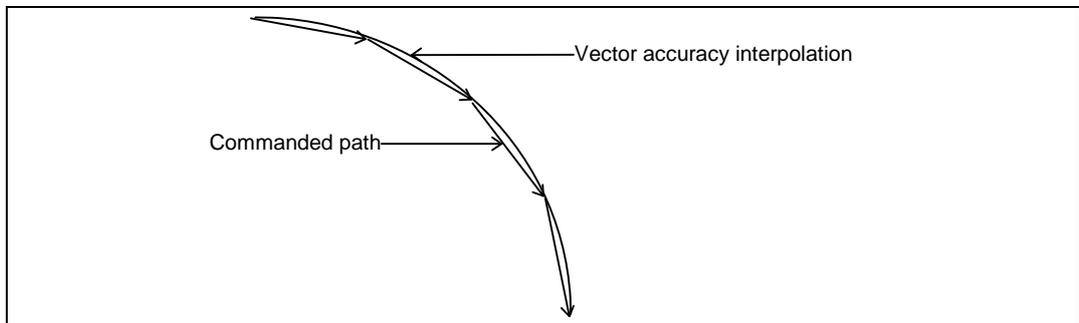
**(Note 5)** The "accuracy coefficient" differs according to parameter "#8021 COMP CHANGE".

#8201 COMP CHANGE	Accuracy coefficient used
0	#8019 R COMPEN
1	#8023 CURVE COMP



**Vector accuracy interpolation**

When a fine segment is commanded and the angle between the blocks is extremely small (when not using optimum corner deceleration), interpolation can be carried out more smoothly using the vector accuracy interpolation.

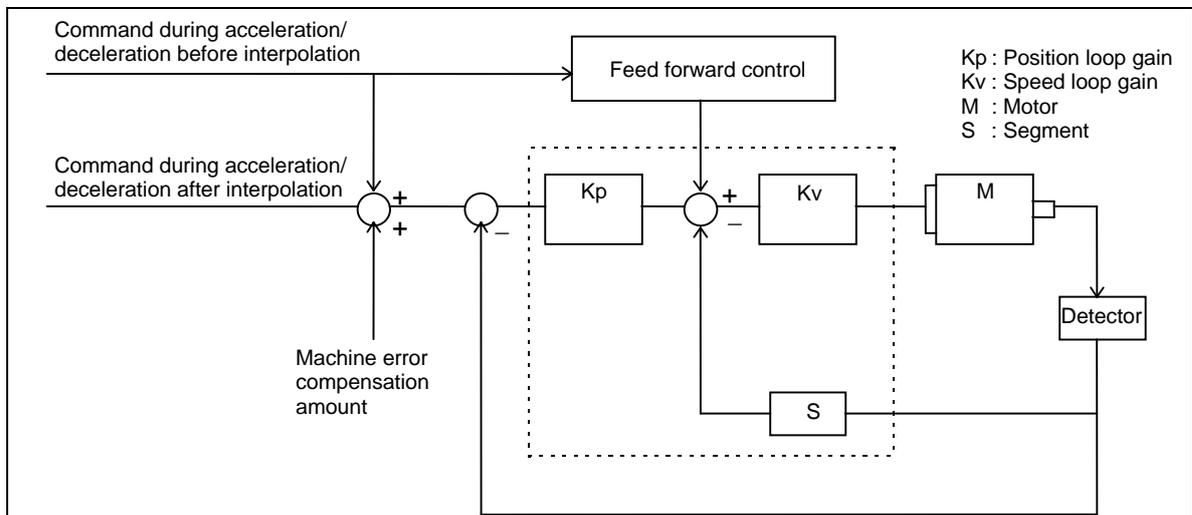


**Feed forward control**

With this function, the constant speed error caused by the position loop control of the servo system can be greatly reduced. However, as machine vibration is induced by the feed forward control, there are cases when the coefficient cannot be increased.

In this case, use this function together with the smooth high gain (SHG) control function and stably compensate the delay by the servo system's position loop to realize a high accuracy. As the response is smoother during acceleration/deceleration, the position loop gain can be increased.

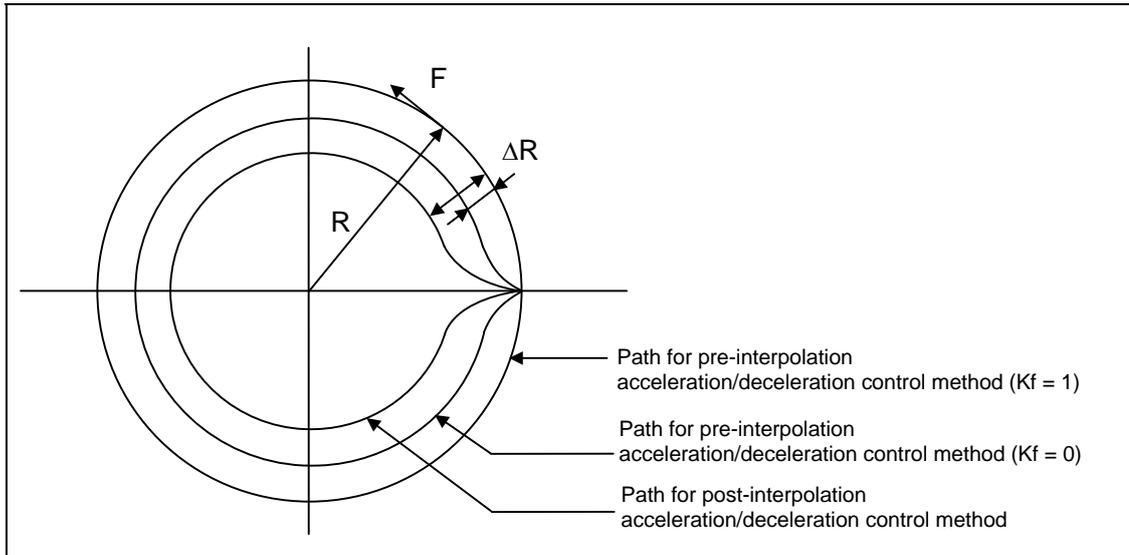
(1) Feed forward control



- (2) Reduction of arc radius reduction error amount using feed forward control  
 With the high-accuracy control, the arc radius reduction error amount can be greatly reduced by combining the pre-interpolation acceleration/deceleration control method above-mentioned and the feed forward control/SHG control.  
 The logical radius reduction error amount  $\Delta R$  in the high-accuracy control mode is obtained with the following expression.

Feed forward control	SHG control + Feed forward control
$\Delta R \leq \frac{1}{2R} \{T_p^2 (1 - K_f^2)\} \left(\frac{F}{60}\right)^2$ <p style="text-align: right;">                     R : Arc radius (mm)                      F : Cutting feedrate (mm/min)                      T<sub>p</sub> : Position loop time constant (s)                      K<sub>f</sub> : Feed forward coefficient (fwd_g/100)                 </p>	
By setting K <sub>f</sub> to the following value, the delay elements caused by the position loop in the servo system can be eliminated, and the logical $\Delta R$ can be set to 0.	
K <sub>f</sub> = 1 (Feed forward gain 100%)	The equivalent feed forward gain to set K <sub>f</sub> to 1 can be obtained with the following expression. $100 \sqrt{1 - \left\{1 - \left(\frac{\text{fwd\_g}}{50}\right)^2\right\} \left\{\frac{\text{PGN1 for conventional control}}{2 \times \text{PGN1 for SHG control}}\right\}^2}$

The feed forward gain can be set independently for G00 and G01.



**(Note)** If the machine vibrates when K<sub>f</sub> is set to 1, K<sub>f</sub> must be lowered or the servo system must be adjusted.



**Arc entrance/exit speed control**

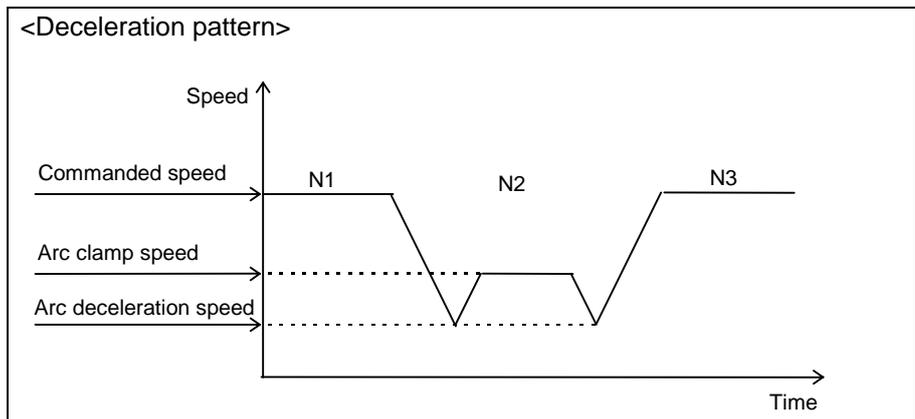
There are cases when the speed fluctuates and the machine vibrates at the joint from the straight line to arc or from the arc to straight line.

This function decelerates to the deceleration speed before entering the arc and after exiting the arc to reduce the machine vibration. If this is overlapped with corner deceleration, the function with the slower deceleration speed is valid.

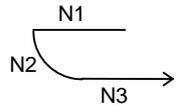
The validity of this control can be changed with the base specification parameter "#1149 cireff". The deceleration speed is designated with the base specification parameter "#1209 cirdcc".

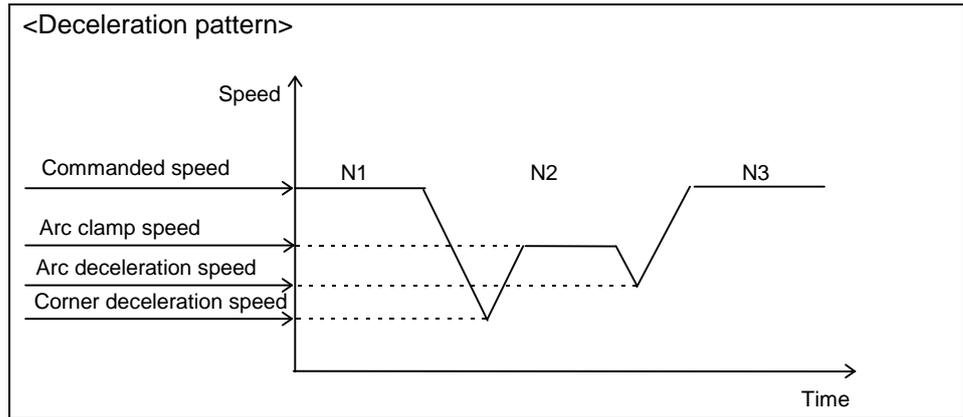
**(Example 1) When not using corner deceleration**

<p>&lt;Program&gt;  G61.1 ;  •  •  N1 G01 X-10. F3000 ;  N2 G02 X-5. Y-5. J-2.5 ;  N3 G01 Y-10. ;  •  •</p>	<p>&lt;Operation&gt;</p>
---------------------------------------------------------------------------------------------------------------------------------------------	--------------------------



(Example 2) When using corner deceleration

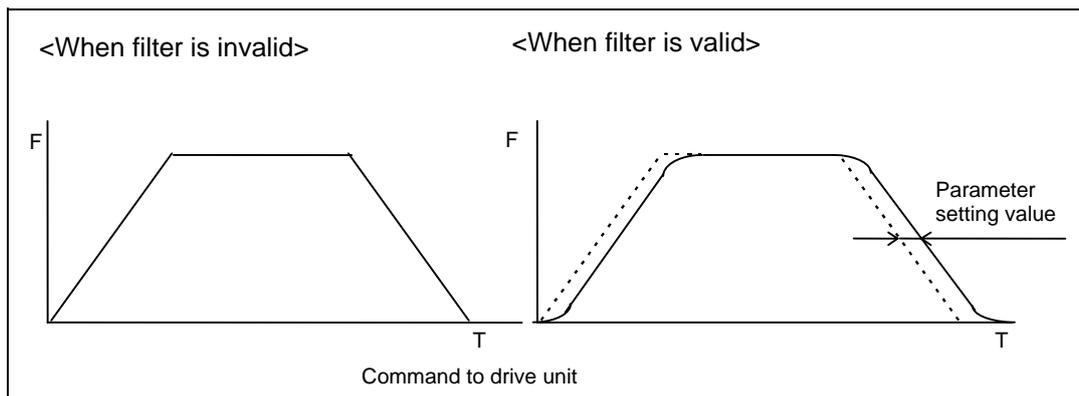
<p>&lt;Program&gt;          G61.1 ;          ·          ·          N1 G01 X-10. F3000 ;          N2 G02 X5. Y-5. I2.5 ;          N3 G01 X10. ;          ·          ·</p>	<p>&lt;Operation&gt;</p> 
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------



**S-pattern filter control**

This control interpolates while smoothing the changes in the segments distributed to each axis element with vector accuracy interpolation. With this, the fluctuation amplified by feed forward control is reduced and the effect onto the machine is reduced.

This can be set in the range of 0 to 200(ms) with the basic specification parameter "#1568 SfiltG1" and "#1569 SfiltG0". With "#1570 Sfilt2", this also enables the acceleration/deceleration fluctuation to further smoothen.

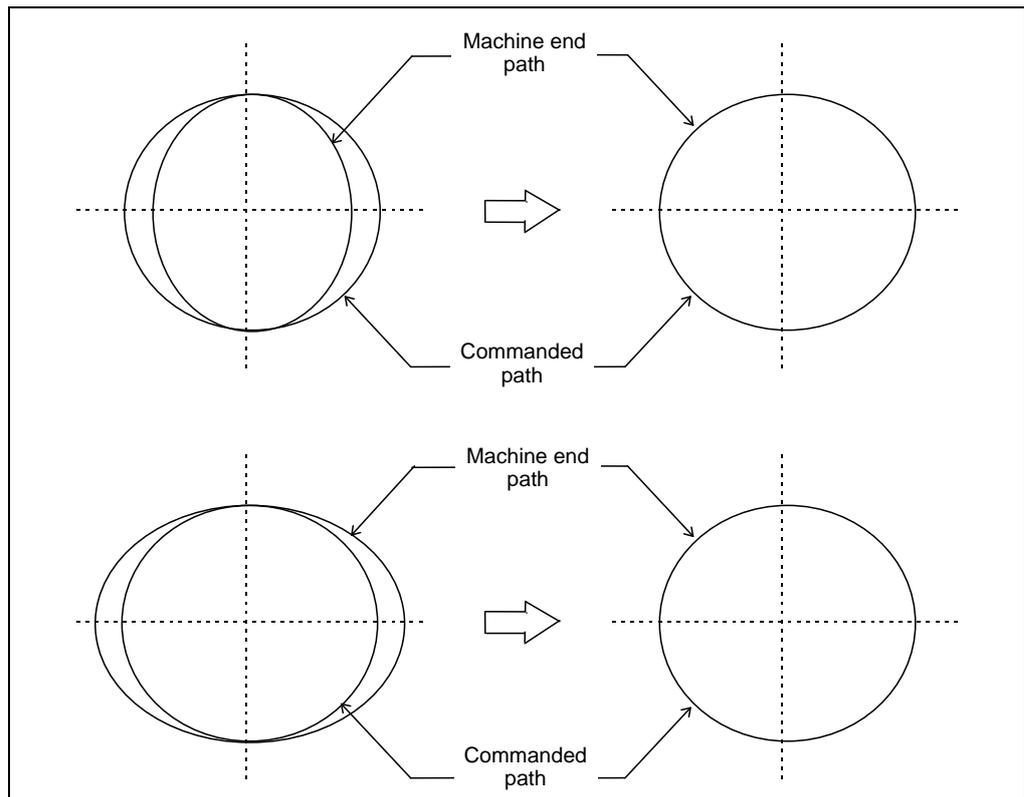




### Circular error radius compensation control for each axis

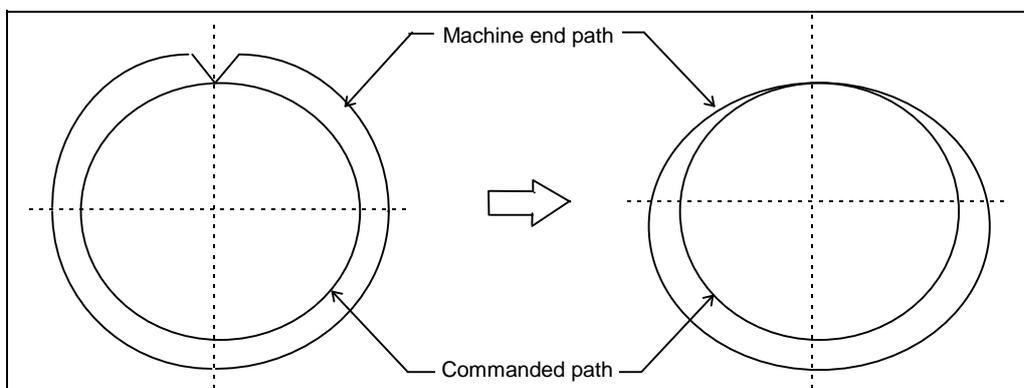
When the roundness at the machine end is, compared to the reference circle, expanded at an axis creating an ellipsis state, compensation is carried out for each axis to make a perfect circle. The validity of this control can be changed with control parameter "#8108 R COMP Select". Note that "#8108 R COMP Select" is valid only when "#8107 R COMPENSATION" is set to "1". The compensation coefficient for each axis is designated with the axis specification parameter "#2069 Rcoeff".

#### (1) Compensation in each axis direction of arc



#### (2) Smooth compensation at entrance and exit

The compensation amount is gradually increased from the arc start point to the 90° position, and 100% compensation is reached at the 90° position. The compensation is gradually decreased from 90° before the end point, and 0% compensation is reached at the end point.





### Relation with other functions

- (1) The modals must be set as shown below when commanding G08P1.

Function	G code
High-speed high-accuracy control II, High-speed machining cancel	G05 P0
Cylindrical interpolation cancel	G07.1
High-accuracy control cancel	G08 P0
Polar coordinate interpolation cancel	G15
Tool radius compensation mode cancel	G40
Normal line control cancel	G40.1
Tool length compensation cancel	G49
Programmable mirror image OFF	G50.1
Mirror image with settings	Cancel
Mirror image with signals	Cancel
No macro modal call	G67
Feed per revolution cancel	G94
Constant surface speed control mode cancel	G97
Interruption type macro mode	IOF

- (2) An alarm will occur if high-accuracy control is commanded in the following modes.
- During milling Program error (P481)
  - During cylindrical interpolation Program error (P481)
  - During polar coordinate interpolation Program error (P481)
  - During normal line control Program error (P29)
- (3) A program error (P29) will occur if the following commands are issued during the high-accuracy control mode.
- Milling
  - Cylindrical interpolation
  - Polar coordinate interpolation
  - Normal line control

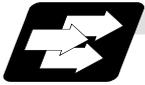


### Precautions

- (1) The "high-accuracy control" specifications are required to use this function.  
If G61.1 is commanded when there are no specifications, a program error (P123) will occur.
- (2) Command G61.1 in an independent block.
- (3) The G61.1 command can be used when the basic parameter "#1267 ext03/bit0" is set to "0". If G61.1 is commanded when the parameter is set to "1", a program error (P34) will occur.
- (4) This function may not be usable depending on the model.
- (5) The #1205 G0bdcc (G0 pre-interpolation) can be used with only one part system.  
If the 2nd or later part system is set to the G0 acceleration/deceleration before interpolation, an error will occur.
- (6) "#1568 SfiltG1", "#1569 SfiltG0" and "#1570 Sfilt2" cannot be changed from the screen during program operation. If these parameters is changed by "parameter input by program", these parameters are valid after the axes stop.

## 13.16 High-speed Machining Mode; G05, G05.1

### 13.16.1 High-speed Machining Mode I,II; G05 P1, G05 P2



#### Function and purpose

This function runs a machining program for which a freely curved surface has been approximated by fine segments at high speed. This is effective in increasing the speed of machining dies of a freely curved surface.

G1 block fine segment capacity for 1mm segment

Mode	Command	Maximum feedrate when 1mm segment G1 block is executed
Standard mode	G05 P0	16.8 m/min
High-speed machining mode I	G05 P1	16.8 m/min
High-speed machining mode II	G05 P2	135.0 m/min

The above performance applies under the following conditions.

- 6-axis system (including spindle) or less
- 1-part system
- 3 axes or less commanded simultaneously in G01
- Block containing only axis name and movement amount (Macro and variable command are not included.)
- During G61.1 high-accuracy control mode, or during cutting mode (G64)
- During tool radius compensation cancel (G40) (Only in the high-speed machining mode II)

When the above conditions are not satisfied, the given feedrate may not be secured.



#### Command format

**G05 P1 ; ..... High-speed machining mode I ON**

**G05 P0 ; ..... High-speed machining mode I OFF**

**G05 P2 ; ..... High-speed machining mode II ON**

**G05 P0 ; ..... High-speed machining mode II OFF**

In addition to the G05 P0 command, the high-speed machining mode I is canceled with the command of High-speed machining mode II (G05 P2). Also, High-speed machining mode II can be canceled with the command of High-speed machining mode I.

**Detailed description**

- (1) The override, maximum cutting speed clamp, single block operation, dry run, manual interruption and graphic trace and high-accuracy control mode are valid even during the high-speed machining mode I/II.
- (2) When using the high-speed machining mode II mode, set "BIT1" of the parameter "#1572 Cirorp" to "1" to eliminate the speed fluctuation at the seams of the arc and straight line or arc and arc.

**Example of program (High-speed machining mode I)**

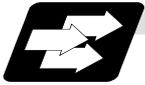
G28 X0. Y0. Z0. ;
G91 G00 X-100. Y-100. ;
G01 F10000 ;
G05 P1 ; ----- High-speed machining mode I ON
:
X0.1 Y0.01 ;
X0.1 Y0.02 ;
X0.1 Y0.03 ;
:
G05 P0 ; ----- High-speed machining mode I OFF
M30 ;

**Restrictions**

- (1) If "G05 P1(P2)" is commanded when the option for high-speed machining mode I/(II) is not provided, a program error (P39) will occur.
- (2) The automatic operation process has the priority in the high-speed machining mode I/II, so the screen display, etc., may be slowed down.
- (3) The speed will decelerate once at the G05 command block, so turn ON and OFF when the tool separates from the workpiece.
- (4) When carrying out operation in the high-speed machining mode I/II by communication or tape operation, the machining speed may be suppressed depending on the program transmission speed limit.
- (5) Command the G05 command in an independent block.
- (6) A decimal point cannot be used for the P address in the G05 command block.
- (7) Only P0, P1, and P2 are valid as P address in the G05 command block.  
If an address other than P is commanded in the G05 block, a program error (P35) will occur. If there is no P command, a program error (P33) will occur.
- (8) The machining speed may be suppressed depending on the number of characters in a designated block.

## 13.17 High-speed High-accuracy Control; G05, G05.1

## 13.17.1 High-speed High-accuracy Control I, II



## Function and purpose

This function runs a machining program that approximates a freely curved surface with fine segments at high speed and high accuracy. This is effective in increasing the speed of machining dies of a freely curved surface.

Simultaneous 3-axis fine segment capacity for 1mm segment

High-speed high-accuracy control I mode	Performance of the fine segment execution		Restriction in the program
	Without radius compensation	With radius compensation	
Invalid	16.8m/min	16.8m/min	No
Valid	33.6m/min	33.6m/min	Yes



## Command format

**G05.1 Q1 ; ..... High-speed high-accuracy control I ON**  
**G05.1 Q0 ; ..... High-speed high-accuracy control I OFF**

**G05 P10000 ; ... High-speed high-accuracy control II ON**  
**G05 P0 ; ..... High-speed high-accuracy control II OFF**

**(Note 1)** The high-speed high-accuracy mode I and II can not be used at the same time.

**(Note 2)** G05.1 Q1 (high-speed high-accuracy mode I) and G05 P10000 (high-speed high-accuracy mode II) are valid when the parameter "#1267 ext03/bit0" is ON.



Detailed description

- (1) The high-speed high-accuracy control I / II can be used during computer link, tape, MDI, IC card or memory operation.
- (2) The override, maximum cutting speed clamp, single block operation, dry run, handle interrupt and graphic trace are valid even during the high-speed high-accuracy control I / II modal.
- (3) The machining speed may drop depending on the number of characters in one block.
- (4) The high-speed high-accuracy control I / II function automatically turns the high-accuracy control mode ON. For high-accuracy control function, refer to "13.14 High-accuracy control"
- (5) Turn the tool radius compensation command ON and OFF during the high-speed high-accuracy control I / II mode.  
If the high-speed high-accuracy control I / II mode is turned OFF without turning the tool radius compensation OFF, program error (P34) will occur.
- (6) Turn the high-speed high-accuracy control I / II mode OFF before commanding data other than the data listed above.
- (7) When using the high-speed high-accuracy control II mode, set parameter "#1572 Cirorp" to "1" to eliminate the speed fluctuation at the seams of the arc and straight line or arc and arc.
- (8) Feedrate command (F) is clamped with the "#2110 Clamp (H-precision)" (Cutting feed clamp speed for high-accuracy control mode) set with parameter.
- (9) "#2109 Rapid (H-precision)" (Rapid traverse speed for high-accuracy control mode) set with parameter will be valid for the rapid traverse speed.
- (10) When "#2109 Rapid (H-precision)" is set to "0", movement is performed with "#2001 rapid" (Rapid traverse rate) set with parameter. Also, when "#2110 Clamp (H-precision)" is set to "0", clamp will be made with "#2002 clamp" (Cutting clamp speed) set with parameter.

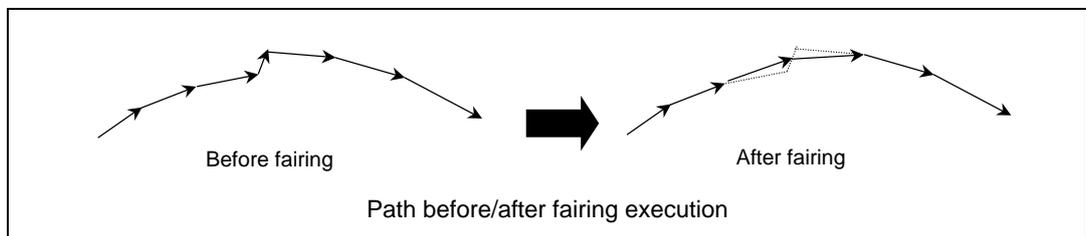


Additional functions when high-speed high-accuracy control 2 mode is ON

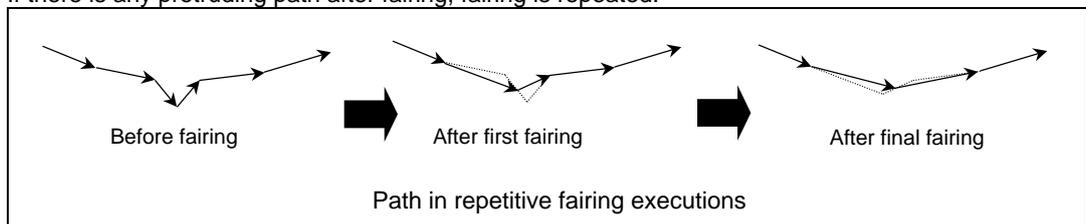
- (1) Fairing

If there is a protruding path (zigzagging path) shorter than the parameter setting values in the machining program generated with a CAM, etc., this function can be used to eliminate the protruding path smaller than the setting value so that the front and back paths are smoothly connected. This function is valid only for continuous linear commands (G1).

Related parameter	Details
#8033 Fairing ON	0: Fairing invalid 1: Execute fairing for the protruding block
#8029 Fairing L	Execute fairing for the shorter block than this setting value



If there is any protruding path after fairing, fairing is repeated.

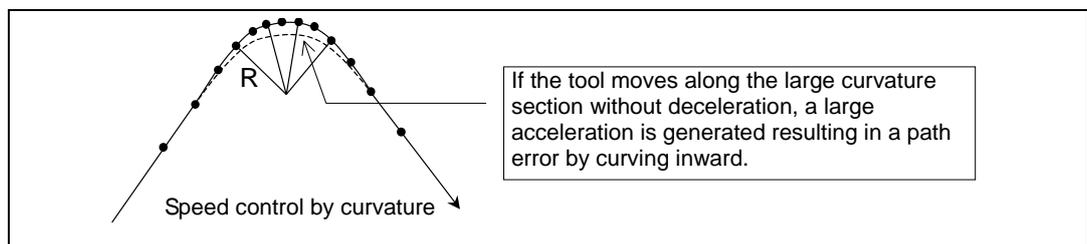


(2) Acceleration clamp speed

With the cutting feed clamp speed during the high-speed high-accuracy control 2 mode, when the following parameter is set to "1", the speed is clamped so that the acceleration generated by each block movement does not exceed the tolerable value. This function clamps the speed optimally even at a section where "angle change at each block is small but entire curvature is large" such as shown below.

The tolerable acceleration value is calculated from the parameter "#1206 G1bF" and "#1207 G1btL" setting values. (Tolerable acceleration = #1206/#1207)

Related parameter	Details
#8034 AccClampt ON	0: Clamp the cutting speed with parameter "#2002 clamp" or the corner deceleration function. 1: Clamp the cutting speed with acceleration judgment.

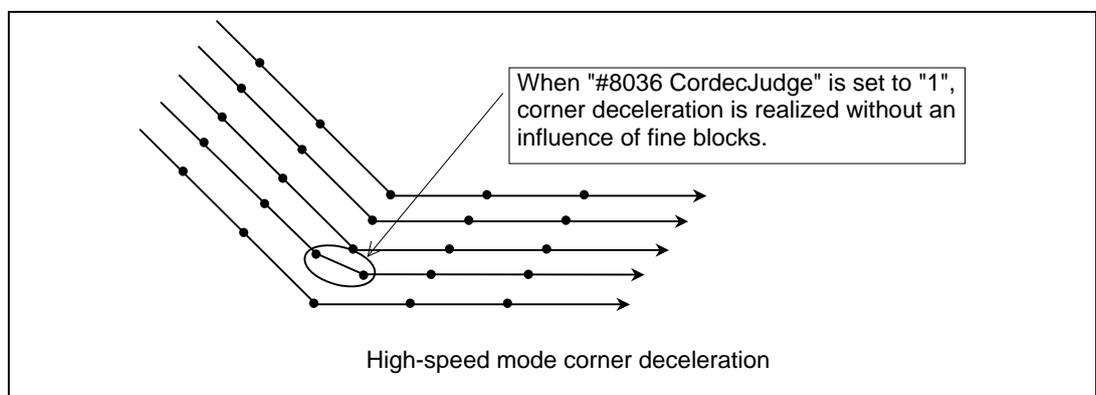


**(Note)** When a speed is set in "#2109 Clamp(H-precision)", clamp is executed at that speed. When the setting value is 0, clamp is executed with "#2002 clamp".

(3) Corner deceleration in high-speed mode

During high-accuracy control, if the angle is large between the adjacent blocks in the machining program, this function, conventionally, automatically decelerates so that the acceleration generated when passing through the corner is within the tolerable value. If a small block is inserted at the corner section in the machining program generated with the CAM, etc., the corner passing speed will not match the periphery. This can affect the machining surface. If this type of small block is inserted when performing corner deceleration in the high-speed mode, the corner will be largely judged by the parameter settings. The small block is excluded when the angle is judged, but is not excluded from the actual movement command.

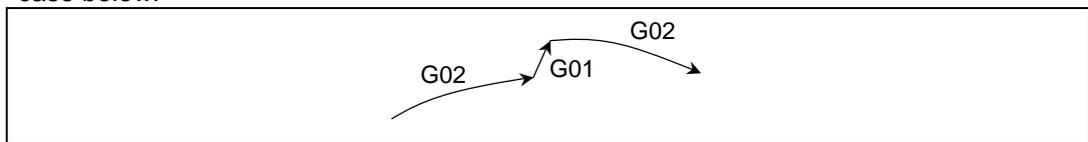
Related parameter	Details
#8036 CordecJudge	0: Judge the corner from the angle of the neighboring block. 1: Judge the corner from the angle of the neighboring block, excluding the minute blocks.
#8037 CorJudgeL	Exclude shorter block than this setting value.





## Precautions

- (1) High-speed high-accuracy control I and II are the optional functions. If "G05.1 Q1" or "G05 P10000" is commanded when the option is not provided, a program error (P39) will occur.
- (2) The automatic operation process has the priority in the high-speed high-accuracy control I/II modal, so the screen display, etc., may be delayed.
- (3) The speed will decelerate once at the "G05.1 Q1"/"G05.1 Q0" and the "G05 P10000"/"P05 P0" command block, so turn ON and OFF when the tool separates from the workpiece.
- (4) If an address other than G/Q or P/N is commanded in the "G05.1 Q1"/"G05.1 Q0" and "G05 P10000"/"G05 P0" command block, a program error (P33) will occur.
- (5) Command the "G05.1 Q1"/"G05.1 Q0" and the "G05 P10000"/"G05 P0" command in an independent block.
- (6) When carrying out high-speed high-accuracy control I/II operation during tape operation, the machining speed may be suppressed depending on the program transmission speed and the number of characters in one block.
- (7) If there is no Q or P command in the G05.1 or G05 command block, a program error (P33) will occur.
- (8) A decimal point cannot be used in the Q or P command.
- (9) If the high-speed high-accuracy control I command is issued in the high-speed high-accuracy control II modal, a program error (P34) will occur.
- (10) If the high-speed high-accuracy control II command is issued in the high-speed high-accuracy control I modal, a program error (P34) will occur.
- (11) Variable commands and user macros cannot be used in the high-speed high-accuracy control II modal.
- (12) Fairing function is valid for the continuous linear command (G1). Fairing is not possible in the case below.



- (13) The G codes for this function are valid when the parameter "#1267 ext03/bit0" setting value is "0". If G05.1 Q1 is commanded when this setting value is "0", a program error (P34) will occur.



Relation with other functions

- (1) The modal state must be as shown below when commanding G05.1 Q1 and G05 P10000. Program error (P34) will occur if the conditions are not satisfied. When commanding a SSS control, refer to "3.16.2 SSS control" for details.

Function	G code
Tool radius compensation cancel	G40
Tool length compensation cancel	G44
G command mirror image cancel	G50.1
Mirror image cancel with parameter settings	Cancel
Mirror image cancel with signals	Cancel
Cutting mode	G64
User macro modal call cancel	G67
Programmable coordinate rotation mode OFF	G69.1
Fixed cycle cancel	G80
Per-minute feed	G94
Constant surface speed control OFF	G97
User macro interrupt cancel	IOF

← For only mode I

Although "G05.1 Q" can be commanded in the modals listed below, correct movement may not be guaranteed.

Function	G code
Exact stop check mode	G61
Automatic corner override	--
Tapping mode	G63
Feed per revolution	G95
Constant surface speed	G96

- (2) The following data can be commanded while the high-speed high-accuracy control I/II mode is ON. Program error will occur if other data is commanded.

Function	High-speed high-accuracy mode		G code			
	I	II				
Positioning	○	○	G00			
Cutting feed	○	○	G01	G02	G03	
Helical interpolation	○	○	G02	G03		
Plane selection	○	○	G17	G18	G19	
Tool radius compensation	○	○	G40	G41	G42	
Tool length compensation	○	-	G43	G44		
Programmable mirror image	○	○	G50.1	G51.1	G62	
Mirror image with parameter settings	-	-	-			
Mirror image with signals	-	-	-			
Absolute command	○	○	G90			
Incremental command	○	○	G91			
Workpiece coordinate system setting	○	-	G92			
Workpiece coordinate system selection	○	-	G54~G59			
Machine coordinate system command	○	-	G53			

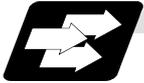
### 13. Program Support Functions

#### 13.17 High-speed High-accuracy Control; G05, G05.1

Function	High-speed high-accuracy mode		G code			
	I	II				
Subprogram call	○	○	G22			
Programmable parameter input	○	-	G10 L50			
Programmable compensation amount input	○	-	G10 L10			
High-speed high-accuracy control I cancel	○	-	G05.1 Q0			
High-speed high-accuracy control II cancel	-	○	G05 P0			
Spline control	-	○	G05.1 Q2	G05.1 Q0		
F code command	○	○	Fxx			
Sequence No. command	○	○	Nxx			
Comment command	○	○	( )			
Optional block skip	○	○	/			
Miscellaneous function (Note 1)	○	○	Mxx	Sxx	Txx	Bxx
I/J/K/R command for circular interpolation	○	○	I	J	K	R
Axis movement command	○	○	X	Y	Z	etc.

**(Note 1)** Macro interruption cannot be used. (High-speed high-accuracy control II only)

## 13.17.2 SSS Control



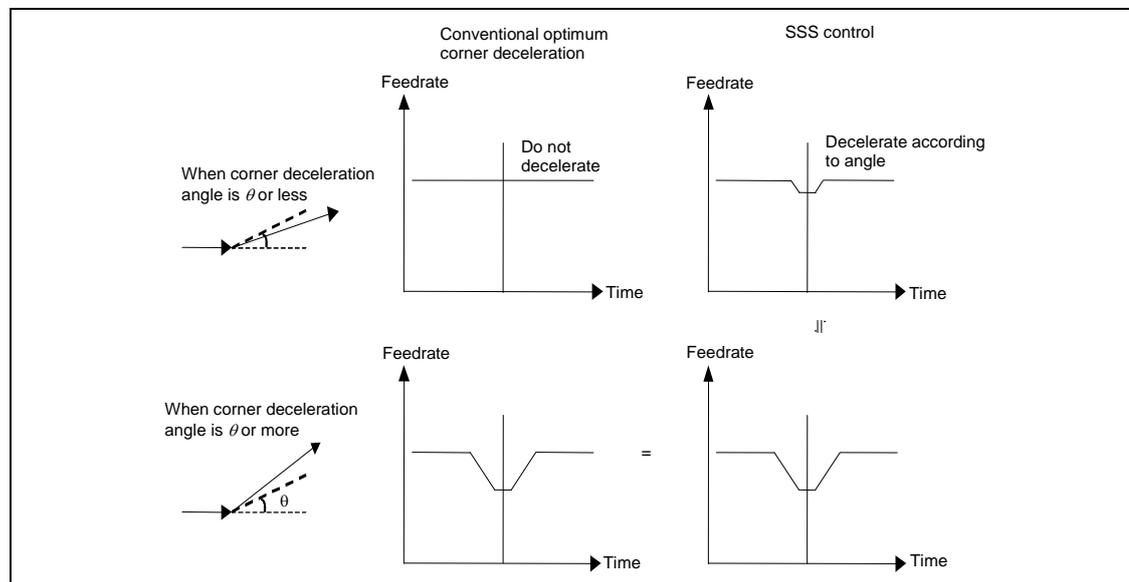
## Function and purpose

With conventional high-accuracy control, the angle between two blocks is compared with the corner deceleration angle to determine whether to execute corner deceleration between the blocks. This can cause the speed to suddenly change between the blocks with an angle close to the corner deceleration angle, resulting in scratches or streaks.

With SSS (Super Smooth Surface) control, the user can predict the optimum machining speed, using the large area of path information. Comparing to the conventional high-accuracy control function, SSS control has more advanced features so that smoother workpiece cutting surface is realized. The followings are some of the features available in SSS control.

- (1) Speed fluctuation caused by the effect of the machining-disturbing blocks (minute stepping or waviness) is suppressed. Thus, scratches caused by these blocks are reduced.
- (2) Even if corner deceleration is not required, the speed is clamped if the predicted acceleration is high.

Furthermore, the machining time can be reduced in machining applications having many corners.



The length of the path direction recognized with SSS control can be adjusted with the machining parameter "#8091 StdLength". The range is increased as the setting value increases, and the effect of the error is reduced.

**(Note)** This function is an option. The high-speed high-accuracy control II option is required to use this function.



## Detailed description

- (1) The following procedures are followed to use SSS control.
  - (a) Turn the following parameters ON beforehand.  
Basic specification parameter "#1267 ext03/bit0"  
Machining parameter "#8090 SSS ON"
  - (b) Command "G05 P10000 ;" (high-speed high-accuracy control II ON).  
→SSS control is valid until "G05 P0 ;" (high-speed high-accuracy control II OFF) is commanded.
- (2) The SSS control can be used during computer link, tape, MDI, IC card or memory operation.
- (3) The machining speed may drop depending on the number of characters in one block.
- (4) To command data other than the valid command data, turn the SSS control mode OFF first.



## Relation with other functions

- (1) The modals must be set as shown below when SSS control starts.  
A program error (P34) will occur if these conditions are not satisfied.

## Modal state

Function	Mode
Tool radius compensation mode	G40
Programmable mirror image	G50.1
Cutting mode	G64
Macro modal call mode	G67
Programmable coordinate rotation mode	G69.1
Canned cycle mode	G80
Per-rotation feed	G94
Constant surface speed control mode	G97
Interrupt type macro mode	IOF

## Status other than modals

Function	Status
Parameter mirror image	OFF
External mirror image	OFF

- (2) The following functions can be commanded during the SSS control mode.  
A program error will occur if any other function is commanded.
- During G code command: Program error (P34)
  - Other cases: Program error (P33)

Function	Command			
Positioning	G00			
Cutting feed	G01	G02	G03	
Helical interpolation	G02	G03		
Spiral interpolation	G02.1	G03.1		
Plane selection	G17	G18	G19	
Tool radius compensation	G40	G41	G42	
Programmable mirror image	G50.1	G51.1		
Absolute command	G90			
Incremental command	G91			
Subprogram call	G22			
High-speed high-accuracy control II cancel	G05 P0			
Spline control	G05.1 Q2	G05.1 Q0		
F code command	Fxxx			
Sequence No. command	Nxxx			
Comment command	( )			
Optional block skip	/			
Auxiliary function (Note 1)	Mxxx	Sxxx	Txxx	Bxxx
I, J, K, R command for circular interpolation	I	J	K	R
Axis movement data	X	Y	Z	etc.

- (3) The F1-digit command function cannot be used.
- (4) Macro interruption cannot be used.
- (5) The override, maximum cutting speed clamp, signal block operation and graphic trace are valid even during the SS control mode.
- (6) The tool radius compensation command should not be used during the SSS control mode.
- (7) Turn the tool radius compensation command ON and OFF during the SSS control mode.  
If the SSS control mode is turned OFF before the tool radius compensation is turned OFF, a program error (P34) will occur.
- (8) The geometric IB command is invalid during SSS control, and will be handled as a normal arc command.



## Parameter standard values

The standard values of the parameters related to SSS control are shown below.

## (1) Machining parameters

#	Item	Standard value
8019	R COMP	0
8020	DCC ANGLE	10
8021	COMP CHANGE	1
8022	CORNER COMP	0
8023	CURVE COMP	-20
8029	Fairing L	0
8033	Fairing ON	0
8034	AccClamp ON	0
8036	CordecJudge	0
8037	CorJudgeL	0

**(Note)** Reference items for adjusting the parameter  
 The relation of each parameter, the accuracy and the speed is shown below.  
 The accuracy and speed required for machining can be adjusted with these settings.  
 When adjusting the parameters, adjust the values within the range in which the machine does not vibrate.

Parameter	Adjustment target	Effect
#8022 CORNER COMP	Accuracy at corner section	Large setting = Accuracy improves, speed drops
#8023 CURVE COMP	Accuracy at curve section	Large setting = Accuracy increases, speed drops
#8092 ClampCoeff	Accuracy at curve section	Large setting = Accuracy drops, speed increases (Note) Usually use the standard value and adjust with "#8023".

## (2) Basic specification parameters

#	Item	Standard value
1148	I_G611 Initial high accuracy	0
1205	G0bdcc G0 before interpolation	0
1206	G1bf Acceleration/deceleration before interpolation Maximum speed	-
1207	G1btL Acceleration/deceleration before interpolation Time constant	-
1209	Cirdcc Arc deceleration speed	-
1267	ext03/bit0 G code type	1
1572	Cirorp Arc command overlap	0
1568	SfiltG1 G1 soft acceleration/deceleration filter	0
1569	SfiltG0 G0 soft acceleration/deceleration filter	0
1570	Sfilt2 Soft acceleration/deceleration filter 2	0

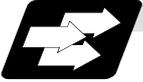
## (3) Axis specification parameters

#	Item		Standard value
2010	fwd_g	Feed forward gain	70
2068	G0fwdg	G00 feed forward gain	70
2096	crncsp	Minimum corner deceleration speed	0

**Restrictions**

- (1) Pre-reading is executed during SSS control, so a program error could occur before the block containing the error is executed.
- (2) Do not correct the buffer during SSS control. The operation will not be guaranteed if the buffer is corrected.
- (3) If automatic/manual simultaneous or automatic handle interrupt are used during SSS control, the machining accuracy will not be guaranteed.
- (4) During SSS control, if override is set to "0" in the single block stop state, the "M01 operation error 0102 override zero" will occur.
- (5) If a fine arc command is issued during SSS control, it may take longer to machine.
- (6) A program error (P33) will occur if external input mirror image is applied during SSS control.
- (7) The same path as single block operation will be used during graphic check.
- (8) The line under the cutting feedrate and arc command block are subjected to the speed control in the SSS control. However, rotary axis command block is not subjected to SSS control. In the command blocks that are not subjected to speed control, decelerates first and automatically switches the SSS control ON and OFF

## 13.18 Spline; G05.1



## Function and purpose

This function automatically generates a spline curve that passes through a sequence of points commanded by the fine segment machining program, and interpolates the path along this curve. This allows highly accurate machining at a high speed.

The spline function can be commanded when the machining parameter "#8025 SPLINE ON" is set to 1 in the high-speed high-accuracy control function II mode (between G05P10000 and G05P0). The following explanation is limited to the spline function in the high-speed high-accuracy control function II mode.



## Command format

<b>G05.1Q2X0Y0Z0 ;</b>	<b>Spline mode ON</b>
<b>G05.1Q0 ;</b>	<b>Spline mode OFF</b>



## Example of program

```

:
G91 ;
G05 P10000 ; ..... High-speed high-accuracy control function II mode ON
:
G05.1 Q2 X0 Y0 Z0 ; ..... Spline mode ON
G01 X1000 Z-300 F1000 ;
X1000 Z-200 ;
Y1000 ;
X-1000 Z-50 ;
X-1000 Z-300 ;
G05.1 Q0 ; ..... Spline mode OFF
:
G05 P0 ; ..... High-speed high-accuracy control function II mode OFF
:

```

- (1) The spline function carries out spline interpolation when the following conditions are all satisfied. If the following conditions are not satisfied, the spline function is canceled once, and it is judged whether to carry out new spline from the next block.
  - When the block length is shorter than the machining parameter "#8030 MINUTE LENGTH".
  - When the movement amount is not 0.
  - When one of the following modes is entered.
    - G01: Linear interpolation, G40: Tool compensation cancel, G64: Cutting mode, G80: Fixed cycle cancel, G94: Per minute feed
  - When only an axis commanded with G05.1Q2 is commanded.
  - Graphic check is not being carried out.
  - A single block is not being executed.
- (2) Command the axis for the spline function mode following G05.1Q2. Note that the command and G05.1Q2. must be in the same block. For example, if the X axis and Y axis are to be commanded in the spline function mode, command "G05.1Q2X0Y0;". The command block containing an axis not designated with this command (G05.1Q2X0Y0) in the spline function mode will carry out linear interpolation instead of spline interpolation.

- (3) If G05.1Q2 is commanded when not in the high-speed high-accuracy control function II mode (between G05P10000 and G05P0), the program error (P34) will occur.
- (4) If the machining parameter "#8025 SPLINE ON" is 0 in the high-speed high-accuracy control function II mode (between G05P10000 and G05P0) and G05.1Q2 is commanded, program error (P34) will occur.
- (5) Up to three axes set as the basic axes I, J and K can be commanded for the spline function.



**Detailed description**

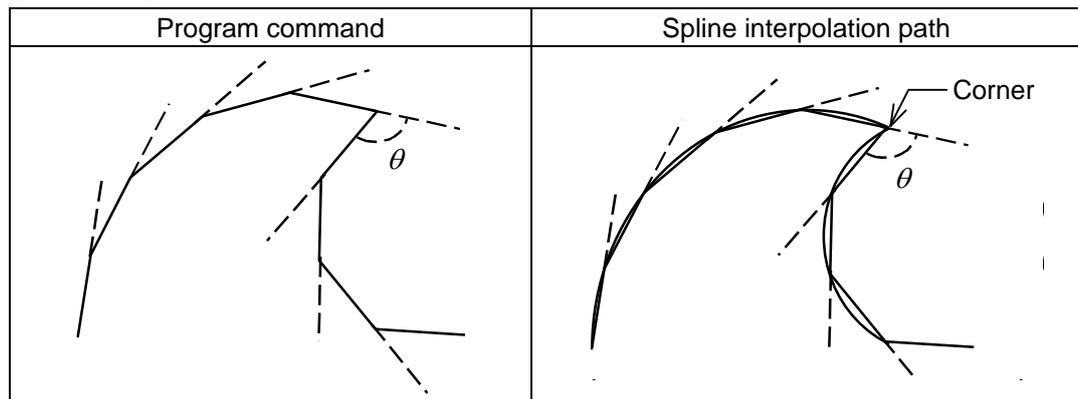
**(1) Temporary cancellation of spline interpolation**

Normally, once the spline function is activated, one curve is generated by smoothly connecting all points until it is canceled. However, if a corner edge is to be created, or if the segment length is long and spline interpolation is not to be carried out, the function can be canceled temporarily with the parameters.

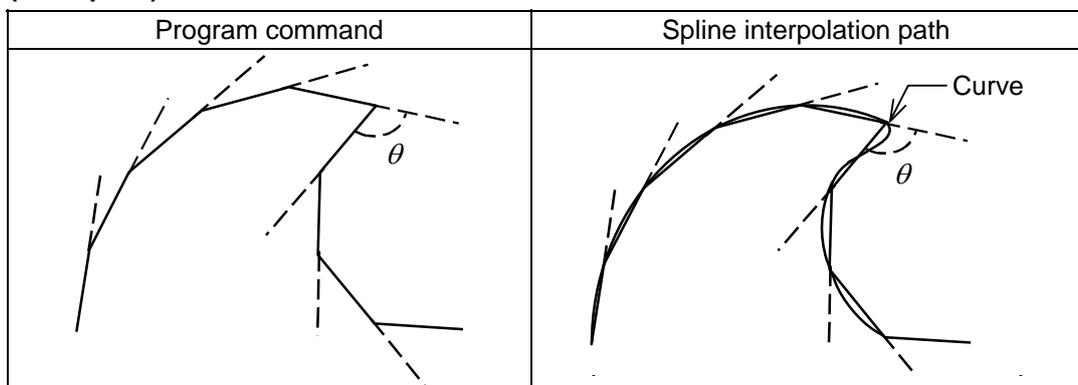
(a) Cancel angle

If the angle  $\theta$  of two consecutive blocks exceeds the value set in parameter "#8026 CANCEL ANG.", the spline function will be temporarily canceled, and optimum corner deceleration will be applied. When this parameter is not set (=0), the spline interpolation will be constantly applied. The parameter of the high-accuracy control function "#8020 DCC ANGLE" is valid during the temporary cancellation, and the optimum corner deceleration will be applied.

**(Example 1) #8026 CANCEL ANG. = 60°**



**(Example 2) #8026 CANCEL ANG. = 0°**

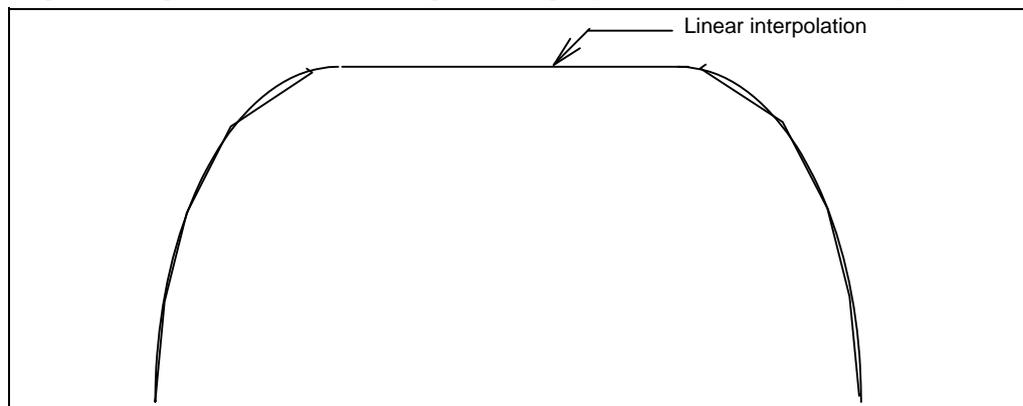


- (Note 1)** If the section to be a corner is smooth when actual machining is carried out, lower the CANCEL ANG. If a smooth section becomes a corner, increase the CANCEL ANG.
- (Note 2)** If the CANCEL ANG.  $\geq$  DCC. ANGLE, the axis will decelerate at all corners which angle is larger than the CANCEL ANG.
- (Note 3)** If the CANCEL ANG.  $<$  DCC. ANGLE, corner deceleration will not be applied if the corner angle is less than the DCC. ANGLE even if the spline interpolation is canceled.

(b) Fine segment length

If the movement amount in a block is longer than the parameter "#8030 MINUTE LENGTH", the spline function will be temporarily canceled, and the linear interpolation will be executed. When this parameter is not set (= 0), the fine segment length will be 1mm. If blocks that satisfies the following condition continue, linear interpolation will be executed.

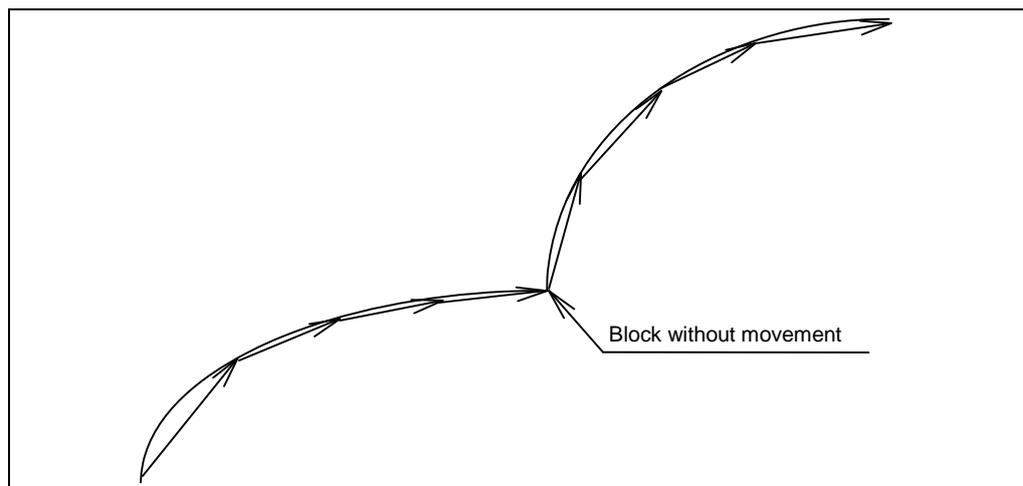
Segment length in a block  $>$  fine segment length (#8030 MINUTE LENGTH)



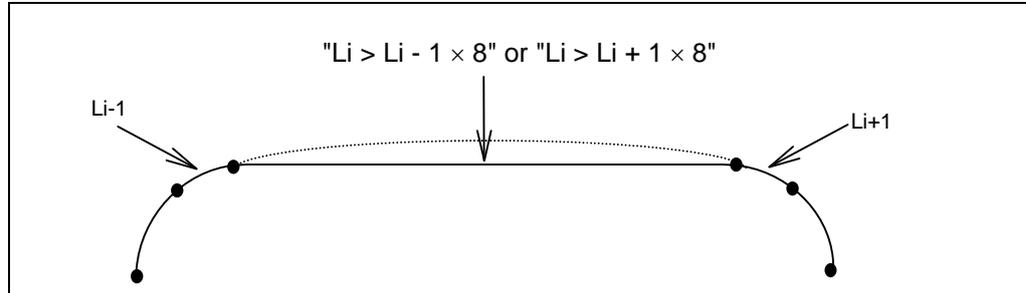
If the #8030 is set to "-1", the spline interpolation will not be canceled according to the block length.

(c) When a block without movement exists

If a block without movement exists in the spline function, the spline interpolation will be canceled temporarily. Note that blocks containing only ";" will not be viewed as a block without movement.



- (d) When a block markedly longer than other blocks exists in spline function  
 If the  $i$ th block length is  $L_i$  in the spline interpolation mode, and it is given as " $L_i > L_{i-1} \times 8$ " or " $L_i > L_{i+1} \times 8$ ", the block will be interpreted as a linear section, and the spline mode will be temporarily canceled. However, if the parameter "#8030 MINUTE LENGTH" is set to "-1", the mode will not be canceled.



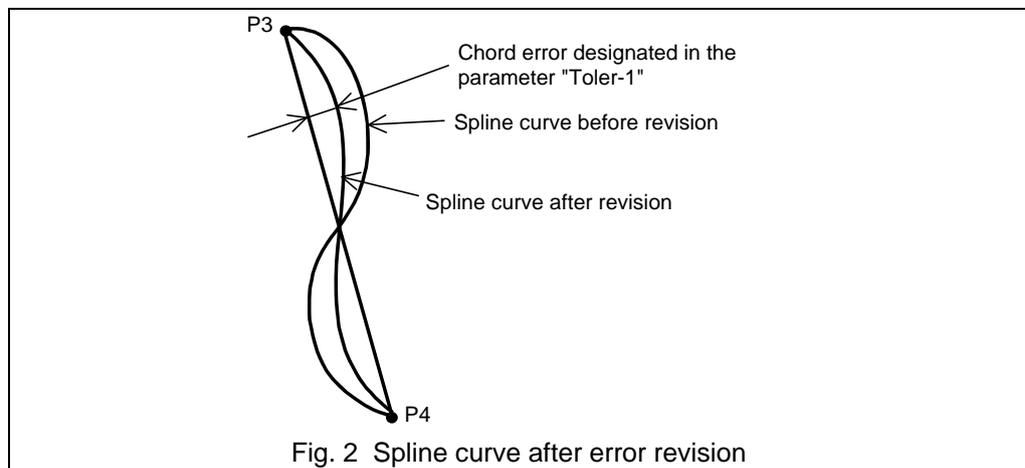
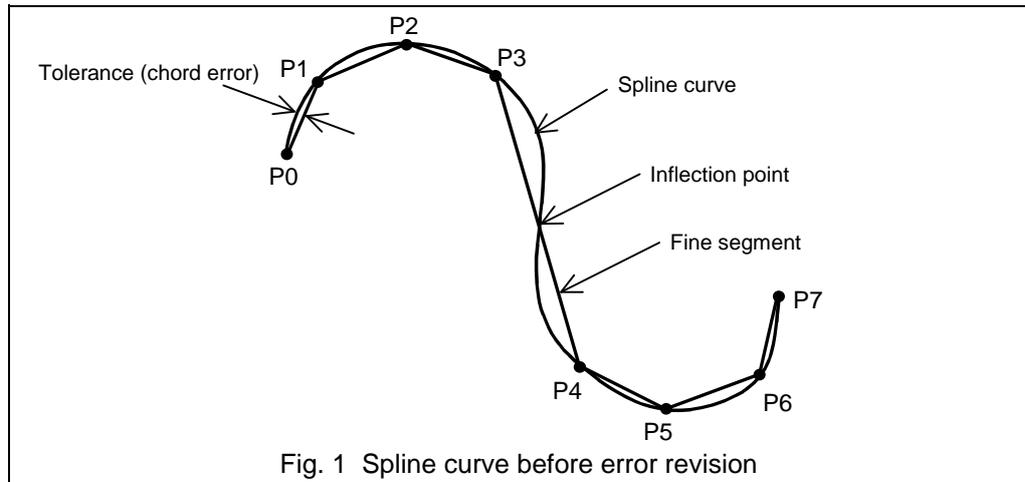
## (2) Spline interpolation curve shape revision

Normally, once the spline function is entered, one curve connecting all points smoothly is generated until the function is canceled. However, if the spline curve shape is to be revised, the spline curve shape can be revised with the parameters.

- (a) Chord error of block containing inflection point  
 When changing the CAD curve data into fine segments with the CAM, normally, the tolerance (chord error) of the curve is approximated in segments that are approx.  $10\mu\text{m}$ . If there is an inflection point in the curve, the length of the block containing the inflection point may lengthen. (Because the tolerance is applied at both ends near the inflection point.) If the block lengths with this block and the previous and subsequent blocks are unbalanced, the spline curve in this block will have a large error in respect to the original curve.  
 At sections where the tolerance (chord error) of the fine segment block and spline curve in a block containing this type of inflection point, if the chord error in the corresponding section is larger than the value set in parameter (#8027 Toler-1), the spline curve shape is automatically revised so that the error is within the designated value.  
 However, if the maximum chord error of the corresponding section is more than five times larger than the parameter "Toler-1" setting value, the spline function will be temporarily canceled.  
 The curve is revised only in the corresponding block.  
 The revisions are carried out with the following conditions for each block in the spline interpolation mode.

There is an inflection point in the spline curve, and the maximum error of the spline curve and linear block is larger than parameter "Toler-1". (Distance between P3-P4 in Fig. 1)

When the above conditions are satisfied, the spline curve will be revised so that the error between P3-P4 in Fig. 2 is within the designated value.



In parameter "Toler-1", set the tolerance for developing into fine segments with the CAM. Set a smaller value if the expansion (indentation) is apparent due to the relation with the adjacent cutting paths.

- (b) Chord error of block not containing inflection point  
Even in blocks that do not contain an inflection point, if the block lengths are not matched, the tolerance of the spline curve may increase. The curve may also expand due to the effect of relatively short blocks.

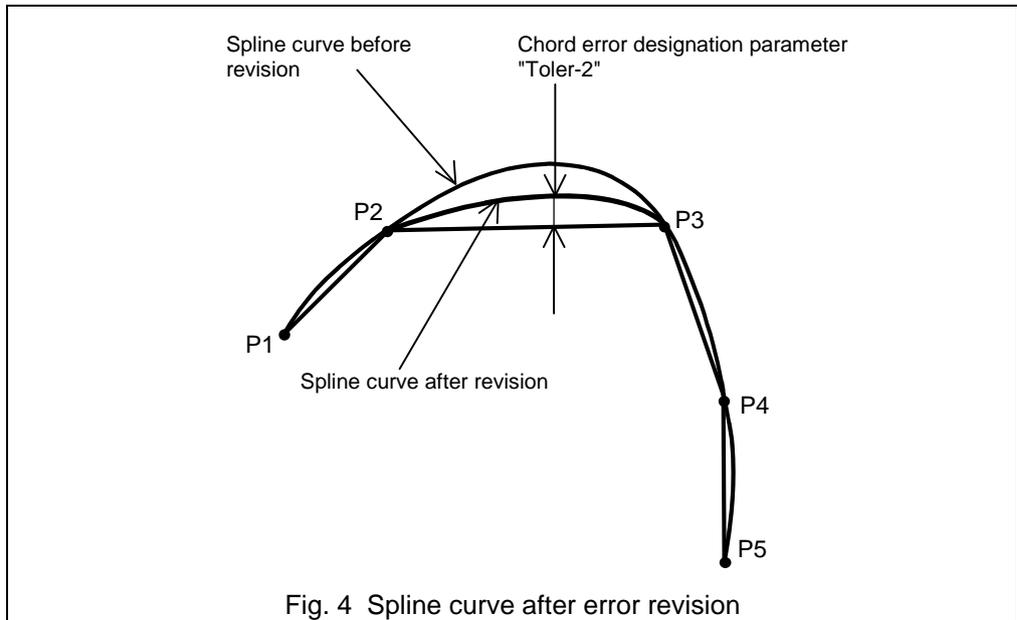
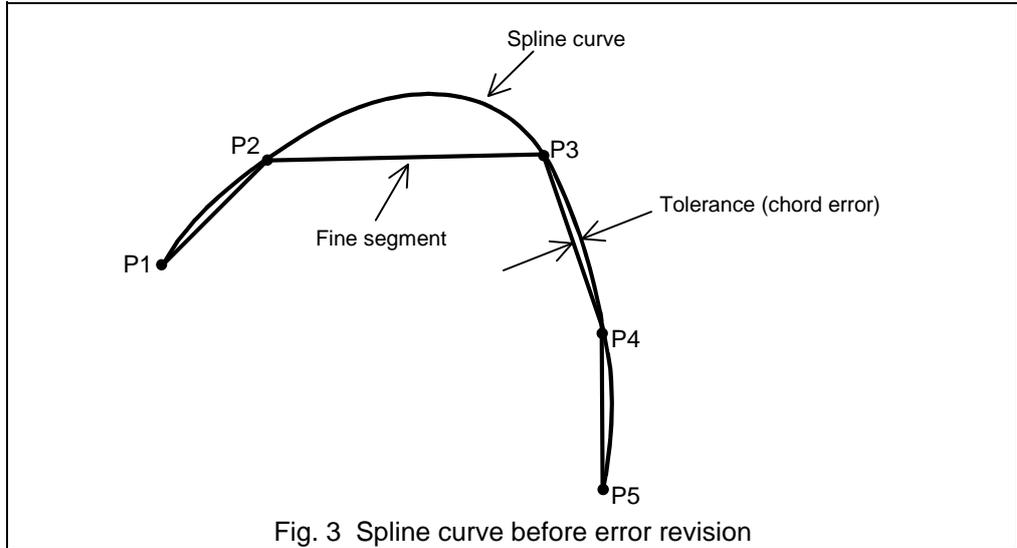
At sections where the tolerance (chord error) of the fine segment block and spline curve in a block not containing an inflection point, if the chord error in the corresponding section is larger than the value set in parameter (#8028 Toler-2), the spline curve shape is automatically revised so that the error is within the designated value. However, if the maximum chord error of the corresponding section is more than five times larger than the parameter "Toler-2" setting value, the spline function will be temporarily canceled.

The curve is revised only in the corresponding block.

The revisions are carried out with the following conditions for each block in the spline interpolation mode.

There is no inflection point in the spline curve, and the maximum error of the spline curve and linear block is larger than parameter "Toler-2". (Distance between P2-P3 in Fig. 3)

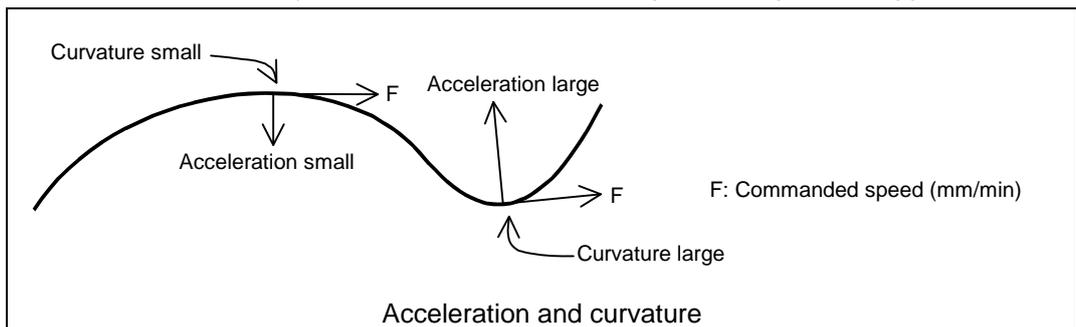
When the above conditions are satisfied, the spline curve will be revised so that the error between P2-P3 in Fig. 4 is within the designated value.



In parameter "Toler-2", set the tolerance for developing into fine segments with the CAM.

**(3) Curvature speed clamp**

The commanded speed F for the spline function during a segment linear arc will be the speed commanded in the previously set modal. However, if the axis is fed with the same speed, excessive acceleration may occur at the sections where the curvature is large (where curvature radius is small) as shown below. Thus, the speed clamp will be applied.

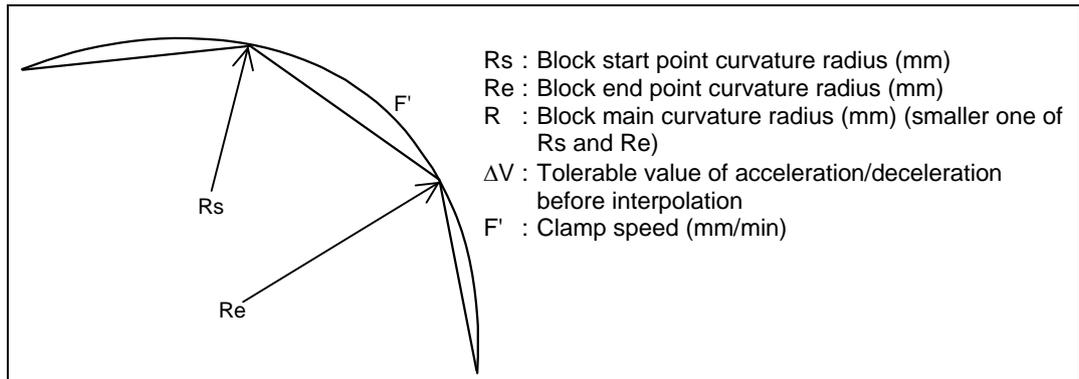


With the spline function, the high-accuracy control function is always valid. Thus, even if the curvature changes such as in this curve, the speed will be clamped so that the tolerable value of acceleration/deceleration before interpolation, which is calculated with the parameters, is not exceeded.

The clamp speed is set for each block, and the smaller of the curvature radius  $R_s$  at the curve block start point and the curvature radius  $R_e$  at the end point of the block will be used as the main curvature radius  $R$ . Using this main curvature radius  $R$ , the clamp speed  $F'$  will be calculated with expression (1).

The smaller of this clamp speed  $F'$  and the commanded speed  $F$  will be incorporated for the actual feedrate.

This allows cutting with an adequate feedrate corresponding the curvature radius along the entire curve.



$R_s$  : Block start point curvature radius (mm)  
 $R_e$  : Block end point curvature radius (mm)  
 $R$  : Block main curvature radius (mm) (smaller one of  $R_s$  and  $R_e$ )  
 $\Delta V$  : Tolerable value of acceleration/deceleration before interpolation  
 $F'$  : Clamp speed (mm/min)

$$F' = \sqrt{R \times \Delta V \times 60 \times 1000} \times \frac{100 - K_s}{100} \dots\dots\dots (1)$$

$$\Delta V = \frac{G1bF \text{ (mm/min)}}{G1btL \text{ (ms)}}$$

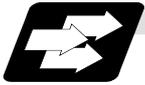
$G1bF$  : Target acceleration/deceleration before interpolation  
 $G1btL$  : Acceleration/deceleration time to reach the target speed  
 $K_s$  : #8019 R COMP (Accuracy coefficient)



**Precautions**

- (1) The spline function will be canceled during graphic check.
- (2) This function is valid when the base specifications parameters "#1267 ext03/bit0" is set to "1". If G05.1 Q2 is commanded when "0" is set, program error (P34) will occur.
- (3) If the specifications for this function are not provided and G05.1Q2 is commanded, the program error (P39) will occur.
- (4) Even if "-1" is set for parameter "#8030 MINUTE LENGTH", the spline function will be temporarily canceled by the cancel conditions (cancel angle, non-movement block, excessive chord error, etc.) other than the block length.
- (5) Command the G05.1 Q2 and G05.1 Q0 commands in independent blocks.  
 A program error (P33) will occur if these are not commanded in independent blocks.
- (6) A program error (P33) will occur if the G05.1 command block does not contain a Q command.
- (7) A program error (P34) will occur if the number of axis in part system does not exceed 3.

## 13.19 High-accuracy Spline Interpolation ; G61.2



## Function and purpose

This function automatically generates a spline curve that passes through a sequence of points commanded by the fine segment machining program, and interpolates the path along this curve. This allows highly accurate machining at a high speed.

This function has two functions; fairing function to delete unnecessary fine blocks, and spline interpolation to connect smoothly a sequence of points commanded by the program.

The high-accuracy control function G61.1 is valid also.

Function	Contents
Fairing	Super-fine blocks often included in the data generated with CAM are deleted. Such a super-fine block scratches the machining surface, and might increase machining time because of acceleration/deceleration. This function avoids these troubles.
Spline interpolation	The spline interpolation connects smoothly a sequence of points commanded by program. As a result, the glossy machining surface can be obtained, and the machining time can be reduced because the frequency of the corner deceleration decreases compared with conventional linear interpolation.



## Command format

**G61.2 X\_\_ Y\_\_ Z\_\_ F\_\_ ;**

or

**G61.2 ;**

G61.2 : High-accuracy spline interpolation mode ON

X : X axis end point coordinates

Y : Y axis end point coordinates

Z : ZX axis end point coordinates

F : Feedrate command

**G64 ;**

G64 : High-accuracy spline interpolation mode OFF



## Detailed description

## (1) Fairing

Refer to "Additional functions when high-speed high-accuracy control 2 mode is ON" in "13.17.1 High-speed high-accuracy control".

## (2) Spline interpolation

Refer to "Detailed description" in "13.18 Spline".



## Example of program

```

:
G91 ;
G61.2 ;..... High-accuracy spline interpolation mode ON
G01 X0.1 Z0.1 F1000 ;
X0.1 Z-0.2 ;
Y0.1 ;
X-0.1 Z-0.05 ;
X-0.1 Z-0.3 ;
G64 ;..... High-accuracy spline interpolation mode OFF
:

```

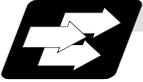
- (1) The spline function carries out spline interpolation when the following conditions are all satisfied. If the following conditions are not satisfied, the spline function is canceled once, and it is judged whether to carry out new spline from the next block.
  - It is the movement only of three axes set to the basic axes I, J and K.
  - When the block length is shorter than the machining parameter "#8030 MINUTE LENGTH".
  - When the movement amount is not 0.
  - The group 1 command is G01 (linear interpolation).
  - It is not during the fixed cycle modal.
  - It is not during hypothetical axis interpolation mode.
  - It is not during 3-dimensional coordinate conversion modal.
  - A single block is not being executed.
- (2) The spline function is modal command of group 13. This function is valid from G61.2 command block.
- (3) The spline function is canceled by group 13 command (G61 to G64).
- (4) The spline function is canceled by NC reset 2, reset & rewind, NC reset 1 (the setting which does not hold modal when NC is reset) or power ON/OFF.



## Precautions

- (1) This function is valid when the base specifications parameters "#1267 ext03/bit0" is set to "1". If "G61.2" is commanded when "0" is set, program error (P34) will occur.
- (2) If the specifications for this function are not provided and "G61.2" is commanded, the program error (P39) will occur.
- (3) Even if "-1" is set for parameter "#8030 MINUTE LENGTH", the spline function will be temporarily canceled by the cancel conditions (cancel angle, non-movement block, excessive chord error, etc.) other than the block length.
- (4) Graphic check draws a shape of spline interpolation OFF.
- (5) A program error (P34) will occur if the number of axis in part system does not exceed 3.

## 13.20 Scaling; G50/G51



## Function and purpose

By multiplying the moving axis command values within the range specified under this command by the factor, the shape commanded by the program can be enlarged or reduced to the desired size.



## Command format

## (1) Scaling ON

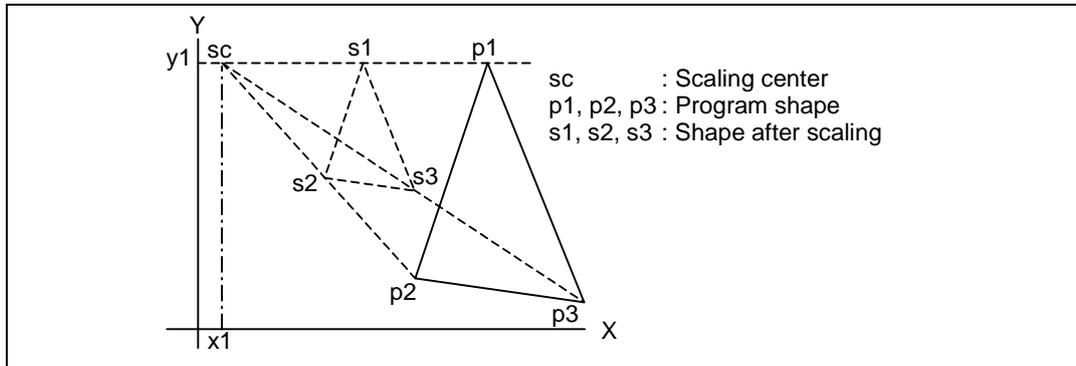
**G51 X\_\_ Y\_\_ Z\_\_ P\_\_ ; Scaling ON**

X, Y, Z : Scaling center coordinates

P : Scaling factor

## (2) Scaling cancel

**G50 ; Scaling cancel**



## (3) When individually set the scaling factor to three basic axes

**G51 X\_\_ Y\_\_ Z\_\_ I\_\_ J\_\_ K\_\_ ; Scaling ON**

X, Y, Z : Scaling center coordinates

I : Scaling factor of basic 1st axis

J : Scaling factor of basic 2nd axis

K : Scaling factor of basic 3rd axis



## Detailed description

**(1) Specifying the scaling axis, scaling center and its factor**

Commanding G51 selects the scaling mode. The G51 command only specifies the scaling axis, its center and factor, and does not move the axis.

Though the scaling mode is selected by the G51 command, the axis actually valid for scaling is the axis where the scaling center has been specified.

**(a) Scaling center**

- Specify the scaling center in accordance with the then absolute/incremental mode (G90/G91).
- The scaling center must be specified also when the current position is defined as a center.
- As described above, the axis valid for scaling is only the axis whose center has been specified.

**(b) Scaling factor**

- Use the address P or I, J, K to specify the scaling factor.
- Minimum command unit : 0.000001
- Command range: Either -99999999 to 99999999 (-99.999999 to 99.999999 times) or -99.999999 to 99.999999 is valid, but the decimal point command is valid only after the G51 command.
- When the factor is not specified in the same block as G51, the factor set with the parameter (#8072 SCALING P) is used.
- When the address P and the address I, J, K are commanded in the same block, a magnification specified by the address I, J, K is applied for the basic three axes. And a magnification specified by the address P is applied for other axes.
- If changed during the scaling mode, the value of this parameter is not made valid. Scaling is performed with the setting value that was used when G51 was commanded.
- When the factor is not specified in both the program and parameter, it is calculated as 1.

**(c) A program error will occur in either of the following cases.**

- Scaling was commanded though there was no scaling specification. (Program error (P350))
- The upper limit of the factor specifying range was exceeded in the same block as G51. (Program error (P 35))  
(If the machining parameter scaling factor is used, the magnification is calculated as one time in the following cases;  $-0.000001 < \text{magnification} < 0.000001$ , more than 99.999999, or less than -99.999999.)

**(2) Scaling cancel**

When G50 is commanded, scaling is canceled.

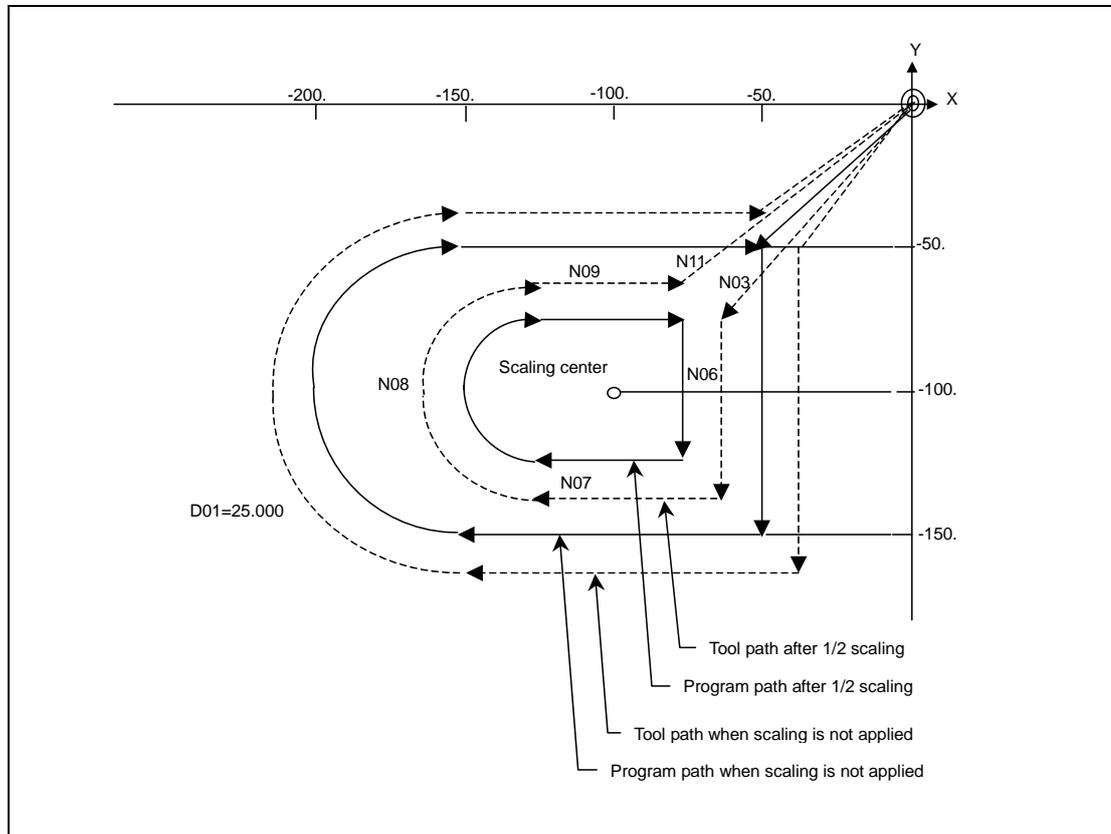
**Precautions**

- (1) Scaling is not applied to the compensation amounts of tool radius compensation, tool position compensation, tool length compensation and the like. (Compensation is calculated for the shape after scaling.)
- (2) Scaling is valid for only the movement command in automatic operation. It is invalid for manual movement.
- (3) For X, Y and Z, scaling is valid for only the specified axes and is not applied to the axes not specified.
- (4) When an arc is commanded and scaling is valid for one of the two axes configuring the arc plane, a program error (P70) will occur.
- (5) When M02 or M30 is commanded, or when NC reset is carried out during the scaling mode, the mode switches to a cancel mode.
- (6) When the coordinate system is shifted (G92, G52 command) during scaling, the scaling center is also shifted by the difference amount.
- (7) If manual interruption is made during scaling, manual ABS selection is ignored for the movement followed by an incremental value command and operation performed is the same as in manual ABS OFF.



## Example of program

(Example 1)



&lt;Program&gt;

```

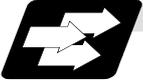
N01 G92    X0 Y0 Z0;
N02 G90 G51 X-100. Y-100. P0.5;
N03 G00 G43 Z-200. H02;
N04 G41    X-50. Y-50. D01;
N05 G01    Z-250. F1000;
N06        Y-150. F200;
N07        X-150.;
N08 G02    Y-50. J50.;
N09 G01    X-50.;
N10 G00 G44 Z0;
N11 G40 G50 X0 Y0;
N12 M02;

```

**Relation with other functions**

- (1) G27 reference position check command  
When G27 is commanded during scaling, scaling is canceled at completion of the command.
- (2) Reference position return command (G28, G29, G30)  
When the G28 or G30 reference position return command is issued during scaling, scaling is canceled at the midpoint and the axis returns to the reference position. When the midpoint is to be ignored, the axis returns to the reference position directly. When G29 is commanded during scaling, scaling is applied to the movement after the midpoint.
- (3) G60 (unidirectional positioning) command  
If the G60 (unidirectional positioning) command is given during scaling, scaling is applied to the final positioning point and is not applied to the creep amount. Namely, the creep amount is uniform regardless of scaling.
- (4) Workpiece coordinate system switching  
When the workpiece coordinate system is switched during scaling, the scaling center is shifted by the difference between the offset amounts of the new and old workpiece coordinate systems.
- (5) During figure rotation  
When scaling is commanded during figure rotation, scaling is applied to the center of the figure rotation and the rotating radius.
- (6) Scaling command in figure rotation subprogram  
When scaling is commanded in the subprogram of the figure rotation, scaling can be applied only to the shape designated by the subprogram without applying scaling to the rotating radius of the figure rotation.
- (7) During coordinate rotation  
When scaling is commanded during coordinate rotation, the scaling center rotates. Scaling is executed at that rotated scaling center.
- (8) G51 command  
When the G51 command is issued during the scaling mode, the axis whose center was newly specified is also made valid for scaling. Also, the factor under the latest G51 command is made valid.

## 13.21 Coordinate Rotation by Program; G68.1/G69.1



## Function and purpose

When machining a complicated shape at a position rotated in respect to the coordinate system, the shape before rotation can be programmed on the local coordinate system, rotation angle designated with the program coordinate rotation command, and the rotated shaped machined.



## Command format

## (1) Coordinate rotation ON

**G68.1 X\_\_ Y\_\_ R\_\_;**      **Coordinate rotation ON**

G68.1 : Coordinate rotation command

X, Y : Rotation center coordinates.

Two axes (X,Y or Z) corresponding to the selected plane are designated with absolute positions.

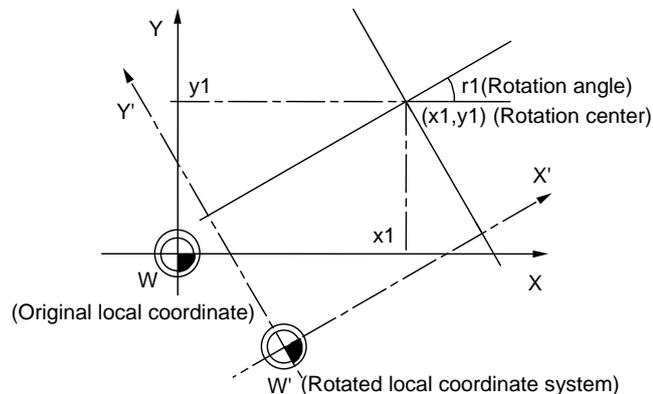
R : Rotation angle

The counterclockwise direction is +.

Select the command plane with G17 to G19.

## (2) Coordinate rotation cancel

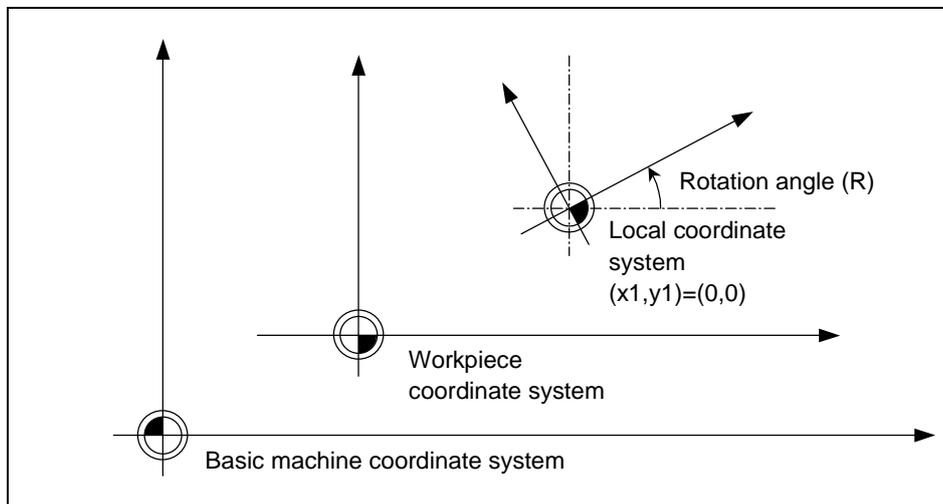
**G69.1 ;**      **Coordinate rotation cancel**





## Detailed description

- (1) Always command the rotation center coordinate (x1, y1) with an absolute value. Even if commanded with an incremental address, it will not be handled as an incremental value. The rotation angle "r" depends on the G90/G91 modal.
- (2) If the rotation center coordinates (x1, y1) are omitted, the position where the G68.1 command was executed will be the rotation center.
- (3) The rotation takes place in the counterclockwise direction by the angle designated in rotation angle r1.
- (4) The rotation angle r1 setting range is -360.000 to 360.000. If a command exceeding 360 degrees is issued, the remainder divided by 360 degrees will be the command.
- (5) Since the rotation angle "r1" is modal data, if once commanded, it will not be changed until the new angle data is commanded. Thus, the rotation angle "r1" can be omitted. If the rotation angle is omitted in spite that G68.1 is commanded for the first time, "r1" will be regarded as "0".
- (6) The program coordinate rotation is a function used on the local coordinate system. The relation of the rotated coordinate system, workpiece coordinate system and basic machine coordinate system is shown below.



- (6) The coordinate rotation command during coordinate rotation is processed as center coordinates and rotation angle changes.
- (7) If M02 or M30 is commanded or the reset signal is input during the coordinate rotation mode, the coordinate rotation mode will be canceled.
- (8) G68.1 is displayed on the modal information screen during the coordinate rotation mode. When the mode is canceled, the display changes to G69.1. (The modal value is not displayed for the rotation angle command R.)
- (9) The program coordinate rotation function is valid only in the automatic operation mode.



Example of program (Program coordinate rotation by absolute command)

<pre> N01 G28 X0. Y0.; N02 G54 G52 X200. Y100. ; N03 T10 ; N04 G68.1 X-100. Y0. R60. ; N05 G22 H101 ; N06 G69.1 ; N07 G54 G52 X0 Y0 ; N08 M02 ;  Subprogram (Shape programmed with original coordinate system) N101 G00 X-100. Y-40.; N102 G83 X-150. R-20. F100 ; N103 G00 Y40. ; N104 G83 X-150. R-20. F100 ; N105 G23                 </pre>	<pre> Local coordinate designation Coordinate rotation ON Subprogram execution Coordinate rotation cancel Local coordinate system cancel End                 </pre>	
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------	--



**Example of program (Operation of only one axis was commanded by first movement command after coordinate rotation command)**

Command basically two axes in the rotation plane by the absolute value immediately after the coordinate rotation command.

When commanding one axis only, the following two kinds of operation can be selected by the parameter "#19003 PRG coord rot type".

- (1) When "#19003 PRG coord rot type" is "1", the operation is the same as that with "G50.Y0." commanded in N04. The end point is calculated on the assumption that the start point rotates along with the coordinates' rotation.

```

N01 G17 G28 X0. Y0.;
N02 G90 G92 G53 X0. Y0.;
N03 G68.1 X40. Y0. R90.;   Coordinate rotation ON
N04 X50.;
N05 Y50.;
N06 G69.1 ;               Coordinate rotation cancel
N07 M02 ;                 End
                
```

Start point:  $(X', Y') = (0, 0)$

- (2) When "#19003 PRG coord rot type" is "0", only axis commanded in N04 (X' Axis) is moved. The start point does not rotate along with the coordinate rotation; therefore the end position is calculated based on the current position on local coordinate system before rotation.

```

N01 G17 G28 X0. Y0.;
N02 G90 G92 G53 X0. Y0.;
N03 G68.1 X40. Y0. R90.;   Coordinate rotation ON
N04 X50.;
N05 Y50.;
N06 G69.1 ;               Coordinate rotation cancel
N07 M02 ;                 End
                
```

Start point:  $(X', Y') = (40, 40)$



Example of program (Local coordinate designation during program coordinate rotation)

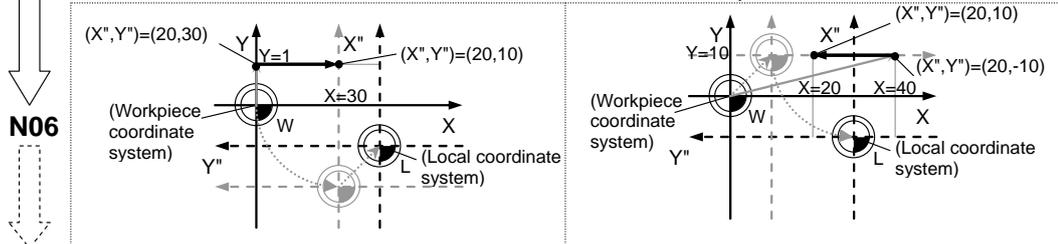
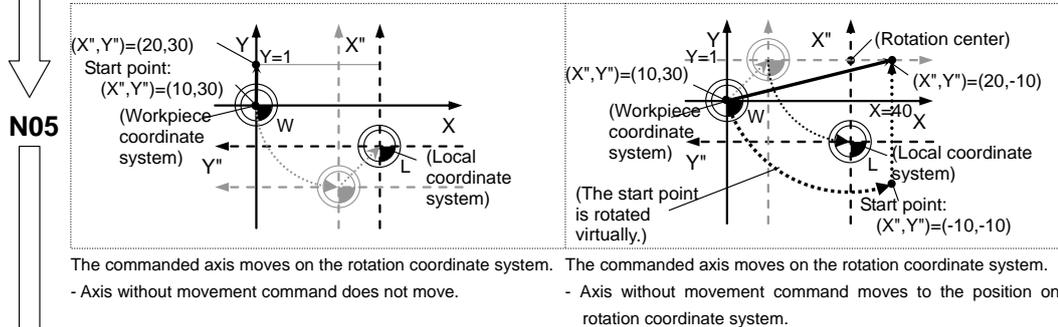
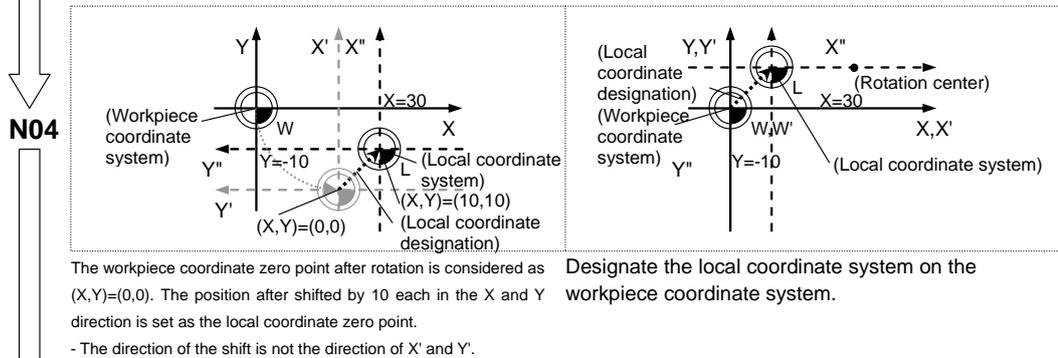
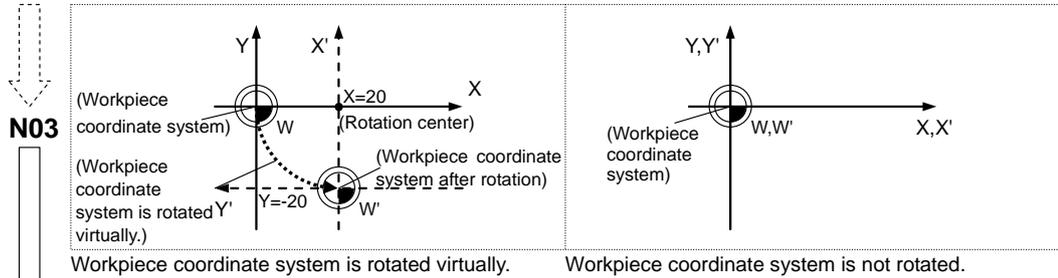
- (1) When "#19003 PRG coord rot type" is "0", it is on the coordinate system after coordinates rotation that the commanded position is set as the local coordinate zero point.
- (2) When "#19003 PRG coord rot type" is "1", it is on the coordinate system before coordinates rotation that the commanded position is set as the local coordinate zero point. Then the coordinates system is rotated.

```

N01 G17 G28 X0. Y0.;
N02 G90 G92 G53 X0. Y0.;
N03 G68.1 X20. Y0. R90.;      Coordinate rotation ON
N04 G52 X10. Y10.;          Local coordinate designation
N05 X20.;
N06 Y10.;
N07 G69.1 ;                Coordinate rotation cancel
N08 M02 ;                    End
    
```

(1) Operation of #19003 = 0

(2) Operation of #19003 = 1

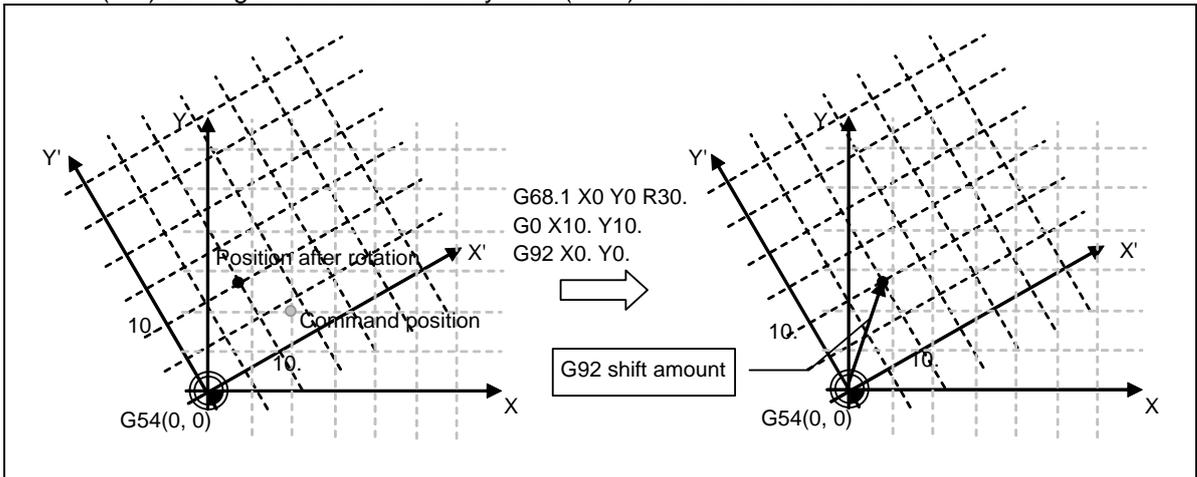




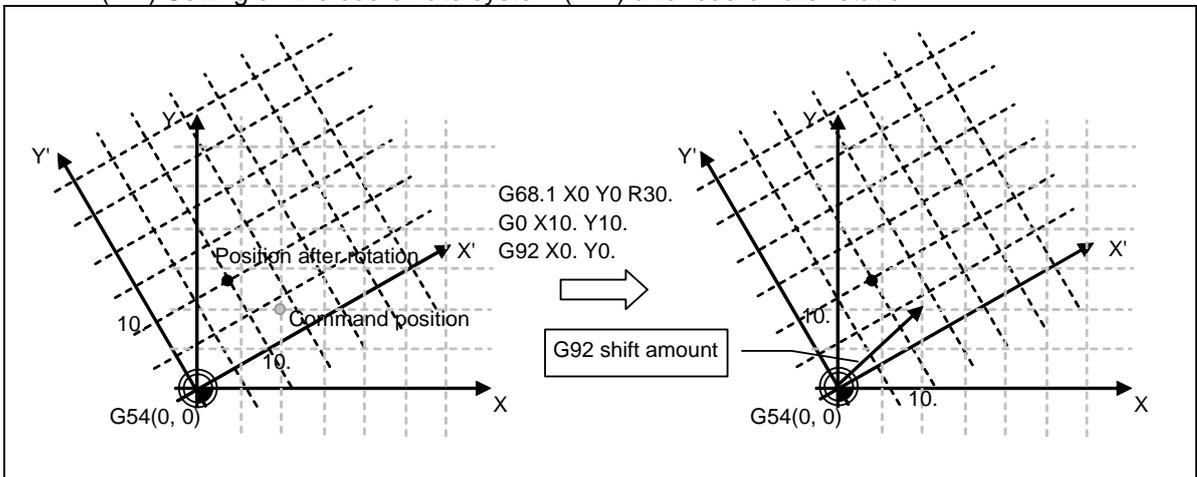
Example of program (Coordinate system designation during program coordinate rotation)

When the coordinate system setting (G92) is executed during program coordinate rotation, this program operates similarly as "Local coordinate designation during program coordinate rotation".

- (1) When "#19003 PRG coord rot type" is "0", it is on the coordinate system after coordinates rotation that the current position is preset to the command position.  
(Ex.) Setting on the coordinate system (X'-Y') after coordinate rotation



- (2) When "#19003 PRG coord rot type" is "1", it is on the coordinate system before coordinates rotation that the current position is preset to the command position. Then the coordinate system is rotated.  
(Ex.) Setting on the coordinate system (X-Y) after coordinate rotation



**(Note 1)** When "#19003 PRG coord rot type" is "1" and the coordinate system setting (G92) is executed during coordinate rotation mode, the rotation center of program coordinate rotation is not shifted. (The same position from the basic machine coordinate system)

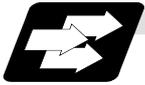
**Precautions**

- (1) Always command an absolute value for the movement command immediately after G68.1 and G69.1.
- (2) If manual absolute is ON and manual interrupt is issued for the coordinate rotation axis, do not use automatic operation for the following absolute value command.
- (3) The intermediate point during reference position return is the position after the coordinates are rotated.
- (4) If the workpiece coordinate system offset amount is changed during the coordinate rotation mode, the rotation center for the program coordinate rotation will be shifted. (The center will follow the coordinate system.)
- (5) If the workpiece coordinates are changed during the coordinate rotation mode (ex., from G54 to G55), the rotation center of the program coordinate rotation will be the position in the commanded coordinate system. (The center position will be the same looking from the basic machine coordinate system.)
- (6) If coordinate rotation is executed to the G00 command for only one axis, two axes will move. If G00 non-interpolation (parameter "#1086 GOIntp" = 1) is set, each axis will move independently at the respective rapid traverse rates. If the axis must be moved linearly (interpolated) from the start point to the end point (such as during the hole machining cycle), always turn G00 non-interpolation OFF (parameter "#1086 GOIntp" = 0). The feedrate in this case is the composite speed of each axis' rapid traverse rate, so the movement speed will be faster than when moving only one axis (before coordinate rotation).
- (7) If the coordinate rotation specifications are not provided, a program error (P260) will occur when coordinate rotation is commanded.
- (8) The offset operation during the coordinate rotation mode compensates the local coordinate system after coordinate rotation. The compensation direction is the coordinate system before rotation.
- (9) Mirror image during the coordinate rotation mode is applied on the local coordinate system after coordinate rotation.
- (10) All position displays show the positions after coordinate rotation with the local coordinate system before rotation.
- (11) When the coordinate value variables are read, the positions are all on the coordinate system before rotation.
- (12) The coordinates can also be rotated for the parallel axis. Select the plane that contains the parallel axis before issuing the G68.1 command. The plane cannot be selected in the same block as the G68.1 command.
- (13) The coordinates can be rotated for the rotation axis. The angle will be interpreted as a length when rotating.

**Relation with other functions**

- (1) Program error (P111) will occur if the plane selection code is commanded during the coordinate rotation mode.
- (2) Program error (P485) will occur if pole coordinate interpolation is commanded during the coordinate rotation mode.
- (3) Program error (P481) will occur if coordinate rotation is commanded during the pole coordinate interpolation mode.
- (4) Program error (P485) will occur if cylindrical interpolation is commanded during the coordinate rotation mode.
- (5) Program error (P481) will occur if coordinate rotation is commanded during the cylindrical interpolation mode.
- (6) Program error (P34) will occur if the workpiece coordinate system preset (G92.1) is commanded during the coordinate rotation mode.
- (7) Program error (P34) will occur if high-accuracy control mode, high-speed machining mode 3, high-speed high-accuracy I or II is commanded during the coordinate rotation mode.

## 13.22 Coordinate Rotation Input by Parameter; G10

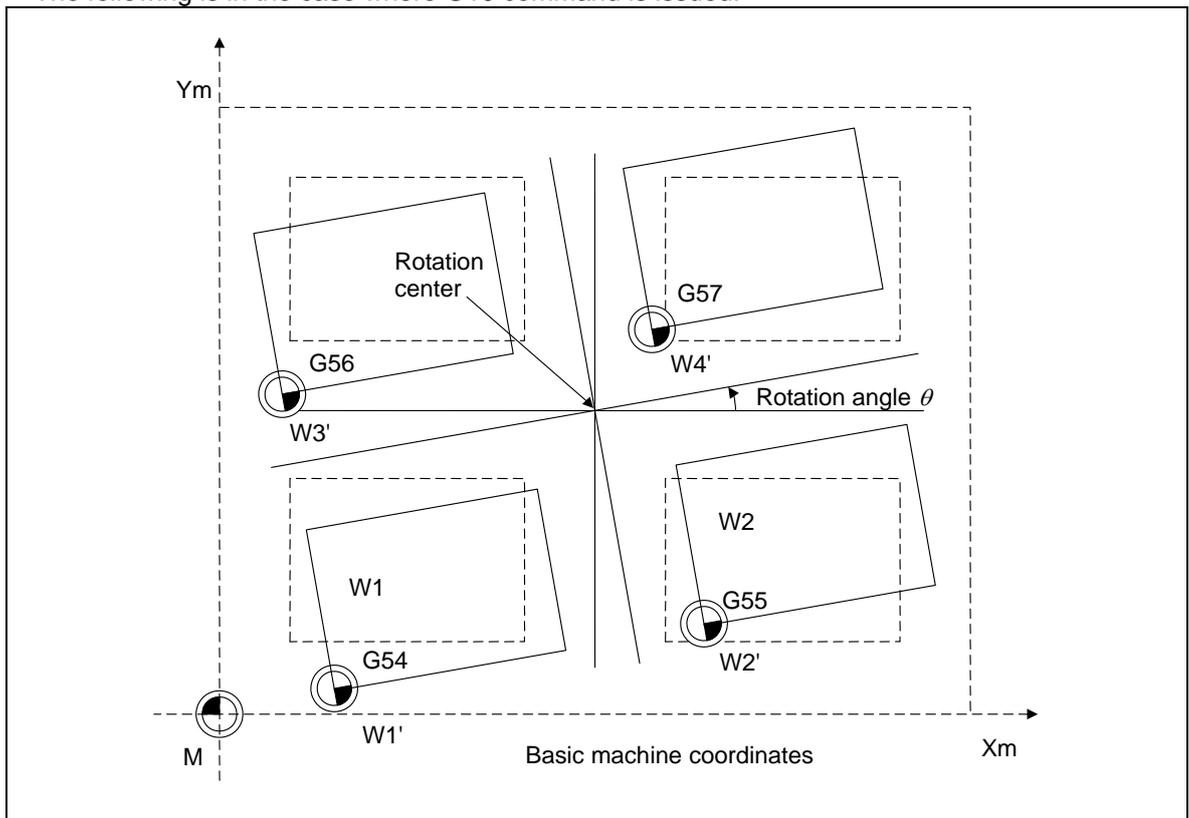


## Function and purpose

If a deviation occurs between the workpiece alignment line and machine coordinate system's coordinate axis when the workpiece is mounted, the machine can be controlled to rotate the machining program coordinates according to the workpiece alignment line deviation. The coordinate rotation amount is set with the parameters. The parameter can be set on the parameter setting screen or with the G10 command.

Note that when the G10 command is used, the separate additional specification "parameter input by program" is required.

The following is in the case where G10 command is issued.





## Command format

**G10 I\_\_ J\_\_ ;**

**G10 K\_\_ ;**

**I** : Horizontal vector. Command a value corresponding to "Coord rot plane (H)" which is set in the parameter input screen. Command range: -999999.999999 to 999999.999999

Coordinate rotation angle is automatically calculated when commanding vector contents.

**J** : Vertical vector. Command a value corresponding to "Coord rot plane (V)" which is set in the parameter input screen. Command range: -999999.999999 to 999999.999999

Coordinate rotation angle is automatically calculated when commanding vector contents.

**K** : Rotation angle. Command a value corresponding to "Coord rot angle" which is set in the parameter input screen. Command range: -360.000000 to 360.000000  
Set the vector contents to "0" when commanding coordinate rotation angle.

Parameters specified in the parameter setting screen can be changed from the machining program.



## Detailed description

- (1) As for the rotation center coordinate position, designate the position on the machine coordinate system.
- (2) All workpiece coordinate systems from G54 to G59 rotate with the rotation command. While the machine coordinate system does not rotate, it can be understood that there is a hypothetical machine coordinate system in the coordinate system after rotation.
- (3) The coordinate value counter does not rotate. The position in the original workpiece coordinate system (non-rotated) is displayed.
- (4) If the setting is made on the parameter setting screen, the setting is validated with the cycle start after the parameter is set. If set with the G10 command, settings are immediately validated.



## Example of program

(1) To use for compensating positional deviation of pallet changer

<p>N01 G28 X0 Y0 Z0 ;            N02 G22 L9000 ; Pallet deviation measurement            N03 G90 G53 X0 Y0 ; Parallel movement amount shift            N04 G92 X0 Y0 ; Parallel movement amount definition            N05 G10 K15. ; Rotation amount definition            N06 G90 G54 G00 X0 Y0 ; G54 workpiece machining            N07 G22 H101 ;            N08 G90 G55 G00 X0 Y0 ; G55 workpiece machining            N09 G22 H101 ;            N10 G90 G56 G00 X0 Y0 ; G56 workpiece machining            N11 G22 H101 ;</p>	<p>N12 G90 G57 G00 X0 Y0 ; G57 workpiece machining            N13 G22 H101 ;            N14 G27 X0 Y0 Z0 ;            N15 M02 ;</p> <p>Machining shape program            N101 G91 G01 G42 D01 F300 ;            N102 X100 ;            N103 G03 Y50. R25. ;            N104 G01 X-100. ;            N105 G03 Y-50. R25. ;            N106 G01 G40 ;            N107 G23 ;</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



## Precautions

- (1) If rotation angle zero is commanded while carrying out coordinate rotation, it will be canceled at the next movement command regardless of the G90 or G91 setting.
- (2) Command the first movement after this command with the G00 or G01 mode. If an arc command is issued, the arc start point will not be rotated. However, only the arc end point will rotate. This will cause the start point radius and end point radius to differ, and the program error (P70) will occur.
- (3) When data is input with the data input/output function, coordinate rotation angle is considered to exist, and automatic calculation from the vector component will not be executed.

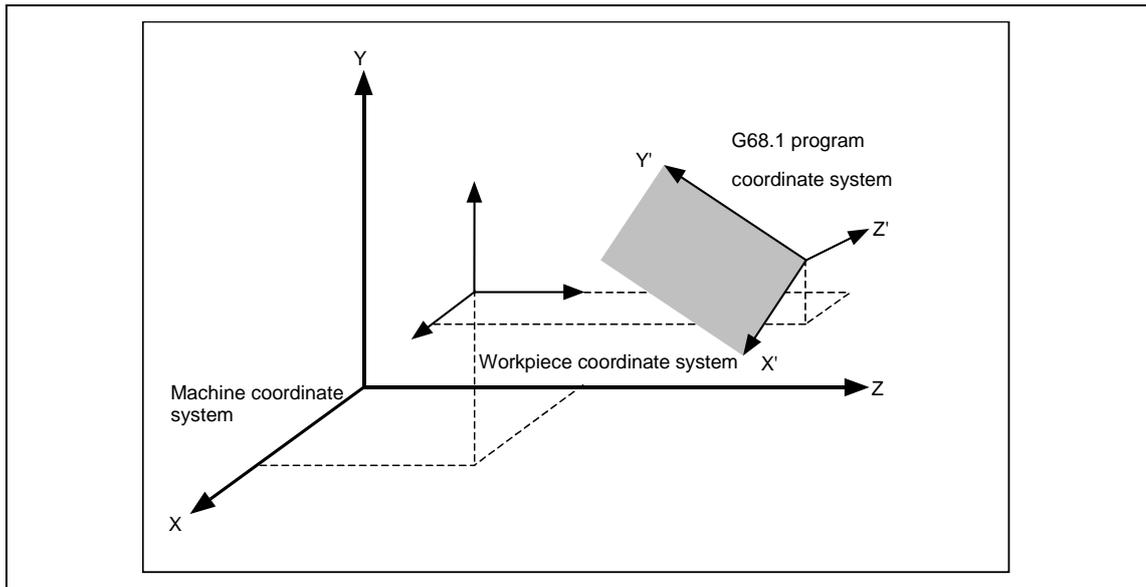
## 13.23 3-dimensional Coordinate Conversion; G68.1/69.1



## Function and purpose

With the 3-dimensional coordinate conversion function, a new coordinate system can be defined by rotating and moving in parallel the zero point in respect to the X, Y and Z axes of the currently set workpiece coordinate system. By using this function, an arbitrary spatial plane can be defined, and machining on that plane can be carried out with normal program commands.

An option is required to validate this function. If the 3-dimensional coordinate conversion is commanded when the option is not added, a program error will occur.



When the G68.1 command is issued, the zero point is shifted by the command value (x, y, z) in respect to the current local coordinate system. A new G68.1 program coordinate system rotated by the designated rotation angle "r" in respect to the commanded rotation center direction (i, j, k) is created.

The local coordinate system is the same as the workpiece coordinate system when the local coordinate system offset is not ON.



## Command format

**G68.1 X\_\_ Y\_\_ Z\_\_ I\_\_ J\_\_ K\_\_ R\_\_ ;**

G68.1 : 3-dimensional coordinate conversion mode command  
 X,Y,Z : Rotation center coordinates  
 Designate with the absolute position of the local coordinate system.  
 I,J,K : Rotation center axis direction (1: Designated 0: Not designated)  
 Note that "1" is designated for only one of the three axes. "0" is designated for the other two axes.  
 R : Rotation angle  
 The counterclockwise direction looking at the rotation center from the rotation center axis direction is positive (+).  
 The setting range is -360 to 360°, and the increment follows the setting and display unit.

**G69.1 ;**

G69.1 : 3-dimensional coordinate conversion mode cancel command

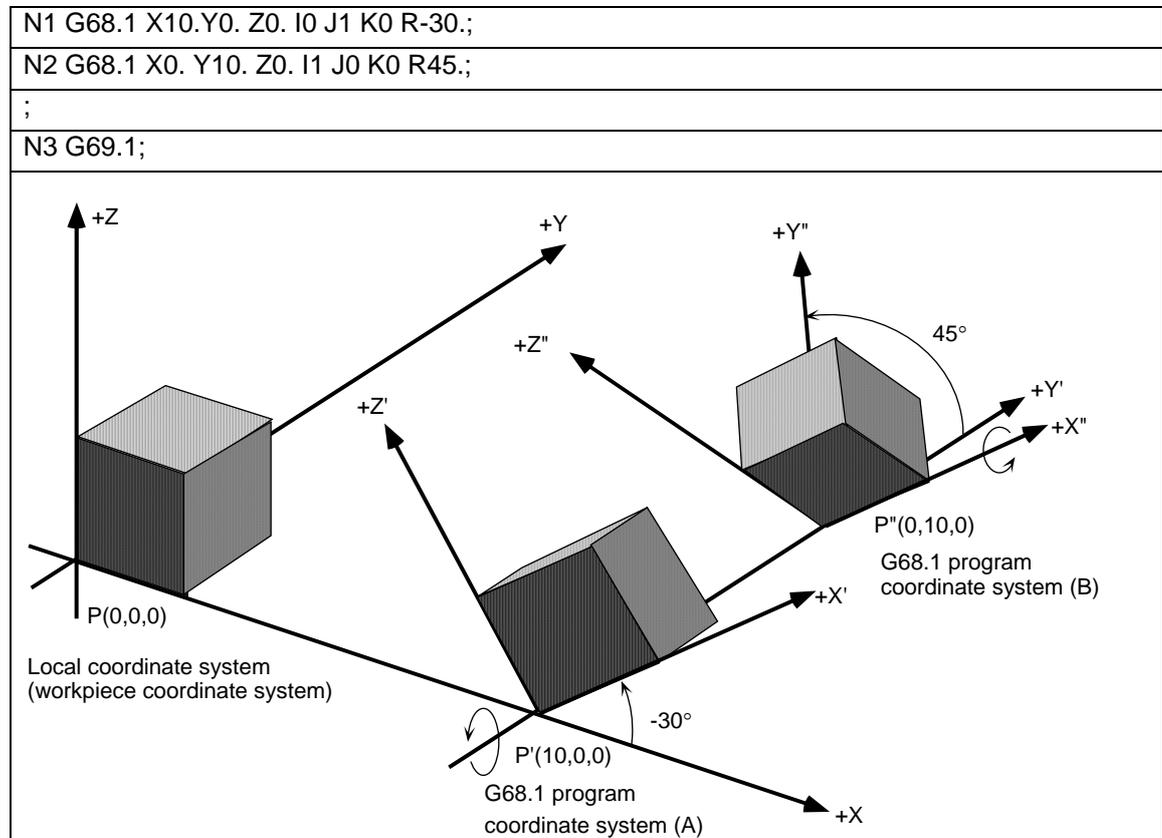


## Detailed description

- (1) Command the rotation center coordinates with absolute values.
- (2) If the rotation center coordinates are omitted, the zero point of the currently set coordinate system will be the rotation center coordinates.
- (3) Designate values for I, J and K. If even one is not designated, program coordinate rotation command will be valid.
- (4) Set "1" in only one of I, J and K, and set "0" for the other two.  
The program error (P33) will occur if "1" is set in two or more.
- (5) The program error (P33) will also occur if "0" is set for all I, J and K.
- (6) When addresses I, J and K are not designated, this will be handled as the program coordinate rotation.
- (7) If a number other than "0" (including two-digit numbers) is designated for addresses I, J and K, this will be handled as "1". If a blank is designated, this will be handled as "0".
- (8) If a G code that cannot be commanded in the 3-dimensional coordinate conversion modal is issued, the program error (P921) will occur. When the 3-dimensional coordinate conversion is commanded during the modal where 3-dimensional coordinate conversion cannot be carried out, the program error (P922) will also occur. Furthermore, the G codes that cannot be commanded together with G68.1 are commanded in the same block as G68.1, the program error (P923) will occur.
- (9) The 3-dimensional coordinate conversion command for the rotary axis will result in the program error (P32).
- (10) If a 3-dimensional coordinate conversion command is issued when there is no specifications for 3-dimensional coordinate conversion, the program error (P920) will occur.



## Example of program 1



- (1) With N1, the origin is shifted by  $[x, y, z] = (10., 0, 0)$  in respect to the currently set local coordinate system. The new G68.1 program coordinate system (A in the figure above) rotated  $-30^\circ$  in the counterclockwise direction using the Y axis as the center is set.
- (2) With N2, the origin is shifted by  $[x, y, z] = (0, 10., 0)$  in respect to the newly set G68.1 program coordinate system (A in the figure above). The new G68.1 program coordinate system (B in the figure above) rotated  $+45^\circ$  in the counterclockwise direction using the X axis as the center is set.
- (3) With N3, the G68.1 program coordinate systems that have been set are all canceled, and the state prior to where the first G68.1 has been commanded is returned.



### Coordinate system

- (1) By issuing the 3-dimensional coordinate conversion command, a new coordinate system (G68.1 program coordinate system) will be created on the local coordinate system.
- (2) The coordinate system for the 3-dimensional coordinate conversion rotation center coordinates is the local coordinate system. Thus, these coordinate systems are affected by the following coordinate system offset and coordinate system shift amount.
  - Local coordinate system offset issued with G52 command
  - G92 shift amount issued with G92 command
  - Coordinate system offset corresponding to workpiece coordinate system selected with command
  - External workpiece coordinate offset
  - Manual interruption amount or manual feed amount when manual ABS is OFF
- (3) If 3-dimensional coordinate conversion is commanded again during the 3-dimensional coordinate conversion modal, a G68.1 program coordinate system is created on the current G68.1 program coordinate system, and is used as a new G68.1 program coordinate system.
- (4) The local coordinate system cannot be created (G52) on the G68.1 program coordinate system. (If G52 is issued, the program error (P921) will occur.)
- (5) G68.1 program coordinate system can be reset either by G69.1 command or reset inputting. (Exclude reset 1 when "0" is set to the parameter "#1151 rsint".)



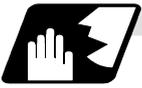
### Rotation angle

- (1) For the rotation angle, the counterclockwise direction looking at the rotation center from the plus direction of the rotation center axis is the plus (+) direction.
- (2) The rotation angle command unit with no decimal point depends on the parameter "#1078 Decpt" (Decimal point type 2).
- (3) If the rotation angle is omitted, the rotation angle will be handled as 0°.



### Rotation center coordinate

- (1) The G68.1 rotation center coordinate system is commanded with the local coordinate system (G68.1 program coordinate system during the 3-dimensional coordinate conversion modal).
- (2) The rotation center coordinate designation is handled as an absolute value designation whether or not an absolute/incremental modal (G90/G91) is currently being executed.
- (3) If the rotation center coordinate is omitted, it will be handled as if the zero point of the current local coordinate (G68.1 program coordinate system during the 3-dimensional coordinate conversion modal) is designated for the omitted address's axis. (The same as when "0" is just set.)



**G68.1 multiple commands**

By commanding 3-dimensional coordinate conversion during the 3-dimensional coordinate conversion modal, two or more multiple commands can be issued.

- (1) The 3-dimensional coordinate conversion command in the 3-dimensional coordinate conversion modal is combined with the conversion in the modal.
- (2) If the 3-dimensional coordinate conversion is overlapped during the 3-dimensional coordinate conversion modal, the overlapped 3-dimensional coordinate conversion will be created on the coordinate system (G68.1 program coordinate system) created with the 3-dimensional coordinate conversion in the modal. Thus, the rotary axis and coordinates must be designated with this G68.1 program coordinate system.

If creating a 90° rotated coordinate system for X axis and Y axis each, commands must be issued as in Example 2, but Example 1.

<b>&lt;Example 1&gt;</b>	G68.1 X0.Y0.Z0.I1J0K0 R90.; G68.1 X0.Y0.Z0.I0J1K0 R90.;	X axis rotation 90° Y axis rotation 90° (Th (Y axis designated here is the same as the Z axis in the original coordinate system.)
<b>&lt;Example 2&gt;</b>	G68.1 X0.Y0.Z0.I1J0K0 R90.; G68.1 X0.Y0.Z0.I0J0K1 R-90.;	X axis rotation 90° Z axis rotation 90° (Th (Z axis -90 rotation designated here is the same as the Y axis +90 rotation in the original coordinate system.)



**Conversion method for 3-dimensional coordinate conversion**

The coordinate values (Xp, Yp, Zp) in the newly set G68.1 program coordinate system and the coordinate values (Xm, Ym, Zm) in the reference workpiece coordinate system are converted as shown below.

<b>First G68.1 command</b>	
$[X_m, Y_m, Z_m, 1] = [X_p, Y_p, Z_p, 1] R1T1 \dots \dots \dots$	(Forward row)
$[X_p, Y_p, Z_p, 1] = [X_m, Y_m, Z_m, 1] (T1^{-1}) (R1^{-1}) \dots \dots \dots$	(Reverse row)
<b>Second G68.1 command</b>	
$[X_m, Y_m, Z_m, 1] = [X_p, Y_p, Z_p, 1] R2T2R1T1$	
$[X_p, Y_p, Z_p, 1] = [X_m, Y_m, Z_m, 1] (T1^{-1}) (R1^{-1}) (T2^{-1}) (R2^{-1})$	

R1, R2: Rotation row calculated from first and second G68.1 parameter  
T1, T2: Movement row calculated from first and second G68.1 parameter

### 13. Program Support Functions

#### 13.23 3-dimensional Coordinate Conversion; G68.1/G69.1

The conversion rows R<sub>n</sub> and T<sub>n</sub> (n = 1, 2) are as follow.

R<sub>n</sub> conversion row

I designation (rotation around X axis)	J designation (rotation around Y axis)	K designation (rotation around Z axis)
$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos R & \sin R & 0 \\ 0 & -\sin R & \cos R & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \cos R & 0 & -\sin R & 0 \\ 0 & 1 & 0 & 0 \\ \sin R & 0 & \cos R & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \cos R & \sin R & 0 & 0 \\ -\sin R & \cos R & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

T<sub>n</sub> conversion row

$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ x & y & z & 1 \end{pmatrix}$
--------------------------------------------------------------------------------------------------

x, y, z : Rotation center coordinates  
(parallel movement amount)

I, J, K : Rotation axis selection

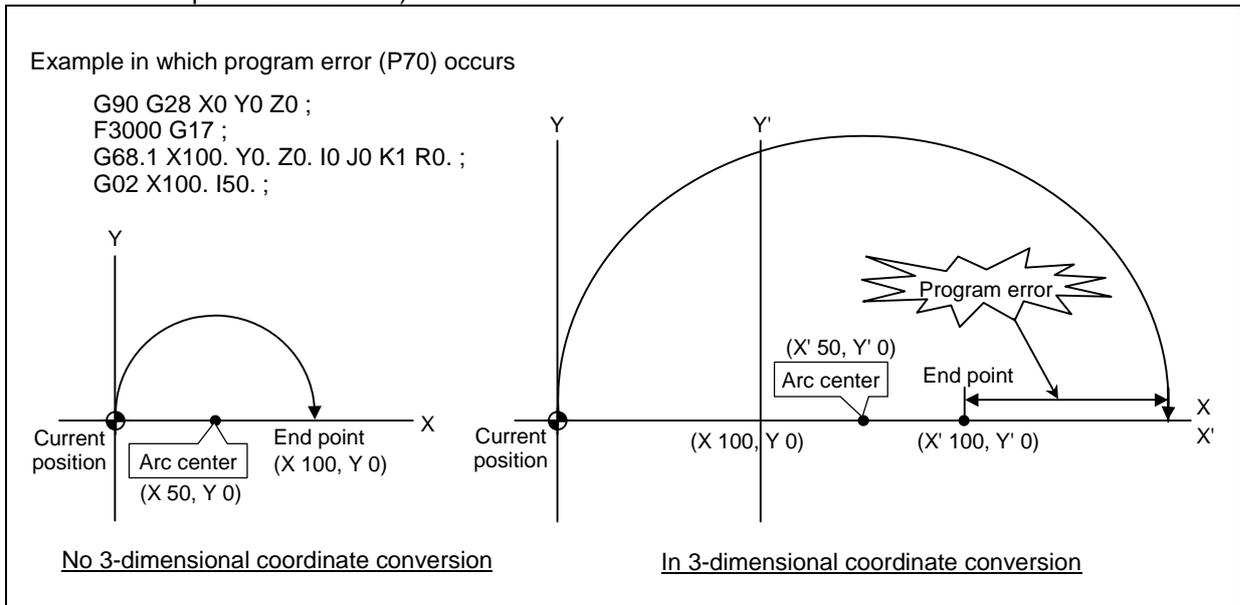
R : Rotation angle



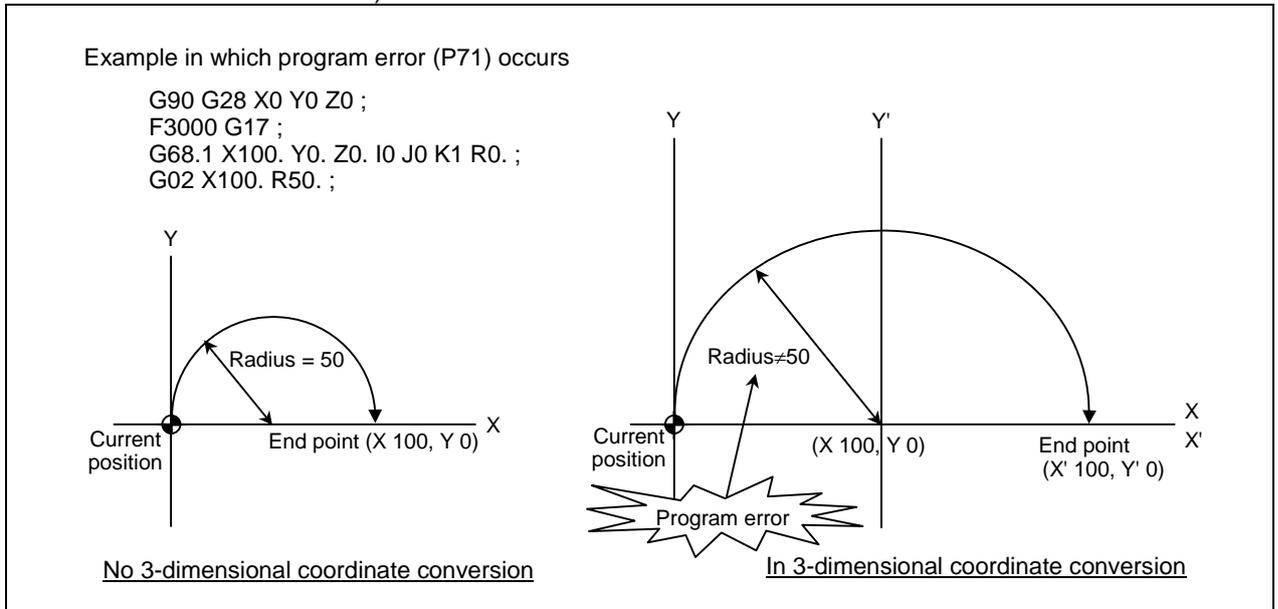
**Precautions related to arc command**

If the first command after the 3-dimensional coordinate conversion command was an arc shape, and the center of the arc did not change before and after the 3-dimensional coordinate conversion, an arc is drawn. However, an error will occur in the following cases:

- (1) For the arc in which the arc center is specified with I and J, if the center coordinate has been deviated followed by the 3-dimensional coordinate conversion, a program error (P70 Major arc end position deviation) will occur.



- (2) For the arc in which the arc radius is specified with R, If the center coordinate has been deviated followed by the 3-dimensional coordinate conversion, a program error (P71 Arc center calculation disabled) will occur.





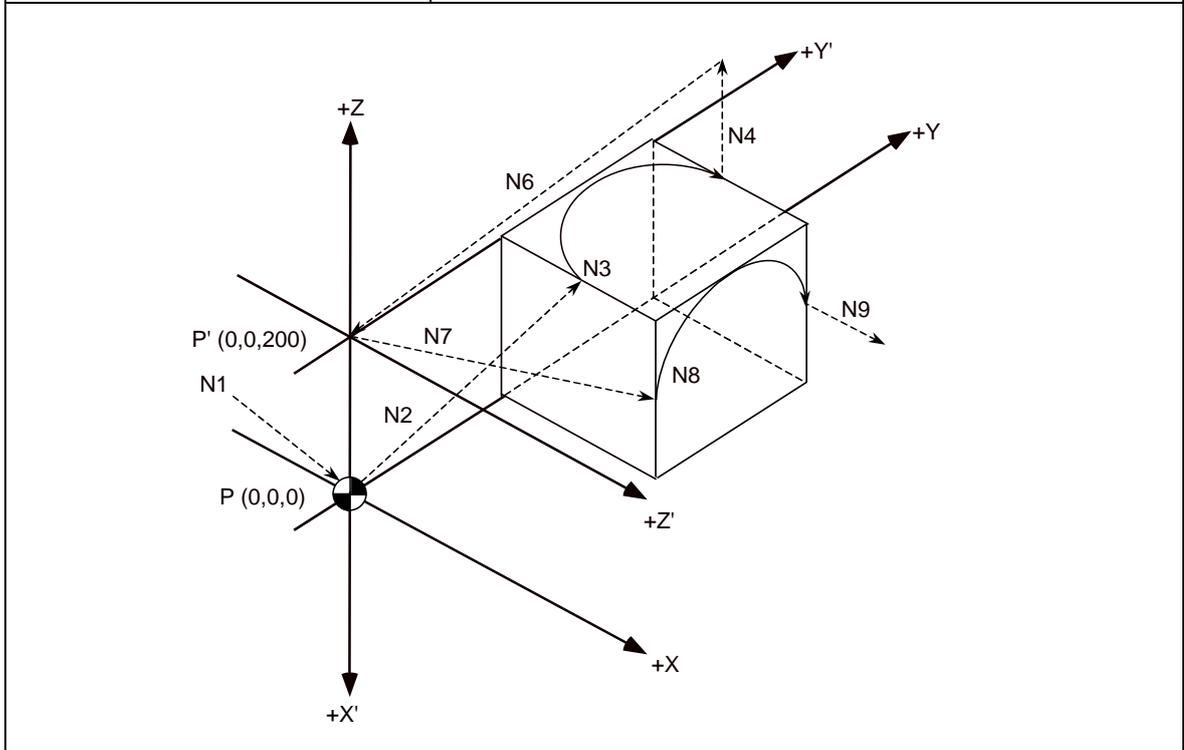
Example of program 2

This is a sample program only to explain about the operations. (To actually proceed with the machining by using this program, the dedicated tools and the tool change functions are required.)

(1) Example of machining program using arc cutting

In the following program example, the arc cutting (N3 block) carried out on the top of the workpiece is also carried out on the side of the workpiece. By using 3-dimensional coordinate conversion, the side can be cut with the same process (N8 block).

N01 G17 G90 G00 X0 Y0 Z0;	Position to the workpiece zero point P.
N02 G00 X100. Y200. Z200.;	Move to (100, 200, 200) with rapid traverse.
N03 G02 X100. Y400. J100. F1000;	Carry out arc cutting on workpiece top.
N04 G00 Z300.;	Escape in +Z direction at +100 rapid traverse.
N05 G68.1 X0 Y0 Z200. I0 J1 K0 R90.;	After shifting program coordinate system to (0, 0, 200), rotate coordinate axis +90° in Y axis direction. Set the program coordinate system (X' Y' Z') which has been rotated +90° in the Y axis direction around the (0,0,200).
N06 G17 G90 G00 X0 Y0 Z0;	Position to the new program zero point P'.
N07 G00 X100. Y200. Z200.;	Move to G68.1 program coordinate system (100, 200, 200) and workpiece coordinate system (200, 200, 100) with rapid traverse.
N08 G02 X100. Y400. J100. F1000;	Carry out arc cutting on workpiece side.
N09 G00 Z300.;	Move +100 in G68.1 program coordinate system + Z' direction with rapid traverse.
N10 G69.1;	
N11 M02;	



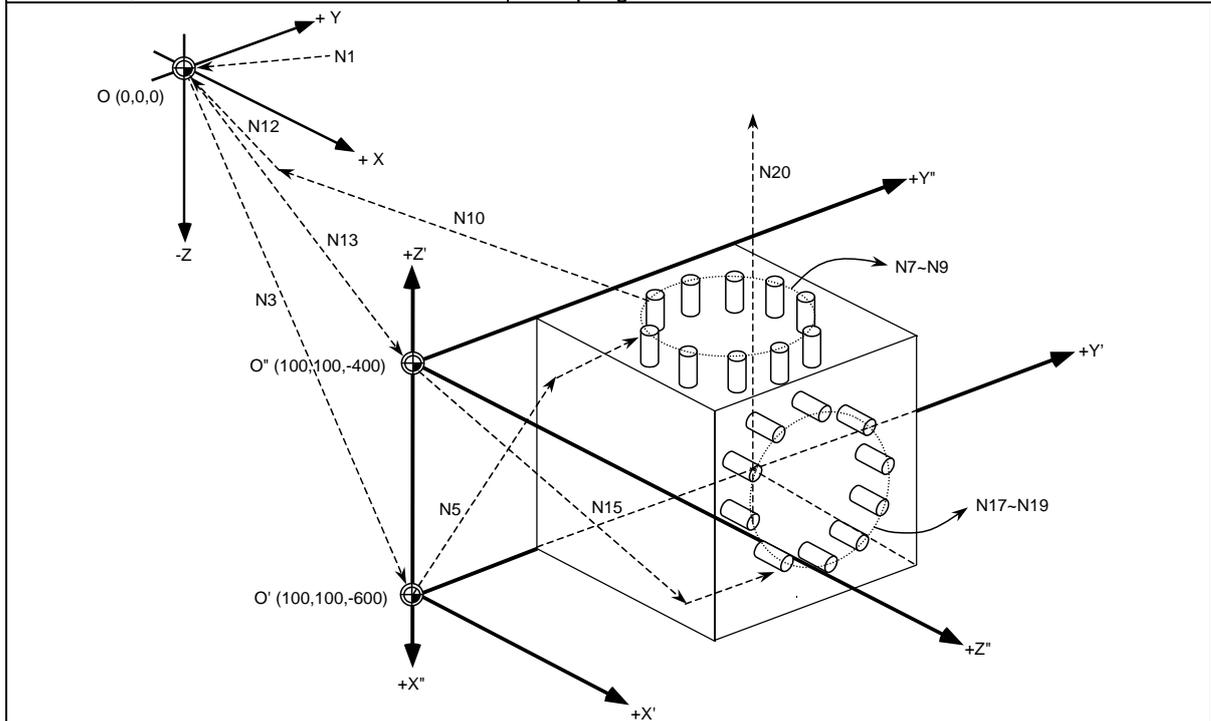
### 13. Program Support Functions

#### 13.23 3-dimensional Coordinate Conversion; G68.1/G69.1

(2) Example of machining program using fixed cycle

In the following program, the bolt hole cycle (N08 block) executed on the top of the workpiece is also carried out on the side of the workpiece. By using 3-dimensional coordinate conversion, the side can be cut with the same process (N18 block).

N01 G90 G00 X0 Y0 Z0;	Position to the workpiece coordinate system's 1st workpiece zero point.
N02 F2000;	
N03 G00 X100. Y100. Z-600.;	Move to (100, 100, -600) with rapid traverse.
N04 G52 X100. Y100. Z-600.;	Set the local coordinate system to the (100, 100, -600) position.
N05 G00 X100. Y10. Z 200.;	Move to the local coordinate system (X',Y',Z')s (100, 10, 200) position with rapid traverse.
N06 G91;	Incremental value command
N07 G81 Z-10. R5. L0 F2000;	Drilling
N08 G34 X100. Y200. I90. J270. K10.;	Bolt hole cycle
N09 G80;	Drilling cancel
N10 G91 G00 X-200.;	Move -200 from machining end point to X axis direction with rapid traverse.
N11 G90 G52 X0 Y0 Z0;	Cancel local coordinate system.
N12 G90 G00 X0 Y0 Z0;	Position to workpiece zero point.
N13 G00 X100. Y100. Z-400.;	Move to (100, 100, -400) with rapid traverse.
N14 G68.1 X100. Y100. Z-400. I0 J1 K0 R90.;	Set G68.1 program coordinate system (X'', Y'', Z'') to position rotated +90° in Y axis direction using (100, 100, -400) position as center.
N15 G00 X100. Y10. Z200.;	Move to (100, 10, 200) position in G68.1 program coordinate system with rapid traverse.
N16 G91;	Incremental value command
N17 G81 Z-10. R5. L0 F200;	Drilling
N18 G34 X100. Y200. I90. J270. K10.;	Bolt hole cycle
N19 G80;	Drilling cancel
N20 G91 G00 X-200.;	Move -200 from machining end point to X axis direction with rapid traverse.
N21 G69.1;	Cancel 3-dimensional coordinate conversion modal.
N22 M02;	End program.



## 13. Program Support Functions

### 13.23 3-dimensional Coordinate Conversion; G68.1/G69.1



#### Relation with other functions (Relation with other G codes)

Pxxx in the list indicates the program error Nos.

Format	Function	When this command is designated during 3-dimensional coordinate conversions	When 3-dimensional coordinate conversion is designated in this modal status	When 3-dimensional coordinate conversion is designated in the same block
G00	Positioning	○	○	P923
G01	Linear interpolation	○	○	P923
G02	Circular interpolation CW	○	○	P923
	Helical interpolation CW	P921	P922	P923
G03	Circular interpolation CW	○	○	P923
	Helical interpolation CCW	P921	P922	P923
G02.3	Exponential interpolation CW	P921	P922	P923
G02.4	3-dimensional circular interpolation CW	P921	P922	P923
G03.3	Exponential interpolation CCW	P921	P922	P923
G03.4	3-dimensional circular interpolation CCW	P921	P922	P923
G04	Dwell	○	-	G04 valid, G68.1 ignored
G05 P0	High-speed machining mode cancel	○	-	P923
G05 P1,2	High-speed machining mode I, II	P34 illegal G code	P34 illegal G code	P923
G05 P10000	High-speed high-accuracy control II	P34 illegal G code	P34 illegal G code	P923
G05.1 Q0	High-speed machining mode/High-speed high-accuracy control cancel	○	○	P923
G05.1 Q1	High-speed high-accuracy control I	○	○	P923
G05.1 Q2	Fine spline	P34 illegal G code	P34 illegal G code	P923
G07.1/G107	Cylindrical interpolation	P921	P481 illegal G code (mill)	P923
G09	Exact stop check	○	-	P923
G10	Parameter input by program	○	P421 parameter input error	P923
	Tool compensation data input by program	○	-	G10 valid, G68.1 ignored
G11.1	Parameter input by program cancel	○	-	P923
G12	Circle cutting CW	○	-	P923
G12.1	Polar coordinate interpolation	P921	P481 illegal G code (mill)	P923
G13	Circle cutting CCW	○	-	P923
G13.1	Polar coordinate interpolation cancel	○	-	P923
G15	Polar coordinate command cancel	○	-	P923

## 13. Program Support Functions

### 13.23 3-dimensional Coordinate Conversion; G68.1/G69.1

Format	Function	When this command is designated during 3-dimensional coordinate conversions	When 3-dimensional coordinate conversion is designated in this modal status	When 3-dimensional coordinate conversion is designated in the same block
G16	Polar coordinate command	○	○	P923
G17	Plane selection X-Y	○	○	○
G18	Plane selection Z-X	○	○	○
G19	Plane selection Y-Z	○	○	○
G20	Inch command	○	○	○
G21	Metric command	○	○	○
G27	Reference position check	○	-	G27 valid, G68.1 ignored
G28	Reference position return	○	-	G28 valid, G68.1 ignored
G29	Start point return	○	-	G29 valid, G68.1 ignored
G30	2nd to 4th reference position return	○	-	G30 valid, G68.1 ignored
G30.1	Tool position return 1	○	-	G30.1 valid, G68.1 ignored
G30.2	Tool position return 2	○	-	G30.2 valid, G68.1 ignored
G30.3	Tool position return 3	○	-	G30.3 valid, G68.1 ignored
G30.4	Tool position return 4	○	-	G30.4 valid, G68.1 ignored
G30.5	Tool position return 5	○	-	G30.5 valid, G68.1 ignored
G30.6	Tool position return 6	○	-	G30.6 valid, G68.1 ignored
G31	Skip	○	-	P923
G31.1	Multi-step skip 1	○	-	P923
G31.2	Multi-step skip 2	○	-	P923
G31.3	Multi-step skip 3	○	-	P923
G33	Thread cutting	P921	P922	P923
G34	Special fixed cycle (bolt hole circle)	○	-	P923
G35	Special fixed cycle (line at angle)	○	-	P923
G36	Special fixed cycle (arc)	○	-	P923
G37	Special fixed cycle (grid)	○	-	P923
G37.1	Automatic tool length measurement	P921	-	G37.1 valid, G68.1 ignored
G38	Tool radius compensation (vector designation)	○	-	P923
G39	Tool radius compensation (corner arc)	○	-	P923
G40	Tool radius compensation (cancel)	○	-	○
G41	Tool radius compensation	○	P922	P923
	3-dimensional tool radius compensation	○	P922	P923

## 13. Program Support Functions

### 13.23 3-dimensional Coordinate Conversion; G68.1/G69.1

Format	Function	When this command is designated during 3-dimensional coordinate conversions	When 3-dimensional coordinate conversion is designated in this modal status	When 3-dimensional coordinate conversion is designated in the same block
G42	Tool radius compensation	○	P922	P923
	3-dimensional tool radius compensation	○	P922	P923
G40.1/G150	Normal line control cancel	P921	-	P923
G41.1/G151	Normal line control (left)	P921	P922	P923
G42.1/G152	Normal line control (right)	P921	P922	P923
G43	Tool length compensation	○	○	P923
G44	Tool length compensation cancel	○	-	P923
G45	Tool position compensation increase	○	-	P923
G46	Tool position compensation decrease	○	-	P923
G47	Tool position compensation 2-fold increase	○	-	P923
G48	Tool position compensation 2-fold decrease	○	-	P923
G43.1	Tool length compensation in tool axis direction	P927	P931	P923
G43.4/G43.5	Tool center point control type 1/2 ON	P941	P922	P923
G50	Scaling cancel	P921	-	P923
G51	Scaling ON	P921	○	P923
G50.1	Mirror image cancel	○	-	P923
G51.1	Mirror image ON	○	○	P923
G52	Local coordinate system setting	P921	-	G52 valid, G68.1 ignored
G53	Machine coordinate system setting	○	-	G53 valid, G68.1 ignored
G54	Workpiece coordinate system 1 selection	P921	○	P923
G55	Workpiece coordinate system 2 selection	P921	○	P923
G56	Workpiece coordinate system 3 selection	P921	○	P923
G57	Workpiece coordinate system 4 selection	P921	○	P923
G58	Workpiece coordinate system 5 selection	P921	○	P923
G59	Workpiece coordinate system 6 selection	P921	○	P923

## 13. Program Support Functions

### 13.23 3-dimensional Coordinate Conversion; G68.1/G69.1

Format	Function	When this command is designated during 3-dimensional coordinate conversions	When 3-dimensional coordinate conversion is designated in this modal status	When 3-dimensional coordinate conversion is designated in the same block
G54.1	Extended workpiece coordinate system selection	P921	○	P923
G60	Unidirectional positioning	P921	-	G60 valid, G68.1 ignored
	Unidirectional positioning (Modal designation)	P921	P922	P923
G61	Exact stop check mode	○	○	P923
G61.1	High-accuracy control	○	○	P923
G63	Tapping mode	P921	P922	P923
G64	Cutting mode	○	○	○
G65	User macro Simple call	○	-	Update modal only (Coordinate rotation by program)
G66	User macro Modal call A	○	○	Update modal only (Coordinate rotation by program)
G66.1	User macro Modal call B	○	Update modal only (Coordinate rotation by program)	Update modal only (Coordinate rotation by program)
G67	User macro Modal call cancel	○	○	Update modal only after macro (Coordinate rotation by program)
G68.1	Coordinate rotation by program ON	P921	P922	-
	3-dimensional coordinate conversion ON	○	○	-
G69.1	Coordinate rotation by program cancel	○ (3-dimensional coordinate conversion cancel)	-	-
	3-dimensional coordinate conversion cancel	○	-	-
G73	Fixed cycle (Step)	○	P922	P923
G74	Fixed cycle (Counter tap) *incl: Synchronous tapping	○	P922	P923
G76	Fixed cycle (Fine balling)	○	P922	P923
G80	Fixed cycle cancel	○	-	P923
G81	Fixed cycle (Drill/spot drill)	○	P922	P923
G82	Fixed cycle (Drill/counter-balling)	○	P922	P923
G83	Fixed cycle (Deep hole drill)	○	P922	P923
G84	Fixed cycle (Tap) *incl: Synchronous tapping	○	P922	P923

### 13. Program Support Functions

#### 13.23 3-dimensional Coordinate Conversion; G68.1/G69.1

Format	Function	When this command is designated during 3-dimensional coordinate conversions	When 3-dimensional coordinate conversion is designated in this modal status	When 3-dimensional coordinate conversion is designated in the same block
G85	Fixed cycle (Balling)	○	P922	P923
G86	Fixed cycle (Balling)	○	P922	P923
G87	Fixed cycle (Back balling)	○	P922	P923
G88	Fixed cycle (Balling)	○	P922	P923
G89	Fixed cycle (Balling)	○	P922	P923
G90	Absolute value command	○	○	○
G91	Incremental value command	○	○	○
G92	Coordinate system setting	P921	-	P923
G94	Asynchronous feed (Feed per minute)	○	○	○
G95	Synchronous feed (Feed per revolution)	○	○	○
G96	Constant surface speed control ON	P921	P922	P923
G97	Constant surface speed control OFF	P921	-	P923
G98	Fixed cycle (initial level return)	○	○	○
G99	Fixed cycle (R point level return)	○	○	○

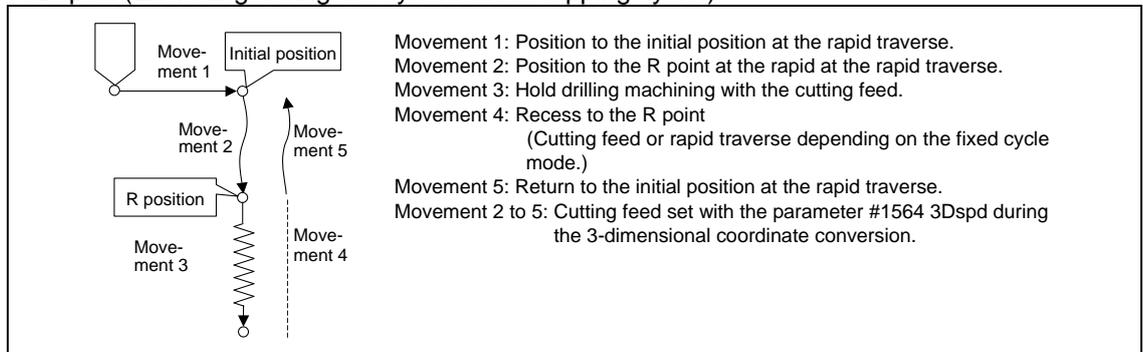
(Note) All the G codes not listed above are disabled.

**Relation with other functions**

- (1) Circular interpolation  
Circular interpolation in the 3-dimensional coordinate conversion modal functions according to the coordinate value resulted by the 3-dimensional coordinate conversion. With G17, G18 and G19 commands, circular interpolation functions normally for all the planes in which 3-dimensional coordinate conversion has been executed.
- (2) Fine spline  
Specify a spline axis for the movement axis after the 3-dimensional coordinate conversion. When a movement occurs to the axis in which spline cannot be specified, spline will be in the pause status.
- (3) Reference position check  
The 3-dimensional coordinate conversion is applied for the position for reference positioning specified with G27 command in the 3-dimensional coordinate conversion modal.
- (4) Reference position return  
The 3-dimensional coordinate conversion is applied for the mid-point specified with G28, G30 command in the 3-dimensional coordinate conversion modal. However, reference position will be returned without the 3-dimensional coordinate conversion carried out.
- (5) Tool position return  
3-dimensional coordinate conversion is not carried out for the tool change position even if a command from G30.1 to G30.6 is issued in the 3-dimensional coordinate conversion modal. The return order and position will be the same as machine coordinate system.
- (6) Tool compensation  
When executing the tool length/radius/position compensation in the 3-dimensional coordinate conversion modal, the 3-dimensional coordinate conversion is carried out after the compensation amount has been applied.
- (7) Machine coordinate system selection  
Coordinate conversion will not be carried out for the machine coordinate system even if G53 command is issued in the 3-dimensional coordinate conversion modal.
- (8) Mirror image  
When issuing the mirror image command in the 3-dimensional coordinate conversion modal, as well as when executing the 3-dimensional coordinate conversion in the mirror image modal, 3-dimensional coordinate conversion will be executed for the coordinate value, which is calculated by the mirror image.
- (9) User macro  
When a user macro call command is issued in the 3-dimensional coordinate conversion modal, the 3-dimensional coordinate conversion will be valid after the macro execution.

## (10) Fixed cycle for drilling

The fixed cycle in the 3-dimensional coordinate conversion can be executed in an oblique direction for the orthogonal coordinate system. In the same manner, synchronous tapping cycle can also be executed. However, the mode for the fixed cycle for hole drilling will be changed from the rapid traverse to the cutting feed at the speed set with the parameter "#1564 3Dspd". (Excluding during the synchronous tapping cycle.)



## (11) Synchronous tapping cycle

The Synchronous tapping cycle in the 3-dimensional coordinate conversion modal will not function even if #1223 BIT3 (synchronous tapping in-position check expansion valid) is valid. Set the synchronous tapping cycle to invalid.

## (12) Geometric command

Geometric command can be issued in the 3-dimensional coordinate conversion modal. However, if the geometric command is issued in the same block as in the 3-dimensional coordinate conversion command (G68.1, G69.1), "P32 Illegal address" will occur.

## (13) Initial constant surface speed

When the 3-dimensional coordinate conversion command is issued while the parameter initial constant surface speed is valid, "P922 3D conversion illegal mode" will occur. This is the same consequence as in the case where the 3-dimensional coordinate conversion command is issued in the constant surface speed (G96) modal.

## (14) Machine lock

The machine lock in the 3-dimensional coordinate conversion modal will be valid for the movement axis for the coordinate value after executing the 3-dimensional coordinate conversion.

## (15) Interlock

The interlock in the 3-dimensional coordinate conversion modal will be valid for the movement axis for the coordinate value after executing the 3-dimensional coordinate conversion.

(16) Coordinate read variable

When reading the workpiece coordinate system/skip coordinate system during the 3-dimensional coordinate system conversion modal, local coordinate system and G68.1 program coordinate system can be switched with the parameter "#1563 3Dcdc".

(17) Manual operation

Manual operation in the 3-dimensional coordinate conversion modal will not execute the 3-dimensional conversion. Manual operation will be executed in the machine coordinate system. Also, when the manual ABS is OFF, G68.1 program coordinate system will move as much as the manual interruption or manual feed amount.

(18) Workpiece coordinate display

Whether the workpiece coordinate position in the 3-dimensional coordinate conversion modal to be displayed in the workpiece coordinate system or in the G68.1 program coordinate system can be switched with the parameter "#1561 3Dcdc". In the same manner, absolute value can be displayed on the special display.

(Note) 1um of display deviation may occur during the 3-dimensional coordinate conversion; however, this is normal.

(19) Remaining command display

Whether the remaining commands in the 3-dimensional coordinate conversion modal to be displayed in the workpiece coordinate system or in the G68.1 program coordinate system can be switched with the parameter "#1562 3Dremc".

(Note) 1um of display deviation may occur during the 3-dimensional coordinate conversion; however, this is normal.

(20) Others

G41, G42, and the fixed cycle command G73 to G89 have to be nested inside the G68.1/G69.1 commands.

For the block next to G68.1, a movement command in the G90 (Absolute value command) mode has to be issued.

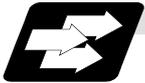
(Example)

```

G68.1 X50. Y100. Z150. I1 J0 K0 R60. ;
G90 G00 X0 Y0 Z0 ; ← Issuance of G90 mode movement commands
G41 D01 ;
      .....
G40 ;
G69.1 ;
    
```

- G00 command during 3-dimensional coordinate conversion modal is the interpolation type regardless of settings of the basic parameter "#1086 G0Intp (G00 non-interpolation)"
- Origin zero cannot be executed during the 3-dimensional coordinate conversion modal.

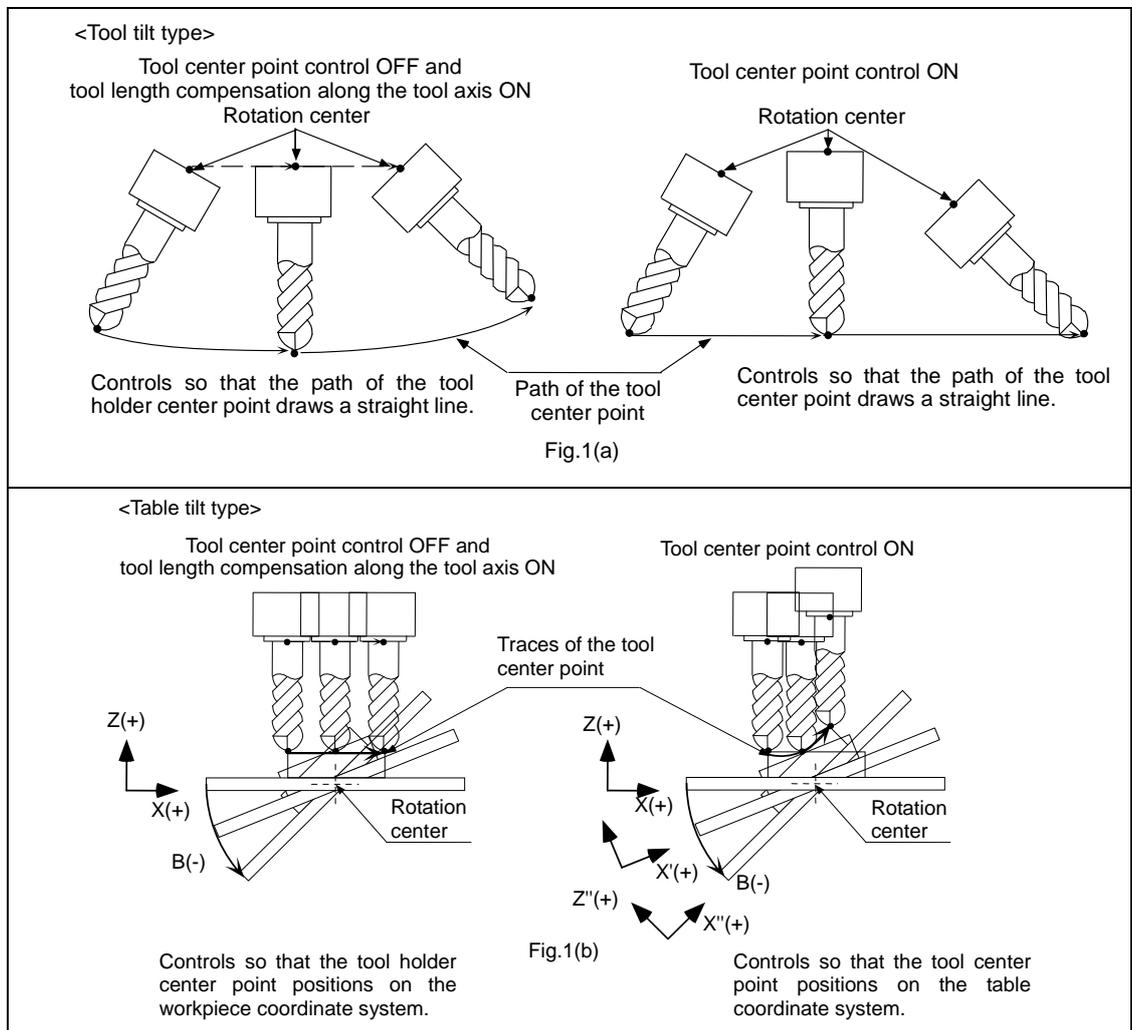
13.24 Tool Center Point Control; G43.4/G43.5

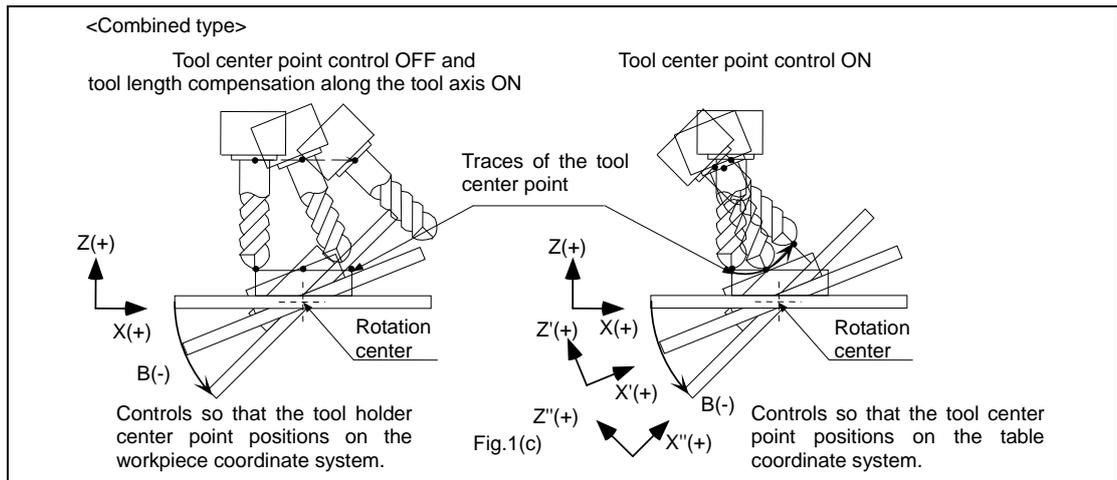


Function and purpose

The tool center point control function controls a commanded position described in the machining program to be the tool center point in the coordinate system that rotates together with a workpiece (table coordinate system). This function can be applied for 5-axis machining, including a tool tilt type (Fig. 1 (a)) with two rotary axes set on the head, table tilt type (Fig. 1 (b)) with two rotary axes set on the table, or combined type (Fig. 1 (c)) with a rotary axis set on each tool and table.

With this function, in the case of using tool tilt type, the tool center point is controlled so that it moves on the programmed path specified on the workpiece coordinate system. In the case of using the table tilt type, the tool center point is controlled so that it moves on the programmed path specified on the table coordinate system (a coordinate system which rotates together with a workpiece).





To use this function, its dedicated option is required. Without the option, a program error (P940) occurs upon executing the tool center point control command.



### Command format

There are two command formats: <Type1>, where tool angle is commanded by the rotary axis; and <Type2>, where tool angle is commanded by the vectors of the workpiece surface, I, J, and K.

#### (1) Tool Center Point Control ON

**G43.4 (X\_\_Y\_\_Z\_\_A\_\_C\_\_)H\_\_ ; Tool center point control type1 ON**

**G43.5 (X\_\_Y\_\_Z\_\_)I\_\_J\_\_K\_\_H\_\_ ; Tool center point control type2 ON**

G43.4 : Tool center point control type1 command  
 G43.5 : Tool center point control type2 command  
 X,Y,Z : Orthogonal coordinate axis movement command  
 A,C : Rotary axis movement command  
 I,J,K : Workpiece surface angle vector  
 H : Tool length offset number

- (Note 1)** When orthogonal coordinate axis movement command or rotary axis movement command is not issued in the same block, start-up without movement command is applied. (No movement for the offset amount.)
- (Note 2)** Commands to I, J, and K will be ignored during the tool center point control type1.
- (Note 3)** Rotary axis movement command cannot be executed during the tool center point control type2. If the command is issued, a program error (P33) occurs.
- (Note 4)** If I, J, or K is omitted when issuing the tool center point control type2 command, the omitted address will be considered as "0".

#### (2) Tool Center Point Control cancel

**G44 (X\_\_Y\_\_Z\_\_A\_\_C\_\_) ; Tool Center Point Control cancel**

- (Note 1)** Instead of using G44, the following G codes in the G group 8 can be used for canceling.  
 G43 (tool length compensation in the forward direction) /  
 G43.1 (tool length compensation along the tool axis)
- (Note 2)** If orthogonal coordinate axis command and rotary axis command are issued in the same block as G44, the tool center point control modal will be canceled on the spot. Then, commanded axis movement will be performed. If G44 is issued alone, the tool center point control modal will be cancelled on the spot, and yet no axis movement (movement for the compensation amount) will be performed.



### Programming coordinate system

The end position of each block looking from the programming coordinate system is specified in the tool center point control mode. In the program, specify the position of the tool center point. The programming coordinate system is a coordinate system used for the tool center point control, and is specified either the table coordinate system (a coordinate which rotates together with a workpiece) or the workpiece coordinate system by the parameter.

(1) Table coordinate system

When 0 is specified for the programming coordinate system selection parameter, the table coordinate system, which is the valid workpiece coordinate system at that time fixed to the table, is specified as the programming coordinate system. Table coordinate system rotates along the table rotation and not the tool axis rotation. The subsequent X,Y,Z addresses are considered to have been issued on the table coordinate system.

When a rotary axis movement is commanded in a block prior to G43.4/G43.5 command, the angle generated by rotary axis movement is regarded as an initial setting at G43.4/G43.5 command.

(2) Workpiece coordinate system

When 1 is specified for the programming coordinate system selection parameter, the programming coordinate system will be the valid workpiece coordinate system at that time. The coordinate system in this case does not rotate along the table rotation. A linear movement is carried out for the table (workpiece) when the subsequent X,Y,Z addresses are issued. The end position looking from the workpiece coordinate system after table rotation is specified to the X, Y and Z.

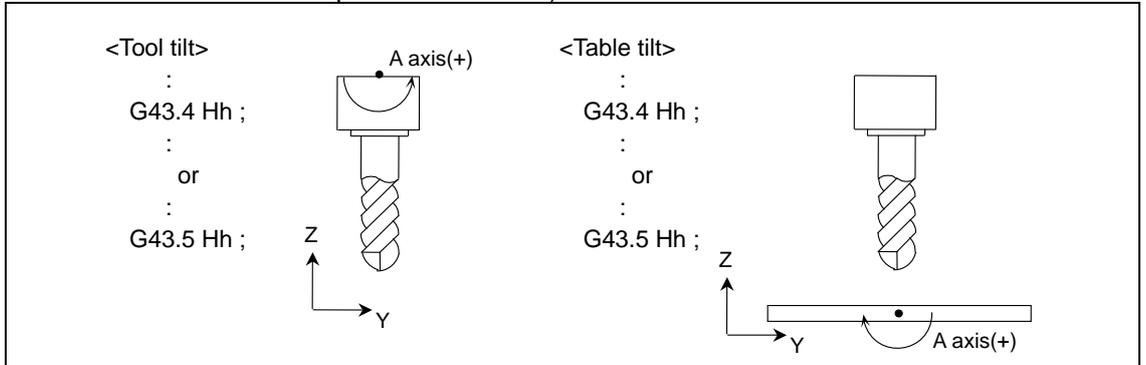


Start-up

(1) Start-up without movement command

(a) Tool center point control type1, type2

When the tool center point control is ON, no axis movement is performed (including movement for the compensation amount).



(b) Tool center point control type2

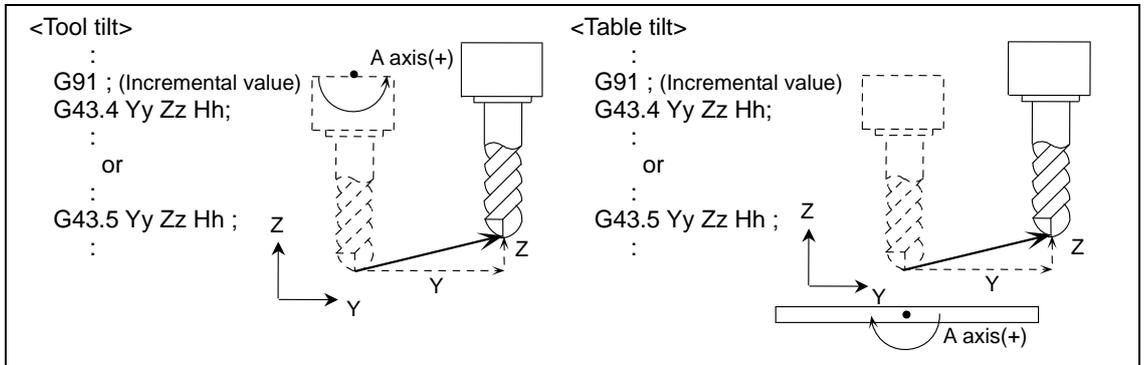
"G43.5 li Jj Kk Hh ;" performs the same movement as the tool center point control type1 in (2).

(2) Start-up with movement command

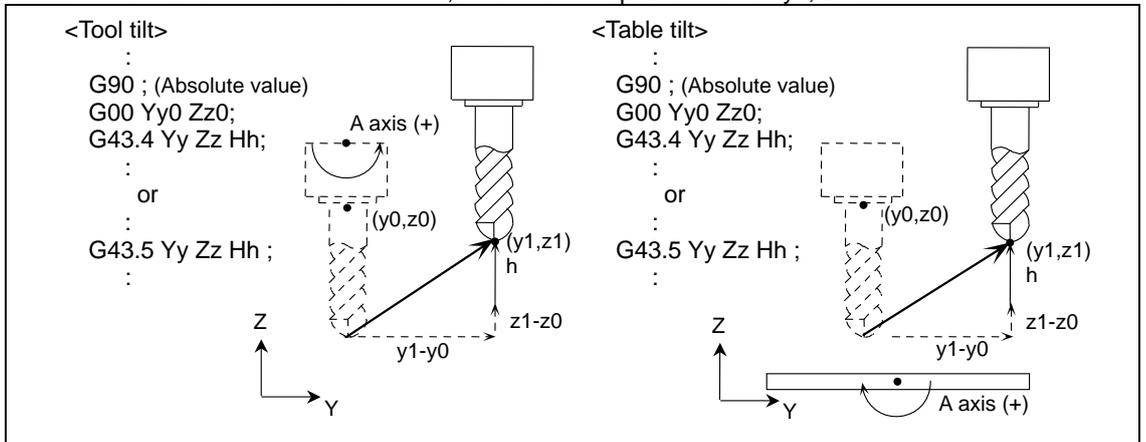
(When orthogonal coordinate axis command is issued in the same block)

(a) Tool center point control type1, type2

When the tool center point control is ON, the tool center point moves only as much as it is ordered under the incremental value command.

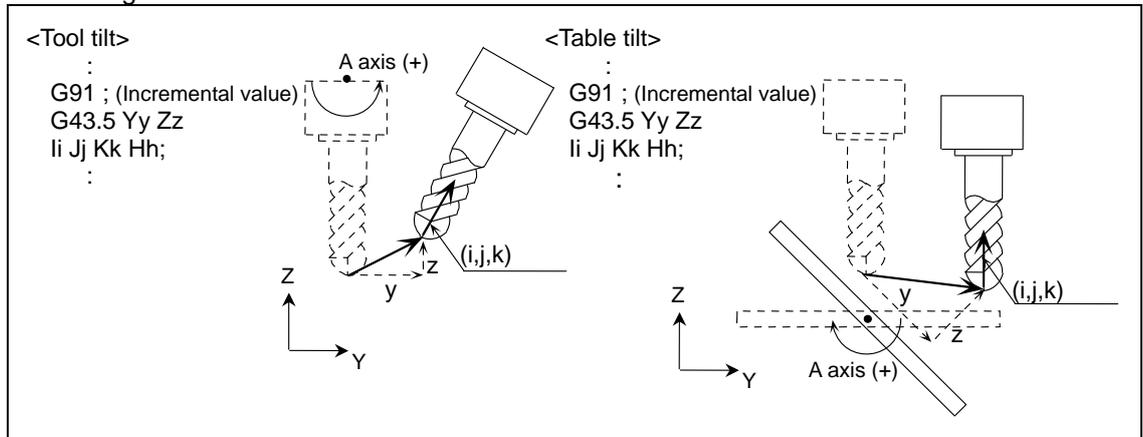


Under the absolute value command, the tool center point moves to y1, z1.



(b) Tool center point control type2

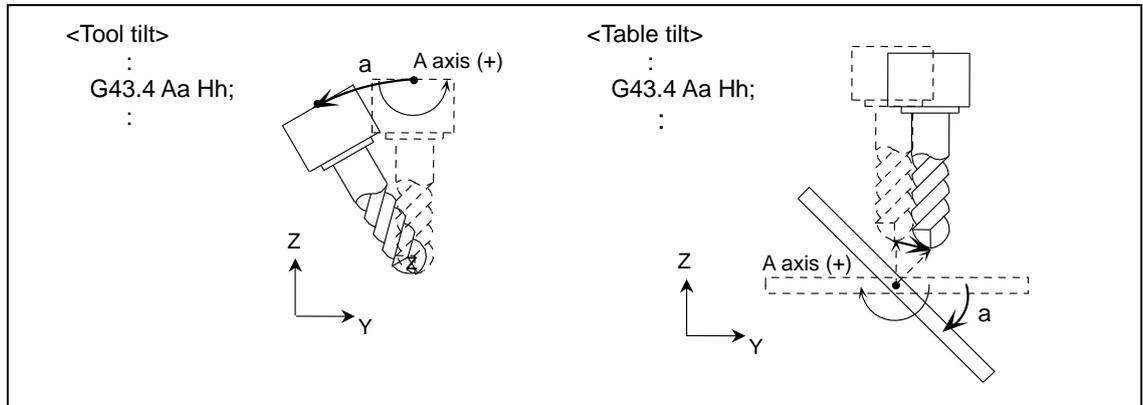
The rotary axis moves toward the commanded workpiece surface vector (I,J,K) direction along the movement command issued.



(3) Start-up with movement command (When rotary axis command is issued in the same block)

(a) Tool center point control type1

In the case of using the tool tilt type, the orthogonal axis moves according to the rotary axis angle while fixing the tool center point to the center. In the case of using the table tilt type, the orthogonal axis moves so that the tool center point locates on the rotated table workpiece coordinate system.



(b) Tool center point control type2

The program error (P33) will occur.

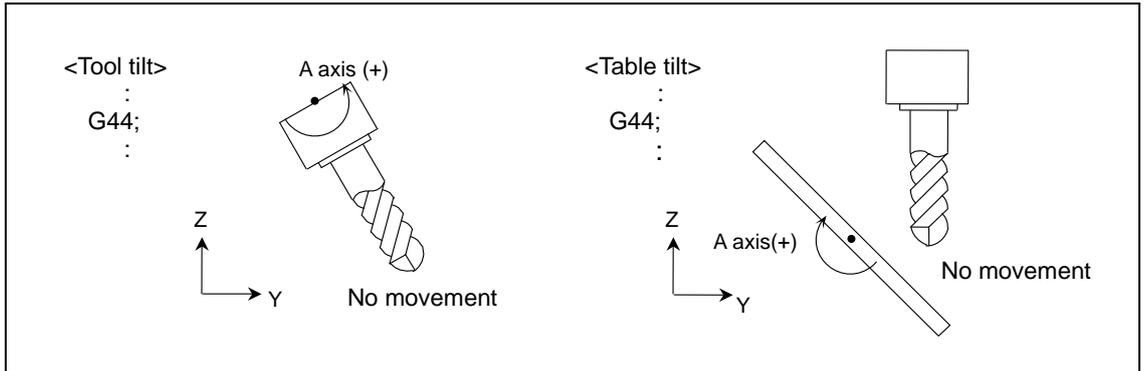


**Cancel**

(1) Cancellation without movement command

(a) Tool center point control type1, type2

Cancellation movement for the compensation amount is not performed regardless of absolute/incremental value command. On the other hand, the tool center point control modal will be cancelled.

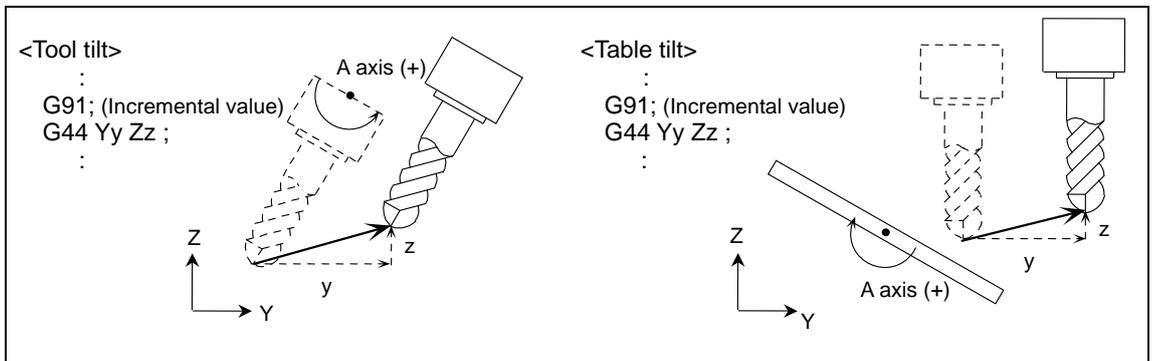


(2) Cancellation with movement command

(When orthogonal coordinate axis command is issued in the same block)

(a) Tool center point control type1, type2

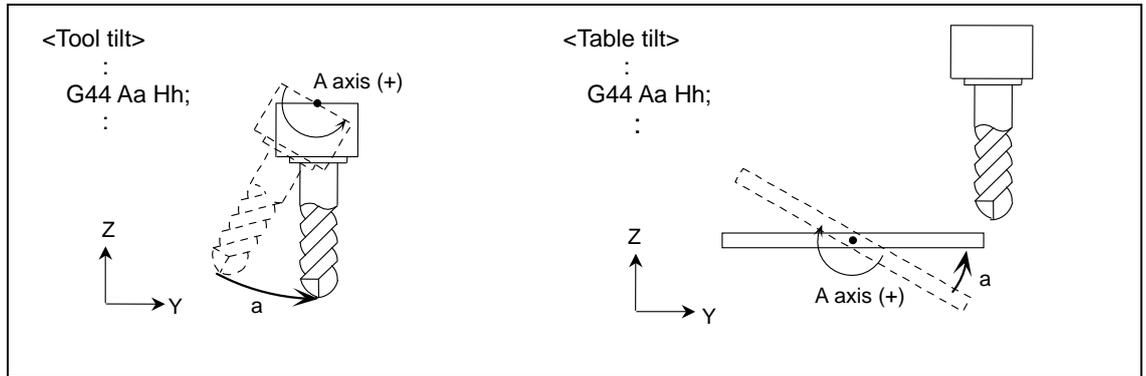
Cancellation movement for the compensation amount is not performed regardless of absolute/incremental value command. Orthogonal coordinate axis movement command is executed upon cancellation of the tool center point control modal.



- (3) Cancellation with movement command  
(When rotary axis command is issued in the same block)

- (a) Tool center point control type1, type2

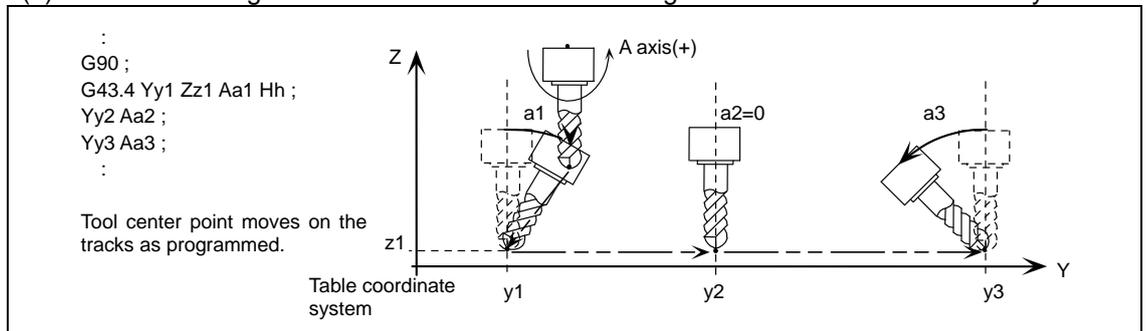
Cancellation movement for the compensation amount is not performed regardless of absolute/incremental value command. Rotary axis movement command is executed upon cancellation of the tool center point control modal



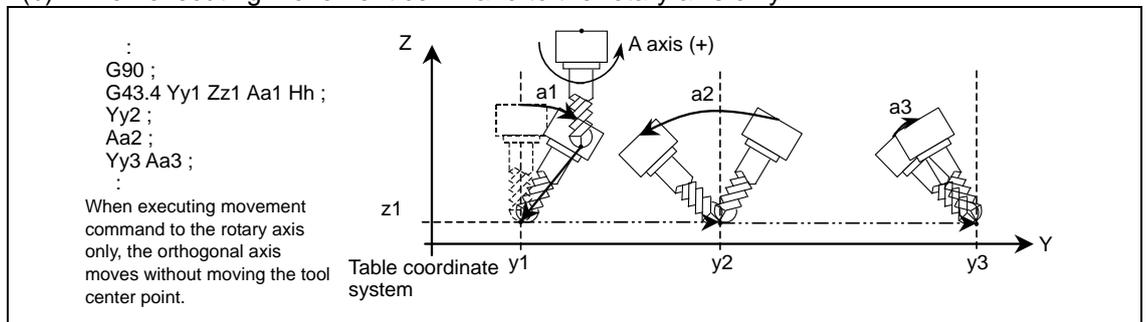
During tool center point control

- (1) Tool center point control type1

- (a) When executing movement command to the orthogonal coordinate axis and rotary axis.

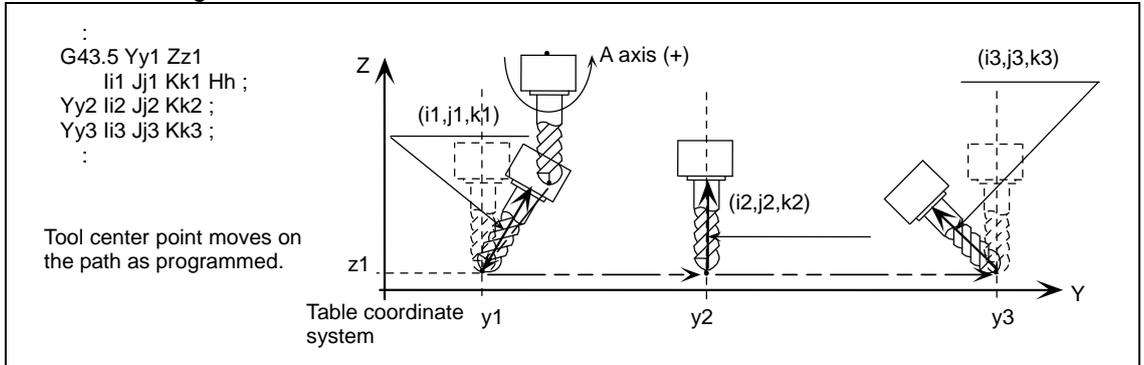


- (b) When executing movement command to the rotary axis only.

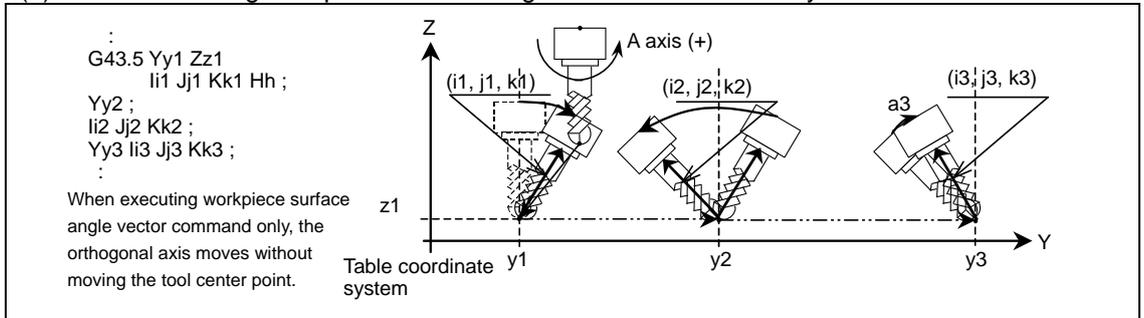


(2) Tool center point control type2

(a) When executing movement command to the orthogonal coordinate axis and workpiece surface angle vector command.



(b) When executing workpiece surface angle vector command only.



Feedrate during tool center point control

Feedrate during the tool center point control is controlled so that the tool center point moves according to the commanded speed.



Interpolation mode

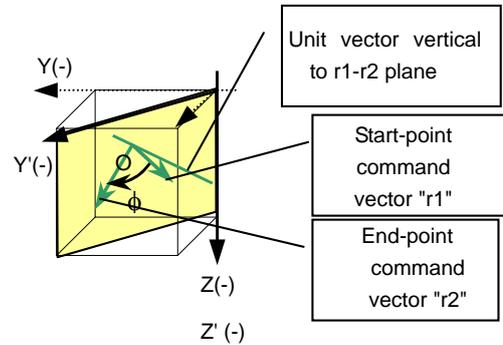
There are two modes of interpolation: single axis rotation interpolation and joint interpolation. You can select one of them by parameter.

(1) Single axis rotation interpolation

When transforming from a start-point angle vector "r1" into an end-point angle vector "r2", interpolate so that the angular rate of the rotary  $\phi$  around the vector k axis, which is vertical to r1-r2 plane, will be constant.

(a) Features

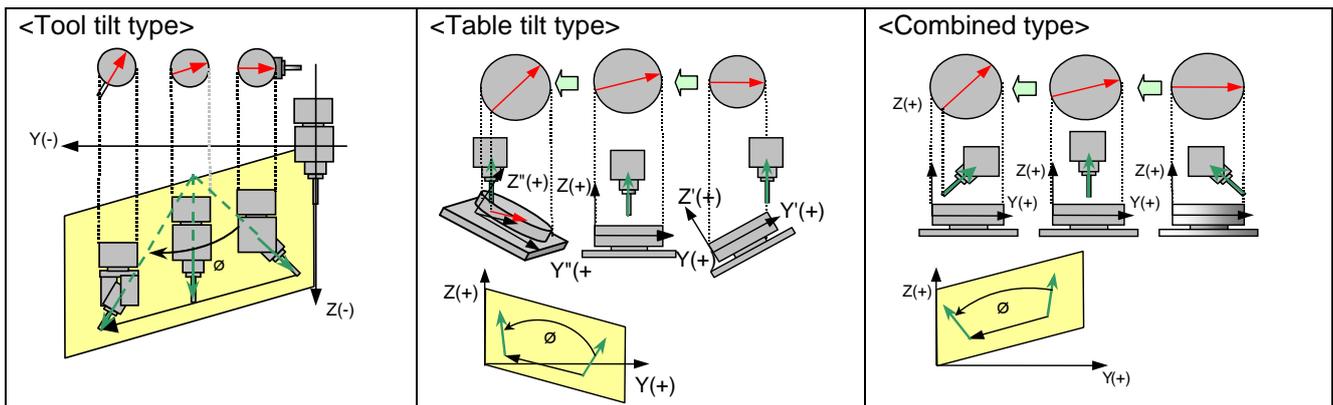
- Tool angle vector always exists on the plane consisting of O, r1 and r2.
- The angular rates of each rotary axis will not be constant.



(b) Operations

(Example) Current position Aa° C0°

When commanding G90 Yy A-a. C45. ; or G90 Yy li Jj Kk ;

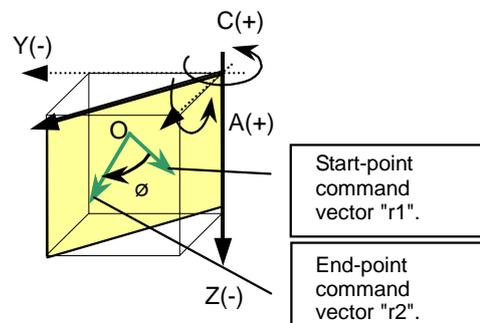


(2) Joint interpolation

A movement from a start-point angle vector "r1" to an end-point angle vector "r2" is interpolated to keep the angular rates of each axis constant.

(a) Features

- The angular rates of each rotary axis become constant.
- As this control aims to keep the angular rates of each rotary axis constant, a tool angle vector may not exist on the plane consisting of O, r1 and r2.

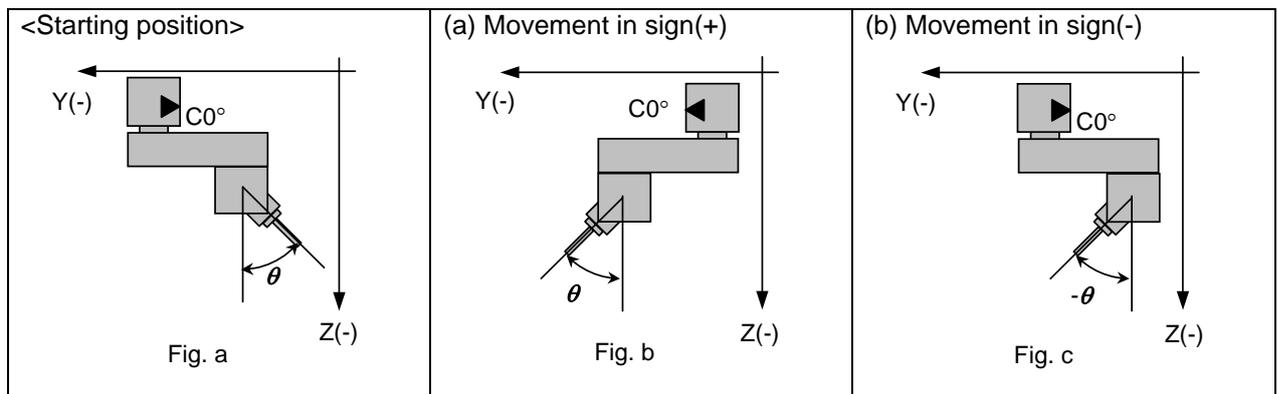




### Passing singular point

When passing the singular point (singular position\*), there are two kinds of movements to be followed from the singular point. When using an A-C axis tilt type machinery, there are two different movements (Fig. b, c) to be followed. In those movements, the rotation angles of the A axis are the same absolute value but different in signs (+/-). The rotation angles of the C axis corresponding the two movements are differed by 180 degrees one another. Determine which one of the two movements are to be selected with parameter.

The figures below are the example of movements seen during tool center point control type 2. When the tool center-point-side rotary axis moves in the sign (+) direction from the starting position (Fig. a), (Fig. b) is representing "passing singular point type1". When the tool center-point-side rotary axis moves in the sign (-) direction from the starting position (Fig. a), (Fig. c) is representing "passing singular point type2".



\*The position in which the tool center-point-side rotary axis or the table base-side rotary axis is 0.

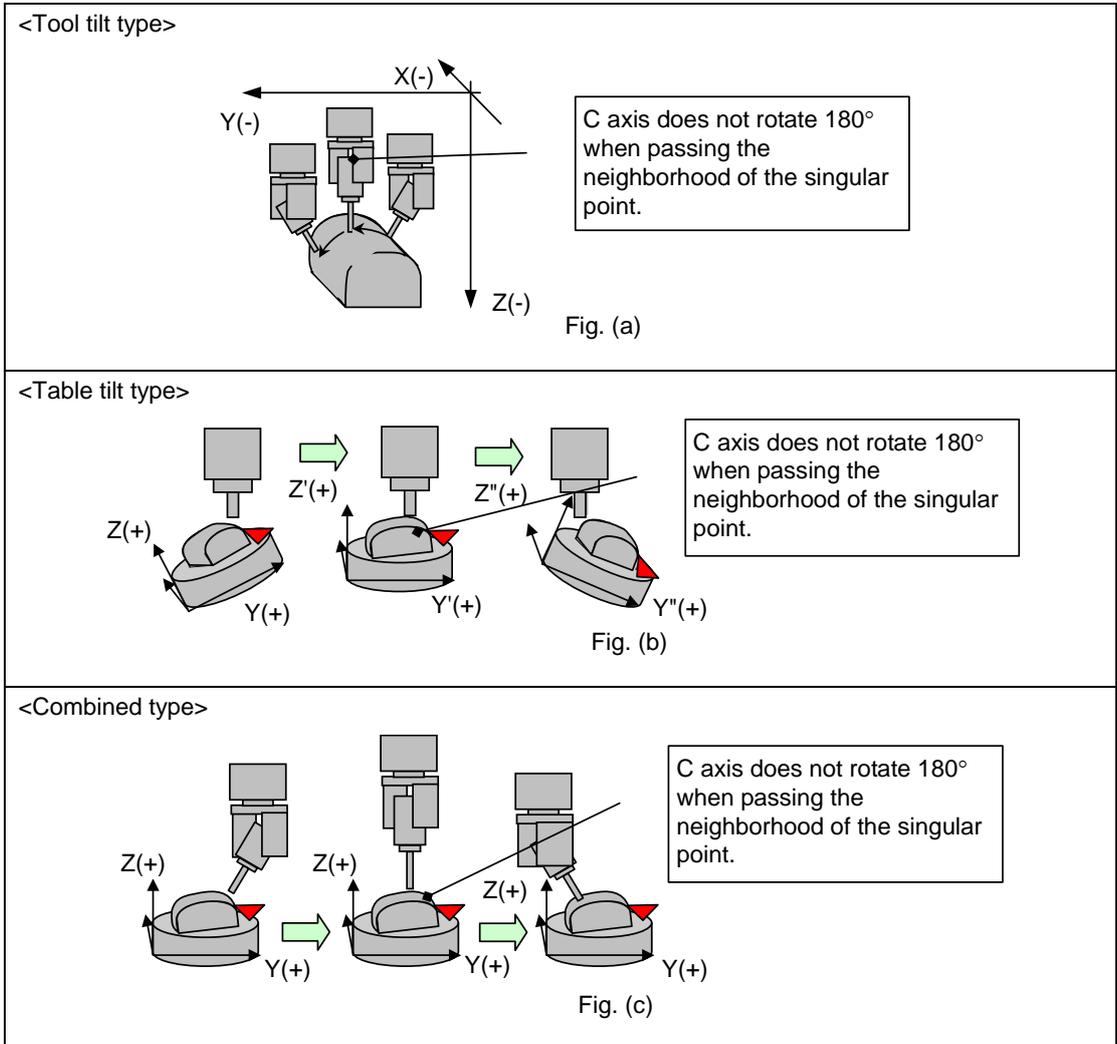
(1) Passing singular point type1

Select the same direction as the start point of the tool base-side rotary axis or table workpiece-side rotary axis in the block where a singular point passing is carried out. When the rotation angle of the start point is 0°, select the wider stroke limit. When the stroke limits are the same, select the one with a minus-coded rotation angle.

<p>&lt;Tool tilt type&gt;</p>	<p>Fig. (a)</p>	<p><b>Singular point</b></p> <p>When passing the neighborhood of the singular point, C axis rotates 180° within the parameter "#7907 CHK_ANG" (Judging angle for the singular point neighborhood.).</p>
<p>&lt;Table tilt type&gt;</p>	<p>Fig. (b)</p>	<p><b>Singular point</b></p> <p>When passing the neighborhood of the singular point, C axis rotates 180° within the parameter "#7907 CHK_ANG" (Judging angle for the singular point neighborhood.).</p>
<p>&lt;Combined type&gt;</p>	<p>Fig. (c)</p>	<p><b>Singular point</b></p> <p>When passing the neighborhood of the singular point, C axis rotates 180° within the parameter "#7907 CHK_ANG" (Judging angle for the singular point neighborhood.).</p>

(2) Passing singular point type 2

Select the one with the smaller rotary movement amount of the tool base-side rotary axis or the table/workpiece-side rotary axis on the singular point. When the tool base-side rotary axis and the table/workpiece have the same rotary movement amount, select the one with the tool base-side rotary axis or the table/workpiece-side rotary axis that are to be rotated in the minus-coded direction.



## (3) Movement in the singular point neighborhood in each interpolation mode

Interpolation mode	Command	Type of passing singular point	Command from a singular point to a non-singular point	Command to pass a singular point
Single axis rotation interpolation	G43.4 (Rotary axis command)	Type 1	Value as commanded. However, in the case where the signs at the start point and end point of either tool center-point-side rotary axis or table base-side rotary axis differ, if tool base-side rotary axis or table workpiece-side rotary axis rotates in the same block, the tool will not pass the singular point, resulting in a program error (P943).	
		Type 2		
	G43.5 (IJK command)	Type 1	Select the one with the wider stroke range. When the stroke range is the same, select a minus direction of the tool center-point-side rotary axis or the table base-side rotary axis.	Select the one with the same-coded end point as the start point of the tool center-point-side rotary axis or the table base-side rotary axis.
		Type 2	Select the one with the smaller movement amount of the tool base side rotary axis or the table workpiece side rotary axis.	
Joint interpolation	G43.4 (Rotary axis command)	Type 1	Value as commanded.	
		Type 2		
	G43.5 (IJK command)	Type 1	Select the one with the wider stroke range. When the stroke range is the same, select a minus direction of the tool center-point-side rotary axis or the table base-side rotary axis.	Select the one with the same-coded end point as the start point of the tool center-point-side rotary axis or the table base-side rotary axis.
		Type 2	Select the one with the smaller movement amount of the tool base-side rotary axis or the table workpiece-side rotary axis.	



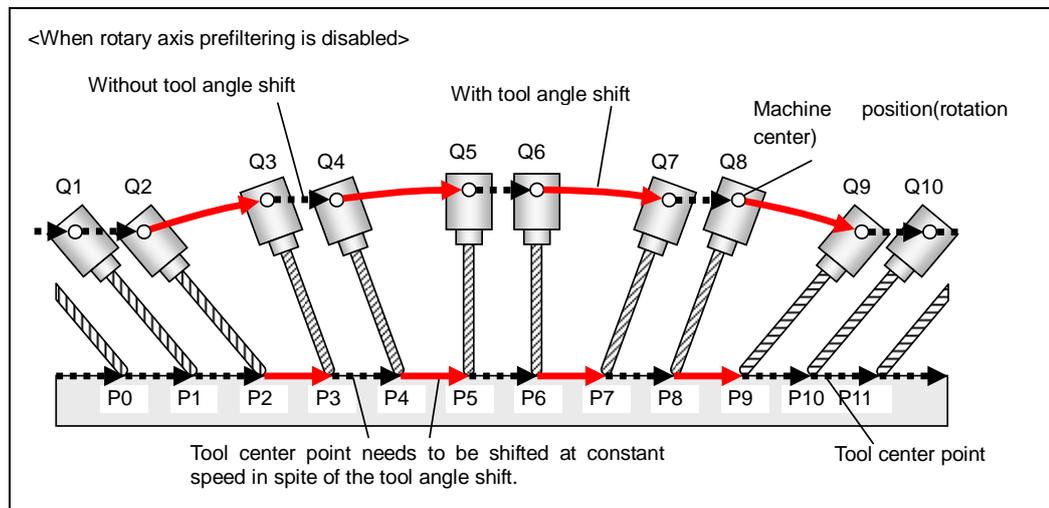
**Rotary Axis Prefiltering**

Rotary axis prefiltering means smoothing (prefiltering) the rotary axis command (tool angle shift) process, which moves the rotary axis smoothly and produces smoother cutting surface. Tool center point moves on the tracks as programmed by the rotary axis command while the command process is smoothed with this function.

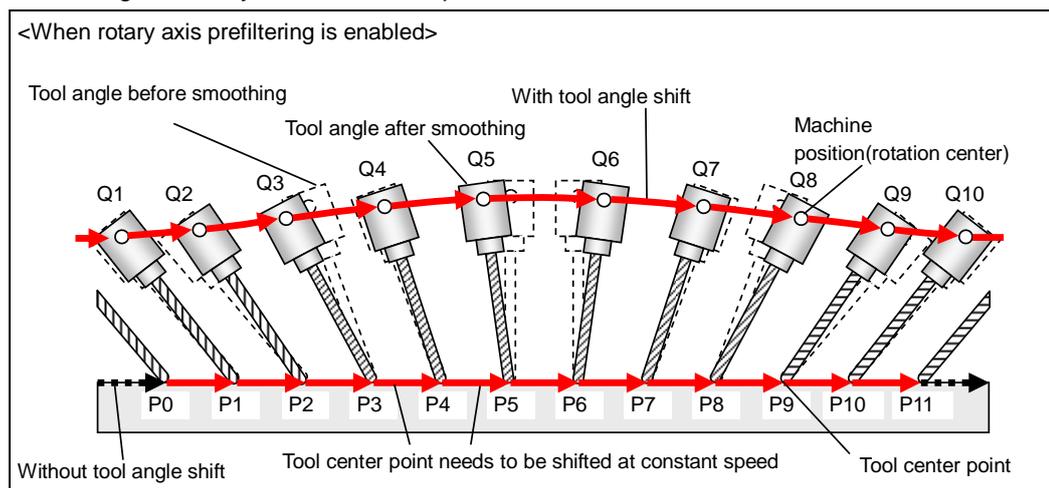
This function is available for the programs which have intermittent rotary axis commands (tool angle shifts) or the programs with inconstant shift amount of rotary axis angle (or tool angle) per unit time.

Set the filter time constant for this function with parameters.

When the rotary axis prefiltering is disabled, the tool center point shift speed may be sharply fluctuated due to the intermitted rotary axis command. See the following image.



As shown below, the rotary axis prefiltering reduces speed fluctuation of tool center point by smoothing the rotary axis command process.



This function is available only when SSS control is enabled.

This function is disabled at G00 command.

The actual angle of the tool may be deviated from the commanded one in the program.



## Relation with other functions (Relation with other G codes)

Pxxx in the list indicates the program error Nos.

Format	Function	The function indicated at the left is commanded in the modal of this function	This function is commanded in the modal indicated at the left	This function is commanded in the same block
G00	Positioning	Switched to a rapid traverse feed rate, and then tool center point Control is performed at the rate.	Perform tool center point control at a rapid traverse feed rate.	Perform tool center point control at a rapid traverse feed rate.
G01	Linear interpolation	Switched to a cutting feed rate, and then tool center point control is performed at the rate.	Perform tool center point control at a cutting feedrate.	Perform tool center point control at a cutting feedrate.
G02/G03	Circular interpolation	P942	P941	P941
G02/G03	Helical Interpolation	P942	P941	P941
G02.1/G03.1	Spiral Interpolation	P942	P941	P941
G02.3/G03.3	Exponential interpolation	P942	P941	P941
G04	Dwell	Dwelling is performed.	-	Tool center point control is ignored as dwell function takes precedence over the tool center point control function.

### 13. Program Support Functions

#### 13.24 Tool Center Point Control; G43.4/G43.5

Format		Function	The function indicated at the left is commanded in the modal of this function	This function is commanded in the modal indicated at the left	This function is commanded in the same block
G05	P1 <b>(Note 1)</b>	High-speed machining mode	Max. feedrate is 16.8m/min when 1mm segment G1 block is commanded with 5 axes simultaneously	Max. feedrate is 16.8m/min when 1mm segment G1 block is commanded with 5 axes simultaneously	P33
	P2 <b>(Note 1)</b>		Max. feedrate is 100m/min when 1mm segment G1 block is commanded with 5 axes simultaneously	Max. feedrate is 100m/min when 1mm segment G1 block is commanded with 5 axes simultaneously	
	P10000 <b>(Note 2)</b>	High-speed high-accuracy control II	Max. feedrate is 100m/min when 1mm segment G1 block is commanded with 5 axes simultaneously	P941	P33
G05.1 <b>(Note 2)</b>		High-speed high-accuracy control I	Max. feedrate is 16.8m/min when 1mm segment G1 block is commanded with 5 axes simultaneously	Max. feedrate is 16.8m/min when 1mm segment G1 block is commanded with 5 axes simultaneously	P33
G06.2		NURBS interpolation	P942	P*** NURBS general error	P941
G07		Hypothetical axis interpolation (Not implemented)	P942	-	P941
G07.1 G107		Cylindrical Interpolation	P942	P941	P941
G08 <b>(Note 2)</b>	P0	High-accuracy control	Tool center point control is performed in the cutting mode.	Tool center point control is performed in the cutting mode.	P33
	P1		Tool center point control is performed in the high-accuracy control mode.	Tool center point control is performed in the high-accuracy control mode.	
G09		Exact Stop Check	Deceleration check is performed at the block end.	-	Deceleration check is performed at the block end.
G10/G11.1		Parameter input by program	P942	-	P941
G10		Compensation data input by program	P942	-	P941
G12/G13		Circular cutting	P942	-	Tool center point control is ignored as the circular cutting takes precedence over the tool center point control function.
G12.1/G13.1		Polar coordinate interpolation	P942	P941	P941
G15/G16		Polar coordinate command	P942	P941	P941

**(Note 1)** It is valid when the parameter "#1267 ext03/bit0" is OFF. If it is commanded when the parameter is ON, the program error (P34) will occur.

**(Note 2)** It is valid when the parameter "#1267 ext03/bit0" is ON. If it is commanded when the parameter is OFF, the program error (P34) will occur.

### 13. Program Support Functions

#### 13.24 Tool Center Point Control; G43.4/G43.5

Format	Function	The function indicated at the left is commanded in the modal of this function	This function is commanded in the modal indicated at the left	This function is commanded in the same block
G17~G19	Plane selection	The modal is switched to the specified plane.	-	The modal is switched to the specified plane.
G20/G21	Inch / Metric	P942	Tool center point control is performed according to the inch / metric modal.	P941
G22.1/G23.1	Stroke check before travel	P942	P941	P941
G27	Reference position check	P942	-	The tool center point control is ignored as the reference position check becomes valid.
G28	Reference position return	P942	-	The tool center point control is ignored as the reference position return becomes valid.
G29	Start position return	P942	-	The tool center point control is ignored as the start position return becomes valid.
G30	2nd, 3rd, 4th reference position return	P942	-	The tool center point control is ignored as the 2nd, 3rd, 4th reference position return becomes valid.
G30.1~G30.6	Tool change position return 1 to 6	P942	-	P941
G31	Skip	P942	-	P941
G31.1~G31.3	Multi-step skip function	P942	-	P941
G33	Thread cutting	P942	P941	P941
G34~G36/G37.1	Special Fixed Cycle	P942	-	P941
G37	Automatic tool length measurement	P942	-	P941
G38	Tool radius compensation vector specification	P942	-	P941
G39	Tool radius compensation corner circular command	P942	-	P941
G40/G41/G42	Tool radius compensation	P942	P941	P941
G40.1/G41.1/G41.2	Normal line control	P942	P941	P941
G43/G44	Tool length compensation	Tool length compensation can be performed upon tool center point control cancellation.	Tool center point control can be performed upon tool length compensation cancellation.	The subsequently commanded modal takes precedence.

### 13. Program Support Functions

#### 13.24 Tool Center Point Control; G43.4/G43.5

Format	Function	The function indicated at the left is commanded in the modal of this function	This function is commanded in the modal indicated at the left	This function is commanded in the same block
G43.1/G44	Tool length compensation along the tool axis	Tool length compensation along the tool axis can be performed upon tool center point control cancellation.	Tool center point control can be performed upon tool length compensation along the tool axis cancellation.	The subsequently commanded modal takes precedence.
G45/G46/ G47/G48	Tool position offset	P942	-	P941
G50/G51	Scaling	P942	P941	P942
G50.1/G51.1/ G62	Mirror image	P942	P941	P941
G52	Local coordinate system setting	P942	-	The tool center point control is ignored as the local coordinate system setting becomes valid.
G53	Machine coordinate system selection	P942	-	The tool center point control is ignored as the machine coordinate system selection becomes valid.
G54–G59/ G54.1	Workpiece coordinate system selection	P942	Tool center point control is performed in the currently selected workpiece coordinate system.	P941
G60	Unidirectional positioning	P942	-	The tool center point control is ignored as the unidirectional positioning becomes valid.
G61	Exact stop check mode	Deceleration check is performed at the block end.	Deceleration check is performed at the block end.	Deceleration check is performed at the block end.

### 13. Program Support Functions

#### 13.24 Tool Center Point Control; G43.4/G43.5

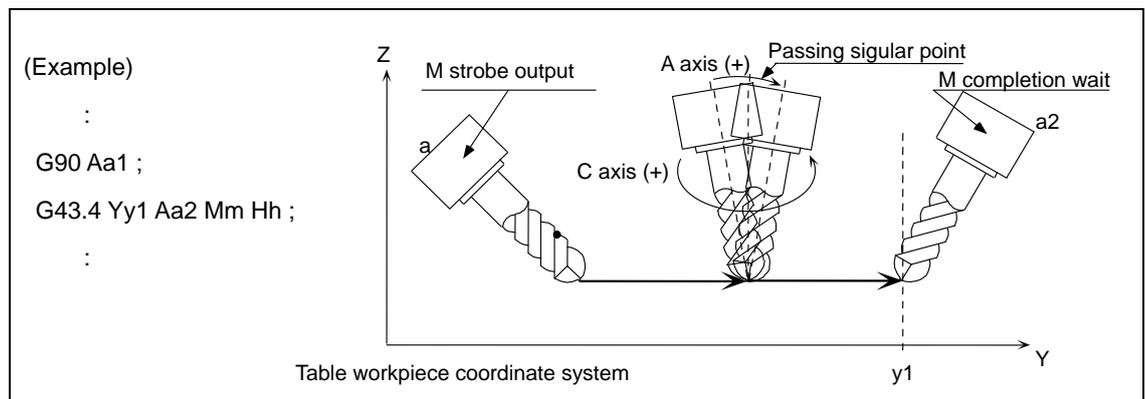
Format	Function	The function indicated at the left is commanded in the modal of this function	This function is commanded in the modal indicated at the left	This function is commanded in the same block
G61.1	High-accuracy control	Tool center point control is performed in the high-accuracy control mode.	Tool center point control is performed in the high-accuracy control mode.	Tool center point control is performed in the high-accuracy control mode.
G61.2	High-accuracy spline interpolation 1	P942	P941	P941
G63	Tapping mode	P942	P941	P941
G64	Cutting mode	Tool center point control is performed in the cutting mode.	Tool center point control is performed in the cutting mode.	Tool center point control is performed in the cutting mode.
G65~ G67/G66.1	User macro	Tool center point control becomes valid even in the user macro program.	Tool center point control becomes valid even in the user macro program.	Tool center point control is ignored as the user macro takes precedence over the tool center point control function.
G68	User macro subprogram termination	User macro subprogram is terminated.	-	Tool center point control is ignored.
G69	End position error check cancellation	The end position error check cancellation becomes valid.	-	Both end position error check cancellation and tool center position control become valid.
G68.1/G69.1	Coordinate rotation	P942	P941	P941
G68.1iJkK/ G69.1	3-dimensional coordinate conversion	P922	P941	P923
G70~G89	Fixed cycle	P942	The tool center point control is ignored as the start fixed cycle becomes valid.	The tool center point control is ignored as the start fixed cycle becomes valid.
G90/G91	Absolute/Incremental value command	The modal is switched to the specified absolute / incremental value command, and then tool center point control is performed.	Tool center point control is performed under the specified absolute / incremental value command.	Tool center point control is performed under the specified absolute / incremental value command.
G92	Machine coordinate system setting	P942	-	P941
G94	Feed per minute	Tool center point control is performed in the feed-per-minute mode.	Tool center point control is performed in the feed-per-minute mode.	Tool center point control is performed in the feed-per-minute mode.
G95	Feed per revolution	P942	P941	P941
G96/G97	Constant surface speed control	P942	P941	P941
G98	Fixed cycle initial level return	The modal is switched to G98 and tool center point control becomes valid.	The modal is switched to G98 and tool center point control becomes valid.	The modal is switched to G98 and tool center point control becomes valid.
G99	Fixed cycle R point level return	The modal is switched to G99 and tool center point control becomes valid.	The modal is switched to G99 and tool center point control becomes valid.	The modal is switched to G99 and tool center point control becomes valid.
G114.1	Spindle synchronization	P942	P941	P941

**(Note)** All the G codes not listed above are disabled.



## Relation with other functions

- (1) F 1-digit feed  
Controls so that the tool center point moves at the commanded speed. Note that speed cannot be changed with the manual handle.
- (2) Buffer correction  
Buffer correction cannot be performed during tool center point control.
- (3) Miscellaneous functions (MSTB)  
Miscellaneous function (MSTB) command can be executed during tool center point control. (When passing the singular point, strobe signal is output at the block start and the completion wait at the block end.)



- (4) Spindle/C axis control  
Axes unrelated to the tool tilt or table tilt can be controlled.
- (5) Manual reference position return  
Do not perform manual reference position return during tool center point control. If performed, the tool moves off the programmed track.
- (6) Machining time calculation  
Machining time calculation is not performed accurately on the machining program in which the tool center point control mode is commanded.
- (7) Graphic trace  
Graphic trace during the tool center point control allows the tool center point to be traced.
- (8) Graphic check  
Graphic check during the tool center point control always allows the tool center point to be viewed for the purpose of programming confirmation.
- (9) Program restart  
Restart search cannot be performed during the tool center point control. If attempted, a program error (P49) occurs.
- (10) Rest modal retention  
Cancelled during the tool center point control.
- (11) Collation stop  
Position in the tool center point control can be collated.

- (12)Automatic operation handle interruption  
Do not perform the automatic operation handle interruption during the tool center point control. If performed, the tool moves off the programmed track.
- (13)Manual / Automatic simultaneous  
Manual / Automatic simultaneous cannot be executed to the axes related to the tool center point control during the tool center point control.
- (14)Tool handle feed & interruption  
Do not perform the tool handle feed & interruption during the tool center point control. If performed, the tool moves off the programmed track.
- (15)Corner chamfering/Corner R  
When the corner chamfering/corner R is performed during the tool center point control, the tool center point control becomes valid to the track after the corner chamfering/corner R.
- (16)Mirror image by parameter setting / external mirror image input  
When the tool center point control command is issued during the mirror image by parameter/external input, a program error (P941) occurs. Also, do not turn the mirror image by parameter/external input ON during the tool center point control.
- (17)Linear angle command  
When A axis is used as a rotary axis, the linear angle command cannot be executed. When A axis is not used as a rotary axis, tool center point control becomes valid to the shape after the linear angle command.
- (18)Geometric command  
When A axis is used as a rotary axis, the geometric command cannot be executed. If executed, a program error When A axis is not used as a rotary axis, tool center point control becomes valid to the shape after the geometric command.
- (19)Figure rotation  
The tool center point control becomes valid to the shape after the figure rotation.
- (20)Coordinate rotation by parameter  
When the tool center point control command is issued during the coordinate rotation by parameter, a program error (P941) occurs. Also, do not turn the coordinate rotation by parameter ON during the tool center point control.
- (21)Chopping  
Chopping operation for the 3 orthogonal axes and 2 rotary axes cannot be performed during the tool center point control.
- (22)Macro interruption  
If the macro interruption command is executed during the tool center point control, a program error (P941) occurs.
- (23)Tool life management  
The compensation amount of the tool center point control during the tool life management is equal to the compensation amount of the tool subjected to the tool life management.
- (24)G00 non-interpolation  
Functions as "G00 interpolation".

- (25)Actual feed rate display  
The final combined feed rate is displayed here.
- (26)Manual interruption  
When the manual interruption is executed during the feed hold or single block stop, the movement will be the one to be observed when the manual ABS is OFF when rebooting regardless of whether an absolute/incremental value command is selected.
- (27)Machine lock  
The each Machine Lock becomes valid to the motor axis.
- (28)Remaining distance counter  
Remaining distance at the tool center point on the programming coordinate system is displayed.
- (29)Interlock  
Interlock is applied for the motor axis.
- (30)Cutting feed / Rapid traverse override  
Override is applied to the feedrate at the tool center point. When the federate is clamped, the override is applied to the clamp speed.
- (31)Manual reference position return  
If the manual reference position return is performed during the tool center point control, the tool moves off the programmed track after that.
- (32)Dry run  
Dry run is applied to the speed at the tool center point.
- (33)NC reset  
Immediately decelerates to stop when the NC reset is executed during the tool center point control. The tool center point control will be canceled even if NC reset 1 and the modal retention.
- (34)Emergency stop  
Immediately stops if the emergency stop is applied during the tool center point control.
- (35)Stored stroke limit  
Stored stroke limit will be valid at the motor axis for all IB, IIB and IC.
- (36)MDI interruption  
When the MDI interruption is performed during the tool center point control, an operation error (O170) occurs.

13.25 Timing-synchronization between Part Systems

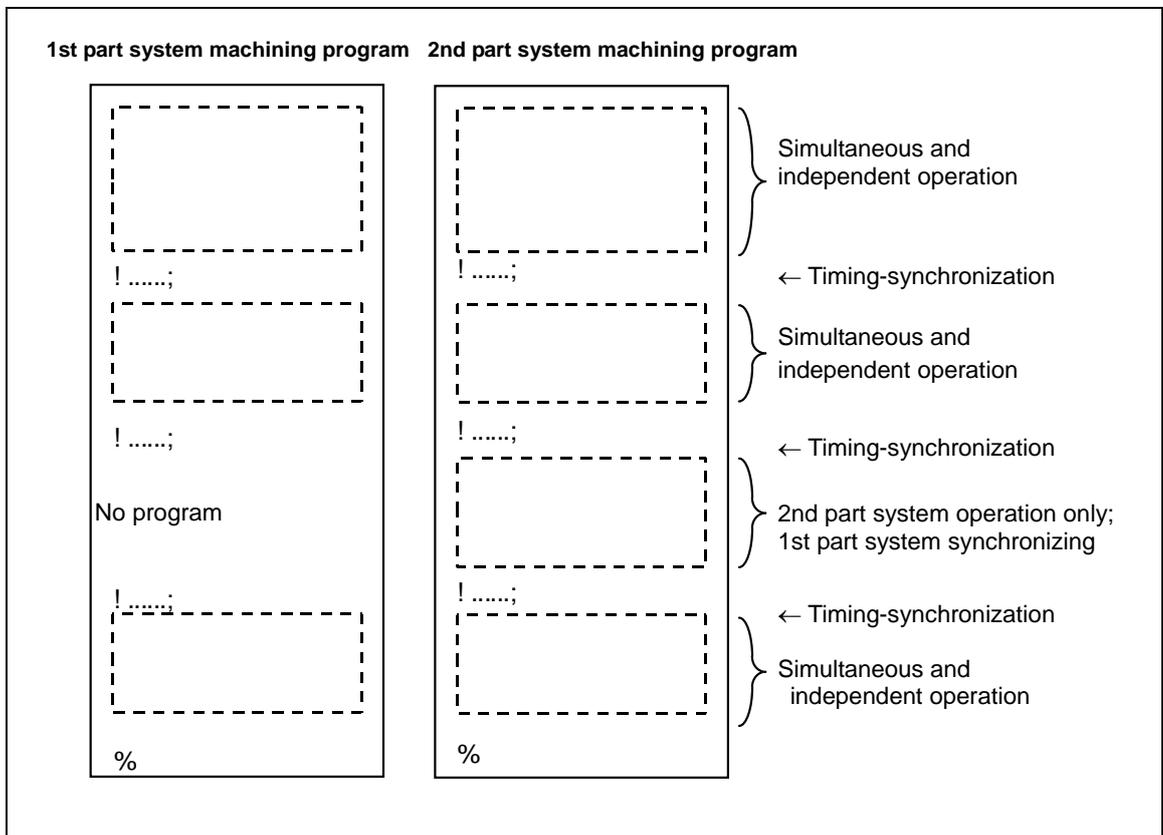
**CAUTION**

⚠ When programming a program of the multi-part system, carefully observe the movements caused by other part systems' programs.



**Function and purpose**

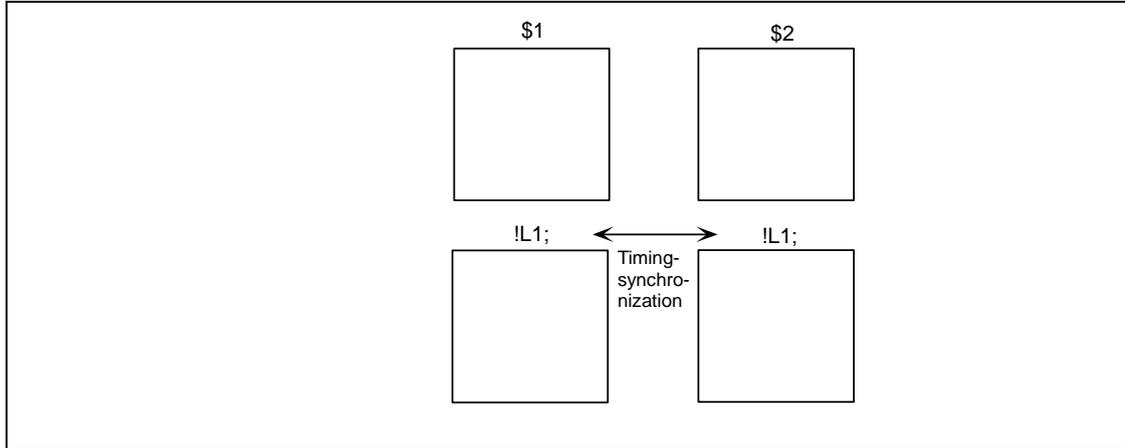
The multi-axis, multi-part system complex control NC system can simultaneously run multiple machining programs independently. The timing-synchronization between part systems function is used in cases when, at some particular point during operation, the operations of 1st and 2nd part systems are to be synchronized or in cases when the operation of only one part system is required.





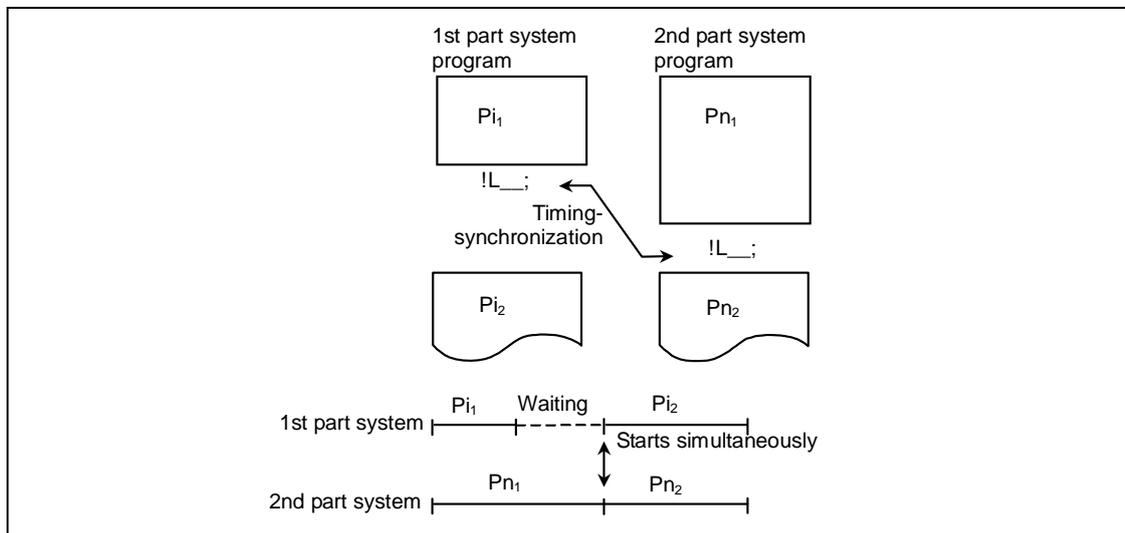
Command format

```
!L__ ;
|           : Synchronizing No. 1 to 9999
```



Detailed description

- (1) If !L\_\_ is commanded from one part system, operation of the first part system's program will wait until !L\_\_ is commanded from the other part system's program. When !L\_\_ is commanded, the programs for the two part systems will start simultaneously.

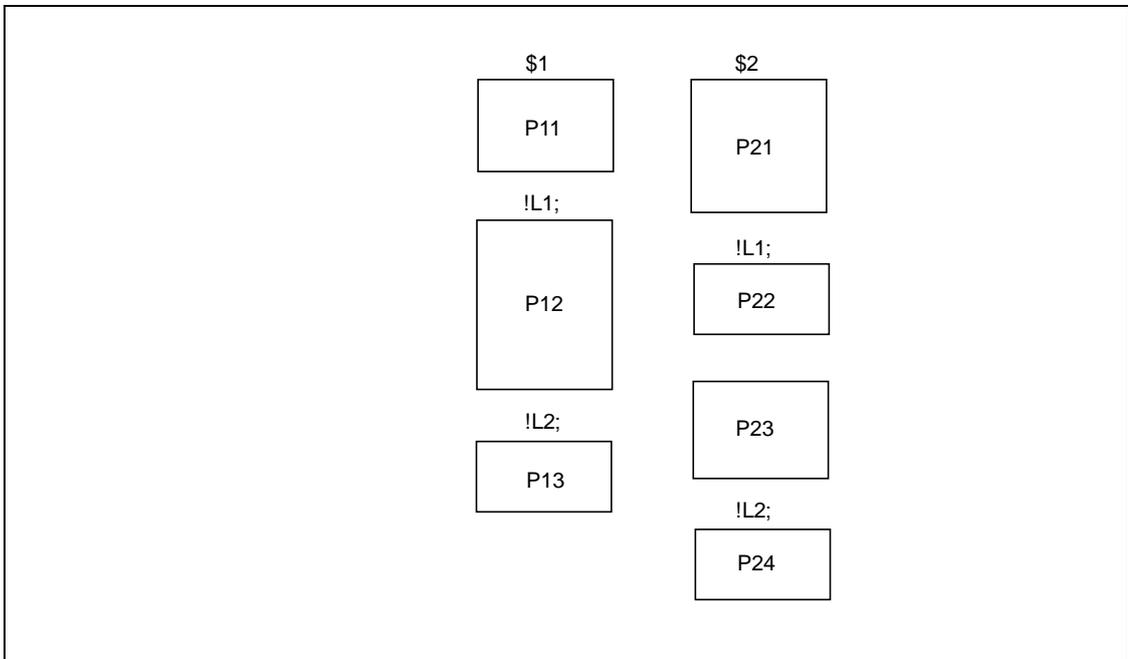


- (2) The timing-synchronization command is normally issued in a single block. However, if a movement command or M, S or T command is issued in the same block, whether to synchronize after the movement command or M, S or T command or to execute the movement command or M, S or T command after synchronization will depend on the parameter (#1093 Wmvfin).  
 #1093 Wmvfin 0: Synchronize before executing movement command.  
 1: Synchronize after executing movement command.

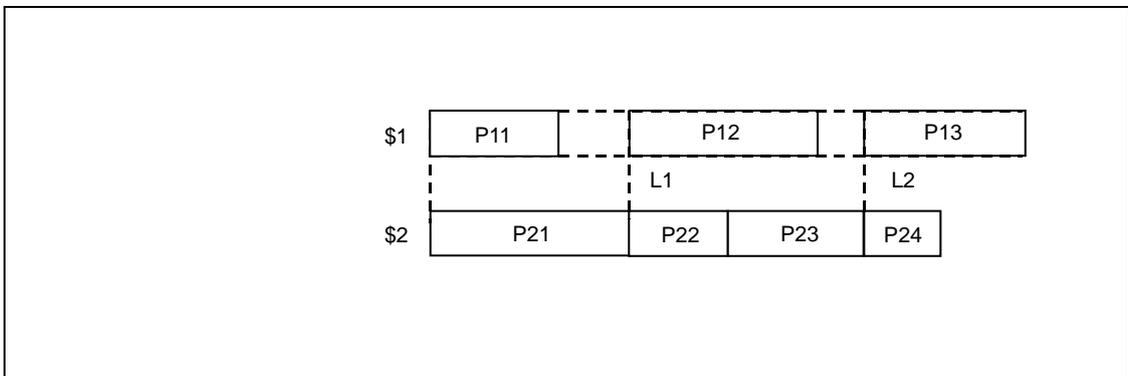
- (3) If there is no movement command in the same block as the timing-synchronization command, when the next block movement starts, timing-synchronization may not be secured between the part systems. To synchronize the part systems when movement starts after the timing-synchronization, issue the movement command in the same block as the synchronizing command.
- (4) Timing-synchronizing is done only while the part system to be synchronized is operating automatically. If this is not possible, the timing-synchronization command will be ignored and operation will advance to the next block.
- (5) The L command is the synchronizing identification No. The same Nos. are synchronized but when they are omitted, the Nos. are handled as L0.
- (6) "SYN" will appear in the operation status section during timing-synchronization. The timing-synchronization signal will be output to the PLC I/F.



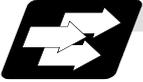
**Example of timing-synchronization between part systems**



The above programs are executed as follows:



## 13.26 End Point Error Check Cancellation; G69



## Function and purpose

If an illegal program is given to NC, a program error will occur.

However, the error point check cancellation command G69 can be used to escape a program error only in the following conditions.

- (1) The end point does not exist on the circumference of a circle in circular interpolation (containing helical cutting).
- (2) With tool radius compensation, the compensation amount is changed during circular interpolation modal.



## Command format

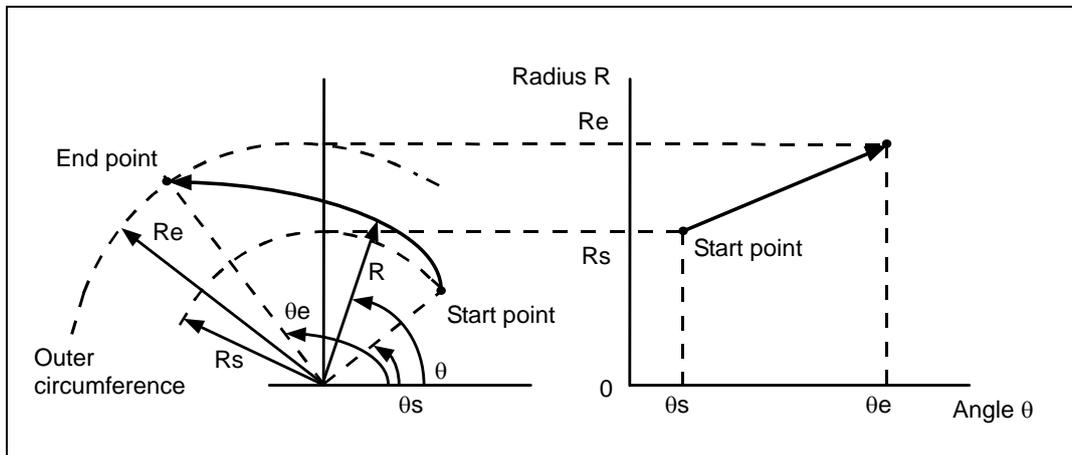
<b>G69 (G03) X__ Y__ Z__ --- ;</b>	<b>End point error check cancellation</b>
G69	:End point error check cancellation (G69 command is unmodal.)
G03	:Commands such as circular command
X Y Z ---	:End point of the circular commands
	Though this point is illegal at usual, it will not be regarded as the error point by using G69.



## Detailed description

## (1) Circular cutting (including helical interpolation)

If the end point does not exist on the circumference of a circle and G69 command is given in circular interpolation, the circular radius changes linearly together with the angle after movement as shown below:



Although the speed of the outer circumference is slightly faster than the commanded speed, the angular speed is constant.

(Ex.1) Heart cum cutting

Displacement is the radius difference at the start and end points ((b - a) in the illustration below). The example program is separated into two blocks of the right and left sides.

	<p>G69 G03 Ya+b Jb Ff<sub>1</sub>;                  (right side)                  G69 G03 Y-a-b J-a ; (left side)</p>
--	-------------------------------------------------------------------------------------------------------------------------------

(Ex.2) Conical cutting

Conical cutting is taper thread cutting application, and its start or end point exists on the center line.

Taper can be carried out by incrementing or decrementing the circular radius. The pitch is determined by  $z1/p1$ .

	<p>G69 G02 X-x<sub>1</sub> Y0 Zz<sub>1</sub> I-x<sub>1</sub> Pp<sub>1</sub> Ff<sub>1</sub> ;                  x<sub>1</sub>: Bottom radius                  z<sub>1</sub>: Height                  p<sub>1</sub>: Number of pitches                  f<sub>1</sub>: Feedrate</p>
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

(2) Tool radius compensation

When some changes such as the compensation amount occur in the circular interpolation modal, use this command (G69).

(Ex.1) Tool radius compensation during circular mode

- G69 G41 G02 Xx<sub>1</sub> Yy<sub>1</sub> Ii<sub>1</sub> Jj<sub>1</sub> Ff<sub>1</sub> Hh<sub>1</sub>: ... Tool radius compensation amount change
- G69 G40 G02 Xx<sub>2</sub> Yy<sub>2</sub> Ii<sub>2</sub> Jj<sub>2</sub>: ..... Tool radius compensation cancel

## 13.27 Coordinate Read Function; G14



## Function and purpose

The G14 command is used to read the end point coordinates of the immediately preceding block, the machine coordinates, the workpiece coordinates, the TLM coordinates, or the skip coordinates.



## Command format

**G14 X\_\_Y\_\_Z\_\_A\_\_P\_\_ ;    Coordinate read command**

X Y Z A                    : Each axis and the variable No. where the axis's coordinates data will be stored.

(Aa means the additional axis.)

P                            : Type of the coordinates to be read



## Detailed description

- (1) The type of coordinates to be read is specified in "p". The coordinates that can be read are listed below:

"P" command value	Contents	Compensation amount	Read during moving
No "p" command is given or a value other than 1 to 4 is specified	End point position of the immediately preceding block (The absolute coordinate system set by G92)	Not contained	Enabled
1	Machine coordinates (Basic machine coordinate system)	Contained	Disabled
2	Zero point of the current workpiece coordinate system (Workpiece coordinate system)	Contained	Disabled
3	TLM coordinates (The point where TLM switch is turned OFF to ON at machine stop is used as the zero point of the TLM coordinate system.)	Contained	Disabled
4	Skip signal input coordinates (Workpiece coordinate system)	Contained	Enabled

- (2) The end point position will be read instead of the skip coordinates when the skip signal is not turned ON in a G31 block.
- (3) G14 command is unmodal, thus, it is valid only in the block where it is commanded.
- (4) The end point and skip coordinates can be read even while moving. Always confirm that the movement has stopped before reading the machine, workpiece, and TLM coordinates.  
**(Note)** To stop the movement, stop the axis movement with the dwell command and so forth.
- (5) The program error (P241) will occur if an illegal variable No. is commanded.

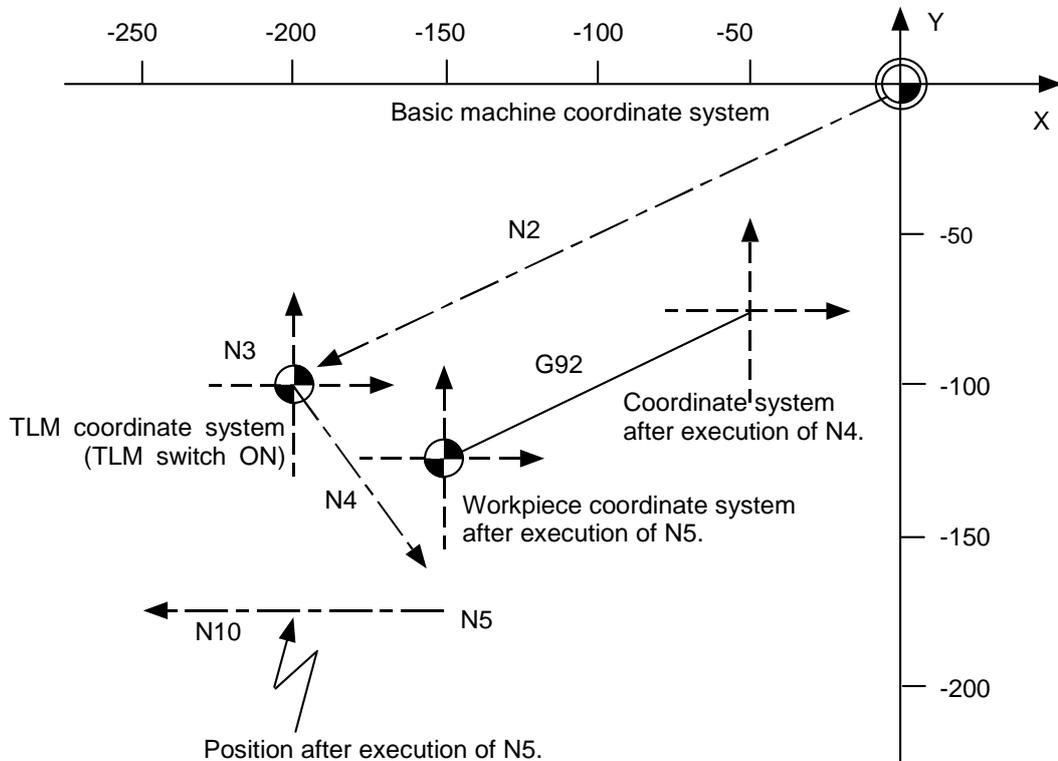


Example of program

(1) An example of p<sub>1</sub> command value and reading coordinates are given.

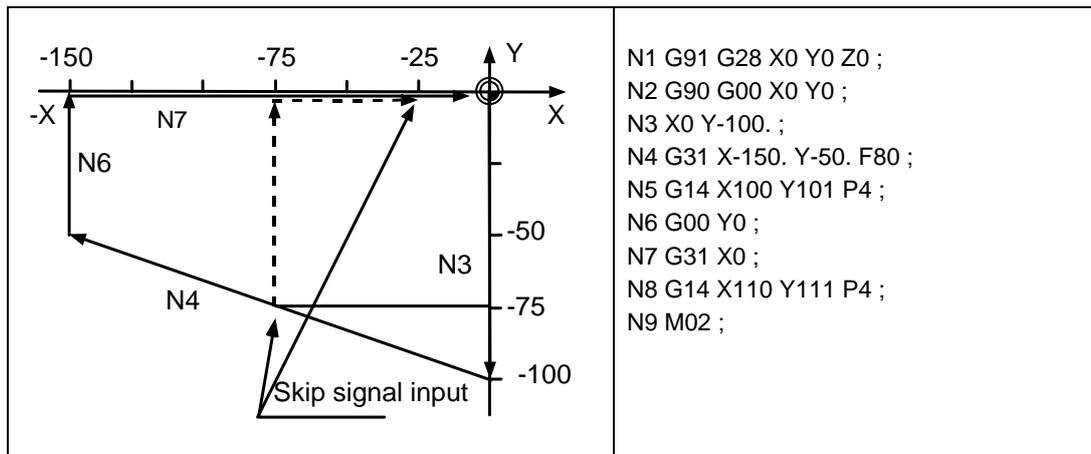
```

N1 G28 X0 Y0 Z0 ;
N2 G90 G00 X-200. Y-100. G53 ;
N3 M60 ; (TLM switch is turned from OFF to ON.)
N4 G00 G54 X-100. Y-100. ;
N5 G92 X0 Y0 Y-50. ;
N6 G00 G43 X-100. H10 ;
N7 G14 X100 Y101 Pp1 ;
  ( G54 : X-50.      H10 : 50.
    Y-75. )
    
```



Coordinates	"P" command value	Read coordinates	Compensation amount
End point position of the immediately preceding block	No "p" command is given or a value other than 1 to 4 is specified	#100 = -100. #101 = -50.	Not contained
Machine coordinates	1	#100 = -200. #101 = -175.	Contained
Workpiece coordinates	2	#100 = -50. #101 = -50.	Contained
TLM coordinates	3	#100 = -0 #101 = -75.	Contained

(2) An example of reading skip coordinates are given.



The coordinates read by the above program is as follows.

#100 = 75. +  $\varepsilon$                       #101 = 75. +  $\varepsilon$   
 #110 = 25. +  $\varepsilon$                       #111 = 75. +  $\varepsilon$

**(Note1)** " $\varepsilon$ " is an error caused by response delay. (For details, see the section of skip function.)

**(Note2)** The skip signal input coordinates of N4 block will be read to #111 because there is no Y command in the N7 block

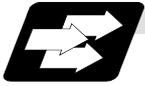


### Precautions

(1) The program error (P241) will occur if an illegal variable No. is commanded.

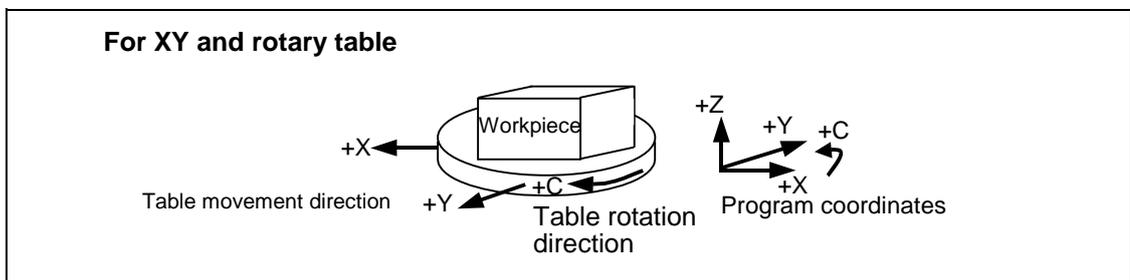
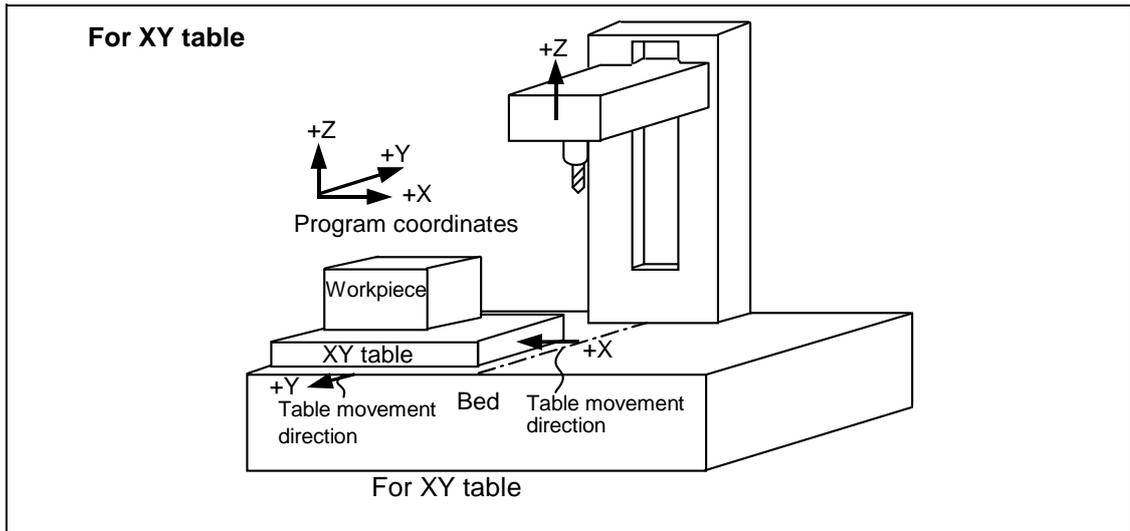
## 14. Coordinates System Setting Functions

### 14.1 Coordinate Words and Control Axes



#### Function and purpose

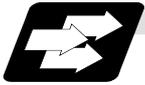
There are three controlled axis for the basic specifications, but when an additional axis is added, up to four axes can be controlled. Pre-determined alphabetic coordinate words that correspond to the axes are used to designate each machining direction.



## 14. Coordinates System Setting Functions

### 14.2 Basic Machine, Workpiece and Local Coordinate Systems

#### 14.2 Basic Machine, Workpiece and Local Coordinate Systems

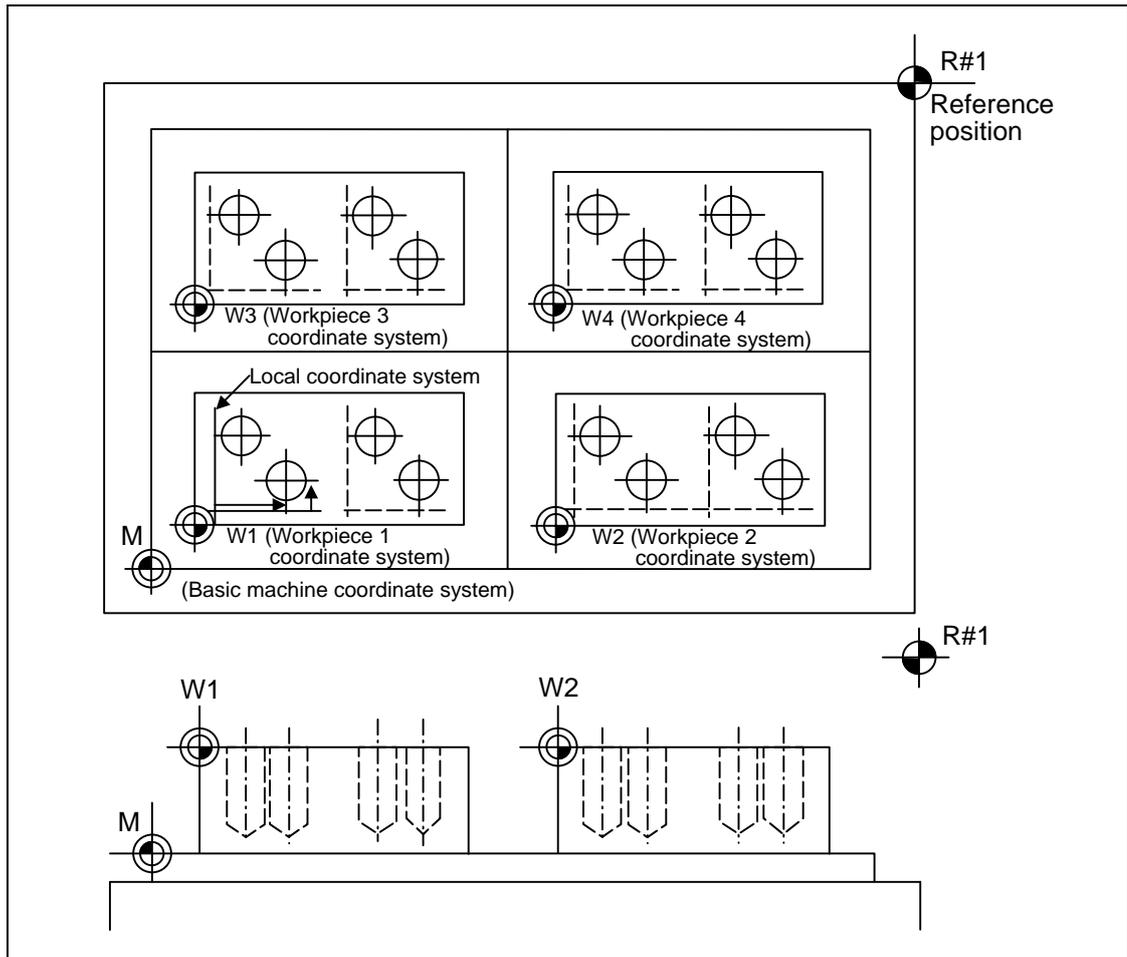


##### Function and purpose

The basic machine coordinate system is fixed in the machine and it denotes that position which is determined inherently by the machine.

The workpiece coordinate systems are used for programming and in these systems the reference position on the workpiece is set as the coordinate zero point.

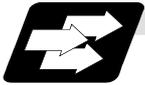
The local coordinate systems are created on the workpiece coordinate systems and they are designed to facilitate the programs for parts machining.



## 14. Coordinates System Setting Functions

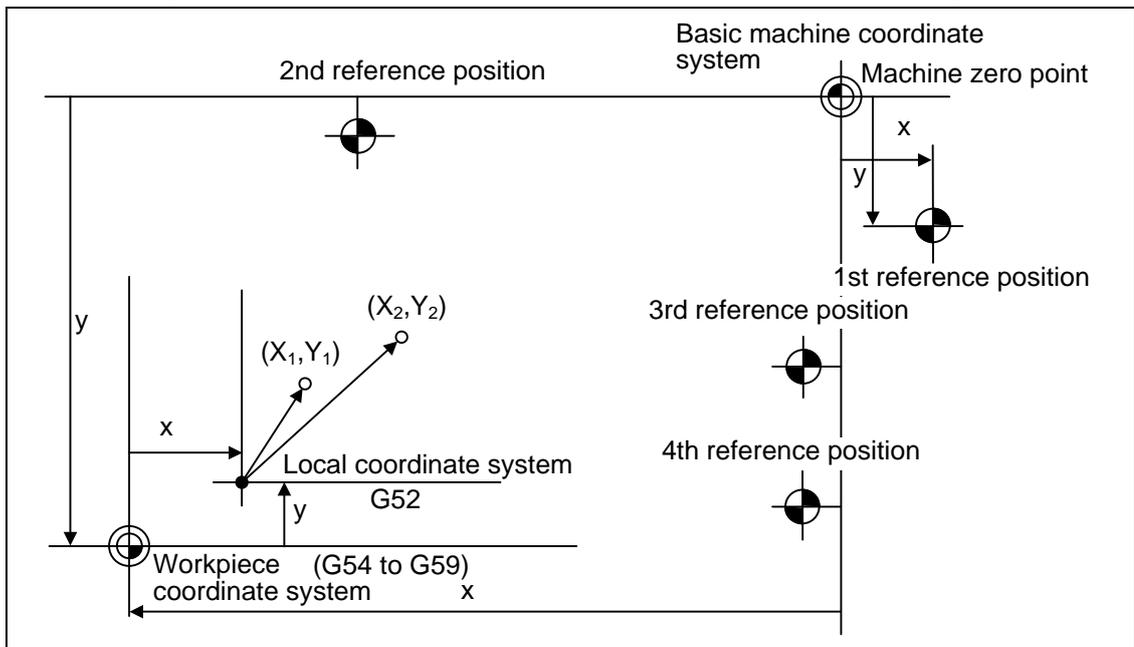
### 14.3 Machine Zero Point and 2nd, 3rd, 4th Reference Positions

#### 14.3 Machine Zero Point and 2nd, 3rd, 4th Reference Positions

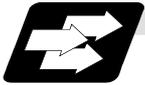


##### Function and purpose

The machine zero point serves as the reference for the basic machine coordinate system. It is inherent to the machine and is determined by the reference (zero) position return. 2nd, 3rd and 4th reference positions relate to the position of the coordinates which have been set beforehand by parameter from the zero point of the basic machine coordinate system.



## 14.4 Basic Machine Coordinate System Selection; G53



## Function and purpose

The basic machine coordinate system is the coordinate system that expresses the position (tool change position, stroke end position, etc.) that is characteristic to the machine.

The tool is moved to the position commanded on the basic machine coordinate system with the G53 command and the coordinate command that follows.



## Command format

Basic machine coordinate system selection

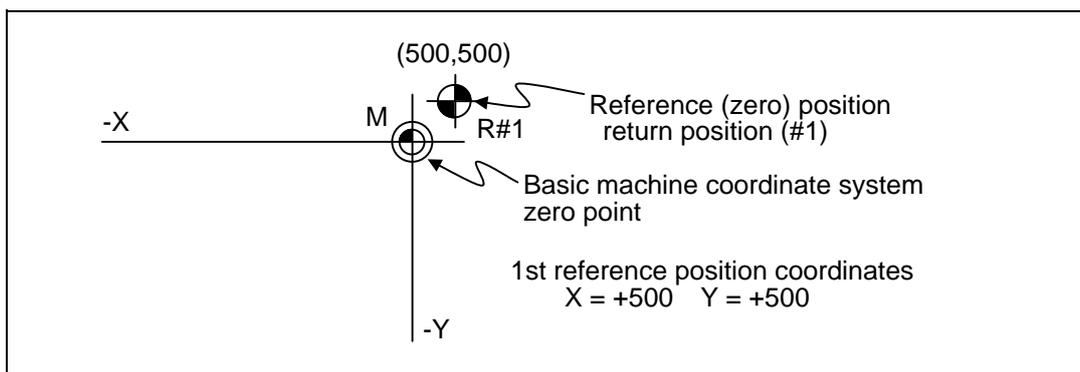
**(G90) G53 X\_\_ Y\_\_ Z\_\_ α\_\_ ;**

α :Additional axis

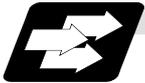


## Detailed description

- (1) When the power is switched on, the basic machine coordinate system is automatically set as referenced to the reference (zero) position return position, which is determined by the automatic or manual reference (zero) position return.
- (2) The basic machine coordinate system is not changed by the G92 command.
- (3) The G53 command is valid only in the block in which it has been designated.
- (4) In the incremental value command mode (G91), the G53 command provides movement with the incremental value in the coordinate system being selected.
- (5) Even if G53 is commanded, the tool radius compensation amount for the commanded axis will not be canceled.
- (6) The 1st reference position coordinate value indicates the distance from the basic machine coordinate system 0 point to the reference position (zero point) return position.
- (7) The G53 command will move with cutting feedrate or rapid traverse following command modal.
- (8) If the G53 command and G28 command (reference position return) are issued in the same block, the command issued last will be valid.



14.5 Coordinate System Setting; G92



Function and purpose

By commanding G92, the absolute value (workpiece) coordinate system and current position display value can be preset in the command value without moving the machine.



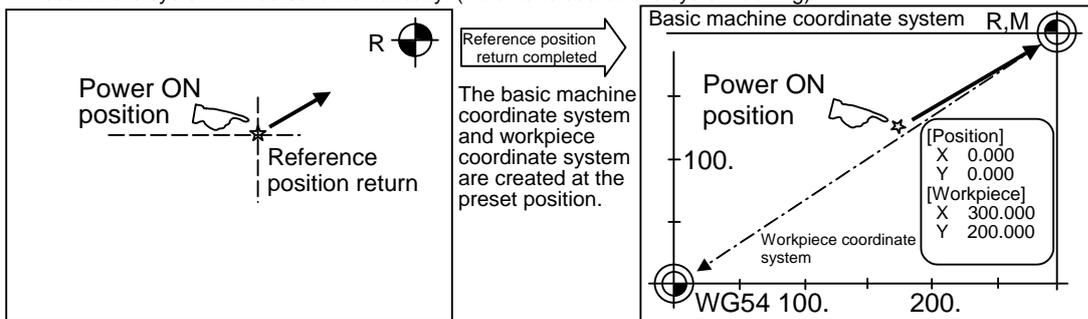
Command format

G92 X\_\_ Y\_\_ Z\_\_ α\_\_ ;  
 α :Additional axis

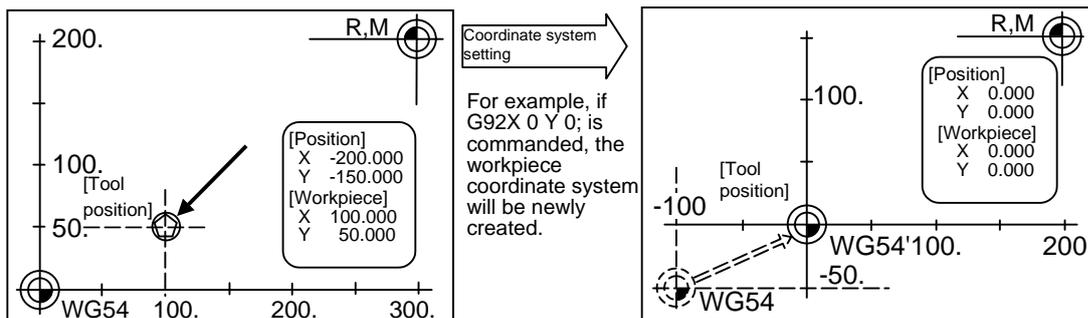


Detailed description

(1) After the power is turned on, the first reference position return will be done with dog-type, and when completed, the coordinate system will be set automatically. (Automatic coordinate system setting)



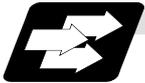
(2) By commanding G92, the absolute value (workpiece) coordinate system and current position display value can be preset in the command value without moving the machine.



**(Note)** If the workpiece coordinate system deviated because the axis is moved manually when the manual absolute position switch is OFF, etc., the workpiece coordinate system can be corrected with the following steps.

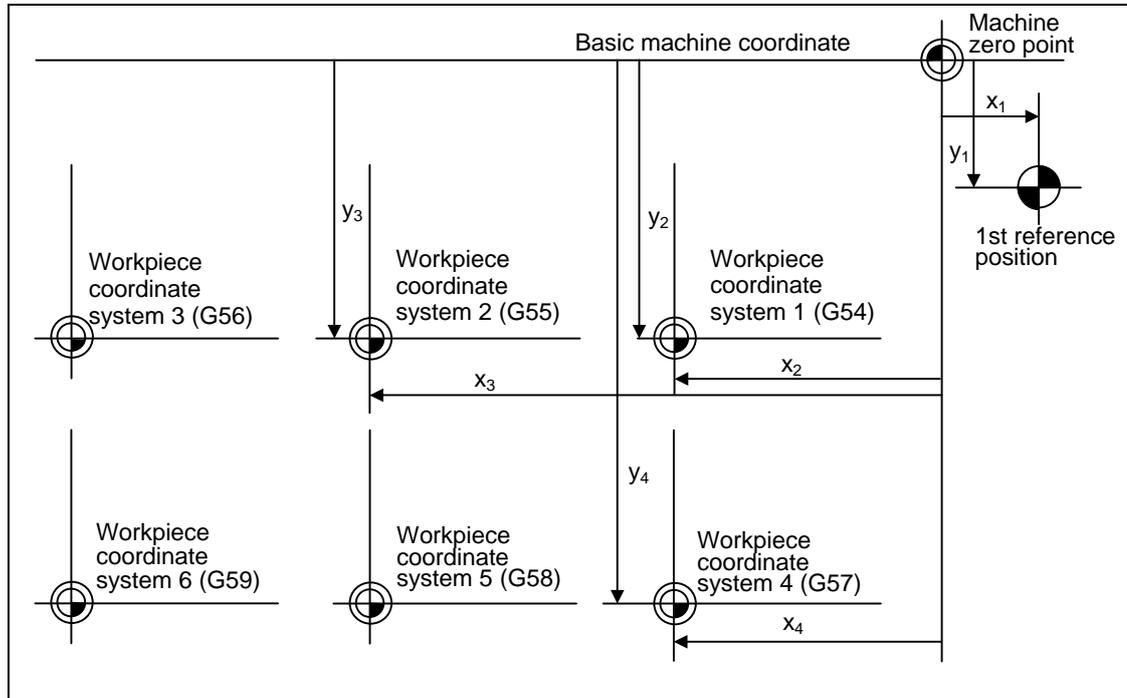
- (1) Execute reference position return while the coordinate system is deviated.
- (2) After that, command G92G53X0Y0Z0;. With this command, the workpiece coordinate position and current position will be displayed, and the workpiece coordinate system will be preset to the offset value.

## 14.6 Automatic Coordinate System Setting



## Function and purpose

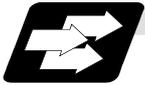
This function creates each coordinate system according to the parameter values input beforehand from the setting and display unit when the reference position is reached with the first manual reference position return or dog-type reference position return when the NC power is turned ON.



## Detailed description

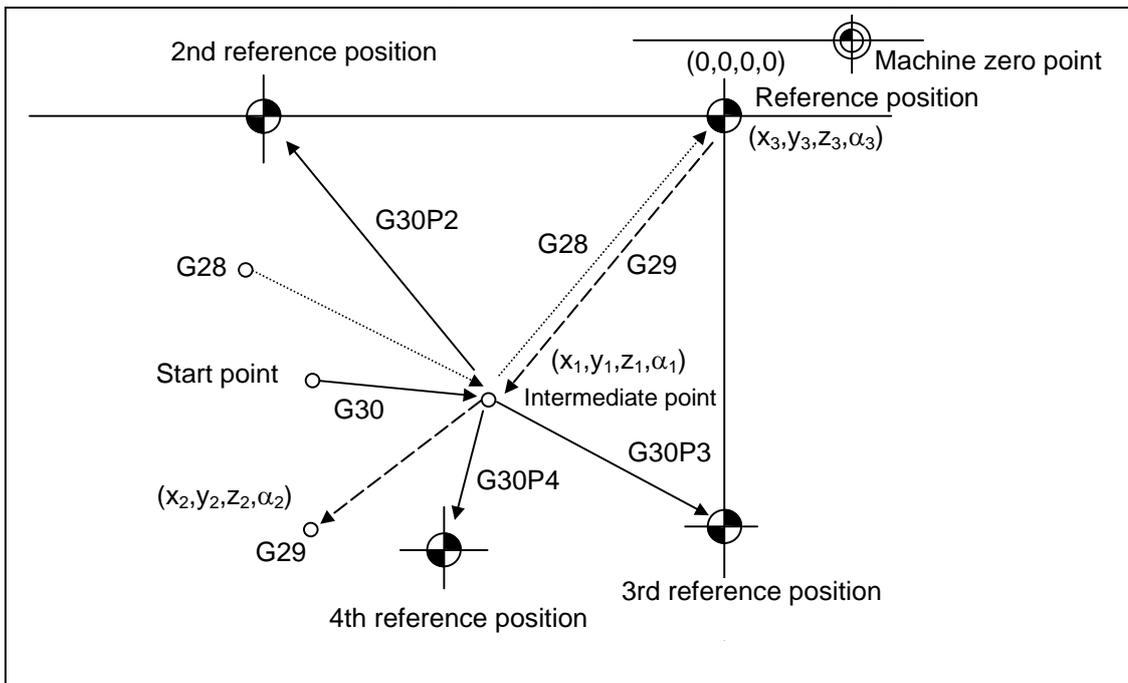
- (1) The coordinate systems created by this function are as follow:
  - (a) Basic machine coordinate system
  - (b) Workpiece coordinate systems (G54 to G59)
- (2) The parameters related to the coordinate system all provide the distance from the zero point of the basic machine coordinate system. Therefore, it is decided at which position in the basic machine coordinate system the first reference position should be set and then the zero point positions of the workpiece coordinate systems are set.
- (3) When the automatic coordinate system setting function is executed, shifting of the workpiece coordinate system with G92, setting of the local coordinate system with G52, shifting of the workpiece coordinate system with origin set, and shifting of the workpiece coordinate system with manual interruption will be canceled.
- (4) The dog-type reference position return will be executed in the following conditions.
  - The first manual reference position return after power ON.
  - The first automatic reference position return after power ON.
  - The second and following manual reference position return when the dog-type is selected with a parameter.
  - The second and following automatic reference position return when the dog-type is selected with a parameter.

## 14.7 Reference (Zero) Position Return; G28, G29



## Function and purpose

- (1) After the commanded axes have been positioned by G0, they are returned respectively at rapid traverse to the first reference (zero) position when G28 is commanded.
- (2) By commanding G29, the axes are first positioned independently at high speed to the G28 or G30 intermediate point and then positioned by G0 at the commanded position.



## Command format

**G28  $X_{x_1}$   $Y_{y_1}$   $Z_{z_1}$   $\alpha_{\alpha_1}$  ; Automatic reference position return**

**G29  $X_{x_2}$   $Y_{y_2}$   $Z_{z_2}$   $\alpha_{\alpha_2}$  ; Start position return**

$\alpha_{\alpha_1}/\alpha_{\alpha_2}$  : additional axis



## Detailed description

- (1) The G28 command is equivalent to the following:

G00 Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> α<sub>1</sub> ;

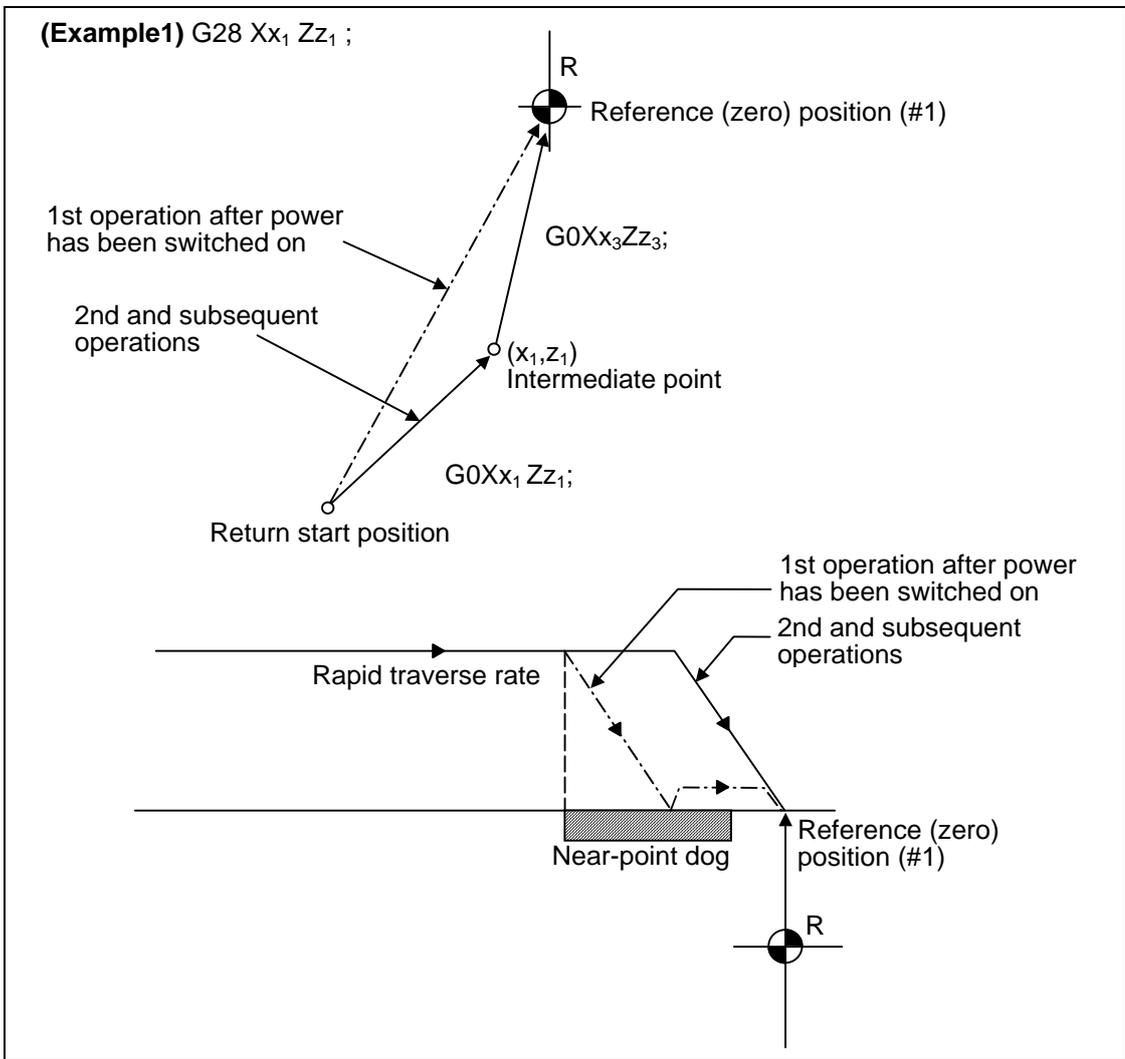
G00 Xx<sub>3</sub> Yy<sub>3</sub> Zz<sub>3</sub> α<sub>3</sub> ;

In this case, x<sub>3</sub>, y<sub>3</sub>, z<sub>3</sub> and α<sub>3</sub> are the reference position coordinates and they are set by a parameter "#2037 G53ofs" as the distance from the zero point of the basic machine coordinate system.

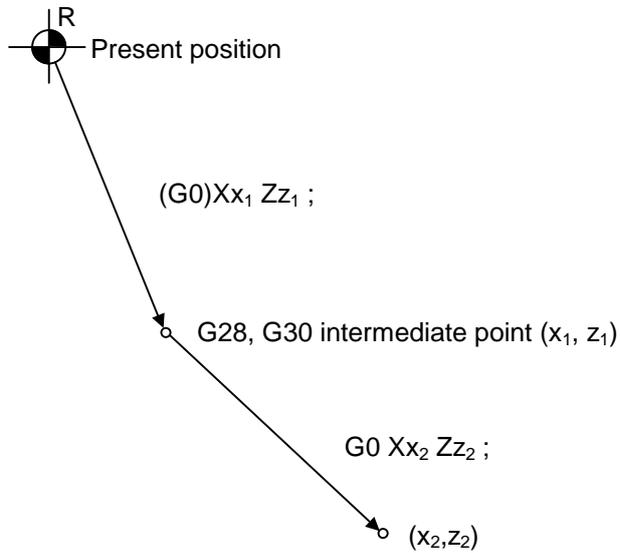
- (2) After the power has been switched on, the axes which have not been subject to manual reference (zero) position are returned by the dog type of return just as with the manual type. In this case, the return direction is regarded as the command sign direction. If the return type is straight-type return, the return direction will not be checked. For the second and subsequent returns, the return is made at high speed to the reference (zero) position which was stored at the first time and the direction is not checked.
- (3) When reference (zero) position return is completed, the zero point arrival output signal is output and also #1 appears at the axis name line on the setting and display unit screen.
- (4) The G29 command is equivalent to the following:
- |                                                                      |                                                                                                                                                          |
|----------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| G00 Xx <sub>1</sub> Yy <sub>1</sub> Zz <sub>1</sub> α <sub>1</sub> ; | } Rapid traverse (non-interpolation type) applies independently for each axis for the positioning from the reference position to the intermediate point. |
| G00 Xx <sub>2</sub> Yy <sub>2</sub> Zz <sub>2</sub> α <sub>2</sub> ; |                                                                                                                                                          |
- In this case, x<sub>1</sub>, y<sub>1</sub>, z<sub>1</sub> and α<sub>1</sub> are the coordinates of the G28 or G30 intermediate point.
- (5) Program error (P430) results when G29 is executed if automatic reference (zero) position return (G28) is not performed after the power has been switched on.
- (6) When the Z axis is canceled, the movement of the Z axis to the intermediate point will be ignored, and only the position display for the following positioning will be executed. (The machine itself will not move.)
- (7) The intermediate point coordinates (x<sub>1</sub>, y<sub>1</sub>, z<sub>1</sub>, α<sub>1</sub>) of the positioning point are assigned by the position command modal. (G90, G91).
- (8) G29 is valid for either G28 or G30 but the commanded axes are positioned after a return has been made to the latest intermediate point.
- (9) The tool compensation will be canceled during reference position return unless it is already canceled, and the compensation amount will be cleared.



Example of program

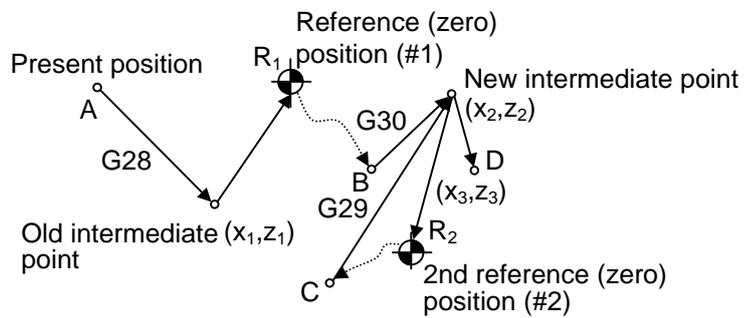


(Example2) G29 Xx<sub>2</sub> Zz<sub>2</sub> ;



(Example 3)

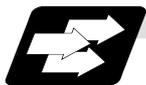
- G28 Xx<sub>1</sub> Zz<sub>1</sub> ; (From point A to reference (zero) position)
- G30 Xx<sub>2</sub> Zz<sub>2</sub> ; (From point B to 2nd reference (zero) position)
- G29 Xx<sub>3</sub> Zz<sub>3</sub> ; (From point C to point D)



## 14. Coordinates System Setting Functions

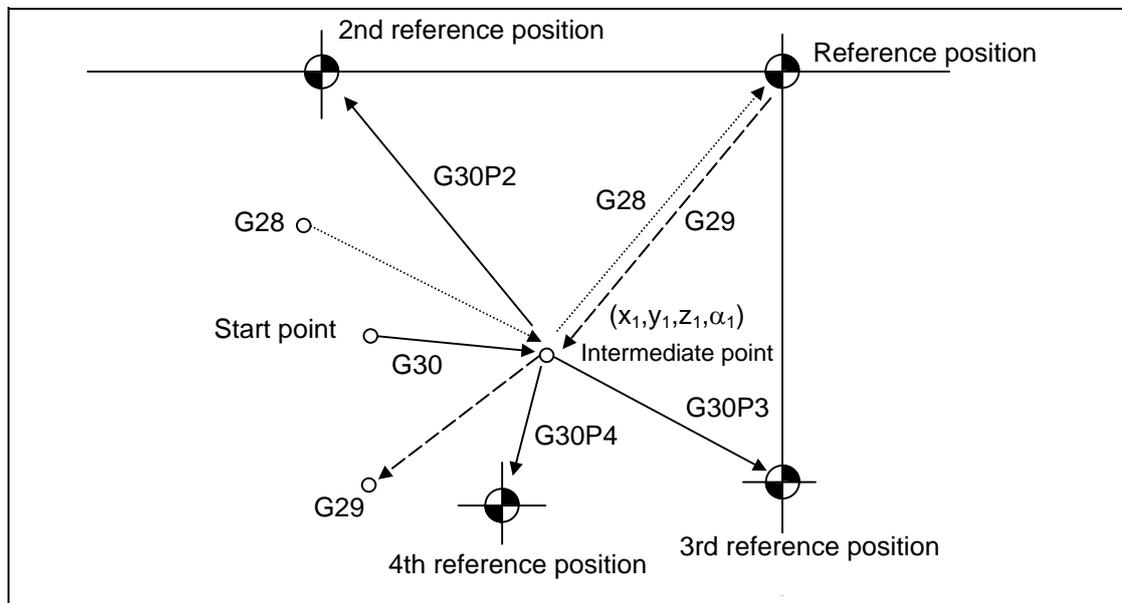
### 14.8 2nd, 3rd and 4th Reference (Zero) Position Return; G30

#### 14.8 2nd, 3rd and 4th Reference (Zero) Position Return; G30



##### Function and purpose

The tool can return to the second, third, or fourth reference (zero) position by specifying G30 P2 (P3 or P4).



##### Command format

**G30 P2 (P3, P4) Xx<sub>1</sub> Yy<sub>1</sub> Zz<sub>1</sub> αα<sub>1</sub>;**

αα<sub>1</sub> : Additional axis

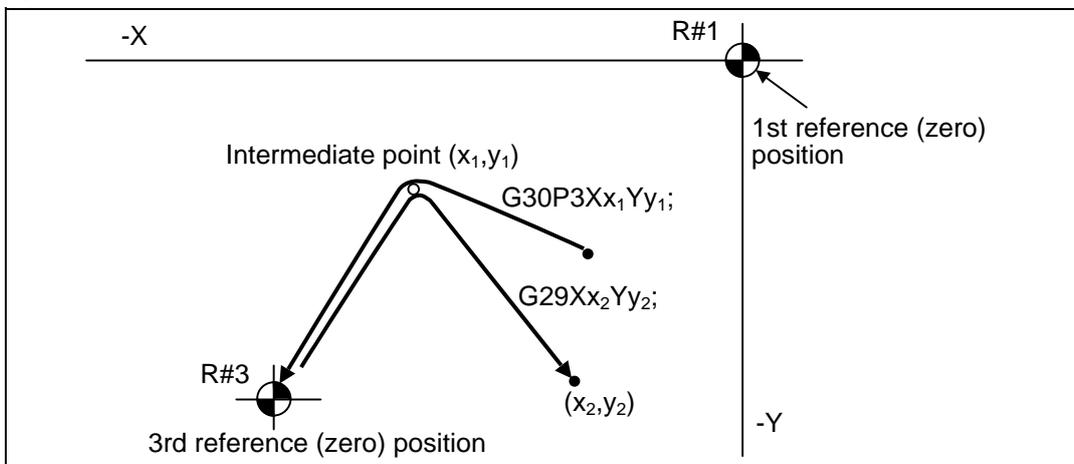
## 14. Coordinates System Setting Functions

### 14.8 2nd, 3rd and 4th Reference (Zero) Position Return; G30

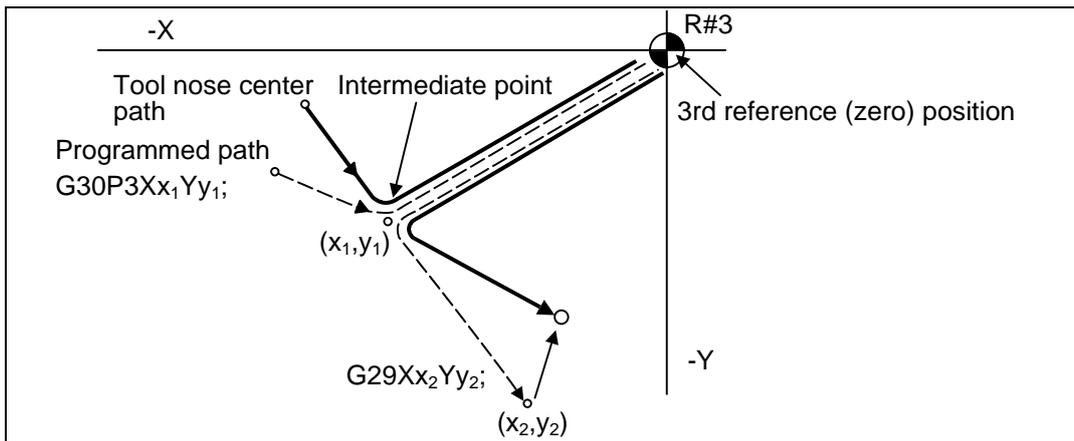


#### Detailed description

- (1) The second, third, or fourth reference (zero) position return is specified by P2, P3, or P4. A command without P or with P0, P1, P5 or a greater P number is ignored, returning the tool to the second reference (zero) position.
- (2) In the second, third, or fourth reference (zero) position return mode, as in the first reference (zero) position return mode, the tool returns to the second, third, or fourth reference (zero) position via the intermediate point specified by G30.
- (3) The second, third, and fourth reference (zero) positions coordinates refer to the positions specific to the machine, and these can be checked with the setting and display unit.
- (4) If G29 is specified after completion of returning to the second, third, and fourth reference (zero) positions, the intermediate position used last is used as the intermediate position for returning by G29.



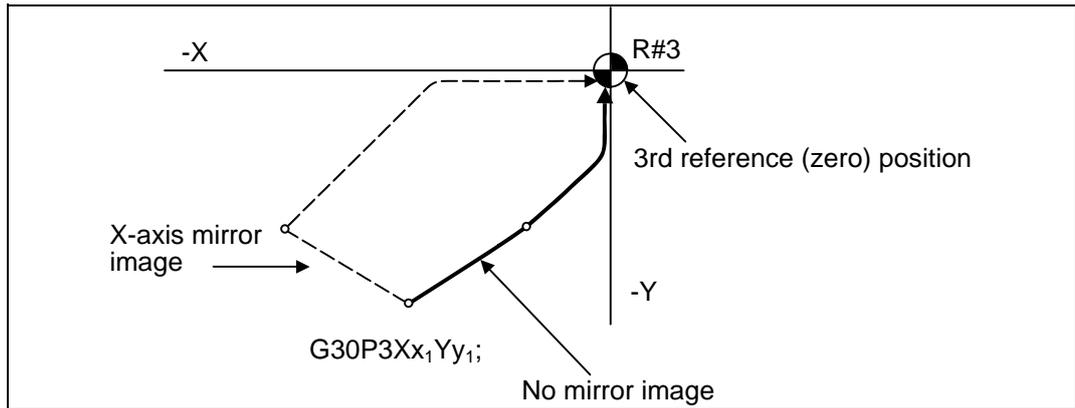
- (5) With reference (zero) position return on a plane during compensation, the tool moves without tool radius compensation from the intermediate point. With a subsequent G29 command, the tool moves with tool radius compensation until the G29 command from the intermediate point.



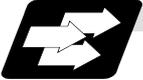
## 14. Coordinates System Setting Functions

### 14.8 2nd, 3rd and 4th Reference (Zero) Position Return; G30

- (6) The tool length compensation amount for the axis involved is canceled after the second, third and fourth reference (zero) position returns.
- (7) With second, third and fourth reference (zero) position returns in the machine lock status, control from the intermediate point to the reference (zero) position will be ignored. When the designated axis reaches as far as the intermediate point, the next block will be executed.
- (8) With second, third and fourth reference (zero) position returns in the mirror image mode, mirror image will be valid from the start point to the intermediate point and the tool will move in the opposite direction to that of the command. However, mirror image is ignored from the intermediate point to the reference (zero) position and the tool moves to the reference (zero) position.



## 14.9 Reference Position Check; G27



## Function and purpose

This command first positions the tool at the position assigned by the program and then, if that positioning point is the first reference position, it outputs the reference position arrival signal to the machine in the same way as with the G28 command. Therefore, when a machining program is prepared so that the tool will depart from the first reference position and return to the first reference position, it is possible to check whether the tool has returned to the reference position after the program has been run.



## Command format

<b>G27 X__ Y__ Z__ P__ ;</b>	
G27	: Check command
X Y Z	: Return control axis
P	: Check number
	P1 : 1st reference position check
	P2 : 2nd reference position check
	P3 : 3rd reference position check
	P4 : 4th reference position check



## Detailed description

- (1) If the P command has been omitted, the first reference position will be checked.
- (2) The number of axes whose reference positions can be checked simultaneously depends on the number of axes which can be controlled simultaneously.  
Note that the display shows one axis at a time from the final axis.
- (3) An alarm will occur if the reference position is not reached after the command is completed.

#### 14.10 Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1)



##### Function and purpose

- (1) The workpiece coordinate systems facilitate the programming on the workpiece, serving the reference position of the machining workpiece as the zero point.
- (2) These commands enable the tool to move to the positions in the workpiece coordinate system. There are 6 workpiece coordinate systems which are used by the programmer for programming. (G54 to G59)  
In addition to the six sets of workpiece coordinate systems between G54 and G59, there are 48 or 96 additional workpiece coordinate system sets. (The 48 sets and 96 sets are optional specifications.)
- (3) By these commands, the workpiece coordinate system will be re-set so that the present position of the tool on the current workpiece coordinate system become the commanded coordinates. (The "present position of the tool" includes the compensation amounts for tool radius, tool length and tool position.)
- (4) An imaginary machine coordinate system with coordinates which have been commanded by the present position of the tool is set by this command.  
(The "present position of the tool" includes the compensation amounts for tool diameter, tool length and tool position compensation.) (G54, G92)



##### Command format

- (1) Workpiece coordinate system selection (G54 to G59)

**(G90) G54 X\_\_ Y\_\_ Z\_\_ α\_\_;**

α : Additional axis

- (2) Workpiece coordinate system setting (G54 to G59)

**(G54) G92 X\_\_ Y\_\_ Z\_\_ α\_\_;**

α : Additional axis

- (3) Extended workpiece coordinate system selection (P1 to P48 or P1 to P96)

**G54.1 Pn ;**

- (4) Extended workpiece coordinate system setting (P1 to P48 or P1 to P96)

**G54.1 Pn ;**

**G92 X\_\_ Y\_\_ Z\_\_ ;**

- (5) Extended workpiece coordinate system offset amount setting (P1 to P48 or P1 to P96)  
When the designated extended workpiece coordinate system offset amount is rewritten

**G10 L20 Pn X\_\_ Y\_\_ Z\_\_ ;**

When the extended workpiece coordinate system is selected, and the offset amount is rewritten

**G10 G54.1 Pn X\_\_ Y\_\_ Z\_\_ ;**

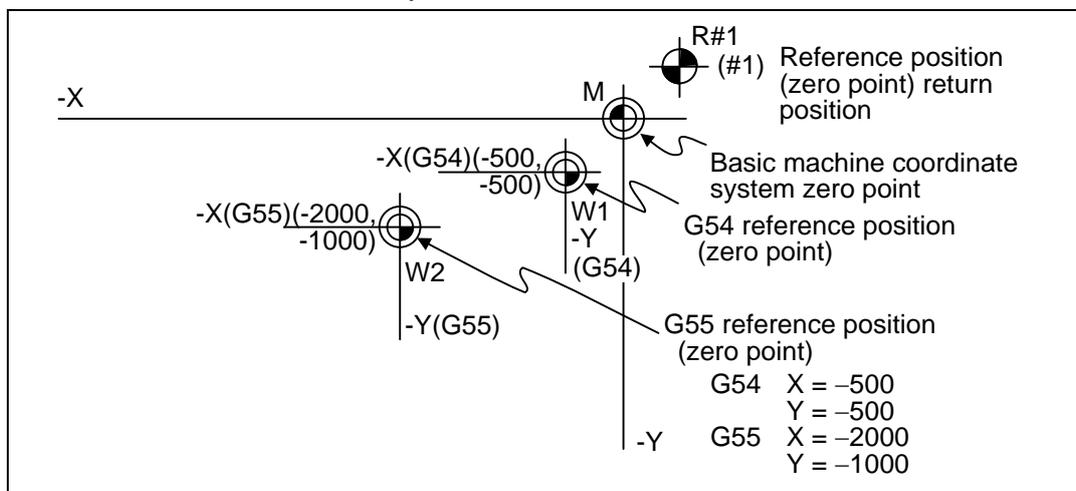
## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset



#### Detailed description

- (1) The tool radius compensation amounts for the commanded axes will not be canceled even if workpiece coordinate system is switched with any of the G54 through G59 or G54.1P1 through G54.1P96 commands
- (2) The G54 workpiece coordinate system is selected when the power is switched ON.
- (3) Commands G54 through G59 and G54.1P1 through G54.1P96 are modal commands (group 12).
- (4) The coordinate system will move with G92 in a workpiece coordinate system.
- (5) The offset setting in a workpiece coordinate system denotes the distance from the zero point of the basic machine coordinate system.



## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

- (6) The offset settings of workpiece coordinate systems can be changed any number of times. (They can also be changed by G10 L2(L20) Pp1 Xx1 Zz1.)

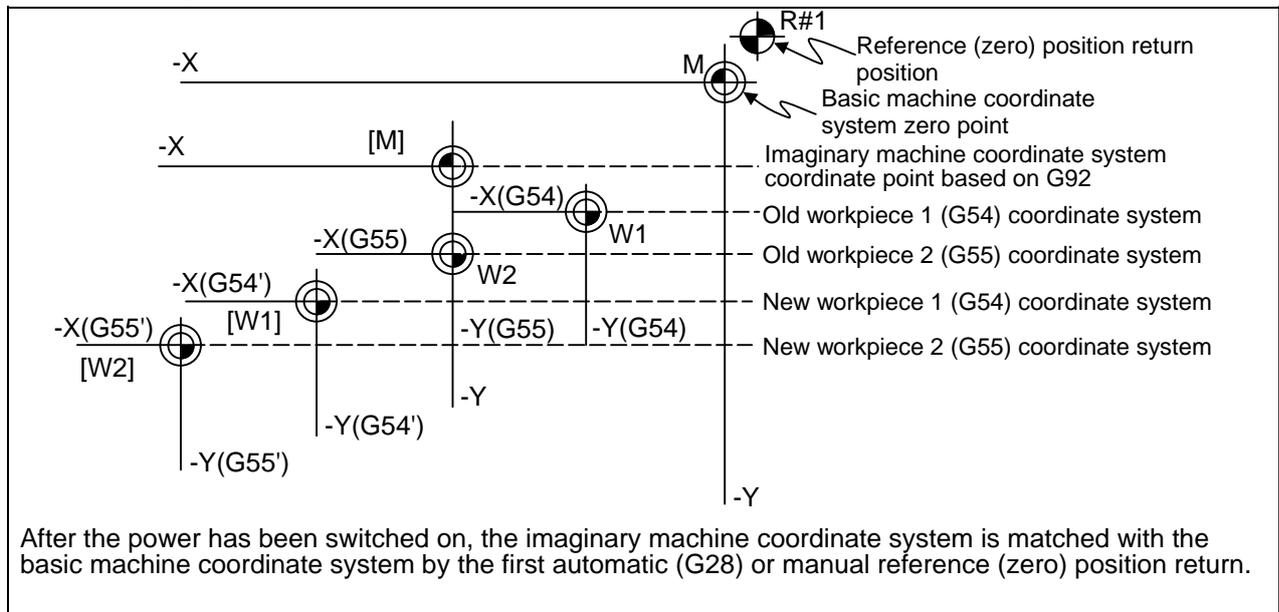
Handling when L or P is omitted

G10 L2 Pn Xx Yy Zz ;	n=0 : Set the offset amount in the external workpiece coordinate system. n=1 to 6 : Set the offset amount in the designated workpiece coordinate system. Others : The program error (P35) will occur.
G10 L2 Xx Yy Zz ;	Set the offset amount in the currently selected workpiece coordinate system. When in G54.1 modal, the program error (P33) will occur.
G10 L20 Pn Xx Yy Zz ;	n=1 to 96 : Set the offset amount in the designated workpiece coordinate system. Others : The program error (P35) will occur.
G10 L20 Xx Yy Zz ;	Set the offset amount in the currently selected workpiece coordinate system. When in G54 to G59 modal, the program error (P33) will occur.
G10 Pn Xx Yy Zz ;	Set the offset amount in the designated coordinate system No. by P code. When the currently selected coordinate system is G54 to G59, P1 to P6 corresponds to G54 to G59 respectively. When the external coordinate system is selected, P No. corresponds to G54.1 P1 to P96. If other values are set, the program error (P35) will occur.
G10 Xx Yy Zz ;	Set the offset amount in the currently selected coordinate system.
G10 G54.1 Xx Yy Zz ;	When there is no P code in the same block as G54.1, the program error (P33) will occur.

## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

- (7) A new workpiece coordinate system 1 is set by issuing the G92 command in the G54 (workpiece coordinate system 1) mode. At the same time, the other workpiece coordinate systems 2 through 6 (G55 to G59) will move in parallel and new workpiece coordinate systems 2 through 6 will be set.
- (8) An imaginary machine coordinate system is formed at the position which deviates from the new workpiece reference (zero) position by an amount equivalent to the workpiece coordinate system offset amount.



- (9) By setting the imaginary basic machine coordinate system, the new workpiece coordinate system will be set at a position which deviates from that imaginary basic machine coordinate system by an amount equivalent to the workpiece coordinate system offset amount.
- (10) When the first automatic (G28) or manual reference (zero) position return is completed after the power has been switched on, the basic machine coordinate system and workpiece coordinate systems are set automatically in accordance with the parameter setting.
- (11) If G54X-Y-; is commanded after the reference position return (both automatic or manual) executed after the power is turned ON, the program error (P62) will occur. (A speed command is required as the movement will be controlled with the G01 speed.)
- (12) Do not command a G code for which a P code is used in the same block as G54.1. The P code will be used in the prioritized G command.
- (13) When number of workpiece offset sets additional specifications is not added, the program error (P39) will occur when the G54.1 command is executed. This error will also occur when one of P49 to P96 is commanded although the specifications allow up to the 48 sets.
- (14) When number of workpiece offset sets additional specifications is not added, the program error (P172) will occur when the G10 L20 command is executed.
- (15) The local coordinate system cannot be used during G54.1 modal. The program error (P438) will occur when the G52 command is executed during G54.1 modal.

## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

- (16) A new workpiece coordinate system P1 can be set by commanding G92 in the G54.1 P1 mode. However, the workpiece coordinate system of the other workpiece coordinate systems G54 to G59, G54.1 and P2 to P48 will move in parallel with it, and a new workpiece coordinate system will be set.
- (17) The offset amount of the extended workpiece coordinate system is assigned to the variable number as shown in Table 1.
- (18) When the parameter "#1151 Reset ini" is OFF, the modal of G54.1 command will be retained even if the reset 1 is carried out.
- (19) P address (coordinate system selection) of G54.1 cannot be commanded alone even in G54.1 modal.  
Even if commanded, the designated extended workpiece coordinate system cannot be selected.
- (Ex) P54.1 P5;                      Changed to P5 workpiece coordinate system.  
      P3;                                Ignored.  
      G92 X0 Y0 Z0;                The current position becomes the zero point of P5 workpiece coordinate system.
- (20) When G92 is commanded in the extended workpiece coordinate system, the coordinate system will be sifted.

Table 1 Variable numbers of the extended workpiece coordinate offset system

	1st axis to nth axis						
P 1	#7001 to #700n	P25	#7481 to #748n	P49	#7961 to #796n	P73	#7001 to #700n
P 2	#7021 to #702n	P26	#7501 to #750n	P50	#7981 to #798n	P74	#7021 to #702n
P 3	#7041 to #704n	P27	#7521 to #752n	P51	#8001 to #800n	P75	#7041 to #704n
P 4	#7061 to #706n	P28	#7541 to #754n	P52	#8021 to #802n	P76	#7061 to #706n
P 5	#7081 to #708n	P29	#7561 to #756n	P53	#8041 to #804n	P77	#7081 to #708n
P 6	#7101 to #710n	P30	#7581 to #758n	P54	#8061 to #806n	P78	#7101 to #710n
P 7	#7121 to #712n	P31	#7601 to #760n	P55	#8081 to #808n	P79	#7121 to #712n
P 8	#7141 to #714n	P32	#7621 to #762n	P56	#8101 to #810n	P80	#7141 to #714n
P 9	#7161 to #716n	P33	#7641 to #764n	P57	#8121 to #812n	P81	#7161 to #716n
P10	#7181 to #718n	P34	#7661 to #766n	P58	#8141 to #814n	P82	#7181 to #718n
P11	#7201 to #720n	P35	#7681 to #768n	P59	#8161 to #816n	P83	#7201 to #720n
P12	#7221 to #722n	P36	#7701 to #770n	P60	#8181 to #818n	P84	#7221 to #722n
P13	#7241 to #724n	P37	#7721 to #772n	P61	#8201 to #820n	P85	#7241 to #724n
P14	#7261 to #726n	P38	#7741 to #774n	P62	#8221 to #822n	P86	#7261 to #726n
P15	#7281 to #728n	P39	#7761 to #776n	P63	#8241 to #824n	P87	#7281 to #728n
P16	#7301 to #730n	P40	#7781 to #778n	P64	#8261 to #826n	P88	#7301 to #730n
P17	#7321 to #732n	P41	#7801 to #780n	P65	#8281 to #828n	P89	#7321 to #732n
P18	#7341 to #734n	P42	#7821 to #782n	P66	#8301 to #830n	P90	#7341 to #734n
P19	#7361 to #736n	P43	#7841 to #784n	P67	#8321 to #832n	P91	#7361 to #736n
P20	#7381 to #738n	P44	#7861 to #786n	P68	#8341 to #834n	P92	#7381 to #738n
P21	#7401 to #740n	P45	#7881 to #788n	P69	#8361 to #836n	P93	#7401 to #740n
P22	#7421 to #742n	P46	#7901 to #790n	P70	#8381 to #838n	P94	#7421 to #742n
P23	#7441 to #744n	P47	#7921 to #792n	P71	#8401 to #840n	P95	#7441 to #744n
P24	#7461 to #746n	P48	#7941 to #794n	P72	#8421 to #842n	P96	#7461 to #746n

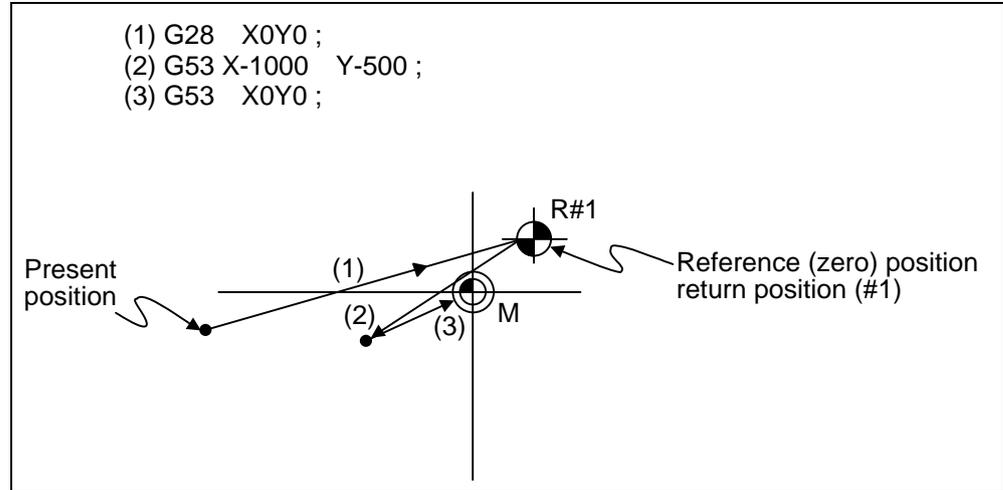
### CAUTION

 If the workpiece coordinate system offset amount is changed during single block stop, the new setting will be valid from the next block.



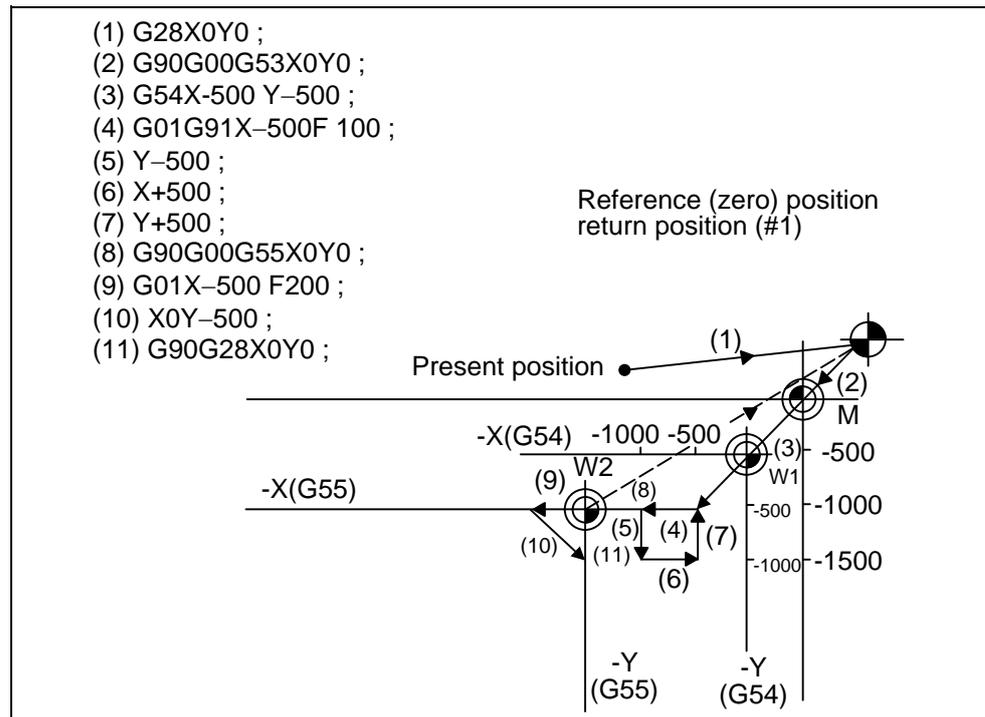
Example of program

(Example 1)



When the first reference position coordinate is zero, the basic machine coordinate system zero point and reference (zero) position return position (#1) will coincide.

(Example 2)

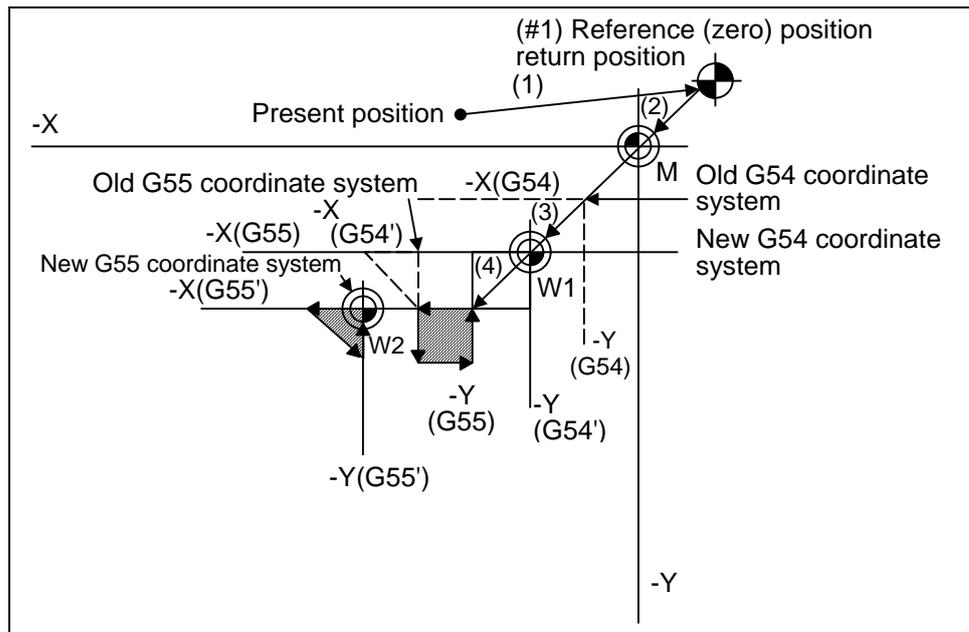


## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

**(Example 3)** When workpiece coordinate system G54 has shifted (-500, -500) in example 2 (It is assumed that (3) through (10) in example 2 have been entered in subprogram 01111.)

(1) G28 X0 Y0 ;	
(2) G90 G53 X0 Y0 ;	(This is not required when there is no G53 offset.)
(3) G54 X -500Y-500 ;	Amount by which workpiece coordinate system deviates
(4) G92 X0 Y0 ;	New workpiece coordinate system is set.
(5) G22 L1111 ;	



**(Note)** The workpiece coordinate system will shift each time steps (3) through (5) are repeated. The reference position return (G28) command should therefore be issued upon completion of the program.

## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

**(Example 4)** When six workpieces are placed on the same coordinate system of G54 to G59, and each is to be machined with the same machining.

#### (1) Setting of workpiece offset data

Workpiece1	X = -100.000	Y = -100.000	.....	G54
2	X = -100.000	Y = -500.000	.....	G55
3	X = -500.000	Y = -100.000	.....	G56
4	X = -500.000	Y = -500.000	.....	G57
5	X = -900.000	Y = -100.000	.....	G58
6	X = -900.000	Y = -500.000	.....	G59

#### (2) Machining program (subprogram)

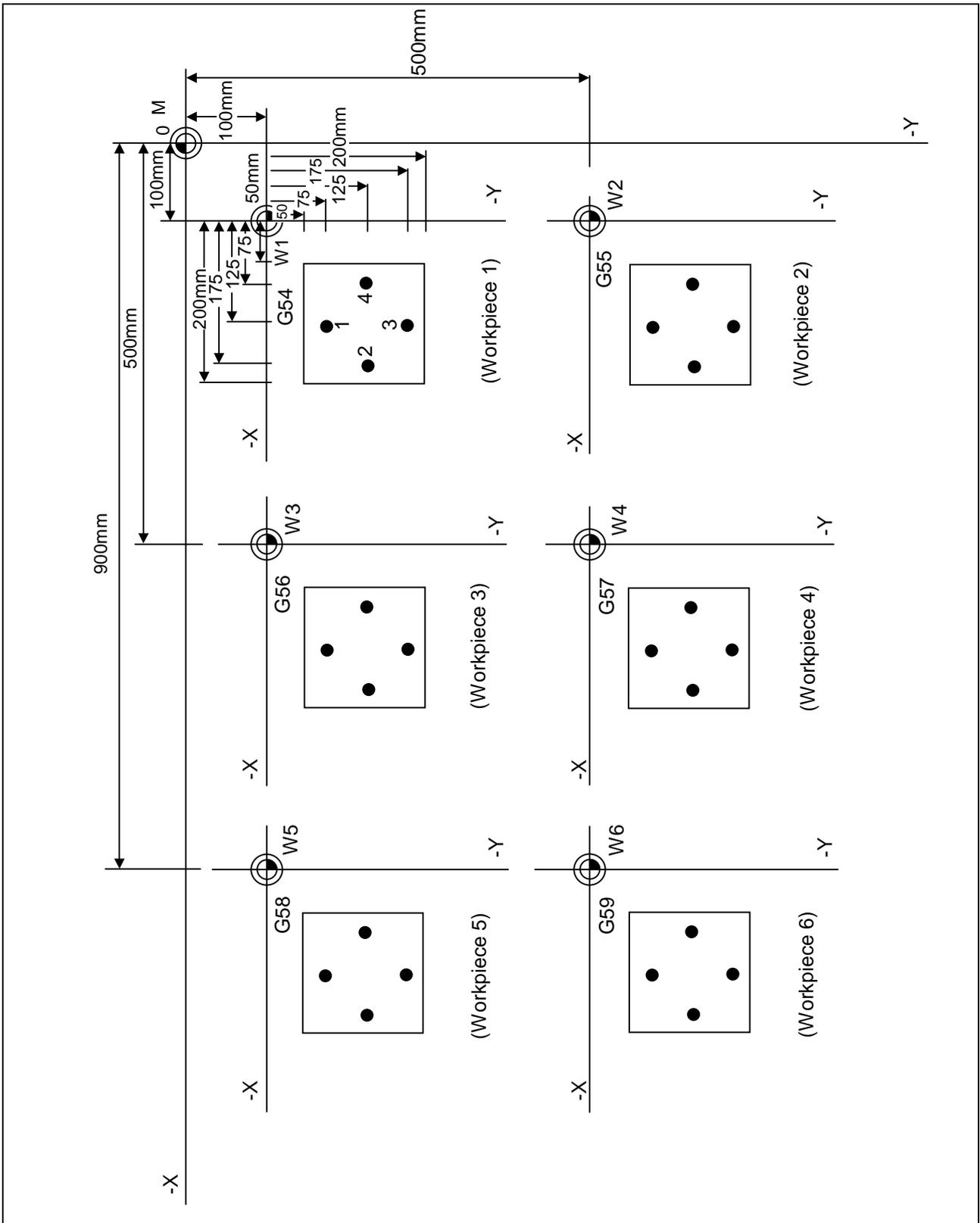
L100;		
N1 G90 G0 G43X-50. Y-50. Z-100. H10;	Positioning	
N2 G01 X-200. F50;	} Surface cutting	
Y-200. ;		
X- 50. ;		
Y- 50. ;		
N3 G28 X0 Y0 Z0 ;		
}		
N4 G98 G81 X-125. Y-75. Z-150. R-100. F40;	} Drilling	1
X-175. Y-125. ;		2
X-125. Y-175. ;		3
X- 75. Y-125. ;		4
G80;		
N5 G28 X0 Y0 Z0 ;		
}		
N6 G98 G84 X-125. Y-75. Z-150. R-100. F40 ;	} Tapping	1
X-175. Y-125. ;		2
X-125. Y-175. ;		3
X- 75. Y-125. ;		4
G80;		
G23;		

#### (3) Positioning program (main)

G28 X0 Y0 Z0 ;	} When power is turned ON
N1 G90 G54 G22 L100 ;	
N2 G55 G22 L100 ;	
N3 G57 G22 L100 ;	
N4 G56 G22 L100 ;	
N5 G58 G22 L100 ;	
N6 G59 G22 L100 ;	
N7 G28 X0 Y0 Z0 ;	
N8 M02 ;	
%	

# 14. Coordinates System Setting Functions

## 14.10 Workpiece Coordinate System Setting and Offset

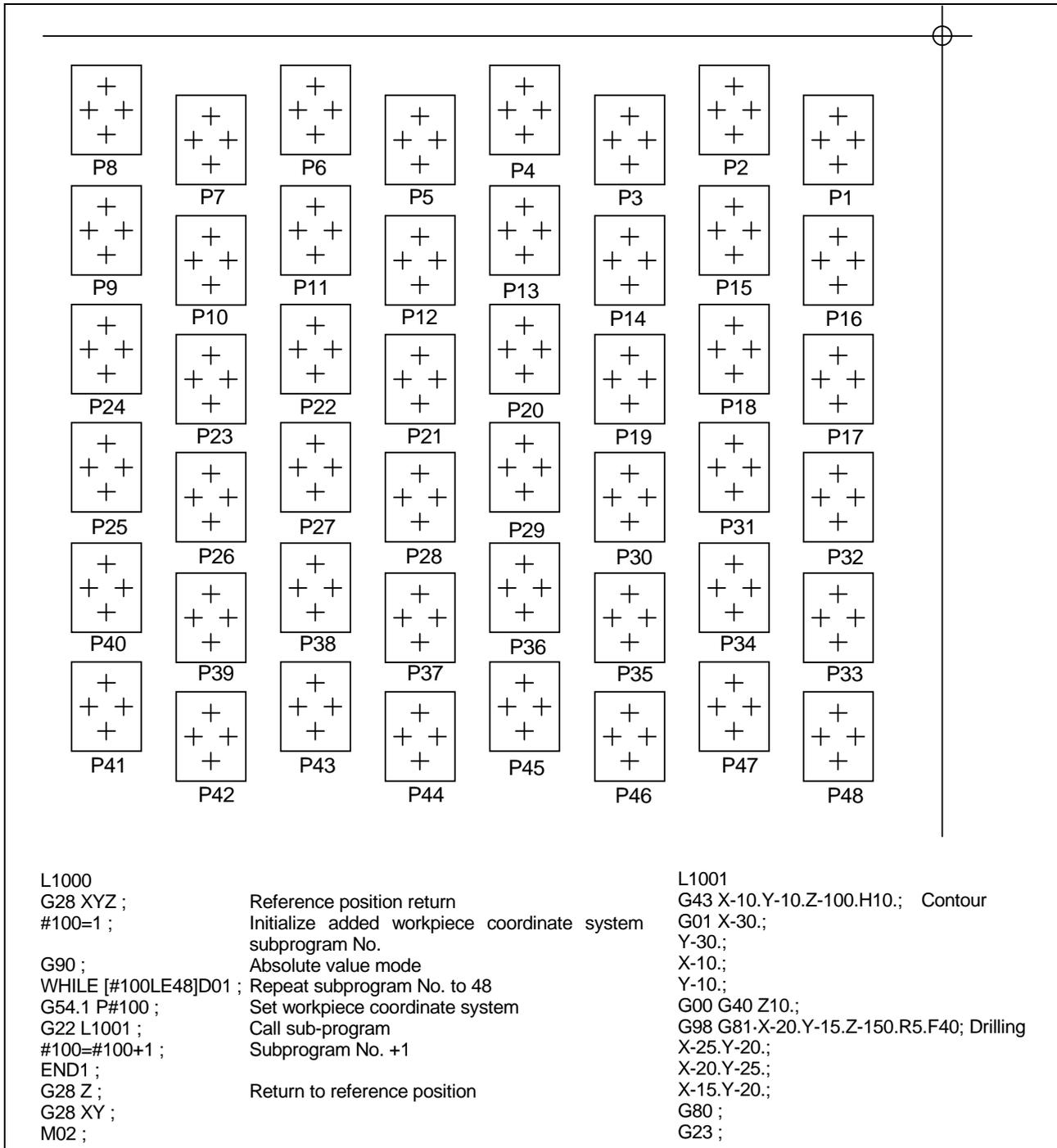


## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

**(Example 5)** Program example when continuously using 48 sets of added workpiece coordinate system offsets.

In this example, the offsets for each workpiece are set beforehand in P1 to P48 when 48 workpieces are fixed on a table, as shown in the drawing below.



## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

**(Example 6)** Program example when the added workpiece coordinate system offsets are transferred to the standard workpiece coordinate system offsets and used.  
In this example, the workpiece coordinate system offsets for each workpiece are set beforehand in P1 to P24 when the workpiece is fixed on a rotating table, as shown in the drawing below.

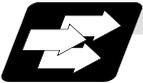
<pre>L20000 (Main) G28 XYZB ; G90 ; G00 B0 ; G65 L2001 A1 ; G22 L2002 ; G00 B90. ; G65 L2001 A7 ; G22 L2002 ; G00 B180. ; G65 L2001 A13 ; G22 L2002 ; G00 B270. ; G65 L2001 A19 ; G22 L2002 ; G28 XYB ; M02 ; %</pre>	<pre>Reference position return Absolute value mode Position table to face 1 Load workpiece offsets Drilling Position table to face 2 Position table to face 3 Position table to face 4 Return to reference position</pre>
<pre>L2001 Transmission of workpiece offsets #2=5221 ; #3=(#1-1)*20+7001 ; #5=0 ; WHILE [#5 LT 6] DO1 ; #6=#6+1 ; #7=#7+1 ; #4=#4+1 ; WHILE [#4 LT 6] DO2 ; #[#6]=#[#7] ; #6=#6+1 ; #7=#7+1 ; #4=#4+1 ; END2 ; #2=#2+20 ; #3=#3+20 ; #5=#5+1 ; END1 ; G23 ; %</pre>	<pre>Leading No. of workpiece coordinate system variables Leading No. of added workpiece coordinate system variables No. of sets counter clear Check No. of sets Set transmission source 1st axis variable No. Set transmission destination 1st axis variable No. Clear No. of axes counter Check No. of axes Transmit variable data Set transmission source next axis Set transmission destination next axis No. of axes counter +1 Transmission source Set lead of next variable set. Transmission destination Set lead of next variable set. No. of sets counter +1</pre>

## 14. Coordinates System Setting Functions

### 14.10 Workpiece Coordinate System Setting and Offset

```
L2002 (Drilling)
G54 G22 H100 ;           Drilling in G54 coordinate system
G55 G22 H100 ;           In G55
G56 G22 H100 ;           In G56
G57 G22 H100 ;           In G57
G58 G22 H100 ;           In G58
G59 G22 H100 ;           In G59
G28 Z0 ;
G23 ;
N100 G98 G81 X-20. Y-15. Z-150. R5. F40 ; Fixed cycle for drilling call
X-25. Y-20. ;
X-20. Y-25. ;
X-15. Y-20. ;
G80 ;
G28 Z ;
G23 ;
%
```

14.11 Local Coordinate System Setting; G52



Function and purpose

The local coordinate systems can be set independently on the G54 through G59 workpiece coordinate systems using the G52 command so that the commanded position serves as the programmed zero point.

The G52 command can also be used instead of the G92 command to change the deviation between the zero point in the machining program and the machining workpiece zero point.



Command format

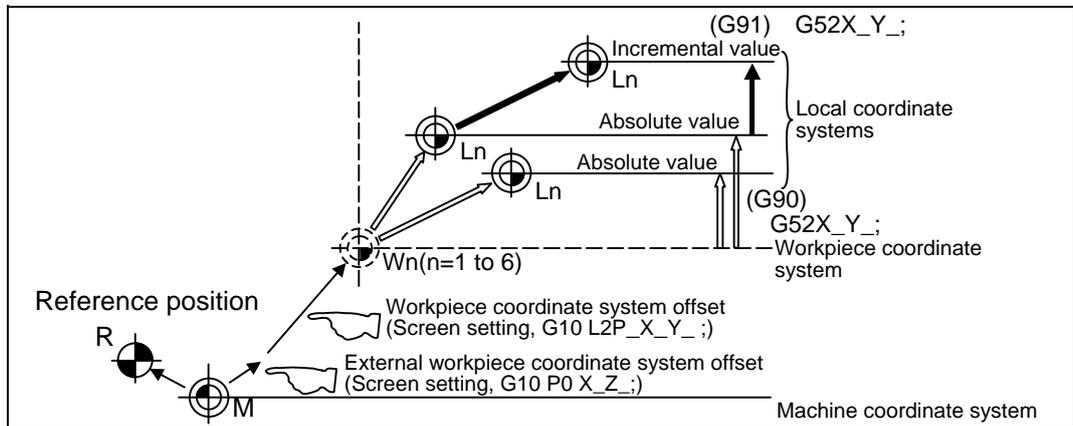
G54 (54 to G59) G52 X\_\_ Y\_\_ Z\_\_ α\_\_ ;

α : Additional axis



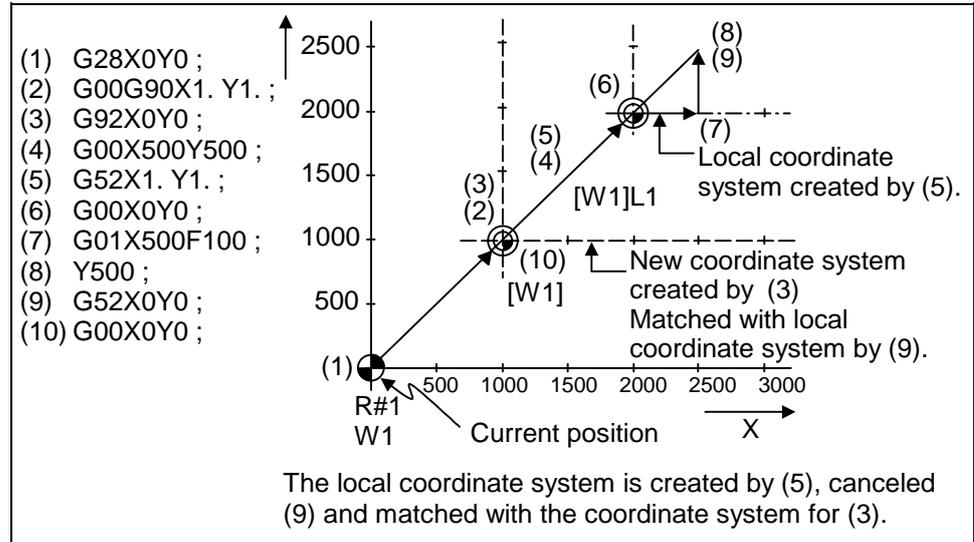
Detailed description

- (1) The G52 command is valid until a new G52 command is issued, and the tool does not move. This command comes in handy for employing another coordinate system without changing the zero point positions of the workpiece coordinate systems (G54 to G59).
- (2) The local coordinate system offset will be cleared by the dog-type manual reference (zero) position return or reference (zero) position return performed after the power has been switched ON.
- (3) The local coordinate system is canceled by (G54 to G59) G52X0 Y0 Z0 α0 ;
- (4) Coordinate commands in the absolute value (G90) cause the tool to move to the local coordinate system position.



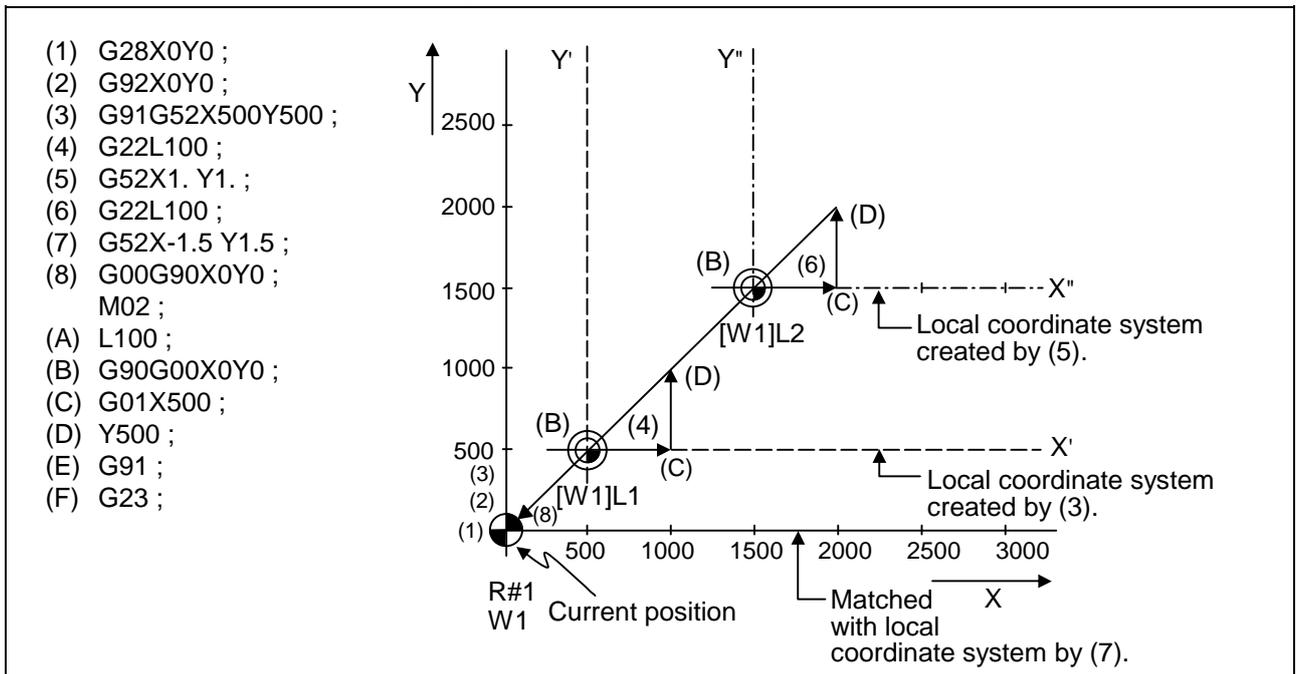
**(Note)** If the program is executed repeatedly, the workpiece coordinate system will deviate each time. Thus, when the program is completed, the reference position return operation must be commanded.

**(Example 1)** Local coordinates for absolute value mode (The local coordinate system offset is not cumulated)



**(Note)** If the program is executed repeatedly, the workpiece coordinate system will deviate each time. Thus, when the program is completed, the reference position return operation must be commanded.

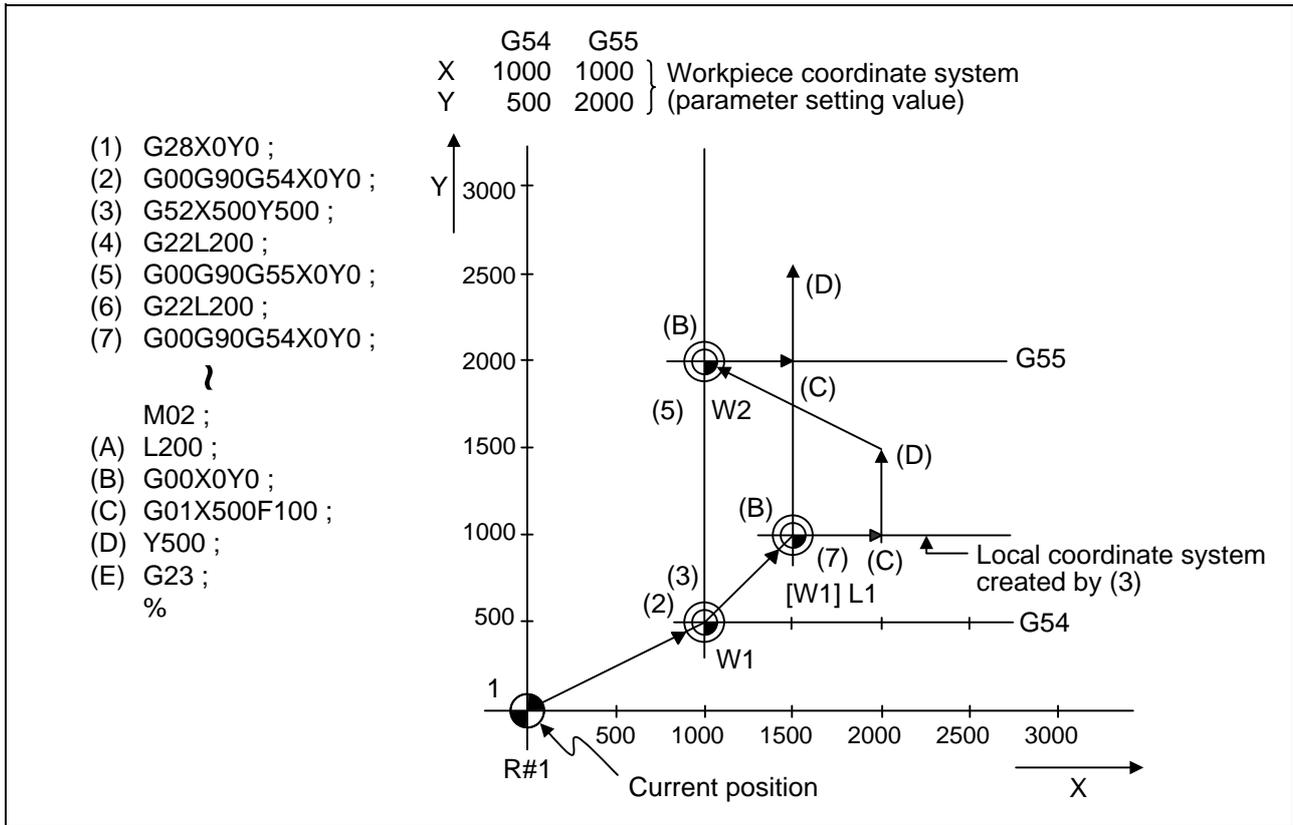
**(Example 2)** Local coordinates for incremental value mode (The local coordinate system offset is cumulated.)



**(Explanation)**

The local coordinate system X'Y' is created at the XY coordinate system (500,500) position by (3). The local coordinate system X''Y'' is created at the X'Y' coordinate system (1000,1000) position by (5). The local coordinate system is created at the X''Y'' coordinate system (-1500, -1500) position by (7). In other words, the same occurs as when the local coordinate system and XY coordinate system are matched and the local coordinate system is canceled.

(Example 3) When used together with workpiece coordinate system



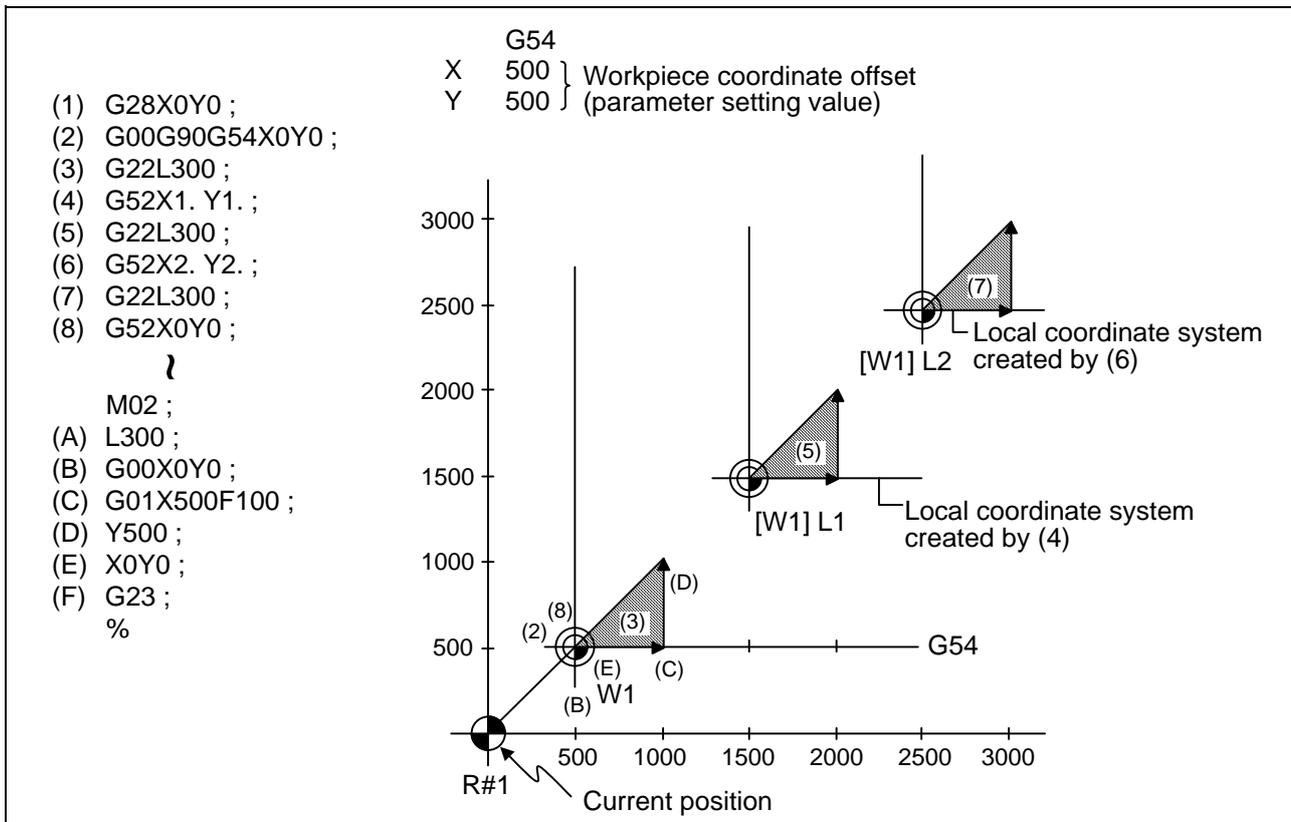
**(Explanation)**

The local coordinate system is created at the G54 coordinate system (500,500) position by (3), but the local coordinate system is not created for the G55 coordinate system.

During the movement for (7), the axis moves to the G54 local coordinate system's reference position (zero point).

The local coordinate system is canceled by G90G54G52X0Y0;.

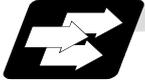
**(Example 4)** Combination of workpiece coordinate system G54 and multiple local coordinate systems



**(Explanation)**

The local coordinate system is created at the G54 coordinate system (1000,1000) position by (4).  
 The local coordinate system is created at the G54 coordinate system (2000,2000) by (6).  
 The G54 coordinate system and local coordinate system are matched by (8).

## 14.12 Workpiece Coordinate System Preset; G92.1



## Function and purpose

- (1) This function presets the workpiece coordinate system shifted with the program command during manual operation to the workpiece coordinate system offset from the machine zero point by the workpiece coordinate offset amount by the program command (G92.1).
- (2) The set workpiece coordinate system will be shifted from the machine coordinate system when the following type of operation or program command is executed.
  - When manual interrupt is executed while manual absolute is OFF
  - When movement command is issued in machine lock state
  - When axis is moved with handle interrupt
  - When operation is started with mirror image
  - When local coordinate system is set with G52
  - When workpiece coordinate system is shifted with G92

This function presets the shifted workpiece coordinate system to the workpiece coordinate system offset from the machine zero point by the workpiece coordinate offset amount. This takes place in the same manner as manual reference position return. Whether to preset the relative coordinates or not can be selected with the parameters.



## Command format

**G92.1 X0 Y0 Z0 α0 ;**

**(G50.3)**

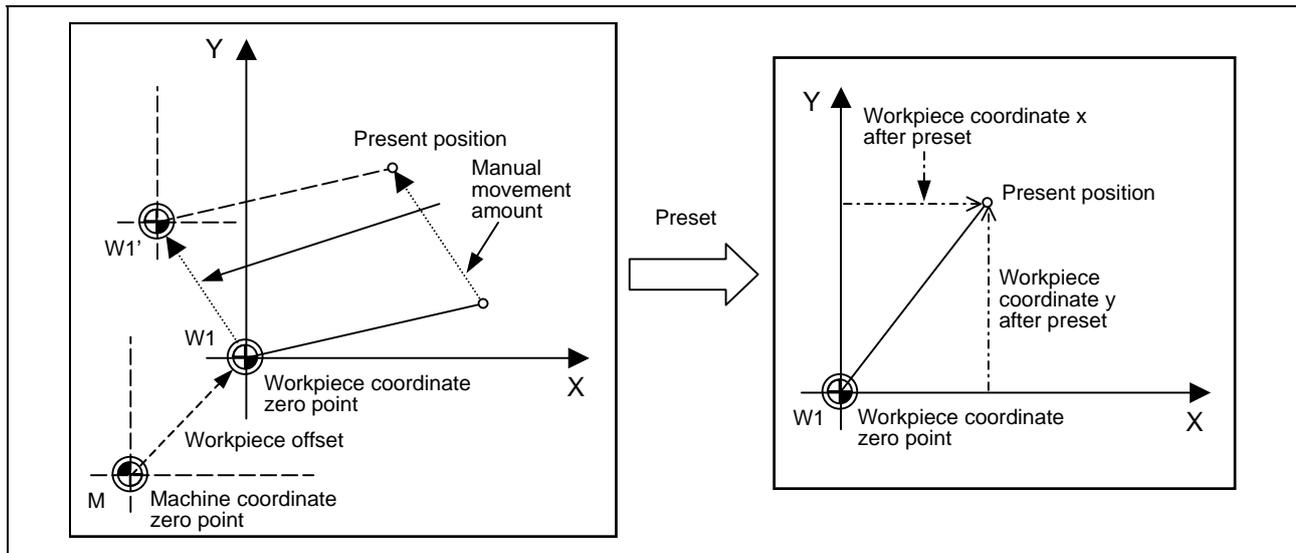
α0

Additional axis



## Detailed description

- (1) Command the address of the axis to be preset. The axis will not be preset unless commanded.
- (2) A program error (P35) will occur if a value other than "0" is commanded.
- (3) This can be commanded in the following G code lists.
  - G code list 2 to 7
  - G code list 1 when the G code changeover parameter (#1267 ext3/bit0 = 1) is set.
- (4) Depending on the G code list, the G code will be "G50.3".
- (5) When manual operation is carried out when manual absolute is set to OFF, or if the axis is moved with handle interrupt



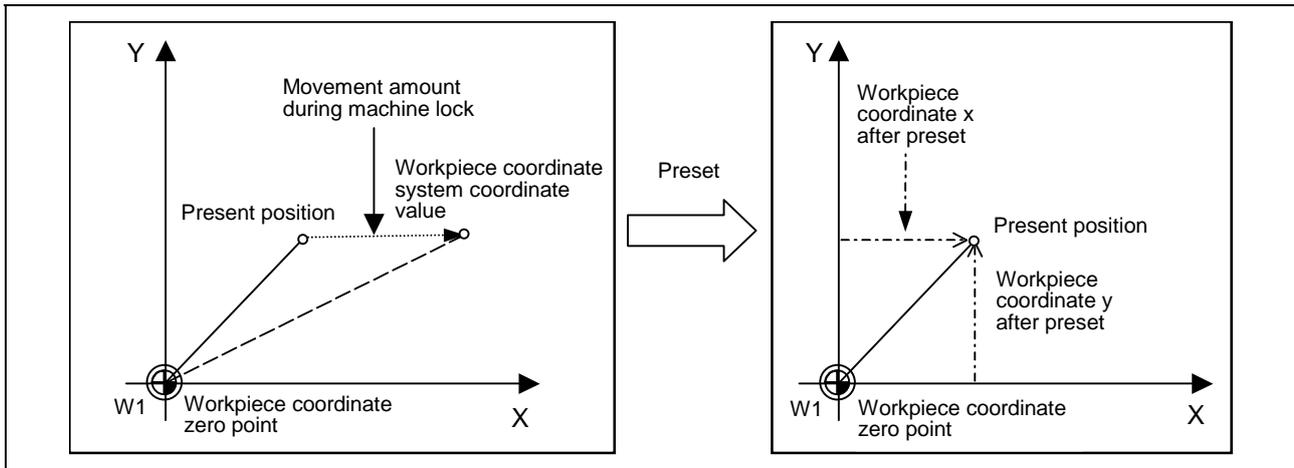
If manual operation is carried out when manual absolute is set to OFF, or if the axis is moved with handle interrupt, the workpiece coordinate system will be shifted by the manual movement amount.

This function returns the shifted workpiece coordinate zero point W1' to the original workpiece coordinate zero point W1, and sets the distance from W1 to the present position as the workpiece coordinate system's present position.

## 14. Coordinates System Setting Functions

### 14.12 Workpiece Coordinate System Preset; G92.1

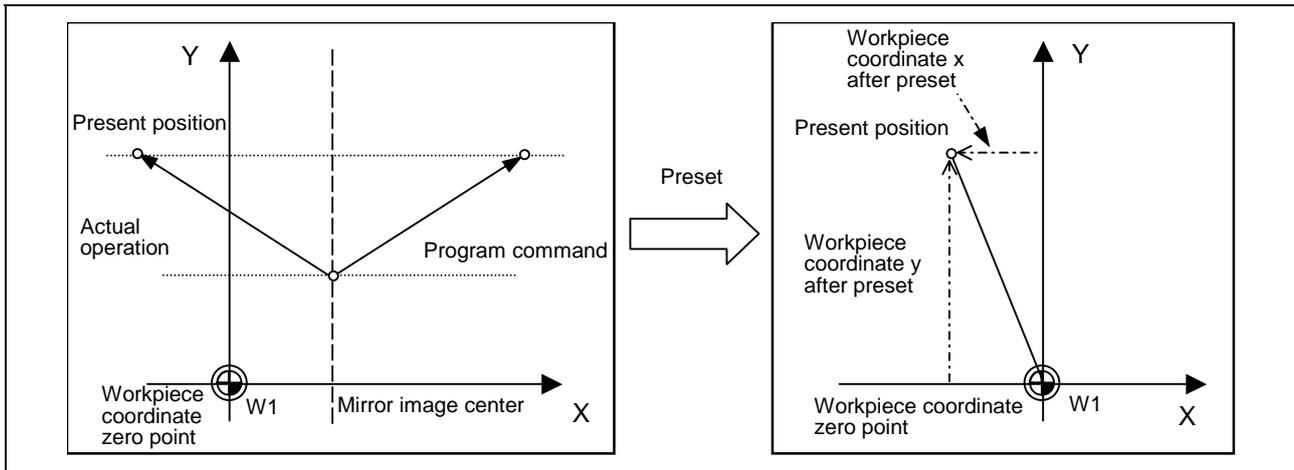
(6) When movement command is issued in machine lock state



If the movement command is issued in the machine lock state, the present position will not move, and only the workpiece coordinates will move.

This function returns the moved workpiece coordinates to the original present position, and sets the distance from W1 to the present position as the workpiece coordinate system's present position.

(7) When operation is carried out with mirror image

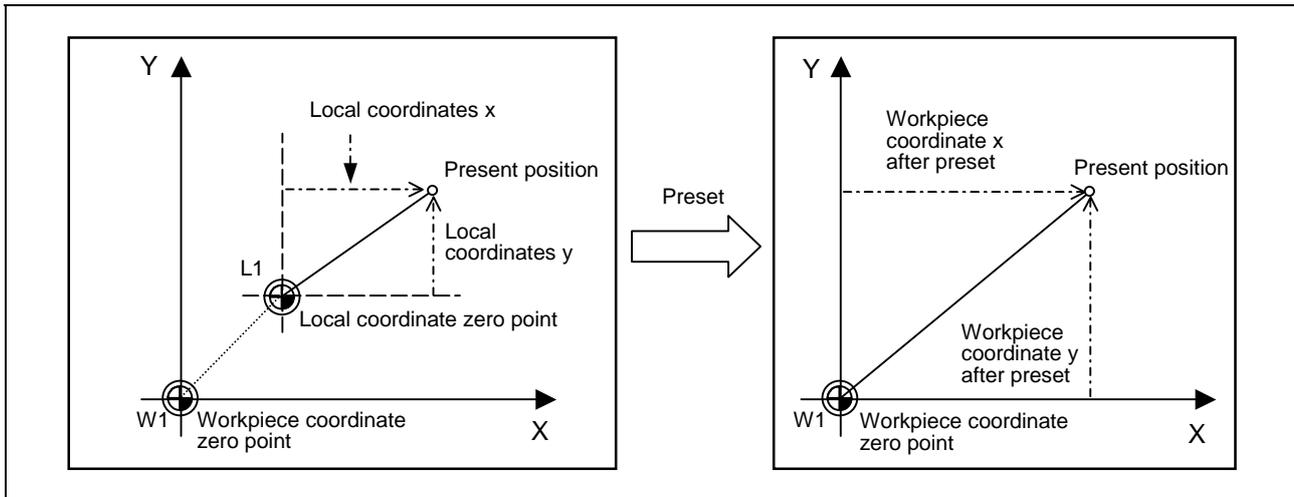


If operation is carried out with mirror image, only the NC internal coordinates are used as the program command coordinates. The other coordinates are the present position coordinates. This function sets the NC internal coordinates as the present position coordinates.

## 14. Coordinates System Setting Functions

### 14.12 Workpiece Coordinate System Preset; G92.1

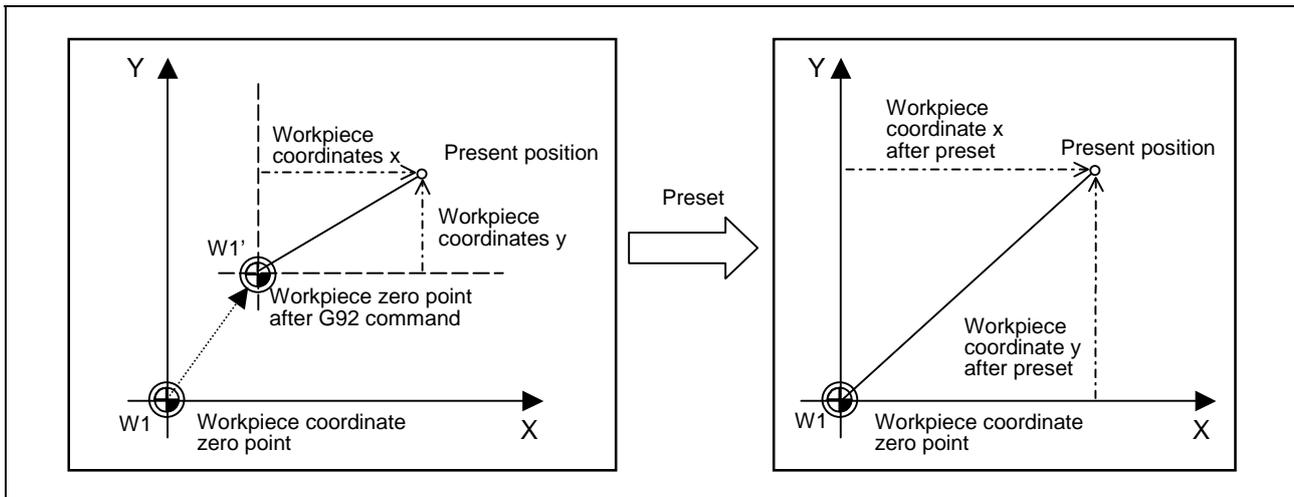
#### (8) Setting local coordinate system with G52



The local coordinate system is set with the G52 command, and the program commands, etc., are issued with the local coordinate system.

With this function, the set local coordinate system is canceled, and the program commands, etc., use the workpiece coordinate system which has W1 as the zero point. The canceled local coordinate system is only the selected workpiece coordinate system.

#### (9) Shifting the workpiece coordinate system with G92

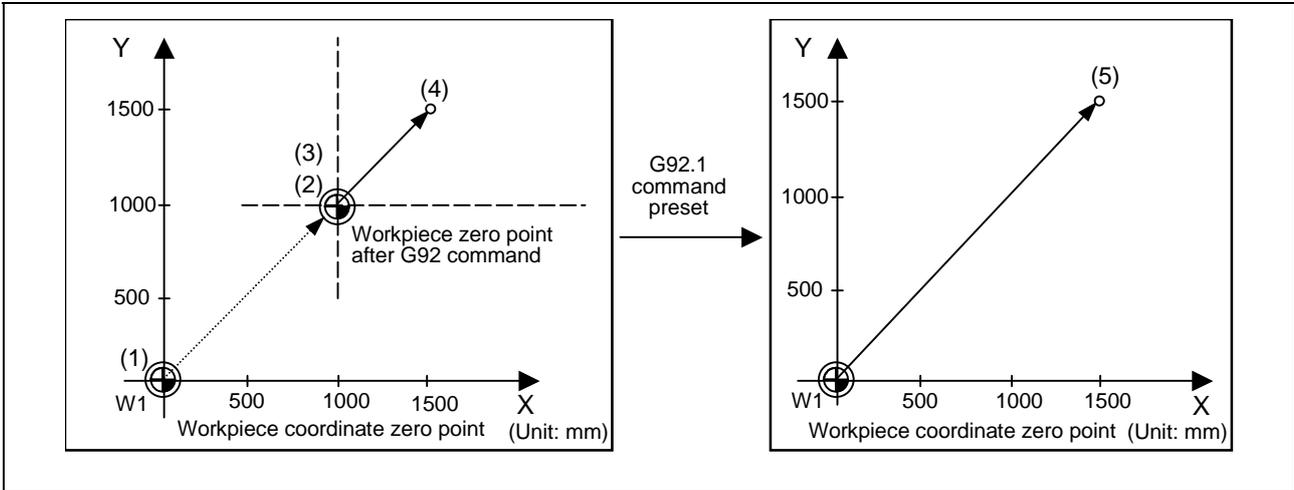


The workpiece coordinate system shifts with the G92 command, and the distance between W1' and the present position is set as the present position of the workpiece coordinate system. This function returns the shifted workpiece coordinate zero point to W1, and sets the distance from W1 to the present position as the workpiece coordinate system's present position. This is valid for all workpiece coordinate systems.



Example of program

The workpiece coordinate system shifted with G92 is preset with G92.1.



(Example)

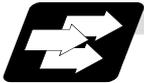
- G28 X0 Y0 ;..... (1)
- G00 G90 X1. Y1. ;..... (2)
- G92 X0 Y0 ;..... (3)
- G00 X500 Y500 ;..... (4)
- G92.1 X0 Y0 ;..... (5)



Precautions

- (1) Cancel tool radius compensation, tool length offset and tool position offset before using this function. If these are not canceled, the workpiece coordinates will be at a position obtained by subtracting the workpiece coordinate offset amount from the machine value. Thus, the compensation vector will be temporarily canceled.
- (2) This function cannot be executed while the program is being resumed.
- (3) Do not command this function during the scaling, coordinate rotation or program mirror image modes.  
A program error (P34) will occur if commanded.

## 14.13 Coordinate System for Rotary Axis



## Function and purpose

The axis designated as the rotary axis with the parameters is controlled with the rotary axis' coordinate system.

The rotary axis includes the rotating type (short-cut valid/invalid) and linear type (workpiece coordinate position linear type, all coordinate position linear type).

The workpiece coordinate position range is 0 to 359.999° for the rotating type, and 0 to ±99999.999° for the linear type.

The machine coordinate position and relative position differ according to the parameters.

The rotary axis is commanded with a degree (°) unit regardless of the inch or metric designation.

The rotary axis type can be set with the parameter "#8213 rotation axis type" for each axis.

	Rotary axis				
	Rotating type rotary axis		Linear type rotary axis		Linear axis
	Short-cut invalid	Short-cut valid	Workpiece coordinate position linear type	All-coordinate position linear type	
#8213 setting value	0	1	2	3	--
Workpiece coordinate value	Displayed in the range of 0° to 359.999°.		Displayed in the range of 0° to ±99999.999°.		
Machine coordinate value/relative position	Displayed in the range of 0° to 359.999°.			Displayed in the range of 0° to ±99999.999°.	
ABS command	The incremental amount from the end point to the current position is divided by 360, and the axis moves by the remainder amount according to the sign.	Moves with a short-cut to the end point.	In the same manner as the normal linear axis, moves according to the sign by the amount obtained by subtracting the current position from the end point (without rounding up to 360 degrees).		
INC command	Moves in the direction of the commanded sign by the commanded incremental amount starting at the current position.				
Reference position return	Until the intermediate point: Depends on the absolute command or the incremental command.				
	From the intermediate point to the reference position: Returns with movement within 360 degrees.			Moves and returns in the reference position direction by the difference from the intermediate point to the reference position.	

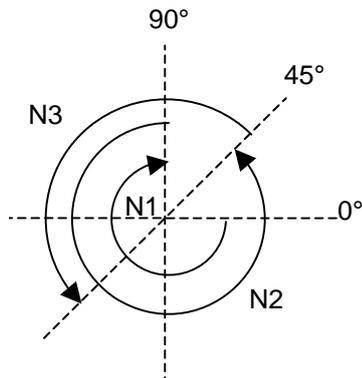


**Example of operation**

Examples of differences in the operation and counter displays according to the type of rotation coordinate are given below.  
(The workpiece offset is set as 0°.)

**(1) Rotary type (short-cut invalid)**

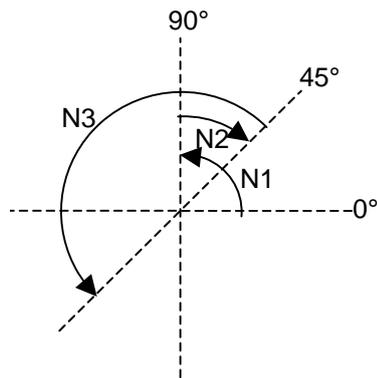
- (a) The machine coordinate position, workpiece coordinate position and current position are displayed in the range of 0 to 359.999°.
- (b) For the absolute command, the axis moves according to the sign by the excessive amount obtained by dividing by 360°.



Program	Workpiece coordinate counter	Machine coordinate counter
G28 C0.		
N1 G90 C-270.	90.000	90.000
N2 C405.	45.000	45.000
N3 G91 C180	225.000	225.000

**(2) Rotation type (short-cut valid)**

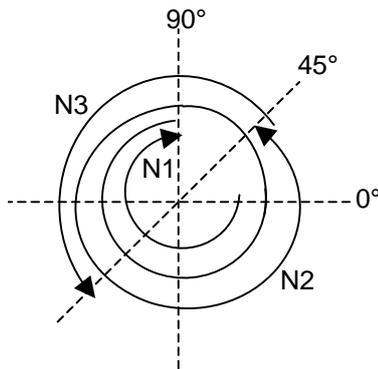
- (a) The machine coordinate position, workpiece coordinate position and current position are displayed in the range of 0 to 359.999°.
- (b) For the absolute command, the axis rotates to the direction having less amount of movement to the end point.



Program	Workpiece coordinate counter	Machine coordinate counter
G28 C0.		
N1 G90 C-270.	90.000	90.000
N2 C405.	45.000	45.000
N3 G91 C180	225.000	225.000

**(3) Linear type (workpiece coordinate position linear type)**

- (a) The coordinate position counter other than the workpiece coordinate position is displayed in the range of 0 to 359.999°. The workpiece coordinate position is displayed in the range of 0 to ±99999.999°.
- (b) The movement is the same as the linear axis.
- (c) During reference position return, the axis moves in the same manner as the linear axis until the intermediate point. The axis returns with a rotation within 360° from the intermediate point to the reference position.
- (d) During absolute position detection, even if the workpiece coordinate position is not within the range of 0 to 359.999°, the system will start up in the range of 0 to 359.999° when the power is turned ON again.



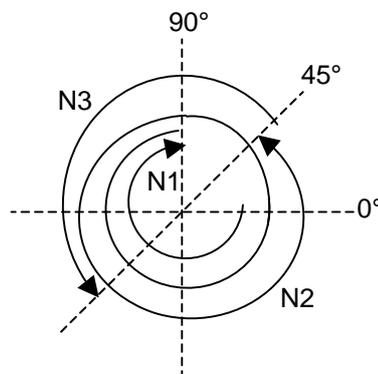
Program	Workpiece coordinate counter	Machine coordinate counter	POSITION counter
G28 C0.			
N1 G90 C-270.	-270.000	90.000	90.000
N2 C405.	405.000	45.000	45.000
N3 G91 C180	585.000	225.000	225.000

When power is turned ON again ↓

Workpiece	Machine
225.000	225.000

**(4) Linear type (all coordinate values linear type)**

- (a) The all-coordinate position counter is displayed in the range of 0 to 99999.999°.
- (b) The movement is the same as the linear axis.
- (c) During reference position return, the axis moves in the same manner as the linear axis until the intermediate point. The axis rotates by the difference from the intermediate point to the reference position and returns to the reference position.
- (d) During absolute position detection, the system starts up at the position where the power was turned OFF when the power is turned ON again.



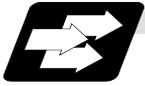
Program	Workpiece coordinate counter	Machine coordinate counter	POSITION counter
G28 C0.			
N1 G90 C-270.	-270.000	-270.000	-270.000
N2 C405.	405.000	405.000	405.000
N3 G91 C180	585.000	585.000	585.000

When power is turned ON again ↓

Workpiece	Machine
585.000	585.000

## 15. Measurement Support Functions

### 15.1 Automatic Tool Length Measurement; G37.1



#### Function and purpose

These functions issue the command values from the measuring start position as far as the measurement position, move the tool in the direction of the measurement position, stop the machine once the tool has arrived at the sensor, cause the NC system to calculate automatically the difference between the coordinate values at that time and the coordinate values of the commanded measurement position and provide this difference as the tool offset amount.

When offset is already applied to a tool, it moves the tool toward the measurement position with the offset still applied, and if a further offset amount is generated as a result of the measurement and calculation, it provides further compensation of the present offset amount.

If there is one type of offset amount at this time, and the offset amount is distinguished between tool length offset amount and wear offset amount, the wear amount will be automatically compensated.



#### Command format

**G37.1 Z\_ R\_ D\_ F\_ ;**

G37.1 : Automatic tool length measurement command

Z : Measuring axis address and coordinates of measurement position ..... X, Y, z,  $\alpha$   
(where,  $\alpha$  is the additional axis)

R : This commands the distance between the measurement position and point where the movement is to start at the measuring speed.

D : This commands the range within which the tool is to stop.

F : This commands the measuring feedrate.

When R\_, D\_ of F\_ is omitted, the value set in the parameter is used instead.

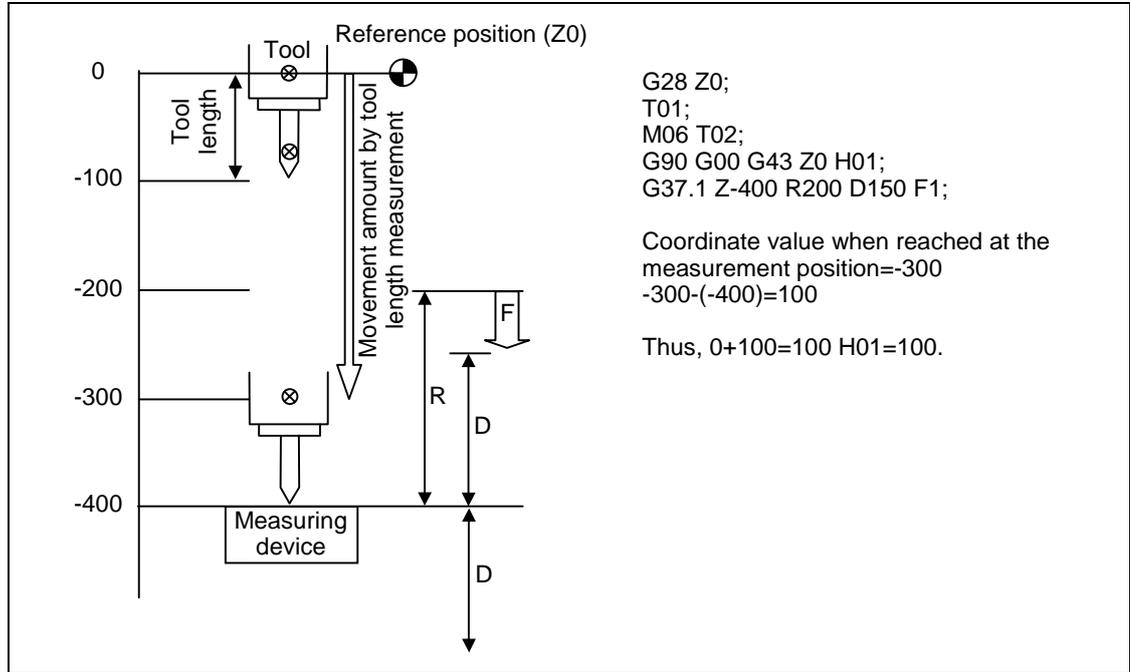
<Parameter> ("TLM" on machining parameter screen)

- #8004 SPEED (measuring feedrate) : 0 to 1000000 (mm/min)
- #8005 ZONE r (deceleration range) : 0 to 99999.999 (mm)
- #8006 ZONE d (measurement range) : 0 to 99999.999 (mm)



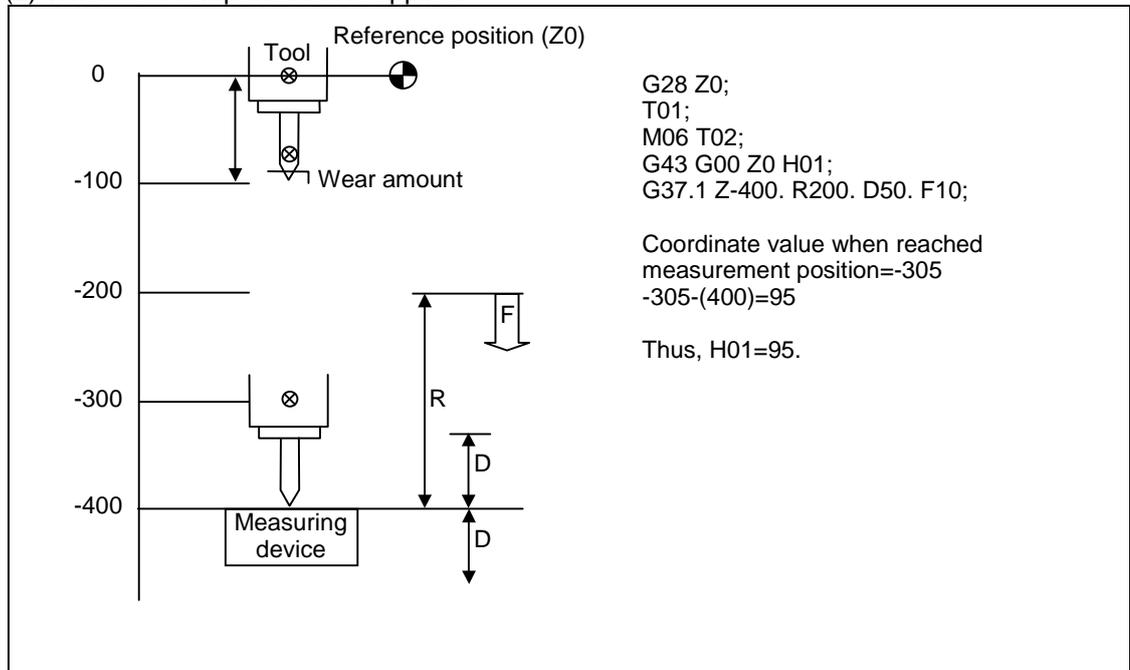
Example of execution

(1) For new measurement



**(Note)** A new measurement is applied when the current tool length compensation amount is zero. Thus, length will be compensated whether or not length dimension by tool compensation memory type and length wear are differentiated.

(2) When tool compensation is applied

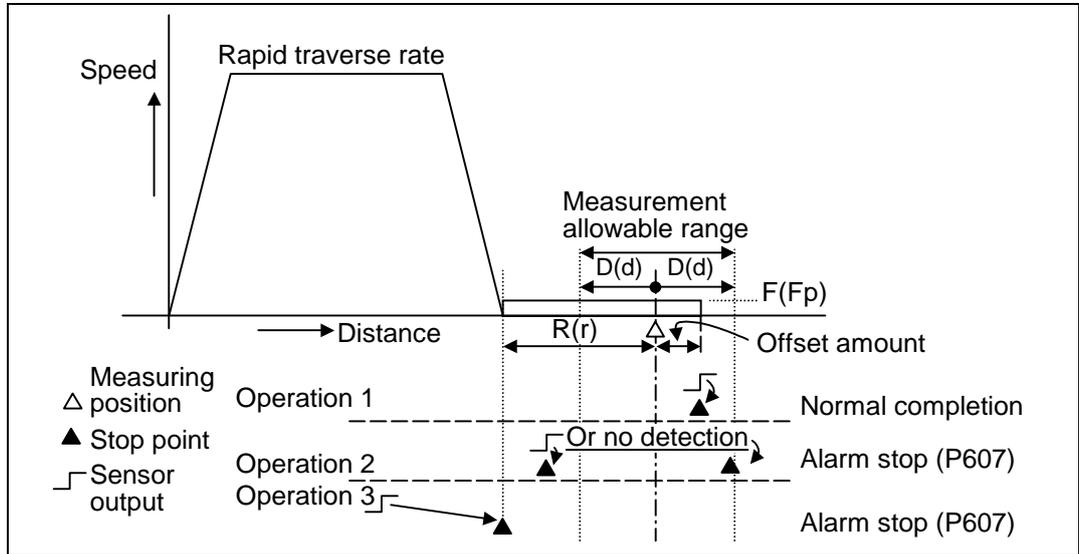


**(Note)** A measurement for the wear amount is applied when the current tool length compensation amount is other than zero. Thus, length wear will be compensated if length dimension by tool compensation memory type and length wear are differentiated. If not differentiated, length dimension will be compensated.



Detailed description

(1) Operation with G37.1 command



- (2) The sensor signal (measuring position arrival signal) is used in common with the skip signal.
- (3) The feedrate will be 1mm/min if the F command and parameter measurement speed are 0.
- (4) An updated offset amount is valid unless it is assigned from the following Z axis (measurement axis) command of the G37.1 command.
- (5) Excluding the delay at the PLC side, the delay and fluctuations in the sensor signal processing range from 0 to 0.2ms.

As a result, the measuring error shown below is caused.

$$\text{Maximum measuring error (mm)} = \text{Measuring speed (mm/min)} \cdot \frac{1}{60} \cdot \frac{0.2 \text{ (ms)}}{1000}$$

- (6) The machine position coordinates at that point in time are ready by sensor signal detection, and the machine will overtravel and stop at a position equivalent to the servo droop.

Maximum overtravel (mm)

$$= \text{Measuring speed (mm/min)} \cdot \frac{1}{60} \cdot \frac{1}{\text{Position loop gain (s}^{-1}\text{)}}$$

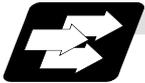
The standard position loop gain is 33 (s<sup>-1</sup>).



## Precautions

- (1) Program error (P600) results if G37.1 is commanded when the automatic tool length measurement function is not provided.
- (2) Program error (P604) results when no axis has been commanded in the G37.1 block or when two or more axes have been commanded.
- (3) Program error (P605) results when the H code is commanded in the G37.1 block.
- (4) Program error (P606) results when G43\_H is not commanded prior to the G37.1 block.
- (5) Program error (P607) results when the sensor signal was input outside the allowable measuring range or when the sensor signal was not detected even upon arrival at the end point.
- (6) When a manual interrupt is applied while the tool is moving at the measuring speed, a return must be made to the position prior to the interrupt and then operation must be resumed.
- (7) The data commanded in G37.1 or the parameter setting data must meet the following conditions:
  - | Measurement point – start point | > R address or parameter r
  - > D address or parameter d
- (8) When the D address and parameter d in (7) above are zero, operation will be completed normally only when the commanded measurement point and sensor signal detection point coincide. Otherwise, program error (P607) will result.
- (9) When the R and D addresses as well as parameters r and d in (7) above are all zero, program error (P607) will result regardless of whether the sensor signal is present or not after the tool has been positioned at the commanded measurement point.
- (10) The automatic tool length measurement command (G37.1) must be commanded together with the G43H\_ command that designates the offset No.
  - [ G43H\_;
  - [ G37.1 Z\_ R\_ D\_ F\_;

## 15.2 Skip Function; G31



## Function and purpose

When the skip signal is input externally during linear interpolation based on the G31 command, the machine feed is stopped immediately, the remaining distance is discarded and the command in the following block is executed.



## Command format

**G31 X\_\_ Y\_\_ Z\_\_ α\_\_ F\_\_ ; (where, a is the additional axis)**

X, Y, Z, α : Axis coordinates; they are commanded as absolute or incremental values according to the G90/G91 modal when commanded.

F : Feedrate (mm/min)

Linear interpolation can be executed using this function. If the skip signal is input externally while this command is being executed, the machine will stop, the remaining commands will be canceled and operation will be executed from the next block.



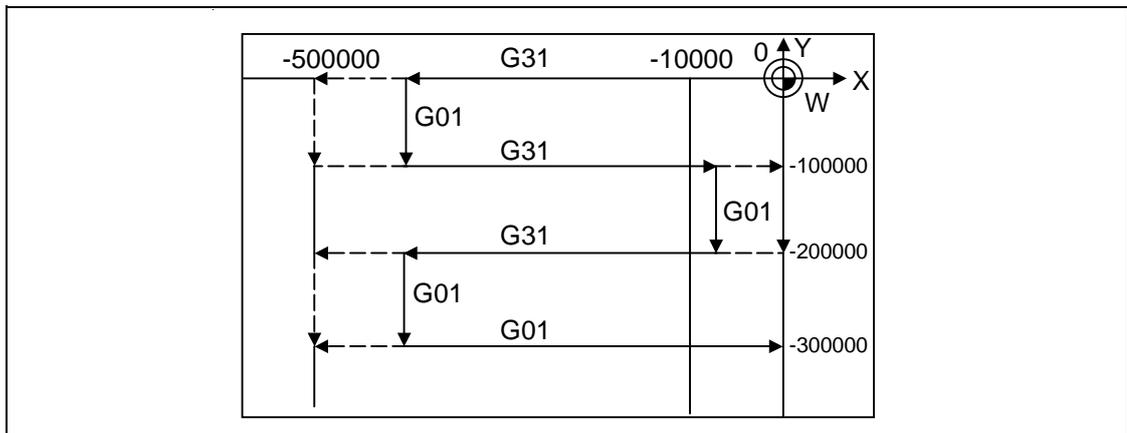
## Detailed description

- (1) If Ff is assigned as the feedrate in the same block as the G31 command block, command feed f will apply; if it not assigned, the value set in the parameter "#1174 Skip\_F" will serve as the feedrate. In either case, the F modal will not be updated.
- (2) The machine will not automatically accelerate and decelerate with the G31 block. The G31 maximum speed will depend on the machine specifications.
- (3) Override is invalid with the G31 command and it is fixed at 100%. Dry run is also invalid. The stop conditions (feed hold, interlock, override zero and stroke end) are valid. External deceleration is also valid.
- (4) The G31 command is unmodal and so it needs to be commanded each time.
- (5) If the skip command is input during G31 command start, the G31 command will be completed immediately.  
When a skip signal has not been input until the G31 block completion, the G31 command will also be completed upon completion of the movement commands.
- (6) When the G31 command is issued during tool radius compensation, program error (P608) will result.
- (7) When there is no F command in the G31 command and the parameter speed is also zero, program error (P603) will result.
- (8) With machine lock or with a command for the Z axis only with the Z axis cancel switch ON, the skip signal will be ignored and execution will continue as far as the end of the block.



Execution of G31

```
G90 G00 X-100000 Y0 ;
G31 X-500000 F100 ;
G01 Y-100000 ;
G31 X0 F100 ;
Y-200000 ;
G31 X-50000 F100 ;
Y-300000 ;
X0 ;
```



Detailed description (Readout of skip coordinates)

The coordinate positions for which the skip signal is input are stored in the system variables #5061 (1st axis) to #506n (n-th axis), so these can be used in the user macros.

<pre>                 } G90 G00 X-100. ; G31 X-200. F60 ; #101 = #5061                 }             </pre>	<p>— Skip command</p> <p>— Skip signal input coordinate values (workpiece coordinate system) are readout to #101.</p>
-------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------



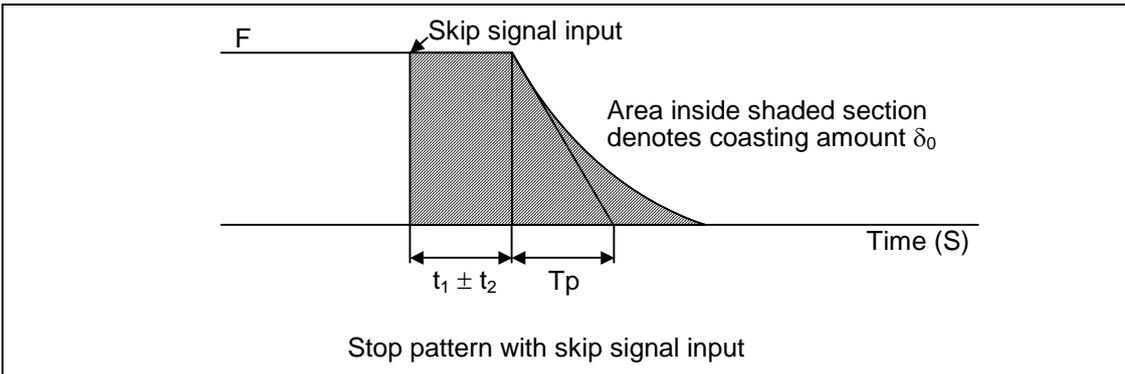
Detailed description (G31 coasting)

The amount of coasting from when the skip signal is input during the G31 command until the machine stops differs according to the parameter "#1174 skip\_F" or F command in G31. The time to start deceleration to a stop after responding to the skip signal is short, so the machine can be stopped precisely with a small coasting amount

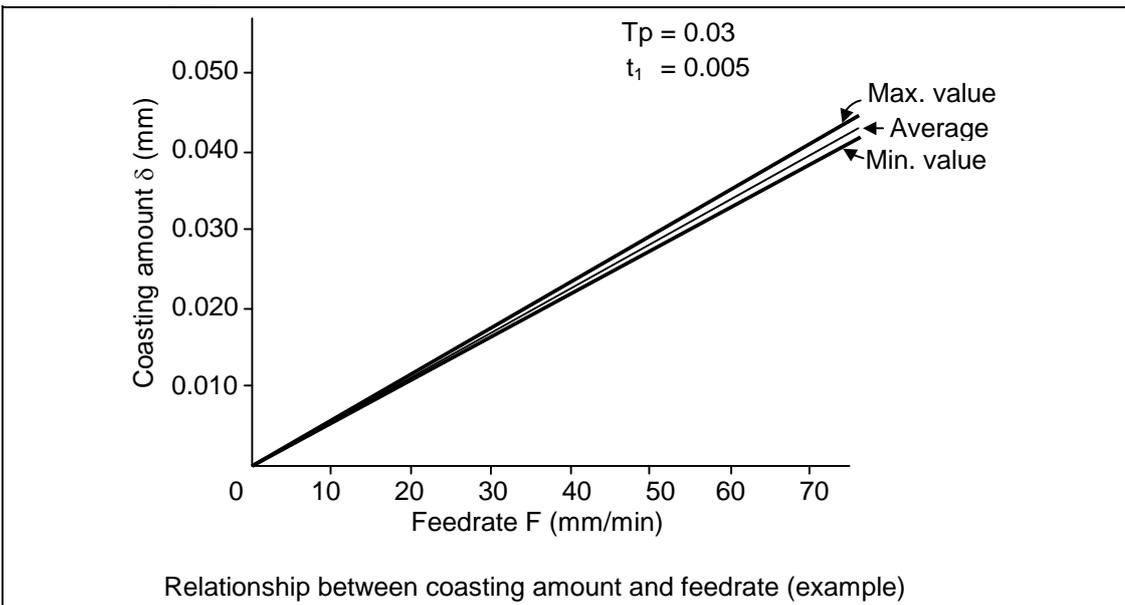
$$\delta_0 = \frac{F}{60} \times T_p + \frac{F}{60} \times (t_1 \pm t_2) = \underbrace{\frac{F}{60} \times (T_p + t_1)}_{\delta_1} \pm \underbrace{\frac{F}{60} \times t_2}_{\delta_2}$$

$\delta_0$  : Coasting amount (mm)  
 F : G31 skip speed (mm/min.)  
 $T_p$  : Position loop time constant (s) = (position loop gain)<sup>-1</sup>  
 $t_1$  : Response delay time (s) = (time taken from the detection to the arrival of the skip signal at the controller via PC)  
 $t_2$  : Response error time (0.001 s)

When G31 is used for calculation, the value calculated from the section indicated by  $\delta_1$  in the above equation can be compensated, however,  $\delta_2$  results in calculation error.



The relationship between the coasting amount and speed when  $T_p$  is 30ms and  $t_1$  is 5ms is shown in the following figure.



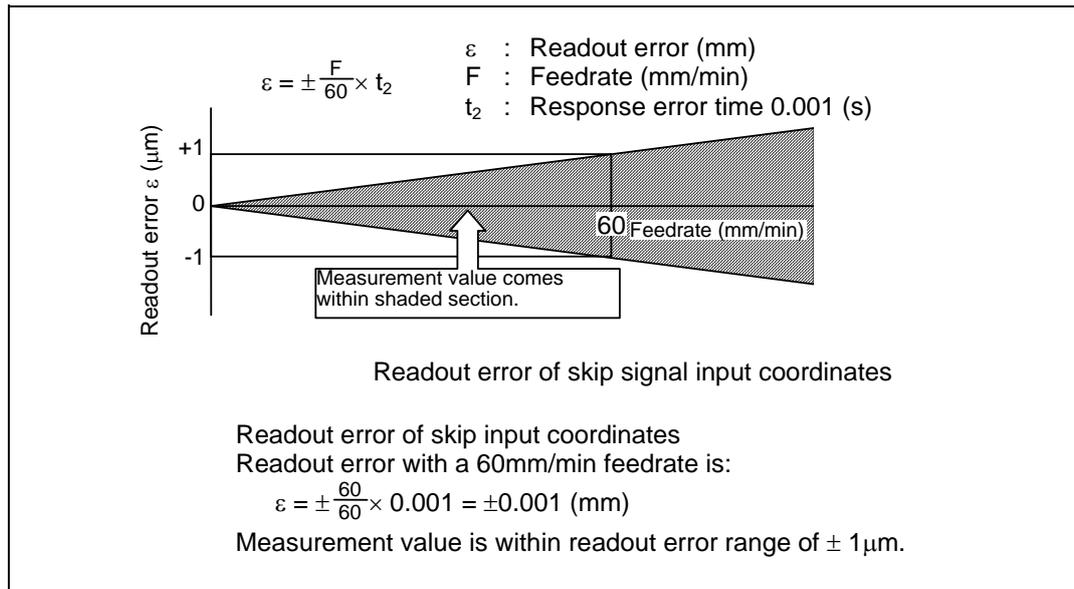


## Detailed description (Skip coordinate readout error)

**(1) Skip signal input coordinate readout**

The coasting amount based on the position loop time constant  $T_p$  and cutting feed time constant  $T_s$  is not included in the skip signal input coordinate values.

Therefore, the workpiece coordinate values applying when the skip signal is input can be read out across the error range in the following formula as the skip signal input coordinate values. However, coasting based on response delay time  $t_1$  results in a measurement error and so compensation must be provided.

**(2) Readout of other coordinates**

The readout coordinate values include the coasting amount. Therefore, when coordinate values are required with skip signal input, reference should be made to the section on the G31 coasting amount and compensation provided. As in the case of (1), the coasting amount based on the delay error time  $t_2$  cannot be calculated, and this generates a measuring error.



## Examples of compensating for coasting

## (1) Compensating for skip signal input coordinates

```

#110 = Skip feedrate ;
#111 = Response delay time  $t_1$  ;

{
  G31 X100. F100 ;
  G04 ;
  #101 = #5061 ;
  #102 = #110*#111/60 ;
  #105 = #101-#102-#103 ;
}

```

— Skip command  
 — Machine stop check  
 — Skip signal input coordinate readout  
 — Coasting based on response delay time  
 — Skip signal input coordinates

## (2) Compensating for workpiece coordinates

```

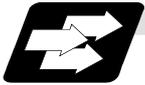
#110 = Skip feedrate ;
#111 = Response delay time  $t_1$  ;
#112 = Position loop time constant  $T_p$  ;

{
  G31 X100. F100 ;
  G04 ;
  #101 = #5061 ;
  #102 = #110*#111/60 ;
  #103 = #110*#112/60 ;
  #105 = #101-#102-#103 ;
}

```

— Skip command  
 — Machine stop check  
 — Skip signal input coordinate readout  
 — Coasting based on response delay time  
 — Coasting based on position loop time constant  
 — Skip signal input coordinates

## 15.3 Multi-step Skip Function; G31.n, G04



## Function and purpose

The setting of combinations of skip signals to be input enables skipping under various conditions. The actual skip operation is the same as with G31.

The G commands which can specify skipping are G31.1, G31.2, G31.3, and G04, and the correspondence between the G commands and skip signals can be set by parameters.



## Command format

**G31.1 X\_\_ Y\_\_ Z\_\_ α\_\_ F\_\_ ;**

X Y Z α ; Command format axis coordinate word and target coordinates  
F ; Feedrate (mm/min)

Same with G31.2 and G31.3 ; Ff is not required with G04

As with the G31 command, this command executes linear interpolation and when the preset skip signal conditions have been met, the machine is stopped, the remaining commands are canceled, and the next block is executed.

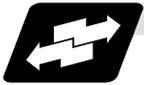


## Detailed description

- (1) Feedrate G31.1 set with the parameter corresponds to "#1176 skip1f", G31.2 corresponds to "#1178 skip2f", and G31.3 corresponds to "#1180 skip3f".
- (2) A command is skipped if it meets the specified skip signal condition.
- (3) The G31.n and G04 commands work the same as the G31 command for other than (1) and (2) above.
- (4) The feedrates corresponding to the G31.1, G31.2, and G31.3 commands can be set by parameters.
- (5) The skip conditions (logical sum of skip signals which have been set) corresponding to the G31.1, G31.2, G31.3 and G04 commands can be set by parameters.

Parameter setting	Valid skip signal		
	1	2	3
1	○		
2		○	
3	○	○	
4			○
5	○		○
6		○	○
7	○	○	○

(Skip when ○ signal is input.)



## Example of operation

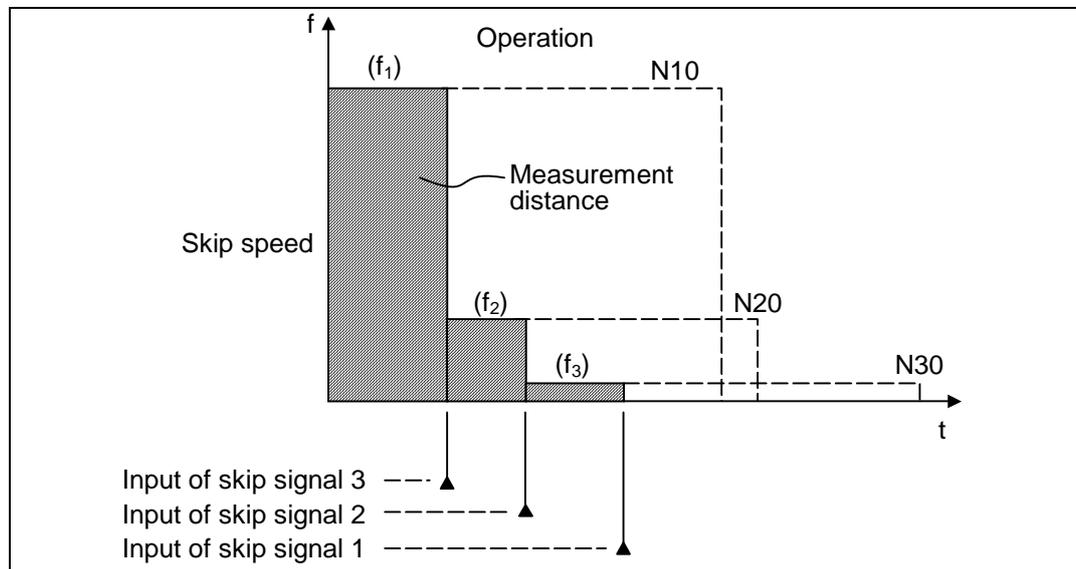
- (1) The multi-step skip function enables the following control, thereby improving measurement accuracy and shortening the time required for measurement.

Parameter settings :

Skip condition	Skip speed
G31.1 : 7	20.0mm/min (f <sub>1</sub> )
G31.2 : 3	5.0mm/min (f <sub>2</sub> )
G31.3 : 1	1.0mm/min (f <sub>3</sub> )

Program example :

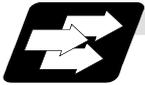
```
N10G31.1 X200.0 ;
N20G31.2 X40.0 ;
N30G31.3 X1.0 ;
```



**(Note 1)** If skip signal 1 is input before skip signal 2 in the above operation, N20 is skipped at that point and N30 is also ignored.

- (2) If a skip signal with the condition set during G04 (dwell) is input, the remaining dwell time is canceled and the following block is executed.

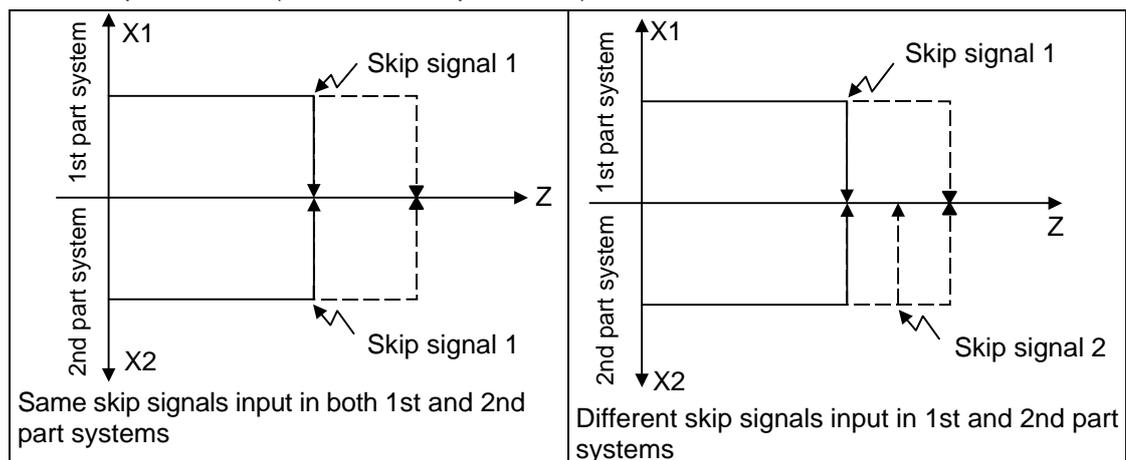
## 15.4 Multi-step Skip Function 2; G31



## Function and purpose

During linear interpolation followed by the skip command (G31), operation can be skipped according to the conditions of the skip signal parameter Pp. Skip signal command P is specified with the external skip signal 1 to 8.

If multi-step skip commands are issued simultaneously in different part systems, both part systems perform skip operation simultaneously if the input skip signals are the same, or they perform skip operation separately if the input skip signals are different. The skip operation is the same as with a normal skip command (G31 without P parameter).



If the skip condition specified by the parameter "#1173 dwlskp" (external skip signals 1 to 8 are used for the specification) is met during execution of a dwell command (G04), the remaining dwell time is canceled and the following block is executed. Similarly, if the skip condition is met during revolution dwelling, the remaining revolution is canceled and the following block is executed.



## Command format

**G31 X\_\_ Z\_\_ α\_\_ P\_\_ F\_\_ ;**

X Z α : Command format axis coordinate word and target coordinates  
 P : Skip signal parameter  
 F : Feedrate (mm/min)



Detailed description

- (1) The skip is specified by command speed f. Note that the F modal is not updated.
- (2) The skip signal is specified by skip signal parameter p. p can range from 1 to 255. If p is specified outside the range, program error (P35) occurs.

Skip signal parameter P	Valid skip signal							
	8	7	6	5	4	3	2	1
1								○
2							○	
3							○	○
4						○		
...	...	...	...	...	...	...	...	...
253	○	○	○	○	○	○		○
254	○	○	○	○	○	○	○	
255	○	○	○	○	○	○	○	○

(Skip when ○ signal is input.)

- (3) The specified skip signal command is a logical sum of the skip signals.

(Example)

```
G31 X100. P5 F100 ;
```

Operation is skipped if skip signal 1 or 3 is input.

- (4) If skip signal parameter Pp is not specified, the skip condition specified by the G31 parameter works. If speed parameter Ff is not specified, the skip speed specified by the G31 parameter works.

Relations between skip and multi-step skip

Skip specifications	x		o	
	Condition	Speed	condition	Speed
G31 X100 ; Without P and F	Program error (P601)		Skip 1	Parameter
G31 X100 P5 ; Without F	Program error (P602)		Command value	Parameter
G31 X100 F100 ; Without P	Program error (P601)		Skip 1	Command value
G31 X100 P5 F100 ;	Program error (P602)		Command value	Command value

(Note) "Parameter" in the above table indicates that specified with a skip command (G31).

- (5) If skip specification is effective and P is specified as an axis address, skip signal parameter P is given priority and axis address P is ignored.

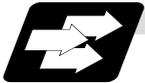
(Example)

```
G31 P500. F100 ;
```

This is regarded as a skip signal parameter and program error (P35) results.

- (6) Those items other than (1) to (5) are the same with the ordinary skip function (G31 without P).

## 15.5 Speed Change Skip; G31



## Function and purpose

When the skip signal is detected during linear interpolation by the skip command (G31), the feedrate is changed.



## Command format

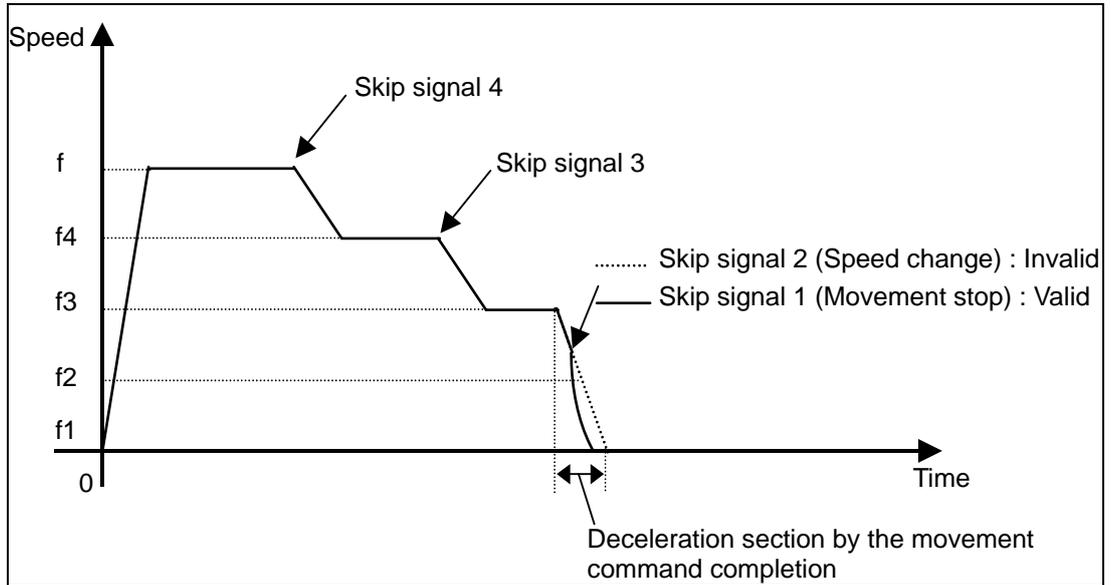
<b>G31</b>	<b>X__ Y__ Z__ α_ F__ F1=__ ... Fn=__ ;</b>	(n is the skip signal 1 to 8)
G31	:	Skip command
X Y Z α	:	Command format axis coordinate word and target coordinates
F	:	Feedrate when starting the cutting feed (mm/min)
Fn=	:	Feedrate after detecting the skip signal (mm/min)
		fn=0: Movement stop
		fn≠0: Changing the feedrate to fn
		F1=Feedrate after inputting the skip signal 1
		:
		F8=Feedrate after inputting the skip signal 8



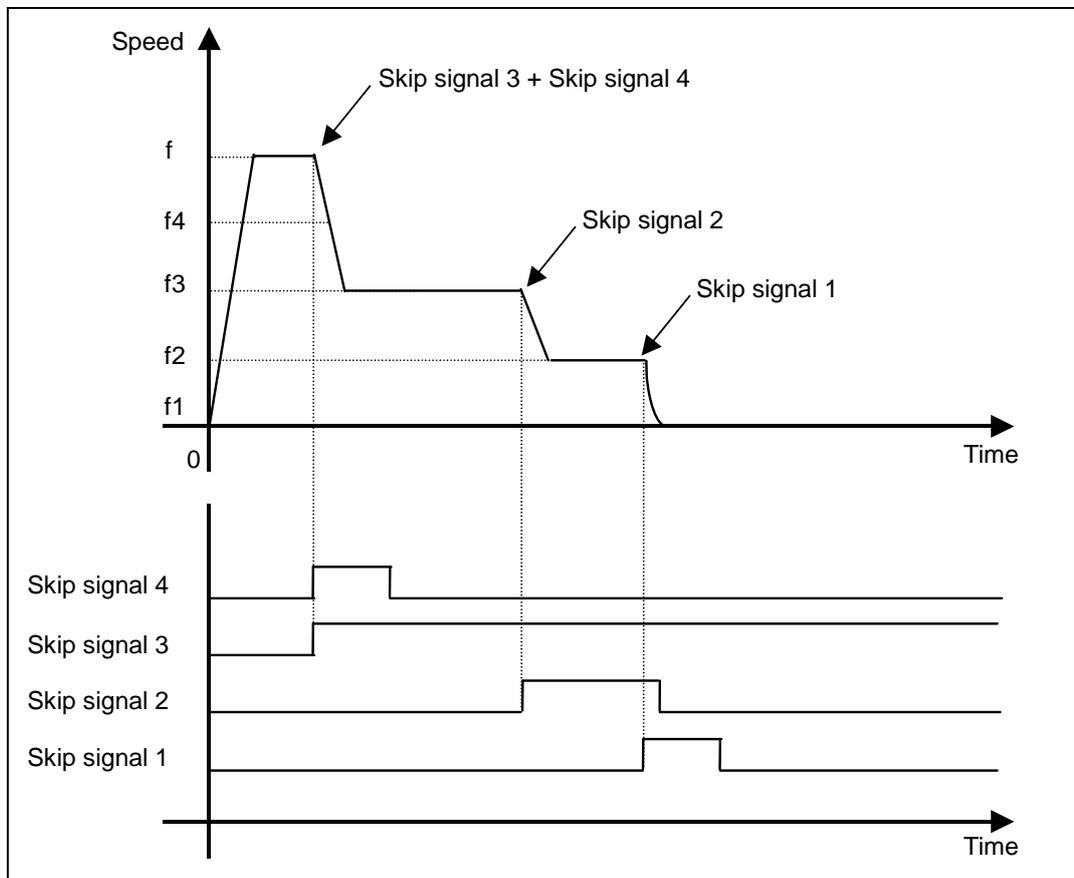
## Detailed description

- (1) When the skip signal for which the feedrate  $fn \neq 0$  is commanded, the speed is changed to the command speed corresponding to the skip signal.
- (2) When the skip signal for which the feedrate  $fn = 0$  is commanded, the movement is stopped. The acceleration and deceleration time constant at the movement stop does not follow the skip time constant, but the normal G31 skip. After the movement is stopped, the remaining movement commands are canceled and the following block executed.
- (3) When the skip signal has not been input until the G31 block completion, the G31 command will be also completed upon completion of the movement command.
- (4) When the skip return is valid, the return operation by the skip signal detection is executed after the movement is stopped.
- (5) Even if the acceleration and deceleration with the inclination constant G1 (#1201 G1\_acc) is valid, the speed change skip will be the operation of the time constant acceleration and deceleration.
- (6) When the feedrate command ( $Fn = fn$ ) is not specified after detecting the skip signal, the normal G31 skip operation will be applied.

- (7) If the skip signal is input during the deceleration by the movement command completion, the speed change will be ignored.



- (8) The skip signal for which the feedrate is not commanded in the program is ignored.
- (9) The speed change or the movement stop is performed when detecting the rising edge of the skip signal. Note that if several rising edges are input at 3.5ms intervals or less, they may be judged the simultaneous input. When they are judged the simultaneous input, the smaller value will be valid.



- (10) If the G31 block is started with the skip signal input, that signal is considered to rise at the same time as the block start.
- (11) If the skip signals for changing the speed and for stopping the movement are simultaneously input, the skip signal for stopping the movement will be valid regardless of the size of the number.
- (12) If the skip time constant "#2102 skip\_tL" is illegal, the "Y51 PARAMETER ERROR 15" will occur, and if the "#2103 skip\_t1" is illegal, the "Y51 PARAMETER ERROR 16" will occur.
- (13) The items other than (1) to (12) are the same with the G31 command.

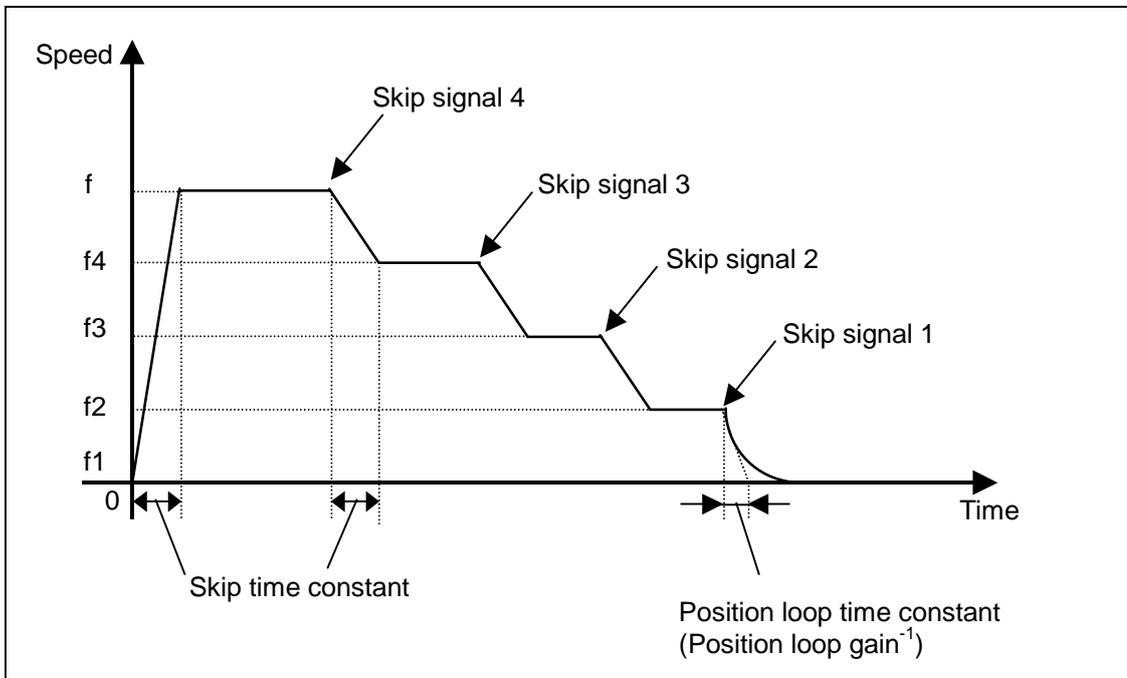


**Example of operation**

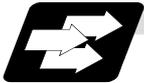
**Example of program**

```
G31 X100. Ff F1=0 F2=f2 F3=f3 F4=f4 ;
```

**Operation**



## 15.6 Programmable Current Limitation



### Function and purpose

This function allows the current limit value of the servo axis to be changed to a desired value in the program, and is used for the workpiece stopper, etc.  
The commanded current limit value is designated with a ratio of the limit current to the rated current.



### Command format

**G10 L14 Xn ;**

L14 : Current limit value setting (+ side/- side)

X : Axis address

n : Current limit value (%) Setting value: 1 to 999



### Precautions and restrictions

- (1) If the current limit value is reached when the current limit is valid, the current limit reached signal is output.
- (2) The following two modes can be used as the operation after the current limit is reached. The external signal determines which mode applies.
  - Normal mode  
The movement command is executed in the current state.  
During automatic operation, the movement command is executed to the end, and then the next block is moved to with the droops still accumulated.
  - Interlock mode  
The movement command is executed in the current state.  
During automatic operation, the operation stops at the corresponding block, and the next block is not moved to.  
During manual operation, the following same direction commands are ignored.
- (3) During the current limit, the position droop generated by the current limit can be canceled when the current limit changeover signal of external signals is canceled.  
(Note that the axis must not be moving.)
- (4) The setting range of the current limit value is 1% to 999%. Commands that exceed this range will cause a program error (P35).
- (5) If a decimal point is designated with the G10 command, only the integer will be valid.  
**(Example)** G10 L14 X10.123 ; The current limit value will be set to 10%.
- (6) For the axis name "C", the current limit value cannot be set from the program (G10 command). To set from the program, set the axis address with an incremental axis name, or set the axis name to one other than "C".

## 15.7 Stroke Check Before Travel; G22.1/G23.1



## Function and purpose

By commanding the boundaries from the program with coordinate values on the machine coordinate system, machine entry into that boundary can be prohibited. This can be set only for the three basic axes.

While the normal stored stroke limit stops entry before the prohibited area, this function causes a program error before movement to the block if a command exceeding the valid movement area is issued.



## Command format

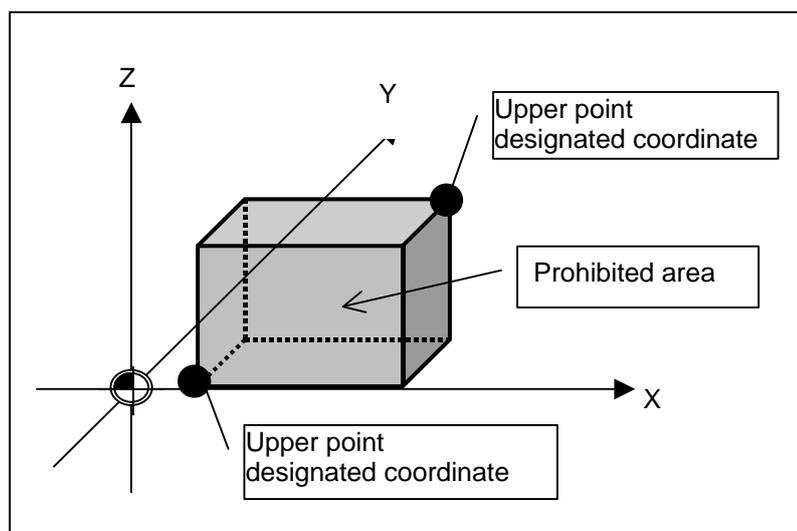
<b>G22.1 ; X__ Y__ Z__ I__ J__ K__;</b>	<b>Stroke check before travel ON</b>
<b>G23.1;</b>	<b>Stroke check before travel cancel</b>
G22.1	: Stroke check before travel ON
G23.1	: Stroke check before travel cancel
X Y Z	: Coordinates of upper point (basic axis name and its coordinate position)
I J K	: Coordinates of lower point (basic axis name and its coordinate position)

**(Note)** In the following command format, the basic axes are X, Y and Z. If the basic axis name differs, command the point 1 coordinate command address with the basic axis name.



## Detailed description

- (1) The inner side of the boundary commanded with the point 1 coordinate and point 2 coordinate is the prohibited area.
- (2) If the command is omitted, "0" will be set for the address.
- (3) The area designated with this function is different from the area designated with the stored stroke limit. However, the area enabled by both functions will be the actual valid movement range.



**(Note)** The upper point and lower point are commanded with coordinate on the machine coordinate system.



## Precautions and restrictions

- (1) This function is valid only when starting the automatic operation. When interrupted with manual absolute OFF, the prohibited area will also be shifted by the interrupted amount.
- (2) An error will occur if the start point or end point is in the prohibited area.
- (3) Stroke check will not be carried out for the axes having the same coordinates set for the upper point and the lower point.
- (4) The stroke check is carried out with the tool center coordinate values.
- (5) If G23.1 X\_Y\_Z\_; etc., is commanded, the command will be interpreted as G23.1; X\_Y\_Z;.(2 blocks) Thus, the stroke check before travel will be canceled, then movement will take place with the previous movement modal.
- (6) During automatic reference position return, the check will not be carried out from the intermediate point to the reference position. With G29, when moving from the start point to intermediate point, the check will not be carried out.
- (7) If there is an address not used in one block, a program error will occur.
- (8) When the rotary-type rotation axis is set as a basic axis, the prohibited area will be converted to the range of from 0° to 360° in the same manner as the movement command. If the setting extends over "0°", the side containing "0°" will be the check area.

## Example

(a) G22.1 Z45. K315.	: Stroke check area $45. \leq Z \leq 315.$
(b) G22.1 Z-115. K-45.	: Stroke check area $225. \leq Z \leq 315.$
(c) G22.1 Z45. K-45.	: Stroke check area $0. \leq Z \leq 45., 315. \leq Z \leq 360.$

## Appendix 1. Program Error

(The bold characters are the message displayed in the screen.)

These alarms occur during automatic operation, and the causes of these alarms are mainly program errors which occur, for instance, when mistakes have been made in the preparation of the machining programs or when programs which conform to the specification have not been prepared.

Error No.	Details	Remedy
P 10	<b>No. of simultaneous axes over</b> The number of axis addresses commanded in the same block exceeds the specifications.	<ul style="list-style-type: none"> <li>• Divide the alarm block command into two.</li> <li>• Check the specifications.</li> </ul>
P 11	<b>Illegal axis address</b> The axis address commanded by the program and the axis address set by the parameter do not match.	<ul style="list-style-type: none"> <li>• Revise the axis names in the program.</li> </ul>
P 20	<b>Division error</b> An axis command which cannot be divided by the command unit has been issued.	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
P 29	<b>Not accept command</b> The normal line control command (G40.1, G41.1, G42.1) has been issued during the modal in which the normal line control is not acceptable.	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
P 30	<b>Parity H error</b> The number of holes per character on the paper tape is even for EIA code and odd for ISO code.	<ul style="list-style-type: none"> <li>• Check the paper tape.</li> <li>• Check the tape puncher and tape reader.</li> </ul>
P 31	<b>Parity V error</b> The number of characters per block on the paper tape is odd.	<ul style="list-style-type: none"> <li>• Make the number of characters per block on the paper tape even.</li> <li>• Set the parameter parity V selection OFF.</li> </ul>
P 32	<b>Illegal address</b> An address not listed in the specifications has been used.	<ul style="list-style-type: none"> <li>• Check and revise the program address.</li> <li>• Check and correct the parameters values.</li> <li>• Check the specifications.</li> </ul>
P 33	<b>Format error</b> The command format in the program is not correct.	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
P 34	<b>Illegal G code</b> A G code not listed in the specifications has been used. An illegal G code was commanded during the coordinate rotation command (G68).	<ul style="list-style-type: none"> <li>• Check and correct the G code address in the program.</li> </ul>
	G51.2 or G50.2 was commanded when the rotary tool axis No. (#1501 polyax) was set to "0". G51.2 or G50.2 was commanded when the tool axis was set to the linear axis (#1017 rot "0").	<ul style="list-style-type: none"> <li>• Check the parameter setting values.</li> </ul>
P 35	<b>Setting value range over</b> The setting range for the addresses has been exceeded.	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
P 36	<b>Program end error</b> "EOR" has been read during tape and memory mode.	<ul style="list-style-type: none"> <li>• Enter the M02 and M30 command at the end of the program.</li> <li>• Enter the M99 command at the end of the subprogram.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
P 37	<b>O, N number zero</b> A zero has been specified for program and sequence Nos.	<ul style="list-style-type: none"> <li>The program Nos. are designated across a range from 1 to 99999999.</li> <li>The sequence Nos. are designated across a range from 1 to 99999.</li> </ul>
P 38	<b>No spec: Add. Op block skip</b> "/n" has been issued even though there are no optional block skip addition specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
P 39	<b>No specifications</b> <ul style="list-style-type: none"> <li>A non-specified G code was specified.</li> <li>The selected operation mode is not used.</li> </ul>	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
P 40	<b>Pre-read block error</b> When tool radius compensation is executed, there is an error in the pre-read block and so the interference check is disabled.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> </ul>
P 48	<b>Restart pos return incomplete</b> Movement command was executed before executing the block that is restart-searched.	<ul style="list-style-type: none"> <li>Carry out program restart again. Movement command cannot be executed before executing the block that is restart-searched.</li> </ul>
P 49	<b>Invalid restart search</b> <ul style="list-style-type: none"> <li>Restart search was attempted for the 3-dimensional circular interpolation.</li> <li>Restart search was attempted during the cylindrical interpolation, polar coordinate interpolation, and tool tip center control.</li> </ul>	<ul style="list-style-type: none"> <li>Reconsider the program.</li> <li>Reconsider the restart search position.</li> </ul>
P 50	<b>No spec: Inch/Metric change</b> Inch/Metric changeover (G20/G21) command was issued even though there is no inch/metric conversion specification.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
P 60	<b>Compensation length over</b> The commanded movement distance is excessive. (Over 2 <sup>31</sup> )	<ul style="list-style-type: none"> <li>Reconsider the axis address command.</li> </ul>
P 61	<b>No spec: Unidirectional posit.</b> Unidirectional positioning (G60) was commanded even though there is no unidirectional positioning specification.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
P 62	<b>No F command</b> <ul style="list-style-type: none"> <li>No feed rate command has been issued.</li> <li>There is no F command in the cylindrical interpolation or polar coordinate interpolation immediately after the G95 mode is commanded.</li> </ul>	<ul style="list-style-type: none"> <li>The default movement modal command at power ON is G01. This causes the machine to move without a G01 command if a movement command is issued in the program, and an alarm results. Use an F command to specify the feed rate.</li> <li>Specify F with a thread lead command.</li> </ul>
P 63	<b>No spec: High-speed machining</b> High-speed machining cancel (G5P0) was commanded even though there is no high-speed machining mode specification.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
P 65	<b>No spec: High speed mode 3</b>	<ul style="list-style-type: none"> <li>Check the high-speed mode III specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P 70</b>	<b>Arc end point deviation large</b> <ul style="list-style-type: none"> <li>There is an error in the arc start and end points as well as in the arc center.</li> <li>The difference of the involute curve through the start point and the end point is large.</li> <li>When arc was commanded, one of the two axes configuring the arc plane was a scaling valid axis.</li> </ul>	<ul style="list-style-type: none"> <li>Check the numerical values of the addresses that specify the start and end points, arc center as well as the radius in the program.</li> <li>Check the "+" and "-" directions of the address numerical values.</li> <li>Check the scaling valid axis.</li> </ul>
<b>P 71</b>	<b>Arc center error</b> <ul style="list-style-type: none"> <li>The arc center is not sought during R-specified circular interpolation.</li> <li>The curvature center of the involute curve cannot be obtained.</li> </ul>	<ul style="list-style-type: none"> <li>Check the numerical values of the addresses in the program.</li> <li>Check whether the start point or end point is on the inner side of the base circle for involute interpolation. When carrying out tool radius compensation, check that the start point and end point after compensation are not on the inner side of the base circle for involute interpolation.</li> <li>Check whether the start point and end point are at an even distance from the center of the base circle for involute interpolation.</li> </ul>
<b>P 72</b>	<b>No spec: Helical cutting</b> A helical command has been issued though it is not included in the specifications.	<ul style="list-style-type: none"> <li>Check the helical specifications.</li> <li>An Axis 3 command was issued by the circular interpolation command. If there is no helical specification, the linear axis is moved to the next block.</li> </ul>
<b>P 73</b>	<b>No spec: Spiral cutting</b> A spiral command was issued despite the fact that such a command does not exist in the specifications.	<ul style="list-style-type: none"> <li>The G02.1 and G03.1 commands are issued for circular interpolation.</li> <li>Check the spiral specifications.</li> </ul>
<b>P 74</b>	<b>Can't calculate 3DIM arc</b> The end block was not specified during 3-dimension circular interpolation supplementary modal, and therefore it is not possible to calculate the 3-dimension circular interpolation. Furthermore, it not possible to calculate the 3-dimension circular interpolation due to an interruption during 3-dimension circular interpolation supplementary modal.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> </ul>
<b>P 75</b>	<b>3DIM arc illegal</b> An unusable G code was issued during 3-dimension circular interpolation modal. Or, a 3-dimension circular interpolation command was issued during a modal for which a 3-dimension circular interpolation command cannot be issued.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> </ul>
<b>P 76</b>	<b>No spec: 3DIM arc interpolat</b> G02.4/G03.4 was commanded even though there is no 3-dimension circular interpolation specification.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P80</b>	<b>No spec: Hypoth ax interpolat</b> Hypothetical axis interpolation (G07) was commanded even though there is no hypothetical axis interpolation specification.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P 90</b>	<b>No spec: Thread cutting</b> A thread cutting command was issued even though there is no thread cutting command specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P 91</b>	<b>No spec: Var lead threading</b> Variable lead thread cutting (G34) was commanded even though there is no variable lead thread cutting specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P 93</b>	<b>Illegal pitch vaule</b> The thread lead (thread pitch) when performing the thread cutting command is incorrect.	<ul style="list-style-type: none"> <li>• Set the correct thread lead command for the thread cutting command.</li> </ul>
<b>P100</b>	<b>No spec: Cylindric interpolat</b> A cylindrical interpolation command was issued even though there is no cylindrical interpolation specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P110</b>	<b>Plane select during figure rot</b> Plane selection (G17/G18/G19) was commanded during figure rotation.	<ul style="list-style-type: none"> <li>• Check the machining program.</li> </ul>
<b>P111</b>	<b>Plane selected while coord rot</b> Plane selection commands (G17, G18, G19) were issued during a coordinate rotation command (G68).	<ul style="list-style-type: none"> <li>• After command G68, always issue a plane selection command following a G69 (coordinate rotation cancel) command.</li> </ul>
<b>P112</b>	<b>Plane selected while R compen</b> <ul style="list-style-type: none"> <li>• Plane selection commands (G17, G18, G19) were issued while tool radius compensation (G41, G42) and nose R compensation (G41, G42, G46) commands were being issued.</li> <li>• Plane selection commands were issued after completing nose R compensation commands when there are no further axis movement commands after G40, and compensation has not been cancelled.</li> </ul>	<ul style="list-style-type: none"> <li>• Issue plane selection commands after completing (axis movement commands issued after G40 cancel command) tool radius compensation and nose R compensation commands.</li> </ul>
<b>P113</b>	<b>Illegal plane select</b> The circular command axis differs from the selected plane.	<ul style="list-style-type: none"> <li>• Issue a circular command after correct plane selection.</li> </ul>
<b>P120</b>	<b>No spec: Feed per rotation</b> Feed per rotation (G95) was commanded even though there is no feed per rotation specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P121</b>	<b>F0 command during arc modal</b> F0 (F 1-digit feed) was commanded during the arc modal (G02/G03).	<ul style="list-style-type: none"> <li>• Check the machining program.</li> </ul>
<b>P122</b>	<b>No spec: Auto corner override</b> An auto corner override command (G62) was issued even though there is no auto corner override specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> <li>• Delete the G62 command from the program.</li> </ul>
<b>P123</b>	<b>No spec: High-accuracy control</b> High-accuracy control command was issued even though there is no high-accuracy control specification	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P124</b>	<b>No spec: Inverse time feed</b> There is no inverse time option.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P125</b>	<b>G93 mode error</b> <ul style="list-style-type: none"> <li>• A G code command that cannot be issued was issued during G93 mode.</li> <li>• G93 command was issued during a modal for which inverse time feed cannot be performed.</li> </ul>	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P126</b>	<b>Invalid cmd in high-accuracy</b> An illegal command was issued during the high-accuracy control mode. <ul style="list-style-type: none"> <li>• A G code group 13 command was issued during the high-accuracy control mode.</li> <li>• Milling, cylindrical interpolation or pole coordinate interpolation was commanded during the high-accuracy control mode.</li> </ul>	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P127</b>	<b>No spec: SSS Control</b> The SSS control valid parameter was set to ON although there is no SSS control specification.	<ul style="list-style-type: none"> <li>• Check the specifications. If there is no SSS control specification, set the parameter #8090 SSS ON to 0.</li> </ul>
<b>P130</b>	<b>2nd M function code illegal</b> The 2nd miscellaneous function address commanded in the program differs from the address set in the parameters. miscellaneous function.	<ul style="list-style-type: none"> <li>• Check and correct the 2nd miscellaneous function address in the program.</li> </ul>
<b>P131</b>	<b>No spec: Cnst surface ctrl G96</b> A constant surface speed control command (G96) was issued even though there is no specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> <li>• Change the constant surface speed control command (G96) to a rotation speed command (G97).</li> </ul>
<b>P132</b>	<b>Spindle rotation speed S=0</b> No spindle rotation speed command has been issued.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P133</b>	<b>Illegal P-No. G96</b> An invalid constant surface speed control axis has been specified.	<ul style="list-style-type: none"> <li>• Reconsider the parameter specified for the constant surface speed control axis.</li> </ul>
<b>P140</b>	<b>No spec: Pos compen cmd</b> The position compensation command (G45 to G48) specifications are not available.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P141</b>	<b>Pos compen during rotation</b> Position compensation was commanded during the figure rotation or coordinate rotation command.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P142</b>	<b>Pos compen invalid arc</b> A position compensation invalid arc command was commanded.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P150</b>	<p><b>No spec: Nose R compensation</b></p> <ul style="list-style-type: none"> <li>• Even though there were no tool radius compensation specifications, tool radius compensation commands (G41 and G42) were issued.</li> <li>• Even though there were no nose R compensation specifications, nose R compensation commands (G41, G42, and G46) were issued.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P151</b>	<p><b>Radius compen during arc mode</b></p> <p>A compensation command (G40, G41, G42, G43, G44, G46) has been issued in the arc modal (G02, G03).</p>	<ul style="list-style-type: none"> <li>• Issue the linear command (G01) or rapid traverse command (G00) in the compensation command block or cancel block. (Set the modal to linear interpolation.)</li> </ul>
<b>P152</b>	<p><b>No intersection</b></p> <p>In interference block processing during execution of a tool radius compensation (G41 or G42) or nose R compensation (G41, G42, or G46) command, the intersection point after one block is skipped cannot be determined.</p>	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P153</b>	<p><b>Compensation interference</b></p> <p>An interference error has arisen while the tool radius compensation command (G41, G42) or nose R compensation command (G41, G42, G46) was being executed.</p>	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P154</b>	<p><b>No spec: 3D compensation</b></p> <p>A three-dimensional compensation command was issued even though there are no three-dimensional compensation specifications.</p>	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P155</b>	<p><b>Fixed cyc exec during compen</b></p> <p>A fixed cycle command has been issued in the radius compensation mode.</p>	<ul style="list-style-type: none"> <li>• The radius compensation mode is established when a fixed cycle command is executed and so the radius compensation cancel command (G40) should be issued.</li> </ul>
<b>P156</b>	<p><b>R compen direction not defined</b></p> <p>At the start of G46 nose R compensation, the compensation direction is undefined if this shift vector is used.</p>	<ul style="list-style-type: none"> <li>• Change the vector to that with which the compensation direction is defined.</li> <li>• Exchange with a tool having a different tip point No.</li> </ul>
<b>P157</b>	<p><b>R compen direction changed</b></p> <p>During G46 nose R compensation, the compensation direction is inverted.</p>	<ul style="list-style-type: none"> <li>• Change the G command to that which allows inversion of the compensation direction (G00, G28, G30, G33, or G53).</li> <li>• Exchange with a tool having a different tip point No.</li> <li>• Turn ON the "#8106 G46 NO REV-ERR" parameter.</li> </ul>
<b>P158</b>	<p><b>Illegal tip point</b></p> <p>During G46 nose R compensation, the tip point is illegal (other than 1 to 8).</p>	<ul style="list-style-type: none"> <li>• Change the tip point No. to a legal one.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P170</b>	<b>No offset number</b> The compensation No. (DOO, TOO, HOO) command was not given when the radius compensation (G41, G42, G43, G46) command was issued. Alternatively, the compensation No. is larger than the number of sets in the specifications.	<ul style="list-style-type: none"> <li>• Add the compensation No. command to the compensation command block.</li> <li>• Check the number of compensation No. sets a correct it to a compensation No. command within the permitted number of tool compensation sets.</li> </ul>
<b>P171</b>	<b>No spec:Comp input by prog G10</b> Compensation data input by program (G10) was commanded even though there is no specification of compensation data input by program.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P172</b>	<b>G10 L number error</b> (G10 L-No. error) The L address command is not correct when the G10 command is issued.	<ul style="list-style-type: none"> <li>• Check the address L-No. of the G10 command and correct the No.</li> </ul>
<b>P173</b>	<b>G10 P number error</b> (G10 compensation error) When the G10 command is issued, a compensation No. outside the permitted number of sets in the specifications has been commanded for the compensation No. command.	<ul style="list-style-type: none"> <li>• First check the number of compensation sets and then set the address P designation to within the permitted number of sets.</li> </ul>
<b>P174</b>	<b>No spec:Comp input by prog G11</b> Compensation data input by program cancel (G11) was commanded even though there is no specification of compensation data input by program.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P177</b>	<b>Tool life count active</b> Registration of tool life management data with G10 was attempted when the used data count valid signal was ON.	<ul style="list-style-type: none"> <li>• The tool life management data cannot be registered when counting the used data. Turn the used data count valid signal OFF.</li> </ul>
<b>P178</b>	<b>Tool life data entry over</b> The number of registration groups, total number of registered tools or the number of registrations per group exceeded the specifications range.	<ul style="list-style-type: none"> <li>• Review the number of registrations.</li> </ul>
<b>P179</b>	<b>Illegal group No.</b> <ul style="list-style-type: none"> <li>• When registering the tool life management data with G10, the group No. was commanded in duplicate.</li> <li>• A group No. that was not registered was designated during the T□□□□99 command.</li> <li>• An M code command must be issued as a single command but coexists in the same block as that of another M code command.</li> <li>• The M code commands set in the same group exist in the same block.</li> </ul>	<ul style="list-style-type: none"> <li>• The group No. cannot be commanded in duplicate. When registering the group data, register it in group units.</li> <li>• Correct to the correct group No.</li> </ul>
<b>P180</b>	<b>No spec: Drilling cycle</b> A fixed cycle command was issued though there are not fixed cycle (G72 - G89) specifications.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> <li>• Correct the program.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P181</b>	<b>No spindle command (Tap cycle)</b> Spindle rotation speed (S) has not been commanded in synchronous tapping.	<ul style="list-style-type: none"> <li>• Command the spindle rotation speed (S) in synchronous tapping.</li> <li>• When "#8125 Check Scode in G84" is set to "1", enter the S command in the same block where the synchronous tapping command is issued.</li> </ul>
<b>P182</b>	<b>Synchronous tap error</b> <ul style="list-style-type: none"> <li>• Connection to the main spindle unit was not established.</li> <li>• The synchronous tapping was attempted with the spindle not serially connected under the multiple-spindle control I.</li> </ul>	<ul style="list-style-type: none"> <li>• Check connection to the main spindle.</li> <li>• Check that the main spindle encoder exists.</li> <li>• Set 1 to the parameter #3024 (sout).</li> </ul>
<b>P183</b>	<b>No pitch/thread number</b> The pitch or thread number command has not been issued in the tap cycle of a fixed cycle for drilling command.	<ul style="list-style-type: none"> <li>• Specify the pitch data and the number of threads by F or E command.</li> </ul>
<b>P184</b>	<b>Pitch/thread number error</b> <ul style="list-style-type: none"> <li>• The pitch or the number of threads per inch is illegal in the tap cycle of the fixed cycle for drilling command.</li> <li>• The pitch is too small for the spindle rotation speed.</li> <li>• The thread number is too large for the spindle rotation speed.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the pitch or the number of threads per inch.</li> </ul>
<b>P185</b>	<b>No spec: Sync tapping cycle</b> Synchronous tapping cycle (G84/G74) was commanded even though there is no synchronous tapping cycle specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P186</b>	<b>Illegal S cmd in synchro tap</b> S command was issued during synchronous tapping modal.	<ul style="list-style-type: none"> <li>• Cancel the synchronous tapping before issuing the S command.</li> </ul>
<b>P190</b>	<b>No spec: Turning cycle</b> A lathe cutting cycle command was input although the lathe cutting cycle was undefined in the specification.	<ul style="list-style-type: none"> <li>• Check the specification.</li> <li>• Delete the lathe cutting cycle command.</li> </ul>
<b>P191</b>	<b>Taper length error</b> In the lathe cutting cycle, the specified length of taper section is illegal.	<ul style="list-style-type: none"> <li>• The radius set value in the lathe cycle command must be smaller than the axis shift amount.</li> </ul>
<b>P192</b>	<b>Chamfering error</b> Chamfering in the thread cutting cycle is illegal.	<ul style="list-style-type: none"> <li>• Set a chamfering amount not exceeding the cycle.</li> </ul>
<b>P200</b>	<b>No spec: MRC cycle</b> The compound type fixed cycle for turning machining I (G70 to G73) was commanded when the compound type fixed cycle for turning machining I specifications were not provided.	<ul style="list-style-type: none"> <li>• Check the specification.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P201</b>	<b>Program error (MRC)</b> <ul style="list-style-type: none"> <li>When called with a compound type fixed cycle for turning machining I command, the subprogram contained at least one of the following commands: <ul style="list-style-type: none"> <li>Reference position return command (G27, G28, G29, G30)</li> <li>Thread cutting (G33, G34)</li> <li>Fixed cycle skip-function (G31, G31.n)</li> </ul> </li> <li>The first move block of the finish shape program in compound type fixed cycle for turning machining I contains an arc command.</li> </ul>	<ul style="list-style-type: none"> <li>Delete the following G codes from this subprogram that is called with the compound type fixed cycle for turning machining I commands (G70 to G73): G27, G28, G29, G30, G31, G33, G34, and fixed cycle G codes.</li> <li>Remove G2 and G3 from the first move block of the finish shape program in compound type fixed cycle for turning machining I.</li> </ul>
<b>P202</b>	<b>Block over (MRC)</b> The number of blocks in the shape program of the compound type fixed cycle for turning machining I is over 50 or 200 (this differs according to the model).	<ul style="list-style-type: none"> <li>Specify 50 or a less value. The number of blocks in the shape program called by the compound type fixed cycle for turning machining I commands (G70 to G73) must be decreased below 50 or 200 (this differs according to the model).</li> </ul>
<b>P203</b>	<b>D cmdnd figure error (MRC)</b> The compound type fixed cycle for turning machining I (G70 to G73) shape program could not cut the work normally because it defined an abnormal shape.	<ul style="list-style-type: none"> <li>Check the compound type fixed cycle for turning machining I (G70 to G73) shape program.</li> </ul>
<b>P204</b>	<b>E cmdnd fixed cycle error</b> A command value of the compound type fixed cycle for turning machining (G70 to G76) is illegal.	<ul style="list-style-type: none"> <li>Check the compound type fixed cycle for turning machining (G70 to G76) command value.</li> </ul>
<b>P210</b>	<b>No spec: Pattern cycle</b> A compound type fixed cycle for turning machining II (G74 to G76) command was input although it was undefined in the specification.	<ul style="list-style-type: none"> <li>Check the specification.</li> </ul>
<b>P220</b>	<b>No spec: Special fixed cycle</b> No special fixed cycle specifications are available.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P221</b>	<b>No. of special fixed holes = 0</b> A 0 has been specified for the number of holes in special fixed cycle mode.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> </ul>
<b>P222</b>	<b>G36 angle error</b> A G36 command specifies 0 for angle intervals.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> </ul>
<b>P223</b>	<b>G12/G13 radius error</b> The radius value specified with a G12 or G13 command is below the compensation amount.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> </ul>
<b>P224</b>	<b>No spec: Circular (G12/G13)</b> There are no circular cutting specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P230</b>	<b>Subprogram nesting over</b> <ul style="list-style-type: none"> <li>A subprogram has been called 8 or more times in succession from the subprogram.</li> <li>The program in the data server contains the M198 command.</li> <li>The program in the IC card has been called more than once (the program in the IC card can be called only once at a time).</li> </ul>	<ul style="list-style-type: none"> <li>Check the number of subprogram calls and correct the program so that it does not exceed 8 times.</li> </ul>
<b>P231</b>	<b>No sequence No.</b> At subprogram call time, the sequence No. set at return from the subprogram or specified by GOTO, was not set.	<ul style="list-style-type: none"> <li>Specify the sequence Nos. in the call block of the subprogram.</li> </ul>
<b>P232</b>	<b>No program No.</b> <ul style="list-style-type: none"> <li>The machining program has not been found when the machining program is called.</li> <li>The file name of the program registered in IC card is not corresponding to O No.</li> </ul>	<ul style="list-style-type: none"> <li>Enter the machining program.</li> <li>Check the subprogram storage destination parameters.</li> <li>Ensure that the external device (including IC card) that contains the file is mounted.</li> </ul>
<b>P235</b>	<b>Program editing</b> Operation was attempted for the file under program editing.	<ul style="list-style-type: none"> <li>Execute the program again after completion of program editing.</li> </ul>
<b>P240</b>	<b>Program editing</b> Operation was attempted for the file under program editing.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P241</b>	<b>No variable No.</b> The variable No. commanded is out of the range specified in the specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> <li>Check the program variable No.</li> </ul>
<b>P242</b>	<b>= not defined at vrble set</b> The "=" sign has not been commanded when a variable is defined.	<ul style="list-style-type: none"> <li>Designate the "=" sign in the variable definition of the program.</li> </ul>
<b>P243</b>	<b>Can't use variables</b> An invalid variable has been specified in the left or right side of an operation expression.	<ul style="list-style-type: none"> <li>Correct the program.</li> </ul>
<b>P244</b>	<b>Invalid set date or time</b> Date or time was set earlier than current date or time in the system variables (#3011, #3012) when the credit system was valid.	<ul style="list-style-type: none"> <li>Date or time cannot be changed.</li> <li>Reconsider the program.</li> </ul>
<b>P250</b>	<b>No spec: Figure rotation</b> Figure rotation (M98 I_J_P_H_L_) was commanded even though there is no figure rotation specification.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P251</b>	<b>Figure rotation overlapped</b> Figure rotation command was issued during figure rotation.	<ul style="list-style-type: none"> <li>Check the machining program.</li> </ul>
<b>P252</b>	<b>Coord rotate in fig. rotation</b> A coordinate rotation related command (G68, G69) was issued during figure rotation.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> </ul>
<b>P260</b>	<b>No spec: Coordinates rotation</b> Even though there were no coordinate rotation specifications, a coordinate rotation command was issued.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P270</b>	<b>No spec: User macro</b> A macro specification was commanded though there are no such command specifications.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P271</b>	<b>No spec: Macro interrupt</b> A macro interruption command has been issued though it is not included in the specifications.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P272</b>	<b>NC and macro texts in a block</b> A statement and a macro statement exist together in the same block.	<ul style="list-style-type: none"> <li>• Reconsider the program and place the executable statement and macro statement in separate blocks.</li> </ul>
<b>P273</b>	<b>Macro call nesting over</b> The number of macro call nests exceeded the specifications.	<ul style="list-style-type: none"> <li>• Reconsider the program and correct it so that the macro calls do not exceed the limit imposed by the specification.</li> </ul>
<b>P275</b>	<b>Macro argument over</b> The number of macro call argument type II sets has exceeded the limit.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P276</b>	<b>Illegal G67 command</b> A G67 command was issued though it was not during the G66 command modal.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> <li>• The G67 command is the call cancel command and so the G66 command must be designated first before it is issued.</li> </ul>
<b>P277</b>	<b>Macro alarm message</b> An alarm command has been issued in #3000.	<ul style="list-style-type: none"> <li>• Refer to the operator messages on the DIAG screen.</li> <li>• Refer to the instruction manual issued by the machine tool builder.</li> </ul>
<b>P280</b>	<b>Brackets [ ] nesting over</b> The number of parentheses "[" or "]" which can be commanded in a single block has exceeded five.	<ul style="list-style-type: none"> <li>• Reconsider the program and correct it so the number of "[" or "]" is five or less.</li> </ul>
<b>P281</b>	<b>Brackets [ ] not paired</b> The number of "[" and "]" parentheses commanded in a single block does not match.	<ul style="list-style-type: none"> <li>• Reconsider the program and correct it so that "[" and "]" parentheses are paired up properly.</li> </ul>
<b>P282</b>	<b>Calculation impossible</b> The arithmetic formula is incorrect.	<ul style="list-style-type: none"> <li>• Reconsider the program and correct the formula.</li> </ul>
<b>P283</b>	<b>Divided by zero</b> The denominator of the division is zero.	<ul style="list-style-type: none"> <li>• Reconsider the program and correct it so that the denominator for division in the formula is not zero.</li> </ul>
<b>P290</b>	<b>IF sentence error</b> There is an error in the IF conditional GOTO□ statement.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P291</b>	<b>WHILE sentence error</b> There is an error in the WHILE conditional DO□-END□ statement.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P292</b>	<b>SETVN sentence error</b> There is an error in the SETVN□ statement when the variable name setting was made.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> <li>• The number of characters in the variable name of the SETVN statement must be 7 or less.</li> </ul>
<b>P293</b>	<b>DO-END nesting over</b> The number of DO-END nesting levels in WHILE conditional DO□-END□ statement has exceeded 27.	<ul style="list-style-type: none"> <li>• Reconsider the program and correct it so that the nesting levels of the DO - END statement does not exceed 27.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P294</b>	<b>DO and END not paired</b> The DO's and END's are not paired off properly.	<ul style="list-style-type: none"> <li>Reconsider the program and correct it so that the DO's and END's are paired off properly.</li> </ul>
<b>P295</b>	<b>WHILE/GOTO in tape</b> There is a WHILE or GOTO statement on the tape during tape operation.	<ul style="list-style-type: none"> <li>During tape operation, a program which includes a WHILE or GOTO statement cannot be executed and so the memory operation mode is established instead.</li> </ul>
<b>P296</b>	<b>No address (macro)</b> A required address has not been specified in the user macro.	<ul style="list-style-type: none"> <li>Review the program.</li> </ul>
<b>P297</b>	<b>Address-A error</b> The user macro does not use address A as a variable.	<ul style="list-style-type: none"> <li>Review the program.</li> </ul>
<b>P298</b>	<b>G200-G202 cmdnd in tape</b> User macro G200, G201, or G202 was specified during tape or MDI mode.	<ul style="list-style-type: none"> <li>Review the program.</li> </ul>
<b>P300</b>	<b>Variable name illegal</b> The variable names have not been commanded properly.	<ul style="list-style-type: none"> <li>Reconsider the variable names in the program and correct them.</li> </ul>
<b>P301</b>	<b>Variable name duplicated</b> The name of the variable has been duplicated.	<ul style="list-style-type: none"> <li>Correct the program so that the name is not duplicated.</li> </ul>
<b>P310</b>	<b>Not use GMSTB macro code</b> G, M, S, T, or B macro code was called during fixed cycle.	<ul style="list-style-type: none"> <li>Review the program.</li> <li>Review the parameter.</li> </ul>
<b>P350</b>	<b>No spec: Scaling command</b> The scaling command (G50, G51) was issued when the scaling specifications were not available.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P360</b>	<b>No spec: Program mirror</b> A mirror image (G50.1 or G51.1) command has been issued though the programmable mirror image specifications are not provided.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P370</b>	<b>No spec: Facing t-post MR</b> The facing turret mirror image specifications are not provided.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P371</b>	<b>Facing t-post MR illegal</b> Mirror image for facing tool posts was commanded to an axis for which external mirror image or parameter mirror image is valid. Mirror image for facing tool posts validating mirror image for a rotary axis was commanded.	<ul style="list-style-type: none"> <li>Check the program.</li> <li>Check the parameters.</li> </ul>
<b>P380</b>	<b>No spec: Corner R/C</b> The corner R/C was issued when the corner R/C specifications were not available.	<ul style="list-style-type: none"> <li>Check the specifications.</li> <li>Remove the corner chamfering/corner rounding command from the program.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P381</b>	<b>No spec: Arc R/C</b> Corner chamfering II /corner rounding II was specified in the arc interpolation block although corner chamfering/corner rounding II is unsupported.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P382</b>	<b>No corner movement</b> The block next to corner chamfering/ corner rounding is not a movement command.	<ul style="list-style-type: none"> <li>Replace the block succeeding the corner chamfering/corner rounding command by G01 command.</li> </ul>
<b>P383</b>	<b>Corner movement short</b> In the corner chamfering/corner rounding command, the movement distance was shorter than the value in the corner chamfering/corner rounding command.	<ul style="list-style-type: none"> <li>Make the corner chamfering/corner rounding less than the movement distance since this distance is shorter than the corner chamfering/ corner rounding.</li> </ul>
<b>P384</b>	<b>Corner next movement short</b> When the corner chamfering/corner rounding command was input, the movement distance in the following block was shorter than the length of the corner chamfering/corner rounding.	<ul style="list-style-type: none"> <li>Make the corner chamfering/corner rounding less than the movement distance since this distance in the following block is shorter than the corner chamfering/corner rounding.</li> </ul>
<b>P385</b>	<b>Corner during G00/G33</b> A block with corner chamfering/corner rounding was given during G00 or G33 modal.	<ul style="list-style-type: none"> <li>Recheck the program.</li> </ul>
<b>P390</b>	<b>No spec: Geometric</b> A geometric command was issued though there are no geometric specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P391</b>	<b>No spec: Geometric arc</b> There are no geometric IB specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P392</b>	<b>Angle &lt; 1 degree (GEOMT)</b> The angular difference between the geometric line and line is 1° or less.	<ul style="list-style-type: none"> <li>Correct the geometric angle.</li> </ul>
<b>P393</b>	<b>Inc value in 2nd block (GEOMT)</b> The second geometric block was specified by an incremental value.	<ul style="list-style-type: none"> <li>Specify this block by an absolute value.</li> </ul>
<b>P394</b>	<b>No linear move command (GEOMT)</b> The second geometric block contains no linear command.	<ul style="list-style-type: none"> <li>Specify the G01 command.</li> </ul>
<b>P395</b>	<b>Illegal address (GEOMT)</b> The geometric format is invalid.	<ul style="list-style-type: none"> <li>Recheck the program.</li> </ul>
<b>P396</b>	<b>Plane selected in GEOMT ctrl</b> A plane switching command was executed during geometric command processing.	<ul style="list-style-type: none"> <li>Execute the plane switching command before geometric command processing.</li> </ul>
<b>P397</b>	<b>Arc error (GEOMT)</b> In geometric IB, the circular arc end point does not contact or cross the next block start point.	<ul style="list-style-type: none"> <li>Recheck the geometric circular arc command and the preceding and following commands.</li> </ul>
<b>P398</b>	<b>No spec: Geometric1B</b> Although the geometric IB specifications are not included, a geometric command is given.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P411</b>	<b>Illegal modal G111</b> <ul style="list-style-type: none"> <li>• G111 was issued during milling mode.</li> <li>• G111 was issued during nose R compensation mode.</li> <li>• G111 was issued during constant surface speed.</li> <li>• G111 was issued during mixed synchronization control.</li> <li>• G111 was issued during fixed cycle.</li> <li>• G111 was issued during polar coordinate interpolation.</li> <li>• G111 was issued during cylindrical interpolation mode.</li> </ul>	<ul style="list-style-type: none"> <li>• Before commanding G111, cancel the following commands. <ul style="list-style-type: none"> <li>• Milling mode</li> <li>• Nose R compensation</li> <li>• Constant surface speed</li> <li>• Mixed synchronization control</li> <li>• Fixed cycle</li> <li>• Polar coordinate interpolation</li> <li>• Cylindrical interpolation</li> </ul> </li> </ul>
<b>P412</b>	<b>P412 No spec: Axis name switch</b> Axis name switch (G111) was issued even though there is no axis name switch (G111) specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P420</b>	<b>No spec: Para input by program</b> Parameter input by program (G10) was commanded even though there is no specification of parameter input by program.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P421</b>	<b>Parameter input error</b> <ul style="list-style-type: none"> <li>• The specified parameter No. or set data is illegal.</li> <li>• An illegal G command address was input in parameter input mode.</li> <li>• A parameter input command was input during fixed cycle modal or nose R compensation.</li> <li>• G10L50, G10L70, G11 were not commanded in independent blocks.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
<b>P430</b>	<b>R-pnt return incomplete</b> <ul style="list-style-type: none"> <li>• A command was issued to move an axis, which has not returned to the reference position, away from that reference position.</li> <li>• A command was issued to an axis removal axis.</li> </ul>	<ul style="list-style-type: none"> <li>• Execute reference position return manually.</li> <li>• The command was issued to an axis for which axis removal is validated so invalidate axis removal.</li> </ul>
<b>P431</b>	<b>No spec: 2,3,4th R-point ret</b> A command for second, third or fourth reference position return was issued though there are no such command specifications.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P432</b>	<b>No spec: Start position return</b> Start position return (G29) was commanded even though there is no start position return specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P433</b>	<b>No spec: R-position check</b> Reference position check (G27) was commanded even though there is no reference position check specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P434</b>	<b>Compare error</b> One of the axes did not return to the reference position when the reference position check command (G27) was executed.	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
<b>P435</b>	<b>G27 and M commands in a block</b> An M command was issued simultaneously in the G27 command block.	<ul style="list-style-type: none"> <li>• An M code command cannot be issued in a G27 command block and so the G27 command and M code command must be placed in separate blocks.</li> </ul>
<b>P436</b>	<b>G29 and M commands in a block</b> An M command was issued simultaneously in the G29 command block.	<ul style="list-style-type: none"> <li>• An M code command cannot be issued in a G29 command block and so the G29 command and M code command must be placed in separate blocks.</li> </ul>
<b>P438</b>	<b>G52 invalid during G54.1</b> A local coordinate system command was issued during execution of the G54.1 command.	<ul style="list-style-type: none"> <li>• Review the program.</li> </ul>
<b>P450</b>	<b>No spec: Chuck barrier</b> The chuck barrier on command (G22) was specified although the chuck barrier was undefined in the specification.	<ul style="list-style-type: none"> <li>• Check the specification.</li> </ul>
<b>P451</b>	<b>No spec: Stroke chk bef travel</b> Stroke check before travel (G22/G23) was commanded even though there is no stroke check before travel specification.	<ul style="list-style-type: none"> <li>• Check the specification.</li> </ul>
<b>P452</b>	<b>Limit before travel exists</b> An illegal command such as the start or end point of the traveling axis is inside the prohibited area or the axis passes through the prohibited area, was detected when Stroke check before travel (G22) was ON.	<ul style="list-style-type: none"> <li>• Review the coordinate values of the axis address commanded in the program.</li> </ul>
<b>P460</b>	<b>Tape I/O error</b> An error has arisen in the tape reader or, alternatively, in the printer during macro printing.	<ul style="list-style-type: none"> <li>• Check the power and cable of the connected devices.</li> <li>• Check the I/O device parameters.</li> </ul>
<b>P461</b>	<b>File I/O error</b> A file of the machining program cannot be read. IC card has not been inserted.	<ul style="list-style-type: none"> <li>• In memory mode, the programs stored in memory may have been destroyed. Output all of the programs and tool data once and format them.</li> <li>• Ensure that the external device (including an IC card, etc) that contains the file is mounted.</li> <li>• Check the parameters for HD operation or IC card operation.</li> </ul>
<b>P462</b>	<b>Computer link commu error</b> A communication error occurred during the BTR operation.	<ul style="list-style-type: none"> <li>• "L01 Computer link error" is displayed simultaneously, so remedy the problem according to the error No.</li> </ul>
<b>P480</b>	<b>No spec: Milling</b> <ul style="list-style-type: none"> <li>• Milling was commanded when the milling specifications were not provided.</li> <li>• Polar coordinate interpolation was commanded when the polar coordinate interpolation specifications were not provided.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the specification.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P481</b>	<b>Illegal G code (mill)</b> <ul style="list-style-type: none"> <li>• An illegal G code was used during the milling mode.</li> <li>• An illegal G code was used during cylindrical interpolation or polar coordinate interpolation.</li> <li>• The G07.1 command was issued during the tool radius compensation.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
<b>P482</b>	<b>Illegal axis (mill)</b> <ul style="list-style-type: none"> <li>• A rotary axis was commanded during the milling mode.</li> <li>• Milling was executed even though an illegal value was set for the milling axis No.</li> <li>• Cylindrical interpolation or polar coordinate interpolation was commanded during mirror image.</li> <li>• Cylindrical interpolation or polar coordinate interpolation was commanded before the tool compensation was completed after the T command.</li> <li>• G07.1 was commanded when cylindrical interpolation was not possible (there is no rotary axis, or external mirror image is ON).</li> <li>• An axis other than a cylindrical coordinate system axis was commanded during cylindrical interpolation.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the machining program, parameters and PLC I/F signal.</li> </ul>
<b>P484</b>	<b>R-pnt ret incomplete (mill)</b> <ul style="list-style-type: none"> <li>• Movement was commanded to an axis that had not completed reference position return during the milling mode.</li> <li>• Movement was commanded to an axis that had not completed reference position return during cylindrical interpolation or polar coordinate interpolation.</li> </ul>	<ul style="list-style-type: none"> <li>• Carry out manual reference position return.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P485</b>	<b>Illegal modal (mill)</b> <ul style="list-style-type: none"> <li>• The milling mode was turned ON during nose R compensation or constant surface speed control.</li> <li>• A T command was issued during the milling mode.</li> <li>• The mode was switched from milling to cutting during tool compensation.</li> <li>• Cylindrical interpolation or polar coordinate interpolation was commanded during the constant surface speed control mode (G96).</li> <li>• The command unacceptable in the cylindrical interpolation was issued.</li> <li>• A T command was issued during the cylindrical interpolation or polar coordinate interpolation mode.</li> <li>• A movement command was issued when the plane was not selected just before or after the G07.1 command.</li> <li>• A plane selection command was issued during the polar coordinate interpolation mode.</li> <li>• Cylindrical interpolation or polar coordinate interpolation was commanded during tool radius compensation.</li> <li>• The G16 plane in which the radius value of a cylinder is 0 was specified.</li> <li>• A cylindrical interpolation or polar coordinate interpolation command was issued during coordinate rotation by program (G68).</li> </ul>	<ul style="list-style-type: none"> <li>• Check the program.</li> <li>• Before issuing G12.1, issue G40 or G97.</li> <li>• Before issuing G12.1, issue a T command.</li> <li>• Before issuing G13.1, issue G40.</li> <li>• Specify the radius value of a cylinder other than 0, or specify the X axis's current value other than 0 before issuing G12.1/G16.</li> </ul>
<b>P486</b>	<b>Milling error</b> <ul style="list-style-type: none"> <li>• The milling command was issued during the mirror image (when parameter or external input is turned ON).</li> <li>• Polar coordinate interpolation, cylindrical interpolation or milling interpolation was commanded during mirror image for facing tool posts.</li> <li>• The start command of the cylindrical interpolation or polar coordinate interpolation was issued during the normal line control.</li> </ul>	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
<b>P511</b>	<b>Synchronization M code error</b> <ul style="list-style-type: none"> <li>• Two or more synchronization M codes were commanded in the same block.</li> <li>• The synchronization M code and "!" code were commanded in the same block.</li> <li>• Synchronization with the M code was commanded in 3rd part system or more. (Synchronization with the M code is valid only in 1st part system or 2nd part system.)</li> </ul>	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
<b>P550</b>	<b>No spec: G06.2(NURBS)</b> There is no NURBS interpolation option.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P551</b>	<b>G06.2 knot error</b> The knot (k) command value is smaller than the value for the previous block.	<ul style="list-style-type: none"> <li>Reconsider the program.</li> <li>Specify the knot by monotone increment.</li> </ul>
<b>P552</b>	<b>Start point of 1st G06.2 err</b> The block end point immediately before the G06.2 command and the G06.2 first block command value do not match.	<ul style="list-style-type: none"> <li>Match the G06.2 first block coordinate command value with the previous block end point.</li> </ul>
<b>P554</b>	<b>Invlid manual interrupt in G6.2</b> Manual interruption using a block was performed while in G06.2 mode.	<ul style="list-style-type: none"> <li>Perform for blocks other than G06.2 mode when manually interrupting.</li> </ul>
<b>P555</b>	<b>Invalid restart during G06.2</b> Restart was attempted from the block in G06.2 mode.	<ul style="list-style-type: none"> <li>Restart from the block other than in G06.2 mode.</li> </ul>
<b>P600</b>	<b>No spec: Auto TLM</b> An automatic tool length measurement command (G37) was execute though there are no such command specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P601</b>	<b>No spec: Skip</b> A skip command (G31) was issued though there are no such command specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P602</b>	<b>No spec: Multi skip</b> A multiple skip command (G31.1, G31.2 or G31.3) was issued though there are no such command specifications.	<ul style="list-style-type: none"> <li>Check the specifications.</li> </ul>
<b>P603</b>	<b>Skip speed 0</b> The skip speed is 0.	<ul style="list-style-type: none"> <li>Specify the skip speed.</li> </ul>
<b>P604</b>	<b>TLM illegal axis</b> No axis or more than one axis was specified in the automatic tool length measurement block.	<ul style="list-style-type: none"> <li>Specify only one axis.</li> </ul>
<b>P605</b>	<b>T &amp; TLM command in a block</b> The T code is in the same block as the automatic tool length measurement block.	<ul style="list-style-type: none"> <li>Specify this T code before the block.</li> </ul>
<b>P606</b>	<b>T cmnd not found before TLM</b> The T code was not yet specified in automatic tool length measurement.	<ul style="list-style-type: none"> <li>Specify this T code before the block.</li> </ul>
<b>P607</b>	<b>TLM illegal signal</b> Before the area specified by the D command or decelerating area parameter d, the measurement position arrival signal went ON. The signal remains OFF to the end.	<ul style="list-style-type: none"> <li>Check the program.</li> </ul>
<b>P608</b>	<b>Skip during radius compen</b> A skip command was specified during radius compensation processing.	<ul style="list-style-type: none"> <li>Specify a radius compensation cancel (G40) command or remove the skip command.</li> </ul>
<b>P610</b>	<b>Illegal parameter</b> <ul style="list-style-type: none"> <li>The parameter setting is not correct.</li> <li>G114.1 was commanded when the spindle synchronization with PLC I/F command was selected.</li> <li>G113 was commanded when the spindle-spindle polygon machining option was OFF and the spindle synchronization with PLC I/F command was selected.</li> </ul>	<ul style="list-style-type: none"> <li>Check whether "#1549 lv0vR1" to "#1553 lv0vR5" are set in descending order (in order of large values).</li> <li>Check whether "#1554 lv0rd2" to "#1557 lv0rd5" are set in descending order.</li> <li>Check and correct "#1514 expLinax" and "#1515 expRotax".</li> <li>Check the program.</li> <li>Check the parameter.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P611</b>	<b>No spec: Exponential function</b> Specification for exponential interpolation is not available.	<ul style="list-style-type: none"> <li>• Check the specification.</li> </ul>
<b>P612</b>	<b>Exponential function error</b> A movement command for exponential interpolation was issued during mirror image for facing tool posts.	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
<b>P700</b>	<b>Illegal command value</b> Spindle synchronization was commanded to a spindle that is not connected serially.	<ul style="list-style-type: none"> <li>• Check the program.</li> <li>• Check the parameter.</li> </ul>
<b>P900</b>	<b>No spec: Normal line control</b> A normal line control command (G40.1, G41.1, G42.1) was issued when the normal line control specifications were not provided.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P901</b>	<b>Normal line control axis G92</b> A coordinate system preset command (G92) was issued to a normal line control axis during normal line control.	<ul style="list-style-type: none"> <li>• Check the program.</li> </ul>
<b>P902</b>	<b>Normal line control axis error</b> <ul style="list-style-type: none"> <li>• The normal line control axis was set to a linear axis.</li> <li>• The normal line control axis was set to the linear type rotary axis II axis.</li> <li>• The normal line control axis has not been set.</li> <li>• The normal line control axis was the same as the plane selection axis.</li> </ul>	<ul style="list-style-type: none"> <li>• Correct the normal line control axis.</li> </ul>
<b>P903</b>	<b>Plane chg in Normal line ctrl</b> The plane selection command (G17, G18, G19) was issued during normal line control.	<ul style="list-style-type: none"> <li>• Delete the plane selection command (G17, G18, G19) from the program for normal line control.</li> </ul>
<b>P920</b>	<b>No spec: 3D coord conv</b> There is no specification for 3-dimensional coordinate conversion.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P921</b>	<b>Illegal G code at 3D coord</b> A G code command that cannot be performed was made during 3-dimensional coordinate conversion modal.	<ul style="list-style-type: none"> <li>• Refer to "Mitsubishi CNC 700/70 Series Programming Instruction Manual (Machining Center Series)" for further details of usable G commands.</li> <li>• When the basic specification parameter "#1229 set01/bit3" is ON, turn the parameter OFF or specify the constant surface speed control cancel (G97).</li> </ul>
<b>P922</b>	<b>Illegal mode at 3D coord</b> A 3-dimensional coordinate conversion command was issued during a modal for which 3-dimensional coordinate conversion cannot be performed.	<ul style="list-style-type: none"> <li>• Refer to "Mitsubishi CNC 700/70 Series Programming Instruction Manual (Machining Center Series)" for further details of usable G commands.</li> </ul>
<b>P923</b>	<b>Illegal addr in 3D coord blk</b> A G code for which G68 to combination could not be performed was specified for the same block.	<ul style="list-style-type: none"> <li>• Refer to "Mitsubishi CNC 700/70 Series Programming Instruction Manual (Machining Center Series)" for further details of usable G commands.</li> </ul>

## Appendix 1. Program Error

Error No.	Details	Remedy
<b>P930</b>	<b>No spec: Tool axis compen</b> A tool length compensation along the tool axis command was issued even though there is no tool length compensation along the tool axis specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P931</b>	<b>Executing tool axis compen</b> A G code that cannot be commanded exists during tool length compensation along the tool axis.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P932</b>	<b>Rot axis parameter error</b> There is a mistake in the linear axis name and rotary axis name in the rotary axis configuration parameters.	<ul style="list-style-type: none"> <li>• Set the correct value and reboot.</li> </ul>
<b>P940</b>	<b>No spec: Tool tip control</b> There is no tool tip center control specification.	<ul style="list-style-type: none"> <li>• Check the specifications.</li> </ul>
<b>P941</b>	<b>Invalid T tip control command</b> A tool tip center control command was issued during a modal for which a tool tip center control command cannot be issued.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P942</b>	<b>Invalid cmd during T tip ctrl</b> A G code that cannot be commanded was issued during tool tip center control.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P943</b>	<b>Tool posture command illegal</b> In the case of tool tip center control type 1, if the signs at the tool-side rotary axis or table base-side rotary axis start and finish points differ, a tool base-side rotary axis or table workpiece-side rotary axis rotation exists for the same block, and does not pass a singular point. In the case of tool tip center control type 2, the posture vector command is incorrect.	<ul style="list-style-type: none"> <li>• Reconsider the program.</li> </ul>
<b>P990</b>	<b>PREPRO error</b> Combining commands that required pre-reading (nose R offset, corner chamfering/corner rounding, geometric I, geometric IB, and compound type fixed cycle for turning machining) resulted in eight or more pre-read blocks.	<ul style="list-style-type: none"> <li>• Reduce the number of commands that require pre-reading or delete such commands.</li> </ul>

## Appendix 2. Order of G Function Command Priority (Command in a separate block when possible)

(Note) Upper level: When commanded in the same block indicates that both commands are executed simultaneously

G Group Commanded G code	01 G00 to G03	02 G17 to G19	03 G90, G91		05 G94, G95	06 G20, G21	07 G40 to G42	08 G43, G44, G49
G00 to G03.1 Positioning/ interpolation	G command commanded last is valid.  Group 1 modal is updated	<input type="radio"/>  <input type="radio"/> Also possible during arc modal	<input type="radio"/>  <input type="radio"/>		<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	Arc and G41, G42 cause error P151  <input type="radio"/> Tool radius is compensated, and then axes move.	Arc and G43 to G49 cause error P70  <input type="radio"/> The G49 movement in the arc modal moves with G01
G04 Dwell	<input type="radio"/> Group 1 modal is updated  G04 is executed  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>		<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	G04 is executed G40 to G42 are ignored  <input type="radio"/>	G04 is executed G43 to G49 are ignored  <input type="radio"/>
G09 Exact stop check	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>		<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>
G10, G11 Program data setting	<input type="radio"/> G10 is priority for axis No movement I, J, K rotation input  <input type="radio"/>	<input type="radio"/> G10 is used for axis, so the selected plan axis will be the basic axis.  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>		<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	G10 to G11 are executed G40 to G42 are ignored  <input type="radio"/>	G10 to G11 are executed G43 to G49 are ignored  <input type="radio"/>
G17 to G19 Plane selection	<input type="radio"/>  <input type="radio"/>	<input type="radio"/> G command commanded last is valid.  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>		<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/> Plane axis changeover during tool radius compensation causes error P112	<input type="radio"/>  <input type="radio"/>

## Appendix 2. Order of G Function Command Priority

G Group Commanded G code	01 G00 to G03	02 G17 to G19	03 G90, G91		05 G94, G95	06 G20, G21	07 G40 to G42	08 G43, G44, G49
G20, G21 Inch/metric changeover	○ ○	○ ○	○ ○		○ ○	Possible in same block ○	○ ○	○ ○
G27 to G30 Reference position compare/ return	○ G00 to G03.1 modals are updated G27 to G30 are executed ○	○ ○	○ ○		○ ○	○ ○	G27 to G30 are executed G40 to G42 are ignored ○	G27 to G30 are executed G43 to G49 are ignored ○
G31 to G31.3 Skip	○ ○	○ ○	○ ○		○ ○	○ ○	Error:P608 Error:P608	○ ○
G33 Thread cutting	G command commanded last is valid. ○	○ ○	○ ○		○ ○	○ ○	○ ○	○ ○
G37 Automatic tool length measurement	G37 is executed G00 to G33 are ignored ○	○ ○	○ ○		○ ○	○ ○	G37 is executed G40 to G42 are ignored	G37 is executed G43 to G49 are ignored ○
G40 to G42 Tool radius compensation	Arc and G41, G42 cause error P151 G41 and G42 in arc modal cause error P151	○ Plane axis changeover during tool radius compensation causes error P112	○ ○		○ ○	○ ○	G command commanded last is valid. ○	○ ○

## Appendix 2. Order of G Function Command Priority

G Group Commanded G code	09 G73 to G89	10 G98, G99	12 G54 to G59	13 G61 to G64	14 G66 to G67	17 G96, G97	19 G50.1 G51.1
G00 to G03.1 Positioning/ interpolation	Group 1 command is executed Group 9 is canceled	○	○	○	G66 to G67 are executed G00 to G03.1 modals are updated	○	○ During the arc command, all axis names become mirror center data Movement with mirror shape ○
G04 Dwell	G04 is executed G73 to G89 are ignored ○	○	G04 is executed Group 12 is changed ○	○	○	○	○ G04 is executed G50.1 and G51.1 are ignored ○
G09 Exact stop check	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G10, G11 Program data setting	G10 to G11 are executed G73 to G89 are ignored ○	○	G10 is executed G54 to G59 modals are updated ○	○	G66 to G67 are executed G10 is ignored ○	○	○ G10 to G11 are executed G50.1 and G51.1 are ignored ○
G17 to G19 Plane selection	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○

## Appendix 2. Order of G Function Command Priority

<b>G Group</b>	<b>09</b>	<b>10</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>17</b>		<b>19</b>
<b>Commanded G code</b>	<b>G73 to G89</b>	<b>G98, G99</b>	<b>G54 to G59</b>	<b>G61 to G64</b>	<b>G66 to G67</b>	<b>G96, G97</b>		<b>G50.1 G51.1</b>
G20, G21 Inch/metric changeover	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○		○ ○
G27 to G30 Reference position compare/ return	○ ○	○ ○	○ ○	○ ○	G66 to G67 are executed G27 to G30 are ignored	○ ○		G27 to G30 are executed G50.1 and G51.1 are ignored ○
G31 to G31.3 Skip	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○		○ ○
G33 Thread cutting	Group 1 command is executed Group 9 is canceled ○	○ ○	○ ○	○ ○	G66 to G67 are executed G33 modals is updated ○	○ ○		○ ○
G37 Automatic tool length measurement		○ ○	○ ○	○ ○	G66 to G67 are executed G37 modals is ignored ○	○ ○		G37 is executed G50.1 and G51.1 are ignored ○
G40 to G42 Tool radius compensation	Error:P155 Error:P155	○ ○	○ ○	○ ○	○ ○	○ ○		○ ○

## Appendix 2. Order of G Function Command Priority

G Group Commanded G code	01 G00 to G03.1 G33	02 G17 to G19	03 G90, G91		05 G94, G95	06 G20, G21	07 G40 to G42	08 G43, G44, G49
G43, G44, G49 Length compensation	Arc and G43, G44 cause error P70  Arc and G43, G44 cause error P70	○	○		○	○	○	○ G command commanded last is valid.  ○
G50.1 G51.1 Program mirror image	○ ○	○ ○	○ ○		○ ○	○ ○	○ ○	○ ○
G52 Local coordinate system	○ ○	○ ○	○ ○		○ ○	○ ○	G52 is executed G40 to G42 are ignored ○	G52 is executed G43 to G49 are ignored ○
G53 Machine coordinate system	○ ○	○ ○	○ ○		○ ○	○ ○	G53 is executed G40 to G42 are ignored ○	G53 is executed G40 to G42 are ignored ○
G54 to G59 Workpiece coordinate system	○ ○	○ ○	○ ○		○ ○	○ ○	○ ○	○ ○
G61 to G64 Mode selection	○ ○	○ ○	○ ○		○ ○	○ ○	○ ○	○ ○
G65 Macro call	G65 is executed G00 to G03.1 modals are updated ○	○ ○	○ ○		○ ○	○ ○	○ ○	G65 is executed G43 to G49 modals are updated ○

## Appendix 2. Order of G Function Command Priority

G Group Commanded G code	01 G00 to G03.1 G33	02 G17, G19	03 G90, G92		05 G94, G95	06 G20, G21	07 G40 to G42	08 G43, G44 G49
G66 to G67 Macro call	G66 to G67 are executed G00 to G03.1 modals are updated  ○	○  ○	○  ○		○  ○	○  ○	○  ○	○ G66 to G67 are executed G43 to G49 modals are updated ○
G73 to G89 Canned cycle	G73 to G89 are canceled G01 to G33 modals are updated ○	○  ○	○  ○		○  ○	○  ○	Error:P155 Canned cycle during compensation  Error:P155	○  ○
G90, G91 Absolute value/ incremental/ value	○ ○	○ ○	Use in same block ○		○ ○	○ ○	○ ○	○ ○
G92 Coordinate system setting	○ ○	○ ○	○ ○		○ ○	○ ○	○ ○	○ ○
G94, G95 Synchronous/ asynchronous	○ ○	○ ○	○ ○		G command commanded last is valid. ○	○ ○	○ ○	○ ○
G96, G97 Constant surface speed control	○ ○	○ ○	○ ○		○ ○	○ ○	○ ○	○ ○
G98, G99 Initial point/ R point return	○ ○	○ ○	○ ○		○ ○	○ ○	○ ○	○ ○

## Appendix 2. Order of G Function Command Priority

G Group Commanded G code	09 G73 to G89	10 G98 to G99	12 G54 to G59	13 G61 to G65	14 G66 to G67	17 G96, G97		19 G50.1 G51.1
G43, G44, G49 Length compensation	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G66 to G67 are executed G43 to G49 modals are updated	<input type="radio"/>		<input type="radio"/>
G50.1 G51.1 Program mirror image	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G66 to G67 are executed G50.1 G51.1 is ignored	<input type="radio"/>		G command commanded last is valid.  <input type="radio"/>
G52 Local coordinate system	G52 is executed G73to G89 are ignored	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		<input type="radio"/>		G52 is executed G50.1 G51.1 is ignored <input type="radio"/>
G53 Machine coordinate system		<input type="radio"/>	<input type="radio"/>	<input type="radio"/>		<input type="radio"/>		G53 is executed G50.1 G51.1 is invalid <input type="radio"/>
G54 to G59 Workpiece coordinate system	<input type="radio"/>	<input type="radio"/>	G command commanded last is valid.	<input type="radio"/>	G66 to G67 are executed G54 to G59 modals are updated	<input type="radio"/>		<input type="radio"/>
G61 to G64 Mode selection	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G command commanded last is valid.	<input type="radio"/>	<input type="radio"/>		<input type="radio"/>
G65 Macro call	G65 is executed G73to G89 are ignored	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Error	<input type="radio"/>		G65 is executed G50.1 G51.1 is ignored <input type="radio"/>

## Appendix 2. Order of G Function Command Priority

G Group Commanded G code	09 G73 to G89	10 G98, G99	12 G54 to G59	13 G61 to G67	14 G66 to G67	17 G96, G97		19 G50.1 G51.1
G66 to G67 Macro call	G66 to G67 are executed G73 to G89 are ignored  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	G66 to G67 are executed G54 to G59 modals are updated  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	G command commanded last is valid.  G command commanded last is valid. <input type="radio"/>	<input type="radio"/>  <input type="radio"/>		G66 to G67 are executed G50.1 G51.1 is ignored <input type="radio"/>
G73 to G89 Canned cycle	G command commanded last is valid.  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	G66 to G67 are executed G73 to G89 are ignored  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>		<input type="radio"/> All axes become mirror center <input type="radio"/>
G90, G91 Absolute value/ incremental value	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>		<input type="radio"/> <input type="radio"/>
G92 Coordinate system setting	G92 is executed G73 to G89 are ignored  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>		<input type="radio"/> Note that G92 is priority for axis <input type="radio"/>
G94, G95 Synchronous/ asynchronous	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>		<input type="radio"/> <input type="radio"/>
G96, G97 Constant surface speed control	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	G command commanded last is valid.  <input type="radio"/>		<input type="radio"/>  <input type="radio"/>
G98, G99 Initial point/R point return	<input type="radio"/>  <input type="radio"/>	G command commanded last is valid.  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>	<input type="radio"/>  <input type="radio"/>		<input type="radio"/>  <input type="radio"/>



INDEX

**Numbers**

2nd, 3rd and 4th Reference (Zero) Position Return; G30 .....515  
 3-dimensional Circular Interpolation; G02.4, G03.4 .....95  
 3-dimensional Coordinate Conversion; G68.1/69.1 .....458

**A**

Actual Examples of Using User Macros .....365  
 Arithmetic Commands .....353  
 ASCII Code Macro .....309  
 Automatic Coordinate System Setting .....510  
 Automatic Corner Override .....138  
 Automatic Tool Length Measurement; G37.1 .....543

**B**

Basic Machine Coordinate System Selection; G53 .....508  
 Basic Machine, Workpiece and Local Coordinate Systems .....506  
 Buffer Register .....22

**C**

Calling Subprogram with G22 and G22 Commands .....287  
 Changing of Compensation No. During Compensation Mode .....203  
 Circle Cutting; G12, G13 .....380  
 Circular Interpolation; G02, G03 .....44  
 Compensation Data Input to Variable by Program; G11 .....241  
 Constant Lead Thread Cutting; G33 .....56  
 Constant Surface Speed Control .....152  
 Constant Surface Speed Control; G96, G97 .....152  
 Control Axes .....1  
 Control Commands .....358  
 Coordinate Read Function; G14 .....502  
 Coordinate Rotation by Program; G68.1/G69.1 .....447  
 Coordinate Rotation Input by Parameter; G10 .....455  
 Coordinate System for Rotary Axis .....540  
 Coordinate System Setting; G92 .....509  
 Coordinate Systems and Coordinate Zero Point Symbols .....2  
 Coordinate Words and Control Axes .....1, 505  
 Coordinates System Setting Functions .....505  
 Corner Chamfering " ,C\_ " .....372  
 Corner Chamfering/Corner Rounding I .....372  
 Corner Rounding " ,R\_ " .....374  
 Cutting Feedrate .....107  
 Cutting Mode; G64 .....143  
 Cylindrical Interpolation; G07.1 .....63

**D**

Data Formats .....7  
 Deceleration Check .....134  
 Decimal Point Input .....28  
 Diameter Designation of Compensation Amount .....214  
 Drilling Cycle with High-Speed Retract .....276  
 Dwell .....144

**E**

End Point Error Check Cancellation; G69 .....500  
 Exact Stop Check Mode; G61 .....134  
 Exact Stop Check; G09 .....131  
 Exponential Function Interpolation; G02.3, G03.3 .....78  
 External Output Commands .....361

**F**

F1-digit Feed .....108  
 Feed Functions .....107  
 Feedrate Designation and Effects on Control Axes .....116  
 Figure Rotation; G22 I\_ J\_ K\_ .....291  
 Fixed Cycles .....248

**G**

G Code Lists .....18  
 G Command Mirror Image; G50.1, G51.1 / G62 .....369  
 G1 -> G0 Deceleration Check .....136  
 G1 -> G1 Deceleration Check .....137  
 G41/G42 Commands and I, J, K Designation .....194  
 General Precautions for Tool Radius Compensation .....202  
 Geometric Command .....376

**H**

Helical Interpolation ; G17 to G19, G02, G03 .....52  
 High-accuracy Control ; G61.1, G08 .....405  
 High-accuracy Spline Interpolation ; G61.2 .....440  
 High-speed High-accuracy Control I, II .....422  
 High-speed High-accuracy Control; G05, G05.1 .....422  
 High-speed Machining Mode I,II; G05 P1, G05 P2 .....419  
 High-speed Machining Mode; G05, G05.1 .....419  
 Hypothetical Axis Interpolation; G07 .....105

**I**

Inch Thread Cutting; G33 .....60  
 Inch/Metric Command Change; G20, G21 .....26  
 Index Table Indexing .....149  
 Indexing Increment .....6  
 Initial Point and R Point Level Return; G98, G99 .....279  
 Input Buffer .....22  
 Input Command Increment Tenfold .....5  
 Input Setting Units .....3  
 Inputting the Tool Life Management Data by G10 L3 Command .....242  
 Inputting the Tool Life Management Data by G10 L30 Command .....244  
 Inputting the Tool Life Management Data; G10, G11 .....242  
 Interference Check .....207  
 Interpolation Functions .....33  
 Interrupts During Tool Radius Compensation .....200  
 Inverse Time Feed; G93 .....112

**L**

Least Command Increments .....3  
 Linear Angle Command .....375  
 Linear Interpolation; G01 .....40  
 Local Coordinate System Setting; G52 .....531

**M**

Machine Zero Point and 2nd, 3rd, 4th Reference Positions .....507  
 Macro Call Command .....300  
 Macro Interrupt; ION, IOF .....383  
 Measurement Support Functions .....543  
 Miscellaneous Functions .....146  
 Miscellaneous Functions (M8-digits BCD) .....146  
 Multiple Spindle Control .....157  
 Multiple Spindle Control II .....158  
 Multi-step Skip Function 2; G31 .....554  
 Multi-step Skip Function; G31.n, G04 .....552

# INDEX

## N

Normal Line Control ; G40.1/G41.1/G42.1 .....	394
NURBS Interpolation .....	100

## O

Optional Block Skip .....	13
Optional Block Skip ; / .....	13
Optional Block Skip Addition ; /n .....	14
Order of G Function Command Priority .....	582
Other Commands and Operations During Tool Radius Compensation .....	185

## P

Parameter Input by Program; G10, G11.1 .....	382
Parity H/V .....	17
Per-minute/Per-revolution Feed (Asynchronous/Synchronous Feed); G94, G95 .....	110
Per-second Dwell ; G04 .....	144
Plane Selection; G17, G18, G19 .....	42
Polar Coordinate Command; G16/G15 .....	84
Polar Coordinate Interpolation; G12.1, G13.1 .....	71
Position Command Methods ; G90, G91 .....	24
Position Commands .....	24
Positioning (Rapid Traverse); G00 .....	33
Precautions .....	363
Precautions Before Starting Machining .....	21
Precautions for Inputting the Tool Life Management Data .....	247
Pre-read Buffers .....	23
Program Error .....	562
Program Formats .....	10
Program Support Functions .....	248
Program/Sequence/Block Numbers ; L(O), N .....	16
Programmable Current Limitation .....	559
Programmed Compensation Input; G10, G11.1 .....	236

## R

Rapid Traverse Constant Inclination Acceleration/Deceleration .....	120
Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration .....	122
Rapid Traverse Rate .....	107
Reference (Zero) Position Return; G28, G29 .....	511
Reference Position Check; G27 .....	518
R-specified Circular Interpolation; G02, G03 .....	49

## S

Scaling; G50/G51 .....	442
Secondary Miscellaneous Functions (B8-digits, A8 or C8-digits) .....	148
Setting of Workpiece Coordinates in Fixed Cycle Mode .....	281
Skip Function; G31 .....	547
Special Fixed Cycle; G34, G35, G36, G37 .....	282
Speed Change Skip; G31 .....	556
Spindle Clamp Speed Setting; G92 .....	153
Spindle Functions .....	151
Spindle Functions (S6-digits Analog) .....	151
Spindle Functions (S8-digits) .....	151
Spindle/C Axis Control .....	154
Spiral/Conical Interpolation; G02.0/G03.1(Type1), G02/G03(Type2) ..	90
Spline; G05.1 .....	433
SSS Control .....	428
Standard Fixed Cycles; G80 to G89, G73, G74, G75, G76 .....	248
Start of Tool Radius Compensation and Z Axis Cut in Operation .....	205
Stroke Check Before Travel; G22.1/G23.1 .....	560
Subprogram Control; G22, G22 .....	287

## T

Tape Codes .....	7
Tape Memory Format .....	13
Tapping Mode; G63 .....	143
Thread Cutting .....	56
Three-dimensional Tool Radius Compensation ; G40/G41,G42 .....	218
Timing-synchronization between Part Systems .....	497
Tool Center Point Control; G43.4/G43.5 .....	475
Tool Change Position Return; G30.1 to G30.6 .....	391
Tool Compensation .....	161
Tool Compensation Functions .....	161
Tool Functions (T command) .....	160
Tool Functions (T8-digit BCD) .....	160
Tool Length Compensation in the Tool Axis Direction ; G43.1/G44 ..	168
Tool Length Compensation/Cancel; G43/G44 .....	165
Tool Position Offset; G45 to G48 .....	229
Tool Radius Compensation Operation .....	176
Tool Radius Compensation; G38, G39/G40/G41,G42 .....	175
Types of Variables .....	315

## U

Unidirectional Positioning; G60 .....	61
User Macro Commands; G65, G66, G66.1, G67, G68(G23) .....	299
User Macro Specifications .....	299

## V

Variable Commands .....	294
Variables .....	313

## W

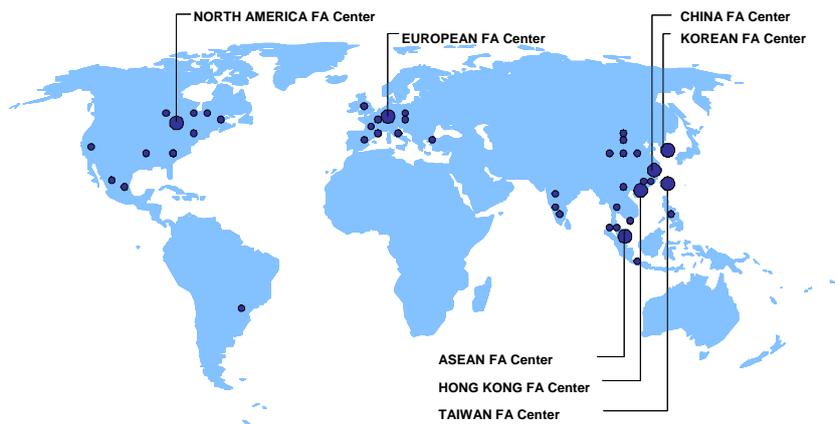
Workpiece Coordinate Changing During Radius Compensation .....	216
Workpiece Coordinate System Preset; G92.1 .....	535
Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1) .....	519

## Revision History

Date of revision	Manual No.	Revision details
Jun. 2005	IB(NA)1500142-A	First edition created.
Sept. 2006	IB(NA)1500142-B	Contents were revised to correspond to Mitsubishi CNC700 Series software version D0. Contents were revised to correspond to Mitsubishi CNC70 Series software version A0. The "Section" (reference) was added in "3.7 G Code Lists". The following sections are added. <ul style="list-style-type: none"> <li>● 13.19 High-accuracy spline interpolation</li> <li>● Index</li> </ul> Mistakes were corrected.
Jun. 2007	IB(NA)1500142-C	The following sections are added. <ul style="list-style-type: none"> <li>7.7 Rapid Traverse Constant Inclination Acceleration/Deceleration</li> <li>7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration</li> <li>12.8 Inputting the tool life management data</li> </ul> Mistakes were corrected.
Aug. 2008	IB(NA)1500142-D	Contents were revised to correspond to Mitsubishi CNC700 Series software version F0. The following sections are added. <ul style="list-style-type: none"> <li>-3.4.2 Optional Block Skip Addition</li> <li>-12.4.10 Workpiece Coordinate Changing During Radius Compensation</li> <li>-13.1.2 Drilling Cycle High-Speed Retract</li> </ul> Mistakes were corrected. The following sections are added.



# Global service network



## North America FA Center (MITSUBISHI ELECTRIC AUTOMATION INC.)

**Illinois CNC Service Center**  
500 CORPORATE WOODS PARKWAY, VERNON HILLS, IL. 60061, U.S.A.  
TEL: +1-847-478-2500 (Se) FAX: +1-847-478-2650 (Se)

**California CNC Service Center**  
5665 PLAZA DRIVE, CYPRESS, CA. 90630, U.S.A.  
TEL: +1-714-220-4796 FAX: +1-714-229-3818

**Georgia CNC Service Center**  
2810 PREMIERE PARKWAY SUITE 400, DULUTH, GA., 30097, U.S.A.  
TEL: +1-678-258-4500 FAX: +1-678-258-4519

**New Jersey CNC Service Center**  
200 COTTONTAIL LANE SOMERSET, NJ. 08873, U.S.A.  
TEL: +1-732-560-4500 FAX: +1-732-560-4531

**Michigan CNC Service Satellite**  
2545 38TH STREET, ALLEGAN, MI., 49010, U.S.A.  
TEL: +1-847-478-2500 FAX: +1-269-673-4092

**Ohio CNC Service Satellite**  
62 W. 500 S., ANDERSON, IN., 46013, U.S.A.  
TEL: +1-847-478-2608 FAX: +1-847-478-2690

**Texas CNC Service Satellite**  
1000, NOLEN DRIVE SUITE 200, GRAPEVINE, TX. 76051, U.S.A.  
TEL: +1-817-251-7468 FAX: +1-817-416-1439

**Canada CNC Service Center**  
4299 14TH AVENUE MARKHAM, ON. L3R 0J2, CANADA  
TEL: +1-905-475-7728 FAX: +1-905-475-7935

**Mexico CNC Service Center**  
MARIANO ESCOBEDO 69 TLALNEPANTLA, 54030 EDO. DE MEXICO  
TEL: +52-55-9171-7662 FAX: +52-55-9171-7698

**Monterrey CNC Service Satellite**  
ARGENTINA 3900, FRACC. LAS TORRES, MONTERREY, N.L., 64720, MEXICO  
TEL: +52-81-8365-4171 FAX: +52-81-8365-4171

**Brazil MITSUBISHI CNC Agent Service Center**  
(AUTOMOTION IND. COM. IMP. E EXP. LTDA.)  
ACESSO JOSE SARTORELLI, KM 2.1 18550-000 BOITUVA – SP, BRAZIL  
TEL: +55-15-3363-9900 FAX: +55-15-3363-9911

## European FA Center (MITSUBISHI ELECTRIC EUROPE B.V.)

**Germany CNC Service Center**  
GOTHAER STRASSE 8, 40880 RATINGEN, GERMANY  
TEL: +49-2102-486-0 FAX: +49-2102486-591

**South Germany CNC Service Center**  
KURZE STRASSE. 40, 70794 FILDERSSTADT-BONLANDEN, GERMANY  
TEL: +49-711-3270-010 FAX: +49-711-3270-0141

**France CNC Service Center**  
25, BOULEVARD DES BOUVETS, 92741 NANTERRE CEDEX FRANCE  
TEL: +33-1-41-02-83-13 FAX: +33-1-49-01-07-25

**Lyon CNC Service Satellite**

**U.K CNC Service Center**  
TRAVELLERS LANE, HATFIELD, HERTFORDSHIRE, AL10 8XB, U.K.  
TEL: +44-1707-282-846 FAX: +44-1707-278-992

**Italy CNC Service Center**  
VIALE COLLEONI 7 - PALAZZO SIRIO, CENTRO DIREZIONALE COLLEONI, 20041 AGRATE BRIANZA - (MI), ITALY  
TEL: +39-039-60531-342 FAX: +39-039-6053-206

**Spain CNC Service Satellite**  
CTRA. DE RUBI, 76-80 -APDO.420 08190 SAINT CUGAT DEL VALLES, BARCELONA SPAIN  
TEL: +34-935-65-2236 FAX:

**Turkey MITSUBISHI CNC Agent Service Center**  
(GENEL TEKNİK SİSTEMLER LTD. STI)  
DARULACEZE CAD. FAMAS İS MERKEZİ A BLOK NO.43 KAT2 80270 OKMEYDANI İSTANBUL, TURKEY  
TEL: +90-212-320-1640 FAX: +90-212-320-1649

**Poland MITSUBISHI CNC Agent Service Center (MPL Technology Sp. z. o. o)**  
UL SŁICZNA 34, 31-444 KRAKOW, POLAND  
TEL: +48-12-632-28-85 FAX:

**Wrocław MITSUBISHI CNC Agent Service Satellite (MPL Technology Sp. z. o. o)**  
UL KOBIERZYCKA 23, 52-315 WROCLAW, POLAND  
TEL: +48-71-333-77-53 FAX: +48-71-333-77-53

**Czech MITSUBISHI CNC Agent Service Center**  
(AUTOCONT CONTROL SYSTEM S.R.O. )  
NEMOCNICNI 12, 702 00 OSTRAVA 2 CZECH REPUBLIC  
TEL: +420-596-152-426 FAX: +420-596-152-112

## ASEAN FA Center (MITSUBISHI ELECTRIC ASIA PTE. LTD.)

**Singapore CNC Service Center**  
307 ALEXANDRA ROAD #05-01/02 MITSUBISHI ELECTRIC BUILDING SINGAPORE 159943  
TEL: +65-6473-2308 FAX: +65-6476-7439

**Thailand MITSUBISHI CNC Agent Service Center (F. A. TECH CO., LTD)**  
898/19,20,21,22 S.V. CITY BUILDING OFFICE TOWER 1 FLOOR 12,14 RAMA III RD BANGPONGPANG, YANNAWA, BANGKOK 10120. THAILAND  
TEL: +66-2-682-6522 FAX: +66-2-682-6020

**Malaysia MITSUBISHI CNC Agent Service Center**  
(FLEXIBLE AUTOMATION SYSTEM SDN. BHD.)  
60, JALAN USJ 10/1B 47620 UEP SUBANG JAYA SELANGOR DARUL EHSAN MALAYSIA  
TEL: +60-3-5631-7605 FAX: +60-3-5631-7636

**JOHOR MITSUBISHI CNC Agent Service Satellite**  
(FLEXIBLE AUTOMATION SYSTEM SDN. BHD.)  
NO. 16, JALAN SHAHBANDAR 1, TAMAN UNGU TUN AMINAH, 81300 SKUDAI, JOHOR MALAYSIA  
TEL: +60-7-557-8218 FAX: +60-7-557-3404

**Indonesia MITSUBISHI CNC Agent Service Center**  
(PT. AUTOTEKININDO SUMBER MAKMUR)  
WISMA NUSANTARA 14TH FLOOR JL. M.H. THAMRIN 59, JAKARTA 10350 INDONESIA  
TEL: +62-21-3917-144 FAX: +62-21-3917-164

**India MITSUBISHI CNC Agent Service Center (MESSUNG SALES & SERVICES PVT. LTD.)**  
B-36FF, PAVANA INDUSTRIAL PREMISES M.I.D.C., BHOASRI PUNE 411026, INDIA  
TEL: +91-20-2711-9484 FAX: +91-20-2712-8115

**BANGALORE MITSUBISHI CNC Agent Service Satellite**  
(MESSUNG SALES & SERVICES PVT. LTD.)  
S 615, 6TH FLOOR, MANIPAL CENTER, BANGALORE 560001, INDIA  
TEL: +91-80-509-2119 FAX: +91-80-532-9480

**Delhi MITSUBISHI CNC Agent Parts Center (MESSUNG SALES & SERVICES PVT. LTD.)**  
1197, SECTOR 15 PART-2, OFF DELHI-JAIPUR HIGHWAY BEHIND 32ND MILESTONE GURGAON 122001, INDIA  
TEL: +91-98-1024-8895 FAX:

**Philippines MITSUBISHI CNC Agent Service Center**  
(FLEXIBLE AUTOMATION SYSTEM CORPORATION)  
UNIT No.411, ALABANG CORPORATE CENTER KM 25, WEST SERVICE ROAD SOUTH SUPERHIGHWAY, ALABANG MUNTINLUPA METRO MANILA, PHILIPPINES 1771  
TEL: +63-2-807-2416 FAX: +63-2-807-2417

**Vietnam MITSUBISHI CNC Agent Service Center (SA GIANG TECHNO CO., LTD)**  
47-49 HOANG SA ST. DAKAO WARD, DIST.1 HO CHI MINH CITY, VIETNAM  
TEL: +84-8-910-4763 FAX: +84-8-910-2593

## China FA Center (MITSUBISHI ELECTRIC AUTOMATION (SHANGHAI) LTD.)

**China CNC Service Center**  
2/F, BLOCK 5 BLDG.AUTOMATION INSTRUMENTATION PLAZA, 103 CAOBAO RD. SHANGHAI 200233, CHINA  
TEL: +86-21-6120-0808 FAX: +86-21-6494-0178

**Shenyang CNC Service Center**  
TEL: +86-24-2397-0184 FAX: +86-24-2397-0185

**Beijing CNC Service Satellite**  
9/F, OFFICE TOWER1, HENDERSON CENTER, 18 JIANGUOMENNEI DAJIE, DONGCHENG DISTRICT, BEIJING 100005, CHINA  
TEL: +86-10-6518-8830 FAX: +86-10-6518-8030

**China MITSUBISHI CNC Agent Service Center**  
(BEIJING JIAYOU HIGHTECH TECHNOLOGY DEVELOPMENT CO.)  
RM 709, HIGH TECHNOLOGY BUILDING NO.229 NORTH SI HUAN ZHONG ROAD, HAIJIAN DISTRICT , BEIJING 100083, CHINA  
TEL: +86-10-8288-3030 FAX: +86-10-6518-8030

**Tianjin CNC Service Satellite**  
RM909, TAIHONG TOWER, NO220 SHIZILIN STREET, HEBEI DISTRICT, TIANJIN, CHINA 300143  
TEL: +86-22-2653-9090 FAX: +86-22-2635-9050

**Shenzhen CNC Service Satellite**  
RM02, UNIT A, 13/F, TIANAN NATIONAL TOWER, RENMING SOUTH ROAD, SHENZHEN, CHINA 518005  
TEL: +86-755-2515-6691 FAX: +86-755-8218-4776

**Changchun Service Satellite**  
TEL: +86-431-50214546 FAX: +86-431-5021690

**Hong Kong CNC Service Center**  
UNIT A, 25/F RYODEN INDUSTRIAL CENTRE, 26-38 TA CHUEN PING STREET, KWAI CHUNG, NEW TERRITORIES, HONG KONG  
TEL: +852-2619-8588 FAX: +852-2784-1323

## Taiwan FA Center (MITSUBISHI ELECTRIC TAIWAN CO., LTD.)

**Taichung CNC Service Center**  
NO.8-1, GONG YEH 16TH RD., TAICHUNG INDUSTRIAL PARK TAICHUNG CITY, TAIWAN R.O.C.  
TEL: +886-4-2359-0688 FAX: +886-4-2359-0689

**Taipei CNC Service Satellite**  
TEL: +886-4-2359-0688 FAX: +886-4-2359-0689

**Tainan CNC Service Satellite**  
TEL: +886-4-2359-0688 FAX: +886-4-2359-0689

## Korean FA Center (MITSUBISHI ELECTRIC AUTOMATION KOREA CO., LTD.)

**Korea CNC Service Center**  
1480-6, GAYANG-DONG, GANGSEO-GU SEOUL 157-200, KOREA  
TEL: +82-2-3660-9631 FAX: +82-2-3664-8668

## **Notice**

Every effort has been made to keep up with software and hardware revisions in the contents described in this manual. However, please understand that in some unavoidable cases simultaneous revision is not possible.

Please contact your Mitsubishi Electric dealer with any questions or comments regarding the use of this product.

## **Duplication Prohibited**

This manual may not be reproduced in any form, in part or in whole, without written permission from Mitsubishi Electric Corporation.

© 2005-2008 MITSUBISHI ELECTRIC CORPORATION  
ALL RIGHTS RESERVED.

# MITSUBISHI CNC



MODEL	700/70 Series
MODEL CODE	008—362
Manual No.	IB-1500142(ENG)